PREDICTING SOLAR CYCLES USING PROBABILISTIC MACHINE LEARNING

By

GUNER AYGIN

1996263

A dissertation submitted to
the University of Birmingham
for the degree of
MASTER OF SCIENCE IN PHYSICS

Solar and Stellar Physics Research Group
School of Physics and Astronomy
College of Engineering and Physical Sciences
University of Birmingham
May 2023

ABSTRACT

The periodic nature of solar cycles has led many to attempt to predict the shape of future cycles, most of which are inaccurate or unreliable. Solar cycle research is critical for the protection of public health and forecasting of space weather, allowing necessary precautions to be taken when planning future space-related missions (Pesnell, 2012). The aim of this project was to develop suitable probabilistic machine learning techniques to improve the accuracy of solar cycle predictions, with a measure of prediction uncertainty. This was performed using Savitzky-Golay filtered daily sunspot number data with both linear regression and Gaussian process regression. Linear regression was used to prove that the existence of a linear relationship between a cycle's amplitude and descending time is unlikely, with a correlation coefficient $r = -0.35$, and gradient $m = -1.74^{+1.44}_{-1.36}$ (Z. Du and S. Du, 2006). Gaussian process regression was performed using a variety of kernel combinations and a custom sine-squared mean function, which proved essential for accurate forecasts. The Matern 5/2 $\times$ Periodic kernel was found to be the most suitable for this challenge, showing a 15-year forecast RMSE of 6.67, which is lower than previous attempts (Gonçalves, Echer, and Frigo, 2020). The overarching conclusion was Gaussian processes may be suitably used for short-term forecasts but fail at predicting further than two cycles due to an inherently short kernel length scale and heightened dependence on the mean function. Solar Cycle 25 is predicted to peak in September 2024, with an amplitude of 122^{+37}_{-28}.

ACKNOWLEDGMENTS

I would like to thank Guy Davies for all his continued support throughout this project, providing me with the resources to expand my knowledge of statistics and Gaussian processes, as well as improving my coding practices and pushing me to my potential.

WORD COUNT

8352

Contents

List of Figures

List of Tables

1

Introduction

Solar magnetic activity cycles, commonly known as solar cycles, occur due to the dynamic evolution of magnetic fluctuations within the Sun's convection zone (Balogh et al., 2015). These solar cycles are quantified by the number of sunspots visible on their surface and have long been known to be *quasi-periodic* in nature. The magnetic activity varies from a minimum to a maximum approximately every 11 years, however, predicting their exact form has remained an unresolved problem.

Periods of high solar activity are accompanied by explosive bursts of energy being released from the solar magnetic field in the form of solar flares and coronal mass ejections (CMEs) (Wang and Jiang, 2014). Changes in solar activity have a significant impact on Earth's atmosphere (Hargreaves, 1992), with heightened solar radiation increasing the exposure to harmful ultraviolet and X-ray radiation for astronauts, airline pilots, and passengers (Feminella and Storini, 1997). Solar flares and CMEs are responsible for extreme space-weather events, which have the potential to cause electrical power outages on Earth, as well as pose a risk to satellites that can result in irreparable damages, costing millions to replace and delaying research (G. Siscoe, 2000). Accurate predictions of future solar cycles can inform preventive measures on Earth and contribute to the success of long-term space missions (Pesnell, 2012).

Despite multiple endeavours, predicting solar cycles has proved a difficult task (Pesnell, 2012). Apart from their periodicity, the inconsistent shape of solar cycles makes any attempt at forecasting particularly difficult. Although there are promising theories explaining the mechanisms behind solar cycles, they are generally regarded to be poorly understood. The consensus reached by previous research is that Solar Cycle (SC) 25 will be similar to its predecessor, with estimates of its peak mean amplitude $\approx 95 \pm 20$ sunspots, forecast to occur around 2024 (Hathaway and Upton, 2016; Kitiashvili, 2016), however, others have forecast slightly higher amplitudes occurring around 2025 (Camacho, Faria, and Viana, 2022; Prasad et al., 2022).

The aim of this project was to utilise various probabilistic machine learning (ML) techniques to simplify the task of solar cycle forecasting, with the hopes of providing accurate predictions for future cycles, including their respective uncertainties.

This report gives a brief overview of the mechanisms driving solar cycles, and how previous

research has fared in the pursuit of solar cycle predictions (Sections 2 and 3). Section 4 explores the fundamental concepts of probabilistic ML, beginning with Bayes' theorem and leading to Gaussian processes (Section 5) - which are the main feature of this report. Probabilistic linear regression was also utilised to determine the existence of a relationship between a cycle's descending time and amplitude, which has the potential to aid our predictions for future cycle amplitudes. The results of these methods are outlined in Section 7, along with their respective discussions in Section 8.

2

Solar Cycle Theory

The solar cycle, which is generally characterised by the periodic variation in the number of sunspots observed on the solar surface, is primarily driven by the complex magnetohydrodynamic processes occurring within the Sun. In this section, we provide a generalised overview of how the solar cycle evolves, and how the number of sunspots serves as a metric for solar activity.

2.1 Solar Dynamo

It is widely accepted that the magnetic activity cycles observed in stars, including the Sun, are driven by a physical mechanism known as the hydromagnetic dynamo. A dynamo process requires rotation, convection, and an electrically charged fluid to stretch and twist the magnetic field (Schrijver and G. L. Siscoe, 2009). A hydrodynamic dynamo can generate and maintain a magnetic field in a conducting fluid (Wang and Jiang, 2014).

The Sun is known to be convective in its outer layers, resulting in a differential rotation of the solar plasma. This differential rotation helps to create a toroidal magnetic field, which is vital for understanding solar magnetic activity cycles. The Coriolis force in the Sun's outer layers stretches and twists the magnetic field lines, creating a structure that can interact with the convective motion to generate sunspots, flares, and other forms of magnetic activity (Schrijver and G. L. Siscoe, 2009). Understanding the hydromagnetic dynamo and the resulting magnetic activity cycles is a crucial step in studying the complex magnetohydrodynamic processes that occur within the Sun and other stars.

Creating a model out of this theoretical framework involves combining Maxwell's equations into a single equation describing the evolution of the magnetic field, known as the magnetohydrodynamical (MHD) induction equation (see Charbonneau, 2020; Davidson, 2002).

2.2 Sunspots

Sunspots are dark regions on the solar surface which manifest due to the prohibition of convection by the high magnetic field. The resulting effect is the emergence of areas cooler than the

surrounding surface, which consequently appear as dark spots (Hathaway, 2015). The number of sunspots varies over the course of a solar cycle and is often used to quantify the intensity of solar activity. In this report, solar cycles are measured using the number of sunspots visible S_N; as such solar cycles are used synonymously with sunspot cycles.

Sunspot numbers are recorded daily in a fashion such that each sunspot group counts as 10, and every umbra within each spot group is individually considered as 1 (SILSO World Data Center, 2023). There exists sunspot data going back as far as 1700, but the much older records are less reliable and only contain the mean sunspot numbers. For this project, daily total sunspot numbers were used, as seen in Fig. 1.

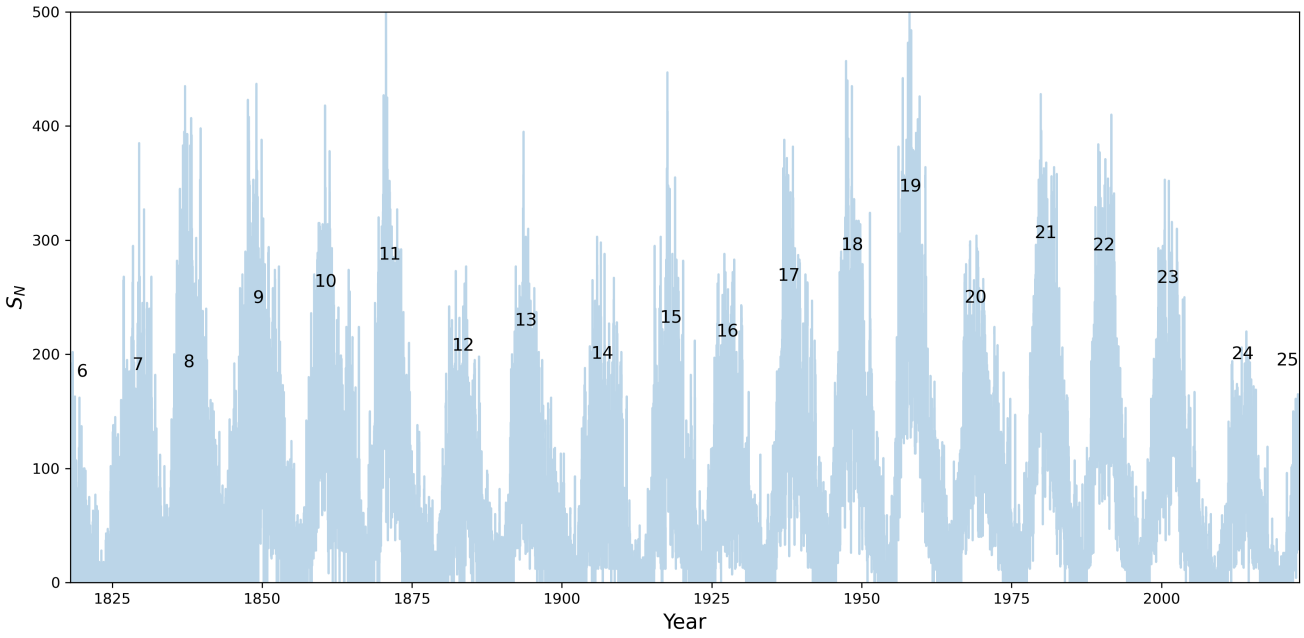

Fig. 1: Daily sunspot number data, from 1818 to 2023 (SC 6-25) SILSO World Data Center, 2023

3

Literature Review

A comprehensive understanding of the previous research methods used to predict solar cycles is essential for understanding the limitations of certain techniques, in order to improve future models and ultimately produce more accurate predictions. The research outlined in this section covers traditional simulation-based models to ML techniques.

3.1 Simulation-based Methods

The paper by Charbonneau, 2020 explores the simulation of the MHD equation, to study magnetic fluctuations and their role in the solar cycle. These simulations were used to predict how the solar cycle would evolve over time, but there were crucial limitations which prevented this approach from providing useful predictions. The first issue was that, due to the complexity of the turbulent processes driving the magnetic fields, there was a high computational cost of running these simulations. In practice, this meant that predictions could only be made for a very short time frame, rather than a complete cycle of 11 years, let alone multiple cycles. The second limitation was that the predictions made by the simulations were not accurate enough to offer any concrete predictions. Charbonneau, 2020 suggests that this was a result of an incomplete theoretical understanding of solar cycles, leading to uncertainties and inaccuracies in the model, which would, in turn, affect the prediction accuracy. The inaccuracies in this model compound with time, resulting in forecasts which are wildly different from the observed values. To address these limitations, Charbonneau, 2020 concludes that long-term predictions should be made using simpler models, which could be more efficient, and which should be less reliant on the incomplete theoretical understanding of the solar dynamo.

3.2 Data Assimilation

An improved method for modelling and forecasting solar cycles was attempted by Kitiashvili, 2016, using a technique known as data assimilation. This combined the numerical simulations of the MHD equation with observations to improve the accuracy and reliability of predictions (Kalnay et al., 1996). It was concluded that using data assimilation significantly improved the accuracy of solar activity predictions. However, the accuracy of this model is ultimately limited by the accuracy of the solar dynamo model, which we have already concluded is not understood

well enough for this task. This logically leads to methods which completely cast aside the physical theory and solely use empirical evidence to inform future predictions.

3.3 Empirical Methods

Machine learning is one such method which learns from the available data in order to make future predictions. The rapidly growing technique has been used in a host of different fields within physics as the availability of data has grown over the decades. ML techniques have recently been used to tackle the problem of solar cycle forecasting.

3.3.1 Linear Regression

The most basic example of an ML technique is linear regression, discussed in Section 6.3. Z. Du and S. Du, 2006 used linear regression on the smoothed monthly mean sunspot number to deduce that a relationship exists between the amplitude of a cycle its descending time (time from maxima to minima) three cycles earlier. The paper finds the significance of this correlation r to be too small when utilising all the available cycle data ($r = -0.383$), and in order to produce a statistically significant correlation a smaller sample is chosen which better agrees with their hypothesis (resulting in $r = -0.811$). The statistical significance of this correlation is explored in more detail in Section 7.2.

3.3.2 Gaussian Processes

Gaussian processes ($\mathcal{GP}$s) are an example of a probabilistic ML technique which has been used to model and forecast the shapes of solar cycles (see Section 5). One of the first instances of its use for solar cycle forecasting was by Gonçalves, Echer, and Frigo, 2020, where a "warped" $\mathcal{GP}$ was used to ensure positive-definite predictions.

The popular SE $\times$ Periodic kernel, commonly referred to as the *quasi-periodic* kernel, was used to model the 'chaotic evolution of the sunspot number, as well as its periodic characteristic', and was taken as a starting point for the models used in this project (Gonçalves, Echer, and Frigo, 2020). One of the drawbacks of their $\mathcal{GP}$ implementation was the lack of an appropriate mean function, which was set to zero. In order to find the optimal model parameters a 'genetic algorithm' was used, differing from the approach taken in this project. However, instead of optimising the period it was automatically set to $P = 10$ years, which may have led to less accurate predictions. Their forecasts had an RMSE value of $25 - 35$ for a 10-year prediction

window, with SC 25 estimated to occur in 2024, with an amplitude of $110 - 117$.

3.3.3 Neural Networks

Neural Networks (NN) have also shown great promise in solving astrophysical problems, being used in a host of classification and regression problems (Bloom et al., 2012). The ability of NN to learn complex patterns and relationships in data has made them a powerful tool for predictive modelling. Prasad et al., 2022 use NN to forecast solar cycles using deep long short-term memory networks. However, the NN is used mainly as a modelling technique throughout the paper, and the predictions made for SC 25 indicate a much higher-than-expected amplitude ($171. \pm 3.4$), suggesting other methods may contain better prospects for accurate forecasts. The paper also varies how predictions are made, predicting SC 25 with a different method to how all other cycles were predicted. This makes the errors calculated for previous cycles unrelated to how well SC 25 would be forecast.

It has been established that a Bayesian NN with a single layer and an infinite network width is equivalent to a $\mathcal{GP}$ (Lee et al., 2018). Using a NN with infinite layers is impossible, therefore it could be argued that any implementation of a NN should, in theory, be inferior to using a $\mathcal{GP}$, which is why $\mathcal{GP}$s were chosen as the focus of this project.

4

Bayesian Inference and Probabilistic Machine Learning

Bayesian inference is a statistical framework that combines prior knowledge with observed data to update beliefs about model parameters or variables. Understanding the key principles of Bayesian inference is crucial to creating a probabilistic ML model with the potential of forecasting solar cycles. This section outlines how probabilistic ML uses the concept of Bayes' theorem to make predictions, and the different methods by which model parameters are optimised, used in Sections 5 and 6.

4.1 Bayes' Theorem

Bayes' theorem is a mathematical expression describing how data should update our prior beliefs given a particular problem (Gelman et al., 1995). In the case of modelling, a *prior distribution* represents an initial belief about the model parameters prior to any data observations. These priors may stem from previous assumptions or investigations, or be selected to be non-informative, to reflect an absence of prior knowledge (Berger, 2006). For example, when defining a prior distribution for the solar cycle period one might choose a normal distribution with a mean of 11 years, and a standard deviation of 1, as this is what previous research suggests the period would be. For other, less understood parameters (such as the length scale of a $\mathcal{GP}$ kernel), one could simply use a broad uniform distribution.

When observed data is incorporated into our model our priors $\mathbf{w}$ are updated using Bayes' theorem, which calculates the *posterior distribution* $p(\mathbf{w}|\mathbf{y}, X)$ of the parameter. This posterior distribution embodies our revised beliefs after accounting for the data. Bayes' theorem can be expressed as:

$$p(\mathbf{w}|\mathbf{y}, X) = \frac{p(\mathbf{y}|X, \mathbf{w})p(\mathbf{w})}{p(\mathbf{y}|X)} \tag{1}$$

where $p(\mathbf{w})$ is the prior distribution, $p(\mathbf{y}|X, \mathbf{w})$ is the likelihood of the data given the parameters, and $p(\mathbf{y}|X)$ is the marginal likelihood, given by:

$$p(\mathbf{y}|X) = \int p(\mathbf{y}|X, \mathbf{w})p(\mathbf{w})d\mathbf{w}. \tag{2}$$

In most cases, the posterior distribution is very quickly dominated by the likelihood of the data, and so an uninformative prior can often be suitable. The advantage of calculating the posterior distributions of model parameters is that it enables a clear assessment of whether the posterior has effectively converged to an optimal value. Additionally, it provides a quantitative measure of the likelihood that the model parameters have been accurately selected.

However, the integrals to calculate the marginal likelihood are often intractable and difficult to solve analytically. Therefore, the posterior distribution is obtained using sampling as an approximation.

4.2 Parameter Optimisation Methods

A key aspect of modelling data is determining the optimal parameter values. This can be done in a variety of ways, but the methods used throughout this project include maximum likelihood estimation (MLE), maximum a posteriori estimation (MAP), and MCMC sampling.

4.2.1 Maximum Likelihood Estimation (MLE)

To optimise a model's parameters one must find the values which maximise the likelihood of the observed data. This technique is known as MLE. The likelihood of a particular set of data is determined based on the properties of that data. As an example, take a simple linear model of the form $y = mx + b$, where each data point follows this equation plus some small Gaussian distributed noise with a mean of zero and variance σ^2 (Hogg, Bovy, and Lang, 2010). The likelihood for this model is Gaussian, given by:

$$p(y_n|x_n, \sigma, m, b) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y_n - mx_n - b)^2}{2\sigma^2}\right] \tag{3}$$

To optimise the model parameters one would maximise the likelihood function, however in practice the easiest way to maximise the likelihood function is to *minimise* the negative of the log-likelihood, given by:

$$\ln p(y_n|x_n, \sigma, m, b) = -\frac{1}{2}\sum_n \left[\frac{(y_n - mx_n - b)^2}{\sigma^2} + \ln(\sigma^2)\right], \tag{4}$$

where y_n is the observed value, x_n represents the predictor variable, m and b are the model parameters.

This minimisation can be achieved analytically for simple functions, or by using an algorithm, such as a gradient descent algorithm (see Appendix A.2) to find the values of m, b and σ which minimise the log-likelihood. The built-in *scipy.optimize* efficiently performs the necessary calculation (Jones, Oliphant, and Peterson, 2007).

4.2.2 Maximum A Posteriori Estimation (MAP)

The MAP values are an alternative way of calculating the optimum model parameters. In this instance, the parameters which maximise the posterior function are calculated. This is achieved by first assigning priors to the parameters, and defining the likelihood (often Gaussian, the same as Eq. 3). The posterior distribution is then calculated with Eq. 1.

The posterior distribution is maximised in the same fashion as the likelihood: by minimising the negative of the log-posterior distribution. This can again be achieved using a gradient descent algorithm, or conveniently calculated using the pre-built 'MAP finder' in PyMC3 (Salvatier, Wiecki, and Fonnesbeck, 2016).

4.2.3 MCMC Sampling

Sampling is used to provide an estimate of a parameter's posterior distribution, which is then used to determine the optimal parameter value. This is generally performed using a method known as a Markov Chain Monte Carlo (MCMC) simulation, which is designed to explore the posterior distribution by generating samples in a sequence (Foreman-Mackey et al., 2013).

MCMC algorithms create a Markov chain by iteratively suggesting new parameter values that depend on the present state. The *proposal distribution* defines the likelihood of moving from the current state to the new state. A proposal distribution with a small variance would lead to many rejected proposals, whereas a large variance may take longer to converge on the posterior distribution. A popular choice for the proposal distribution is a Gaussian distribution centred around the current state (Roberts and Rosenthal, 2009).

At each iteration, the likelihood and prior distribution are evaluated for the new state, and an *acceptance probability* is calculated based on the ratio of the posterior densities at the new and current states. The new state is either accepted with a probability equal to the acceptance probability; otherwise, the current state is retained. After a fixed number of iterations, the resulting Markov chain converges to the target distribution, which is the posterior distribution of the model parameters. The samples from the Markov chain are then used to determine the

optimal model parameters.

Fig. 2 shows an example of an MCMC chain, which is given a random starting point. The subplot shows the 'burn-in period' of the algorithm, where it explores the parameter space as it tries to find the target distribution. After a short while the algorithm converges on the target distribution and produces the noisy plot in Fig. 2. The burn-in period is discarded and the remaining chain is used to estimate the posterior distribution of the model parameters.

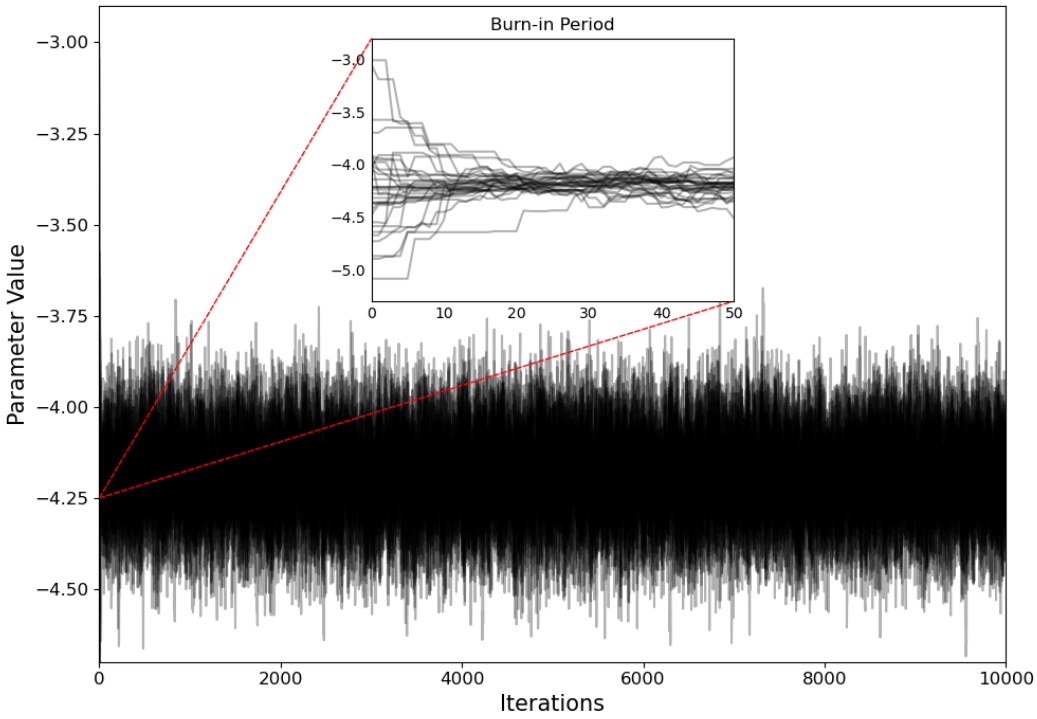

Fig. 2: MCMC chain, used for a linear regression example where the true parameter value is -4.2. The subplot shows the burn-in period of the first 50 MCMC iterations, and how it eventually converges on the optimal value.

In general, MCMC sampling is the most robust method of determining the optimal model parameter as it provides us with the posterior distribution. However, the sampling process can be computationally intensive and time-consuming, which often makes them less favourable than MLE or MAP.

4.3 Probabilistic Machine Learning

Machine learning involves using algorithms to identify patterns and relationships in a data set. Probabilistic ML incorporates uncertainty into the learning process. Unlike traditional ML models, which typically output a single deterministic prediction, probabilistic models output a probability distribution over possible outcomes, using Bayes' theorem and sampling. This

allows probabilistic models to not only make predictions but also provide a measure of the uncertainty or confidence in those predictions.

For the challenge of solar cycle forecasting, it is vital to have a robust understanding of the limitations and uncertainties associated with a given model, so as not to prematurely draw an incorrect conclusion. Probabilistic ML models provide these uncertainties, and as such are gaining favour over traditional ML models. However, one of the main drawbacks of using probabilistic models is that they are generally more computationally expensive than traditional ML models. Another is that model selection is generally considered to be quite difficult, as seen in Section 6.4.2 (Duvenaud, 2014).

Probabilistic ML is particularly well-suited to the problem of predicting solar cycles due to their inherently uncertain nature and would allow us to quantify the uncertainty and provide a confidence level in our predictions. In the following section, we delve into the theoretical framework behind Gaussian Processes. By providing this background on Bayesian inference, we aim to help readers better understand the foundations and motivations behind our chosen methods.

5

Gaussian Processes

Gaussian processes are a flexible and powerful probabilistic ML technique which utilises Bayes' theorem to output predictions and their uncertainties. This technique has been growing in popularity for use in regression and classification tasks. They are particularly useful for modelling data where the functional form is not known *a priori*, such as the solar cycles.

5.1 Training and Predicting with $\mathcal{GP}$s

The definition of a $\mathcal{GP}$ is "a collection of random variables, any finite number of which have a joint Gaussian distribution" (Williams and Rasmussen, 2006). In other words, the values of the variables are related to each other in such a way as to allow predictions to be made regarding their future behaviour. A $\mathcal{GP}$ is defined by a mean function $\mu(\mathbf{x})$, which encodes the long-term average of the data (often set to zero), and a covariance function (kernel) $k(\mathbf{x}, \mathbf{x}')$ which specifies the degree of similarity between different data points.

For a given set of n 'training' data points

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \ \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix},$$

the covariance between points $\mathbf{x}$ and $\mathbf{x}'$ can be written as an $n \times n$ matrix[1]

$$k(\mathbf{x}, \mathbf{x}') = \begin{bmatrix} k(x_1, x_1) & k(x_1, x_2) & \dots & k(x_1, x_n) \\ k(x_2, x_1) & k(x_2, x_2) & \dots & k(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_n, x_1) & k(x_n, x_2) & \dots & k(x_n, x_n) \end{bmatrix}. \tag{5}$$

Each observation $\mathbf{y}$ is said to be related to a function $f(\mathbf{x})$. However, our real sunspot data contains noise, which can be incorporated into our models as a Gaussian-distributed term like:

$$\mathbf{y} = f(\mathbf{x}) + \mathcal{N}(0, \sigma_n^2), \tag{6}$$

[1] In Eq. 8 the kernels are size $n \times n_*$.

where we assume

$$f(\mathbf{x}) \sim \mathcal{GP}\left(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right). \tag{7}$$

Thus, any finite set of function values $f(\mathbf{x}_*)$ are said to be jointly Gaussian distributed like:

$$\begin{bmatrix} \mathbf{y} \\ f(\mathbf{x}_*) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{a} \\ \mathbf{c} \end{bmatrix}, \begin{bmatrix} A & B \\ B^T & C \end{bmatrix}\right), \tag{8}$$

where $\mathbf{a} = \mu(\mathbf{x})$, $\mathbf{c} = \mu(\mathbf{x}_*)$, $A = k(\mathbf{x}, \mathbf{x}) + \sigma_n^2 I$, $B = k(\mathbf{x}, \mathbf{x}_*)$, and $C = k(\mathbf{x}_*, \mathbf{x}_*)$. $\mathbf{x}$ and $\mathbf{x}_*$ are the observed and input data points respectively (Williams and Rasmussen, 2006).

Optimising the $\mathcal{GP}$ involves maximising the marginal likelihood, which typically requires an integral. However, the marginal likelihood of a $\mathcal{GP}$ can be more efficiently calculated by finding the logarithm of the marginal likelihood, given by:

$$\ln p(\mathbf{y}|\mathbf{x}) = -\frac{1}{2}\mathbf{y}^T A^{-1}\mathbf{y} - \frac{1}{2}\ln|A| - \frac{n}{2}\ln 2\pi. \tag{9}$$

This operation scales as $\mathcal{O}(n^3)$ due to the matrix inversion, which is one of the crucial drawbacks of $\mathcal{GP}$s, making them only practical for use on relatively small data sets. Once a $\mathcal{GP}$ has been successfully trained, it can be used to make predictions.

The joint distribution is used to calculate the conditional probability distribution $p\left(f(\mathbf{x}_*)|\mathbf{y}\right)$ for new test inputs $\mathbf{x}_*$, given by:

$$p\left(f(\mathbf{x}_*)|\mathbf{y}\right) = \mathcal{N}\left(\mathbf{c} + BA^{-1}(\mathbf{y} - \mathbf{a}), C - BA^{-1}B^T\right). \tag{10}$$

This distribution gives both the mean prediction (first term), as well as the uncertainty in each prediction (second term). Thus using a $\mathcal{GP}$ allows one to obtain predictions without having to calculate a computationally intensive integral.

It should be noted that the uncertainty in a $\mathcal{GP}$ model's predictions greatly increase the further away the input value is from the last training point. A visualisation of how a $\mathcal{GP}$ learns from data can be seen in this GIF, where each additional data point improves the fit, but we see that the uncertainties quickly increase the further the model predicts from the final training point.

5.2 The Kernel

As already outlined, a $\mathcal{GP}$ is fully described by its mean and covariance function. In order to capture the key structures of the solar cycles the choice of kernel is critical for creating an accurate $\mathcal{GP}$ model, alongside defining an appropriate mean.

Kernels can be combined linearly by adding or multiplying two or more kernels (k_1, k_2) together to form a new kernel. When one adds two kernels together the resulting kernel has the potential to have the properties of k_1 or k_2, due to one being more dominant over the other in terms of magnitude. Multiplying two kernels represents an interaction between them, resulting in a kernel with properties of both k_1 and k_2.

The covariance functions utilised in this project are defined in Appendix A.5. The kernel is often chosen based on some prior knowledge or assumptions about a particular data set. The solar cycles are often modelled using a *quasi-periodic* kernel, which is chosen to capture both the periodic patterns observed for each cycle, as well as the erratic behaviour of the cycle amplitudes. There are many kernel combinations which can be considered *quasi-periodic*, however, the most common one used for forecasting solar cycles is the SE $\times$ Periodic kernel (Gonçalves, Echer, and Frigo, 2020), the equation for which is given by:

$$k(\mathbf{x}, \mathbf{x}') = A^2 \exp\left[-\frac{(\mathbf{x}-\mathbf{x}')^2}{2l^2} - \frac{1}{l_P^2}\sin^2\left(\frac{\pi(\mathbf{x}-\mathbf{x}')}{P}\right)\right] + \sigma^2\delta_{\mathbf{x},\mathbf{x}'}, \tag{11}$$

where A is the amplitude, l and l_P are the respective length scales of the SE and periodic kernel, P is the period, σ is the uncertainty, and $\delta_{\mathbf{x},\mathbf{x}'}$ is the delta function. These parameters are also used for many of the other kernels in Appendix A.5. The length scale of a kernel determines how far apart two points must be before the covariance between them becomes negligible. Kernels which are optimised to have long length scales are therefore able to predict further into the future than kernels with short length scales.

Kernels can be categorised into two types: stationary and non-stationary kernels. A stationary kernel is a class of kernel which depends solely on the relative distance between the input points, such that $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}')$. This gives them the property of translational invariance, creating models with a constant level of correlation throughout the input space. These are useful when modelling data with consistent patterns. Solar cycles are assumed to be stationary on the timescale of a few hundred years, which is a sensible assumption due to stellar evolution taking place over millions of years. For this reason, only stationary kernels were used in solar cycle modelling.

6

Methods

6.1 Sunspot Number Smoothing

The daily sunspot number data (from 1818 - 2023) used in this project contained a high degree of noise which had a significant negative impact on computation time during modelling (SILSO World Data Center, 2023). To reduce the computational cost during training (specifically for $\mathcal{GP}$ modelling) the data was smoothed using a Savitzky-Golay (SG) filter, differing from other methods which use the rolling average or annual sunspot number (Z. Du and S. Du, 2006; Gonçalves, Echer, and Frigo, 2020).

The SG filter works by applying a local polynomial regression to each data point within a moving window, resulting in a smoothed value for that specific point. This process is repeated for each data point in the data set, ultimately generating a new, smoothed data set (Press and Teukolsky, 1990). This approach can be more effective at reducing noise than global smoothing techniques, such as moving averages because it is less likely to obscure important local features in the data, which is why it was chosen for this research.

The window length was set to 4 years, with three different polyorders used, each providing a different level of smoothing as seen in Fig. 3. The smoothest signal (polyorder = 1) was chosen for all other modelling described in this report. Whilst the smoothest function necessarily loses some of the intricate details of the solar cycles it's a much simpler function for the ML models to work with. The philosophy was, if the smoothest version of the solar cycle cannot be successfully predicted, then a more complex shape would fare even worse. If the models were able to successfully predict the smooth signals, then the higher polyorder data could be substituted in its place.

6.2 Solar Cycle Period

The widely quoted 11-year period of solar cycles was investigated to asses how consistent each cycle's period was with this value. The period was calculated in two ways, by finding the time difference between two successive maxima and minima, where each maxima/minima was located using *scipy*. The mean maxima period and mean minima period were used to find the mean overall period P.

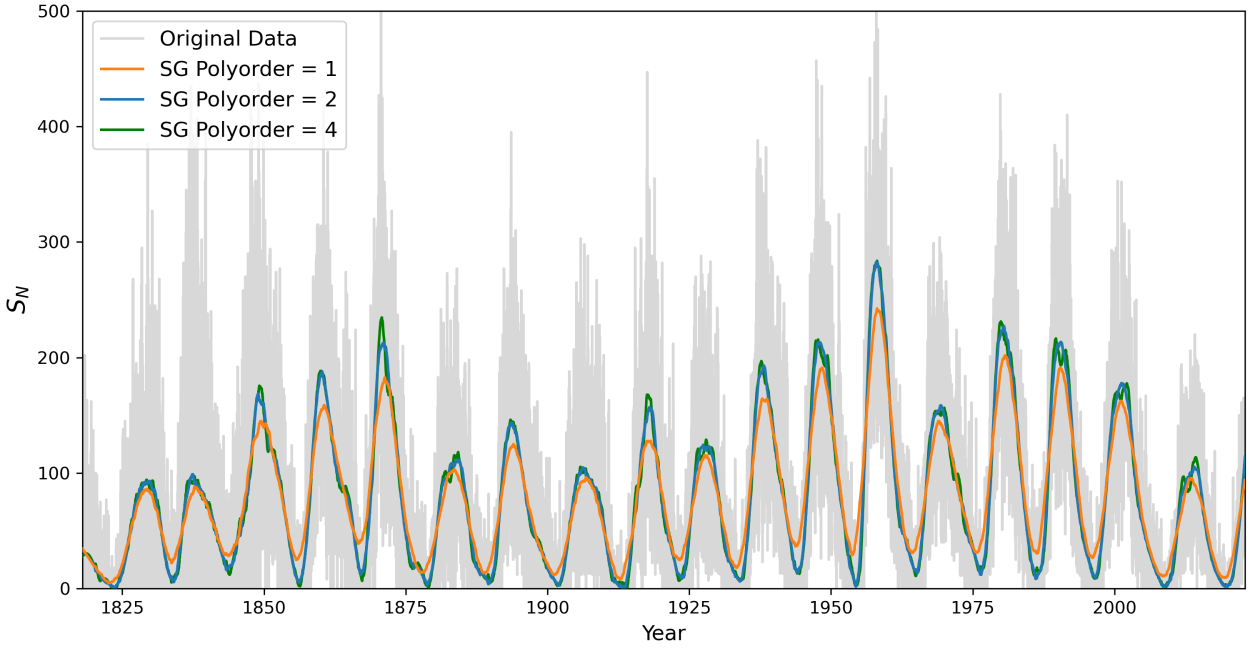

Fig. 3: Plot of SG filtered signals over the original, raw data. Window length = 1461 for each signal, with the corresponding polyorder displayed in the legend.

Two phase diagrams were constructed by first splitting the data into sections of length P and $2P$ years. The cycles were then plotted over each other, creating the diagram. If the cycles each had a period of P then they would perfectly overlap.

6.3 Descending Time and Cycle Amplitude

To investigate the relationship between a cycle's descending time and amplitude, as discussed in Section 3.3.1, the Pearson correlation coefficient r was calculated, alongside performing a simple linear regression analysis to determine the optimal model parameters, using the techniques outlined in Section 4.2. The aim was to determine whether there exists a statistically significant non-zero gradient, indicating a relationship between the two variables.

Recreating the experiment required calculating the descending times, achieved by finding the time between a cycle's maxima and its subsequent minima, using the locations found in Section 6.2. The amplitudes were determined as the maximum value of each cycle. Both were calculated using the SG signal of polyorder = 1. Due to the investigation requiring a cycle's descending time three cycles earlier, the only cycles available for investigation were SC 9-24.

To create the linear regression model, the log-likelihood function was defined as in Eq. 4, and the priors were chosen to be non-informative, uniform priors ($\mathcal{U}$) as shown in Table 1.

Table 1: Prior distributions for the parameters used in the linear regression between amplitude and descending time.

Parameter	**Prior**
m	$\mathcal{U}[-20, 20]$
b	$\mathcal{U}[-500, 500]$
σ	$\mathcal{U}[0, 20]$

These functions were used to calculate the MLE and MAP model parameters, which were plotted alongside 2500 MCMC samples[2]. The posterior distribution of the gradient m contains the meaningful information related to the correlation's significance, with the mean corresponding to the optimal parameter value for m.

6.4 $\mathcal{GP}$ Regression

$\mathcal{GP}$ regression was used to model the periodic shape of solar cycles, using sampling to determine the optimal model parameters. $\mathcal{GP}$s were chosen due to their unique ability to sample over all possible functions, with the aim of converging on the function best describing the evolution of the solar cycles. The theoretical workings of a $\mathcal{GP}$ are outlined in more detail in Section 5.

6.4.1 Data Pre-Processing

The full solar cycle data set contains nearly $75,000$ points, making modelling with a $\mathcal{GP}$ extremely computationally expensive and time-consuming due to the $\mathcal{O}(n^3)$ dependency. In order to reduce the computation time the data was pre-processed such that only every 400 points were used as part of the modelling process. Due to the data already being smoothed (Section 6.1), only taking every 400 points had a minimal effect on the gross structure of the solar cycles, which was vital for efficient modelling. This would not have been achievable without SG smoothing, as the raw data is too noisy to allow every 400 points to effectively capture the overall cycle shape.

The data was further divided into two sections: training data, and validation data. In order to assess our model's forecasting ability the $\mathcal{GP}$ first learnt from a given set of data (training data), and made predictions for some unseen data (validation data). The $\mathcal{GP}$ was trained using

[2]out of 10,000, discarding a burn-in period of 100 iterations

three different training-validation splits, as indicated in Table 2. This was done to assess how efficiently the different $\mathcal{GP}$ models were able to learn from different amounts of data, and whether the best model for Split 1 remained the best model for Split 3.

Table 2: Training and Validation data splits for the $\mathcal{GP}$ models.

Split	**Training Data (%)**	**Validation Data (%)**
1	92.5	7.5
2	86.5	13.5
3	80.5	19.5

Sunspot numbers are always ≥ 0, so to ensure positive definite predictions the logarithm of the data was taken before using it with the $\mathcal{GP}$ model (adjusting parameter priors accordingly); this constrained the model to only allow positive sunspot number predictions without the need of a warping function. The training data splits can be seen in Fig. 4. Predictions were returned to non-log values by using the exponential function.

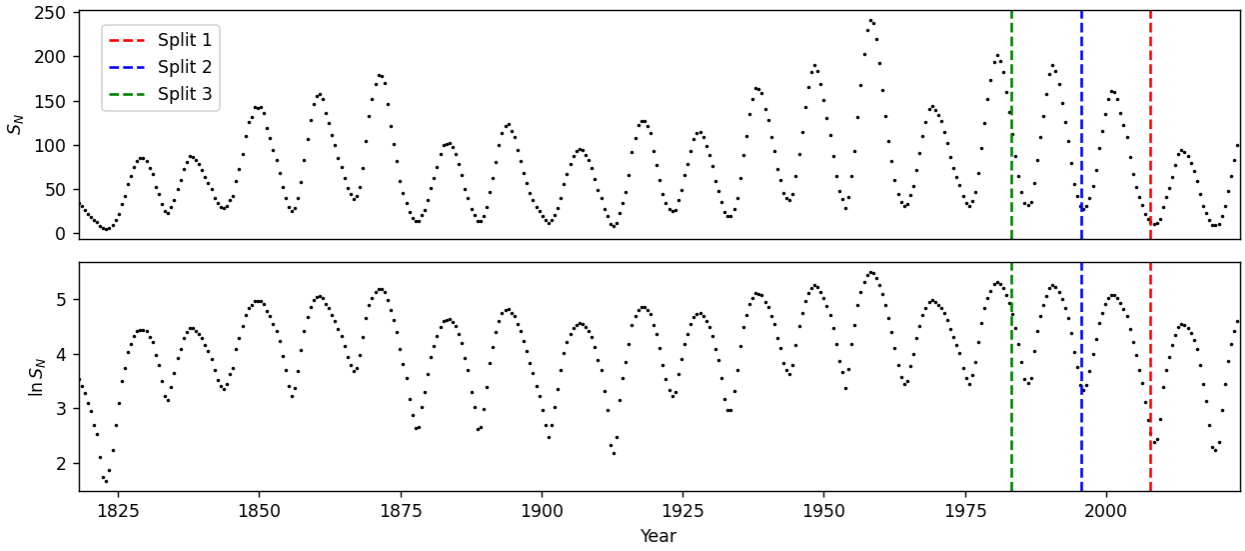

Fig. 4: Sunspot and ln sunspot data, taking every 400 points. The dashed lines show the three different training limits used.

6.4.2 Kernel Choice

As outlined in Section 5.2 the choice of kernel is critical for the success of any $\mathcal{GP}$ model. Whilst the SE $\times$ Periodic kernel is a popular choice for this particular problem, due to the limited accuracy of current predictions a range of different kernel combinations were experi-

mented with to test if any produced better results. Each of the kernels was chosen to display some form of periodicity via a linear combination with the Periodic kernel. The Cosine kernel was also used to see if it would serve as a suitable substitute for the Periodic kernel. The kernel combinations used can be seen in Fig. 7, with each kernel defined in Appendix A.5. Each kernel is called in PyMC3 using the built-in, pre-defined kernels, which each take different input parameters.

6.4.3 *Mean Function Choice*

The default $\mathcal{GP}$ mean function is zero, and this, along with a constant mean function, are the typical choices for $\mathcal{GP}$ models attempting to forecast solar cycles. The issue with using either of these methods is that the solar cycles appear to exhibit a time-dependent average rather than a constant long-term average.

In order to better model the solar cycles, a custom-built sine-squared mean function was employed, which more closely resembles the shape of a solar cycle (see Section 8.1). By using a mean function which more accurately captures the general shape of the solar cycles the kernel should be better able to capture the pattern in the varying amplitudes and provide more accurate forecasts. The mean function is given by:

$$\mu(\mathbf{x}) = \Lambda^2 \sin^2\left(\frac{\pi \mathbf{x}}{P} + \phi\right) + c, \tag{12}$$

where Λ is the amplitude, ϕ is the phase, c is the offset, and P is the period: which is the same as that used in the kernel.

The $\mathcal{GP}$ model was tested with and without this mean function, to assess whether it had a significant impact on the forecasting ability. Split 3 was used to train the models, along with the same priors as in Table 3.

6.4.4 *$\mathcal{GP}$ Optimisation*

The $\mathcal{GP}$ model was constructed in PyMC3 by stating the priors for each of the parameters, before defining the $\mathcal{GP}$'s kernel and mean function. Priors were adjusted until the model outputted suitable predictions, with an attempt to keep them as wide as possible so as to eliminate personal bias. The final priors used for the different $\mathcal{GP}$ model parameters are shown in Table 3, where $\mathcal{U}$ and $\mathcal{N}$ are the uniform and normal distributions respectively. Parameter α is specific to the Rational Quadratic (RQ) kernel, all the others are applicable to the other kernels given in Appendix A.5.

Table 3: Table of parameters and their priors used for all of the $\mathcal{GP}$ models.

Parameter	**Prior**
Λ	$\mathcal{U}[0, 100]$
ϕ	$\mathcal{N}[1.73, 0.5]$
c	$\mathcal{U}[0, 100]$
P	$\mathcal{U}[10.5, 11.5]$
l_P	$\mathcal{U}[0, 150]$
A	$\mathcal{U}[0, 250]$
l	$\mathcal{U}[0, 150]$
σ	$\mathcal{U}[0, 1]$
α	$\mathcal{U}[0, 1000]$

PyMC3 was used to calculate the ln marginal likelihood (Eq. 9) and optimise the model parameters. MCMC sampling was performed to obtain the posterior distributions for each of the parameters, to ensure they converged properly. Predictions were generated for times between 1818 and 2023 for each split by computing the conditional distribution (Eq. 10), of which 1000 samples were taken from the predictive distribution to represent the possible functions which best fit the data. The accuracy of each model's forecast was quantified by calculating the RMSE and MAE values (see Appendix A.4).

6.4.5 Long-term Forecasts

The best fitting $\mathcal{GP}$ model from the kernel experimentation was trained using all of the available data, again skipping every 400 points, and predictions were made until the year 2100. These predictions were used to assess whether long-term solar cycle forecasting was feasible and to obtain predictions for SC 25.

The amplitude and time of maxima were calculated by taking the mean of the predictions and locating the peaks and minima using the same technique as outlined in Section 6.3, with the uncertainties taken from the $\mathcal{GP}$'s predictive distribution.

7

Results

7.1 Cycle Period Analysis

The solar cycles were all found to have different periods, in the range of $8.48 \leq P_n \leq 12.77$ years. The mean period was calculated to be $P = 10.88 \pm 0.01$ years, which is slightly less than the widely quoted 11 years (Charbonneau, 2020). The phase diagram can be seen in Fig 5,

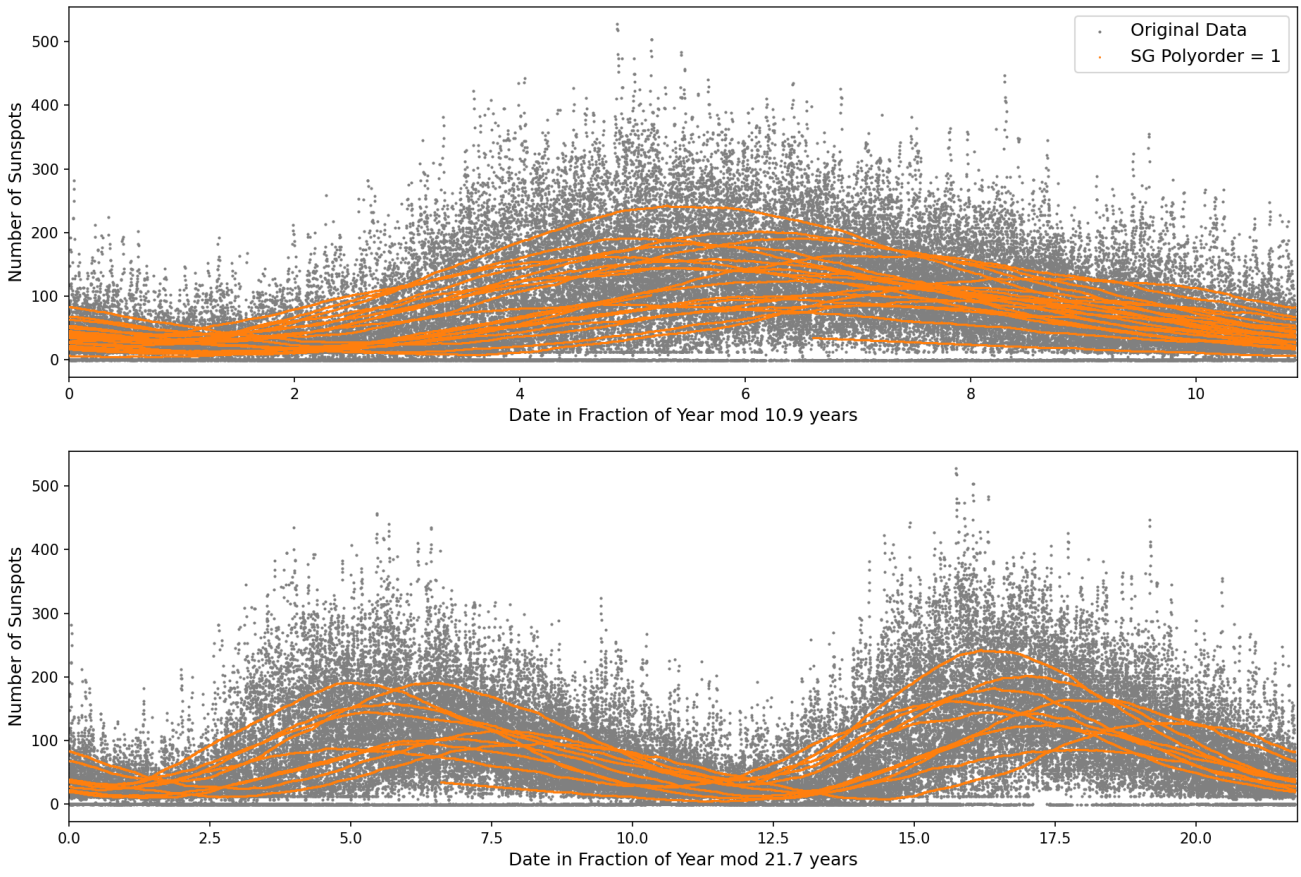

Fig. 5: Phase diagrams of the cycle period for SC 7-24, with both raw data and SG signal with polyorder = 1.

showing how well each cycle's period agrees with the mean. The diagram also shows a range in the cycle amplitudes between $86 \leq A \leq 242$.

7.2 Relationship between descending time and cycle amplitude

The Pearson correlation coefficient for the data was found to be $r = -0.35$. Fig. 6 shows the results of the three linear regression methods used to find a relationship between the amplitude and descending time of a cycle (see Z. Du and S. Du, 2006).

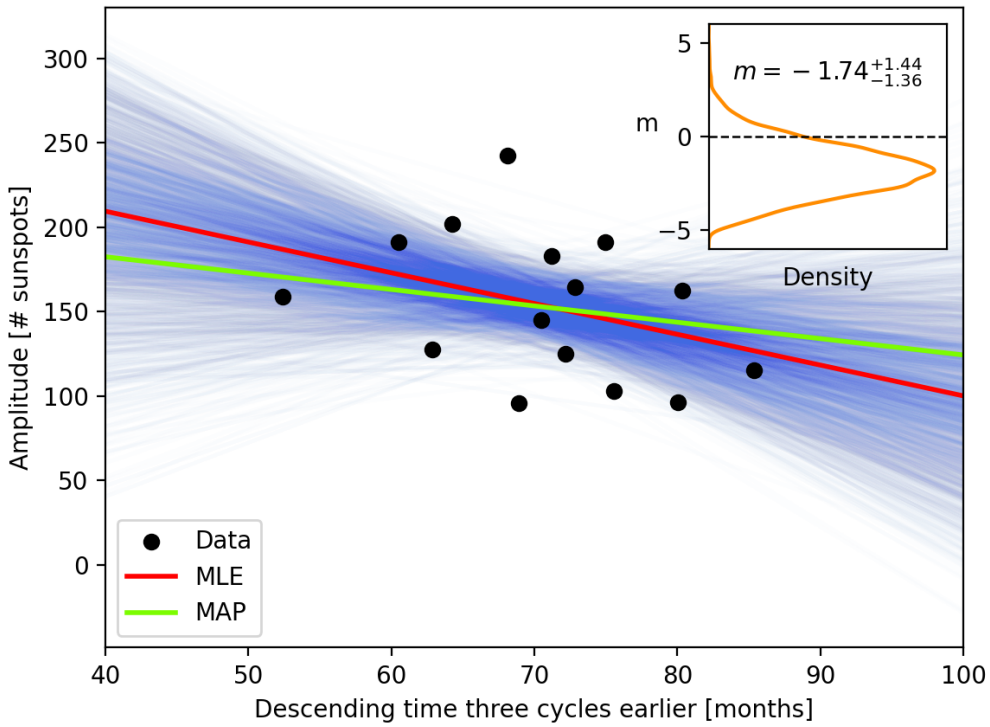

Fig. 6: Plot of Amplitude against Descending Time three cycles earlier with MLE, MAP and MCMC methods. MCMC samples are plotted in blue. MLE: $m = -1.82$, MAP: $m = -0.96$. The subplot shows the posterior distribution of m. The black dotted line shows the location of $m = 0$ in the posterior distribution.

The MLE and MAP fit both show a negative correlation between the two variables, as found by Z. Du and S. Du, 2006. The samples show a wide range of possible fits, most of which indicate a negative correlation, with some indicating no or even a positive correlation.

The posterior distribution of the gradient can be seen in the top right corner of the figure. The mean of the distribution is $m = -1.74^{+1.44}_{-1.36}$.

7.3 $\mathcal{GP}$ Results

7.3.1 Kernel Comparison

The errors from each of the different $\mathcal{GP}$ forecast can be seen in Fig. 7. The Matern 5/2 $\times$ Periodic kernel model had the overall lowest errors, with an RMSE of 6.67 for Split 1. The SE $\times$ Cosine had the highest errors for Split 1, with an RMSE of 33.85.

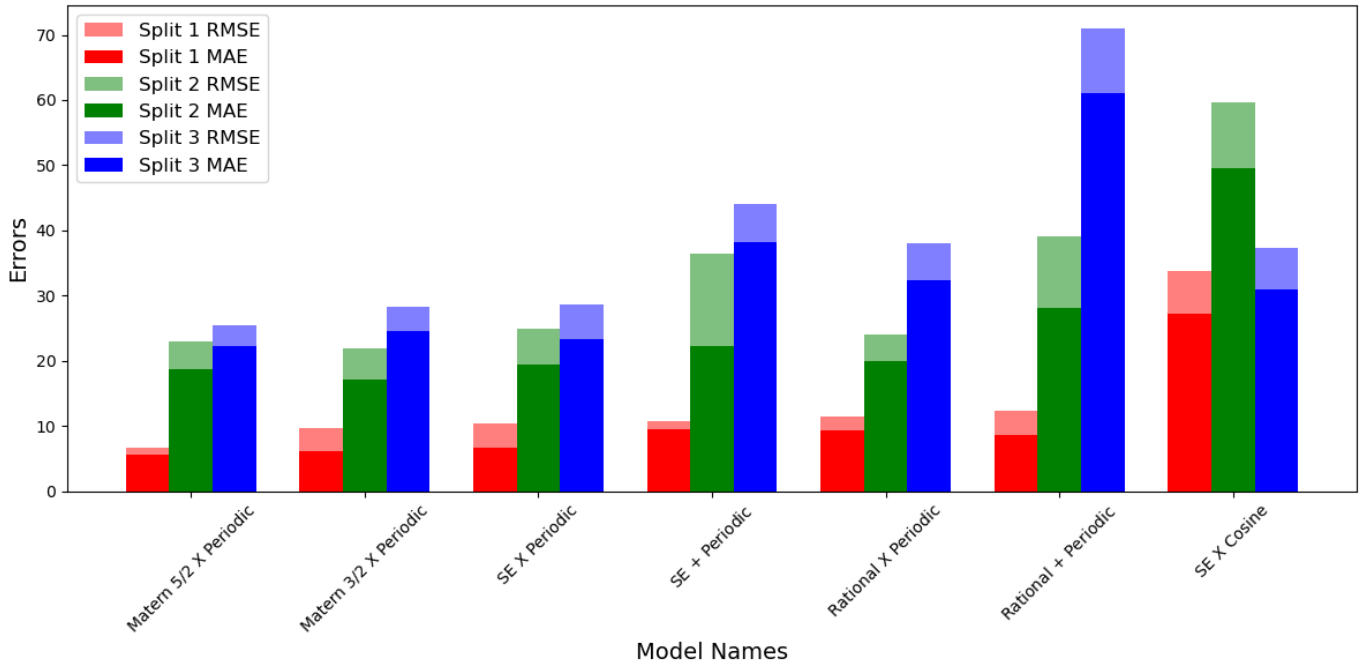

Fig. 7: RMSE values for each of the different $\mathcal{GP}$ models trained, with the MAE values overlapping. Values are given for each of the three training splits, given in Table 2. The bars are ordered in ascending order of Split 1 RMSE.

7.3.2 Matern 5/2 $\times$ Periodic kernel Predictions

The full $\mathcal{GP}$ model for the Matern 5/2 model can be seen in Fig. 8, where each subplot indicates a different training split. The traces represent the possible functions that fit with the data, with areas of high density representing the functions which are most consistent with the observed data. The range of the traces reflects the range of possible functions that can fit the data within the uncertainty bounds of the model. In other words, if the range of the traces is wider, it means that the model is less certain about the shape of the underlying function and therefore has a higher level of uncertainty in its predictions. Conversely, a narrower range of traces indicates that the model is more certain about the shape of the underlying function and has a lower level of uncertainty in its predictions.

A consequence of training the $\mathcal{GP}$ with the $\ln$ data is that some traces have very high amplitudes when reverting $\ln$ predictions back to normal sunspot numbers. As a result, the figure is cut to only show up to 270 sunspots, in order to better judge the model's accuracy.

7.3.3 Optimal Covariance Parameters

The posterior distributions for the Split 1 $\mathcal{GP}$ model's parameters can be seen in Fig. 9. Each of the parameters can be seen to effectively converge onto their mean values, apart from the period. The optimal period was found to be 11 years, however, the posterior shows two different peaks for the period: 10.83 and 11.01 years, supported by the period inconsistency found in Section 7.1. The optimal values of the two length scales are particularly important for assessing how long the kernel is able to effectively capture the correlations in the data.

A visualisation of the kernel can be seen in Fig. 10. This plot illustrates how the covariance between two points changes as a function of distance from the final training point and a new input, or in other words how far in the future our models can successfully forecast. The covariance contains a periodicity, as well as a changing amplitude due to the Matern term.

7.3.4 Mean Function Implementation

A comparison between the $\mathcal{GP}$ models for the same kernel as above, with and without implementing the custom *sine-squared* mean function can be seen in Fig. 11. The figure clearly shows that having a mean function of zero makes forecasts dramatically worse than those which assume a time-dependent mean, and the forecasts very quickly drop to zero. The uncertainties in the zero mean model also increase much faster than the uncertainties in the sine-squared mean model. The RMSEs for the mean and zero-mean models are 25.40 and 59.11 respectively. Excluding the mean function increases errors by approximately 133%.

7.3.5 SC 25 and Long-Term Predictions

The mean prediction for SC 25 suggests that the solar maximum will occur in September 2024, with an SG-filtered peak amplitude of $A = 122^{+37}_{-28}$, and will reach its subsequent minima in February 2030. The longer-term predictions can be seen in Fig. 12, and all show a consistent shape, unlike the previous solar cycles, with very large uncertainties in the cycle amplitudes.

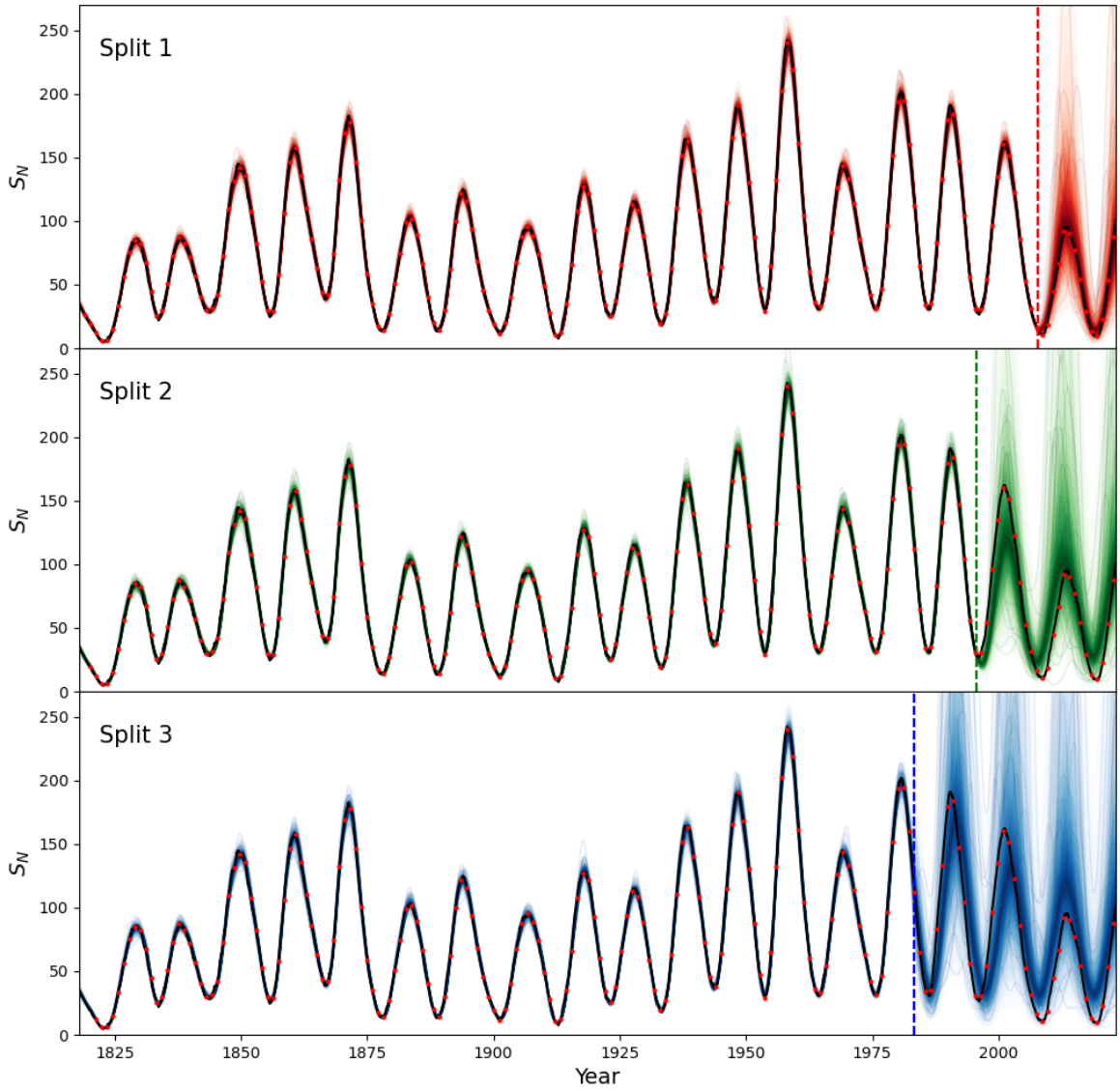

Fig. 8: Plot 1000 samples of the Matern 5/2 $\times$ Periodic model for Splits 1-3 between 1818-2023. Coloured traces represent the $\mathcal{GP}$ model. The red dots denote the training and validation data, with the dashed line indicating the training limit. The complete data is plotted in black for comparison.

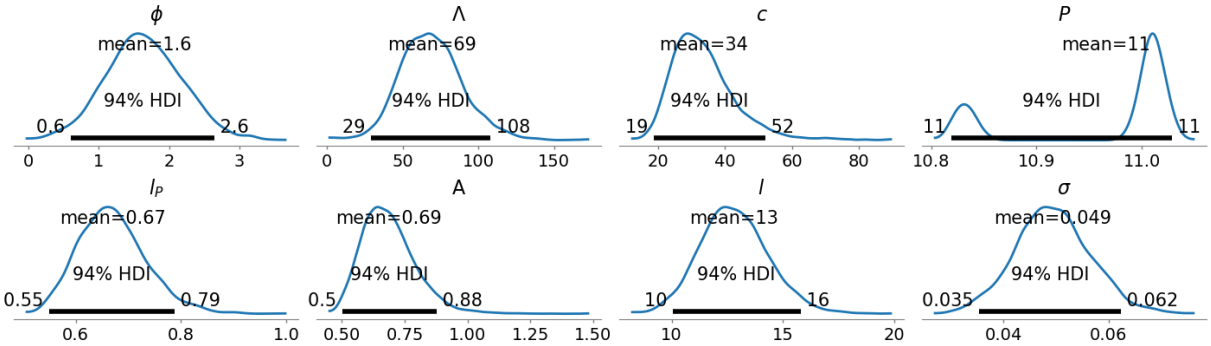

Fig. 9: Posterior distributions for the parameters used in the Matern 5/2 $\times$ Periodic kernel, as well as the mean function, for the $\mathcal{GP}$ in Fig. 8. The graphs show the 94% highest density interval (HDI).

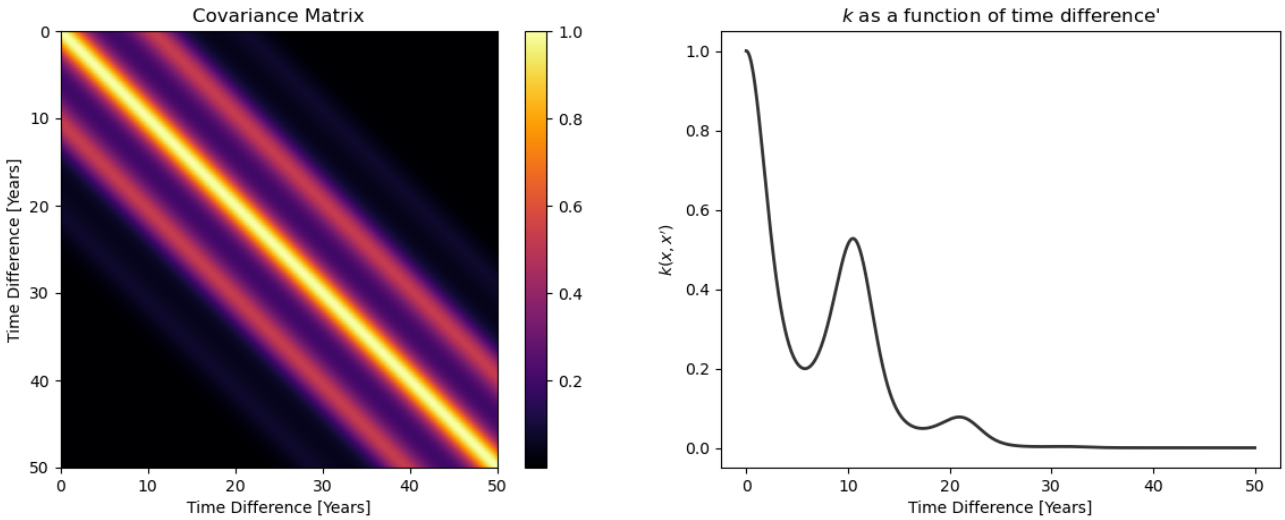

Fig. 10: (Left) Plot of the covariance matrix for the Matern 5/2 $\times$ Periodic kernel, with optimal model parameters shown in Fig 9. (Right) Plot showing how the covariance varies with the time difference between the last training point and the prediction.

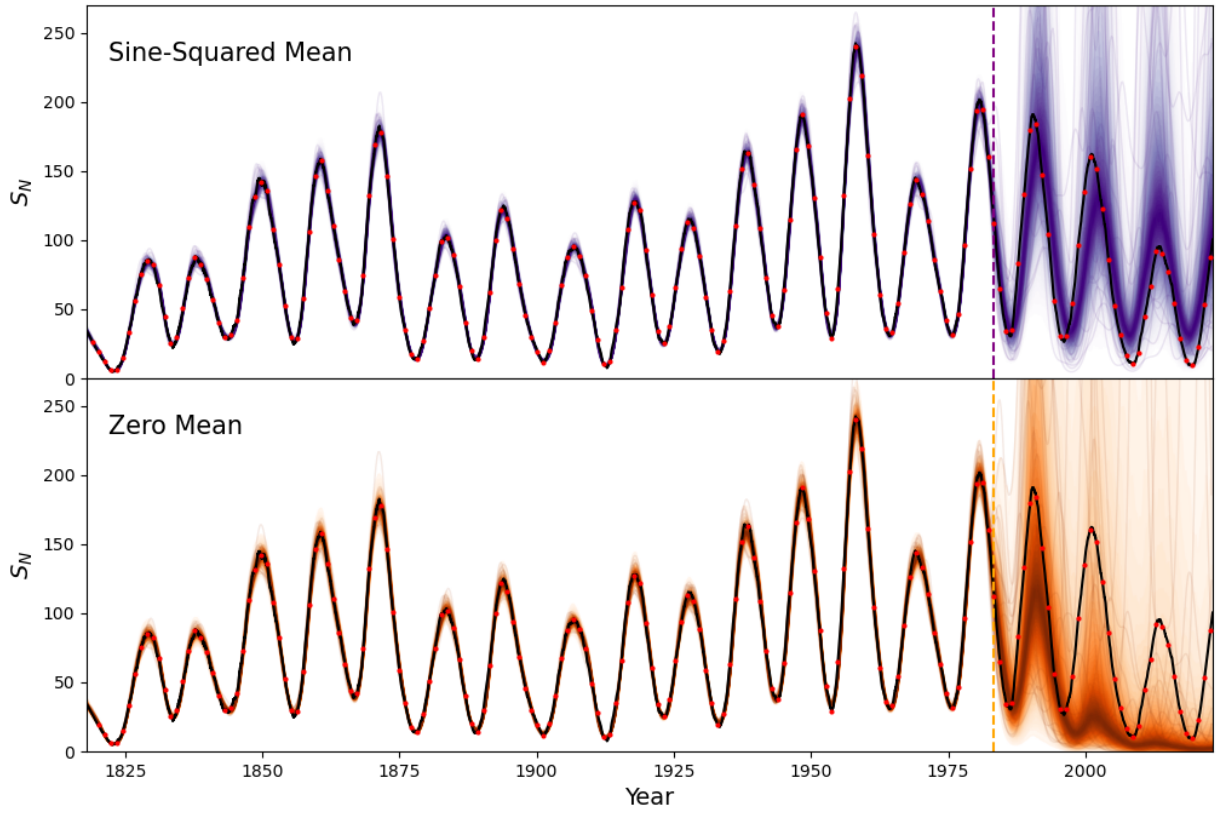

Fig. 11: Plot of the Matern 5/2 $\times$ Periodic kernel for Split 3 with a sine-squared mean function, and a zero mean function. The dashed line shows the training limit.

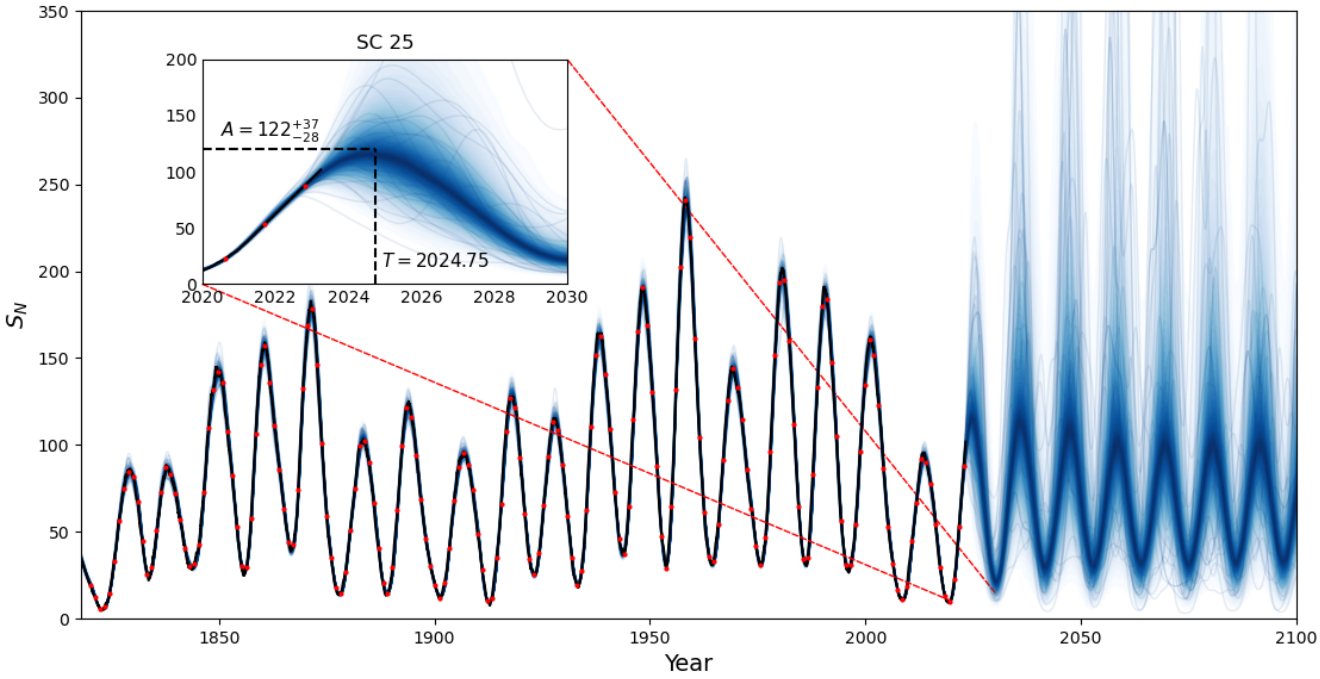

Fig. 12: $\mathcal{GP}$ predictions with the Matern 5/2 $\times$ Periodic kernel, trained between 1818 and 2023, with predictions up to 2100. SC 25 can be seen in the subplot, with the amplitude and time of maxima presented.

8

Discussion

8.1 Solar Cycle Consistency

The phase diagram in Fig. 5 illustrates how each cycle has a varying period and amplitude. The random distribution of cycle amplitudes supports the notion that solar cycles are difficult to predict due to their variability. Although no two cycles exhibit the exact same period, the similarity of the periods seems to confirm the presence of an underlying cycle, which is consistent with the widely quoted 11-year period. This reaffirms the choice to model the solar cycles using a *quasi-periodic* kernel.

The shape of the cycles when plotted in this phase diagram shows a strong similarity to a sine-squared curve. This feature was used in determining the mean function used for the $\mathcal{GP}$, which is why a sine-squared mean was chosen.

8.2 Amplitude & Descending Time relationship

The correlation coefficient $r = -0.35$ of the descending time data suggests that there is a statistically weak relationship between the cycle amplitude and descending time, and was found to be similar to the initial r value calculated by Z. Du and S. Du, 2006 ($r = -0.383$).

Each of the linear regression models calculated a different value for m, which is most likely due to how poorly the data seems to fit with this model. Both the MLE and MAP gradients suggest a negative correlation between the amplitude and descending time of a solar cycle. Whilst most of the MCMC samples also display a negative correlation, the wide distribution of the samples indicates that there is a large uncertainty in that value. This uncertainty can be seen more clearly when analysing the posterior distribution of m, which shows $m = -1.74^{+1.44}_{-1.36}$. These uncertainties suggest that there is a non-negligible probability that the optimal gradient for this data is actually zero, as indicated by the dashed line in Fig. 6.

The analysis of this relationship seems to suggest that there does not exist and linear relationship between the amplitude of a solar cycle and its descending time, and therefore cannot be used in forecasting the shapes of future solar cycles. It also highlights the capability of the probabilistic ML approach for modelling.

8.3 $\mathcal{GP}$ Predictions

8.3.1 Model Error Evaluation

The Matern 5/2 $\times$ Periodic kernel $\mathcal{GP}$ model exhibited the lowest errors overall, and can therefore be deemed the 'best' model. However, Fig. 7 shows us that the Matern 3/2 and SE kernels also had very low errors, which were very similar to that of the Matern 5/2 values. These could also be deemed suitable models for solar cycle forecasting, depending on the data.

The model which consistently performed the poorest (apart from Split 3) was the SE $\times$ Cosine kernel. This is most likely due to the lack of a periodic term in the kernel, which is present in all other models. This suggests that the use of the Periodic kernel is necessary for the task of predicting solar cycles, further supporting the notion that the solar cycles are indeed periodic in nature. Another indication of how poorly the Cosine kernel fits with the data is the fact that the Split 2 errors are larger than those for Split 3. This goes against the trend seen in the other models, which all show a more accurate forecast when trained with more data. An explanation for this peculiarity could be that when training with Split 3, the model fits so poorly that the optimal length scales converged to a short value ($l = 1.1$ and $l_P = 2.2$ years). This short length scale implies the model very quickly reverts back to the mean, which could be why the errors for Split 3 are lower than for Split 2 and the RQ + Periodic Split 3 model (discussed further in Section 8.3.3).

Split 1 was used to forecast approximately 15 years into the future, with an RMSE value of 6.67. The paper by Gonçalves, Echer, and Frigo, 2020 calculated the RMSE values for different prediction lengths, with an RMSE of $25-35$ for a 10-year forecast. Whilst the data they utilised was different (annual average sunspot numbers), the combination of our mean function, kernel choice, and parameter optimisation technique was most likely the main reason for our model's improved forecasts.

Neither the Matern 5/2 nor the Matern 3/2 kernel are typically used for solar cycle predictions. The Matern 5/2 kernel is well suited for forecasting data which is noisier than that which is best described by the SE kernel, but less noisy than the Matern 3/2 kernel. This is an interesting result, as the data used for this project was specifically made to be very smooth by the SG filter, and so the improved performance of a kernel which is better suited to noisier data suggests that others who have modelled the cycles with noisier data and the SE kernel could use this to improve their results. However, the difference in forecast accuracy between the SE and Matern kernels is quite small, so further testing with different levels of data smoothness would be required to conclusively say whether the Matern 5/2 kernel is better suited to solar cycle

modelling. If so, it could also indicate that the use of the Matern 5/2 kernel could provide improved predictions for stellar cycle predictions, which also use the standard SE $\times$ Periodic kernel (Nicholson and Aigrain, 2022).

8.3.2 Modelling and Forecasting Ability

The Split 1 RMSEs for each model (apart from the SE $\times$ Cosine) were consistently low, making short-term predictions with the $\mathcal{GP}$ seemingly possible. The plot of the Matern 5/2 $\mathcal{GP}$ predictions in Fig. 8 shows just how well the $\mathcal{GP}$ predictions agree with the data up to the training limit, with the traces showing very little uncertainty. This was also true for many of the other $\mathcal{GP}$ models, illustrating how powerful $\mathcal{GP}$s are at modelling, and why they have been chosen for solar cycle forecasting.

For each split, after the training limit, the uncertainties very quickly increased. This was expected of our $\mathcal{GP}$ models, as they rely on data to constrain them and reduce their uncertainties. While the mean predictions are closely aligned with the true values, the uncertainties suggest that the actual cycles may be significantly larger or smaller than the predicted values, although those values are much less probable than the mean predictions. The argument could be made that the shape of the solar cycles remains largely unpredictable, with the exception of the period.

The mean predictions for Split 1 are found to be in agreement with the true data, which is impressive given the limited data the model was trained on. It was also able to successfully predict the lower amplitude of SC 24, albeit with large uncertainties. Splits 2 and 3 fared slightly worse in their accuracy, with Split 2 initially failing to capture the large amplitude of SC 23; Split 3 predicted SC 23's maxima slightly late, and too high an amplitude for SC 24. However, even the RMSE of Split 3, which forecasts approximately 40 years, is less than the 10-year forecast RMSE by Gonçalves, Echer, and Frigo, 2020.

Depending on which Split we look at, the subsequent cycle is predicted to an inconsistent level of accuracy. For example, for Splits 1 and 3 the subsequent cycle is very accurately forecasted, but for Split 2 the second cycle is more accurately predicted that the first. This result shows that even though a $\mathcal{GP}$ model may be able to obtain low errors, there is no guarantee that it can perfectly predict the next cycle, and should only be treated as an estimate.

8.3.3 *The Effect of the Mean Function and Kernel*

As previously stated, when a $\mathcal{GP}$ is defined with an appropriate mean function it should be better able to model and forecast the data. This can clearly be seen with the comparison in Fig. 11, where we can see that, for the zero-mean model the predictions very quickly tend to zero. The sine-squared mean model however is able to successfully capture the structure of the future solar cycles.

The kernel parameters used for both of these $\mathcal{GP}$s were virtually identical, which begs the question: does the model only perform well because of the mean function? By analysing the posterior distributions of the kernel we notice some key properties which determine how predictions are made with this model. Firstly, the amplitude of the mean function Λ is much greater than the amplitude of the kernel A, which suggests the mean has a greater effect on the predictions. Secondly, the length scales $l = 0.68$ and $l_P = 13$ years indicate the kernel loses its ability to effectively capture correlations after a relatively short period, given a solar cycle lasts 11 years. This can further be seen in Fig. 10, where the covariance is seen to diminish completely after approximately 25 years.

This, in my opinion, is the key result of our $\mathcal{GP}$ modelling, as it indicates the lack of forecasting ability of our models for long timescales; even though the model produced quantifiably accurate results. These length scales are also found to be shorter than those found by Gonçalves, Echer, and Frigo, 2020, whose kernel doesn't tend to zero until a time difference of 100 years.

Overall, this result shows how essential the use of a suitable mean function is to a $\mathcal{GP}$ model for solar cycle predictions. Whilst the key driver of the forecasts seems to be the sine-squared mean function rather than the covariance function, it undoubtedly improves the forecasts made for the short to medium term and should be considered for use in other solar cycle models. We can conclude that the short length scales suggest that the solar cycles may be inherently unpredictable.

8.3.4 *SC 25 Predictions and Beyond*

Whilst we have established that these $\mathcal{GP}$ models are able to model solar cycles very well, and provide low RMSE forecasts for the near to medium term, we have also demonstrated how vital the role of the mean function was in making these forecasts and concluded that the short length scale of the kernel is the main determinant of how far we can sustainably predict solar cycles.

With this in mind, the mean near-term predictions for SC 25 seem to approximately coincide

with other predictions of the next solar maxima. Our mean amplitude of 122^{+37}_{-28} is very similar to the amplitude of 116 predicted by Gonçalves, Echer, and Frigo, 2020. Predictions using alternative methods have forecast a range of amplitudes, from 100 ± 15 (Camacho, Faria, and Viana, 2022). It should be noted that these predictions are not directly comparable due to the differences in training data, which gives an unfair advantage to models which have trained using the latest data (such as in this project).

However, when we look to forecast cycles beyond SC 25 it is clear that these predictions are unlikely to pan out, as the forecasts seem to revert to the mean function and remain uniform; where we would expect the cycles to once again display some form of varying amplitude. This implies the covariance has a negligible effect on the prediction outcomes, and consequently, the variation in the amplitude cannot be predicted. Therefore we must conclude that solar cycles may be forecasted over a short window of up to 20 years, with a great degree of uncertainty, but forecasting further than that relies too heavily on the mean function.

8.4 Future Developments

Probabilistic ML serves as an especially useful tool for solar cycle predictions, with $\mathcal{GP}$s showing the most promise at successfully predicting solar cycles. Whilst the predictions remain far from perfect, it's highly likely that any improvements in these predictions would come from another $\mathcal{GP}$ model, with a slightly different architecture.

A future modification of the $\mathcal{GP}$ model may include the use of a non-stationary kernel, to investigate whether there is a fundamentally varying length scale, which could explain why the optimal length scales were found to be so short. For this project it was assumed that the driving processes of the solar cycles happen over long timescales, thus over a few hundred years it could be considered a stationary process, however, this could be investigated thoroughly. This would involve creating a suitable function for a length scale which evolves with time, which itself would need to be optimised in a similar fashion to the rest of the $\mathcal{GP}$ model.

Another improvement could come in the form of a multi-input $\mathcal{GP}$, which takes various data inputs and uses them to create improved predictions, with less uncertainty. An ideal choice of data would be the sunspot area and the Sun's radio flux data, both of which vary due to the solar activity cycles. These data sets may be found to have an underlying pattern in relation to S_N which a $\mathcal{GP}$ could determine and use to predict future cycles.

The final refinement could come in the form of a deep $\mathcal{GP}$, which is a type of hierarchical

Bayesian model that can model the relationships between two variables through a series of hidden layers, similar to neural networks. These deep $\mathcal{GP}$s may be able to capture more complex relationships in the data, leading to more accurate predictions, whilst still providing a measure of the prediction uncertainties.

These improvements could lead to incrementally better predictions, in the form of reduced prediction uncertainty, or longer-term forecast capabilities.

9

Conclusion

Throughout this project, the aim has been to assess the use of various probabilistic machine learning techniques, with a focus on Gaussian processes, to predict the shape of future solar cycles so as to inform preventative measures against the harmful effects of extreme space-weather events.

A preliminary analysis of the SG-smoothed sunspot numbers revealed the cycles to have an inconsistent period, in the region of 11 years, as shown by the phase diagram in Fig. 5. This period was also found to be approximately 11 years from the mean of the posterior distribution of P in the $\mathcal{GP}$ model, however, this too suggested there could be multiple optimal periods.

We can conclude that there is unlikely to be a linear relationship between the solar cycle amplitude and descending time, supported by both a correlation coefficient $r = -0.35$, and an optimised gradient $m = -1.74^{+1.44}_{-1.36}$, which suggests the probability of a significant correlation is low. Thus the descending time of a cycle cannot be used to make accurate cycle predictions.

Most $\mathcal{GP}$ models were able to model the data extremely well, with the Matern 5/2 $\times$ Periodic kernel combined with a sine-squared mean function found to produce the best forecasts. The forecast errors were found to be much lower than other work, with a 15-year prediction window displaying an RMSE of 6.67, and a 40-year RMSE of 25.40. These low errors suggest that forecasts within these time windows are possible, but the further we wish to forecast, the larger our errors inevitably become.

A key limitation of our model was its heavy dependence on the mean function, displaying a 133% increase in errors without the presence of the sine-squared mean. This was thought to be a result of the relatively short length scales of the kernel, which were optimised to be $l = 0.68$ and $l_P = 13$ years. These values suggest that, whilst we may be able to obtain acceptable predictions for the near term, our $\mathcal{GP}$ model cannot reliably forecast further than two cycles into the future, as shown in Fig. 12, suggesting that solar cycles may be inherently unpredictable; however further model improvements may lead to slightly better forecasts.

SC 25 was predicted to occur in September 2024, with a peak sunspot number amplitude of $A = 122^{+37}_{-28}$, with its minima occurring in February 2030. Future predictions were deemed unsuitable due to the lack of amplitude variation.

References

Balogh, A et al. (2015). “Introduction to the solar activity cycle: Overview of causes and consequences”. In: *The Solar Activity Cycle*, pp. 1–15.

Berger, James (2006). “The case for objective Bayesian analysis”. In.

Bloom, JS et al. (2012). “Automating discovery and classification of transients and variable stars in the synoptic survey era”. In: *Publications of the Astronomical Society of the Pacific* 124.921, p. 1175.

Camacho, JD, JP Faria, and PTP Viana (2022). “Modelling stellar activity with Gaussian process regression networks”. In: *arXiv preprint arXiv:2205.06627*.

Charbonneau, Paul (2020). “Dynamo models of the solar cycle”. In: *Living Reviews in Solar Physics* 17.1, pp. 1–104.

Davidson, Peter Alan (2002). *An introduction to magnetohydrodynamics*.

Du, Zhanle and Shouyu Du (2006). “The relationship between the amplitude and descending time of a solar activity cycle”. In: *Solar Physics* 238.2, pp. 431–437.

Duvenaud, David (2014). “Automatic model construction with Gaussian processes”. PhD thesis. University of Cambridge.

Feminella, Francesco and Marisa Storini (1997). “Large-scale dynamical phenomena during solar activity cycles.” In: *Astronomy and Astrophysics* 322, pp. 311–319.

Foreman-Mackey, Daniel et al. (2013). “emcee: the MCMC hammer”. In: *Publications of the Astronomical Society of the Pacific* 125.925, p. 306.

Gelman, Andrew et al. (1995). *Bayesian data analysis*. Chapman and Hall/CRC.

Gonçalves, Ítalo G, Ezequiel Echer, and Everton Frigo (2020). “Sunspot cycle prediction using warped Gaussian process regression”. In: *Advances in Space Research* 65.1, pp. 677–683.

Hargreaves, John Keith (1992). *The solar-terrestrial environment: an introduction to geospace-the science of the terrestrial upper atmosphere, ionosphere, and magnetosphere*. Cambridge university press.

Hathaway, David H (2015). “The solar cycle”. In: *Living reviews in solar physics* 12.1, pp. 1–87.

Hathaway, David H and Lisa A Upton (2016). “Predicting the amplitude and hemispheric asymmetry of solar cycle 25 with surface flux transport”. In: *Journal of Geophysical Research: Space Physics* 121.11, pp. 10–744.

Hogg, David W, Jo Bovy, and Dustin Lang (2010). “Data analysis recipes: Fitting a model to data”. In: *arXiv preprint arXiv:1008.4686*.

Jones, Eric, Travis E. Oliphant, and Pearu Peterson (2007). "SciPy: Open source scientific tools for Python". In: *Computing in Science & Engineering* 9.3, pp. 10–20. DOI: 10.1109/MCSE.2007.58.

Kalnay, Eugenia et al. (1996). "The NCEP/NCAR 40-year reanalysis project". In: *Bulletin of the American meteorological Society* 77.3, pp. 437–472.

Kitiashvili, Irina N (2016). "Data assimilation approach for forecast of solar activity cycles". In: *The Astrophysical Journal* 831.1, p. 15.

Lee, Jaehoon et al. (2018). "Deep neural networks as gaussian processes". In: *arXiv preprint arXiv:1711.00165*.

Nicholson, Belinda A and Suzanne Aigrain (2022). "Quasi-periodic Gaussian processes for stellar activity: From physical to kernel parameters". In: *Monthly Notices of the Royal Astronomical Society* 515.4, pp. 5251–5266.

Pesnell, W Dean (2012). "Solar cycle predictions (invited review)". In: *Solar Physics* 281.1, pp. 507–532.

Prasad, Amrita et al. (2022). "Prediction of solar cycle 25 using deep learning based long short-term memory forecasting technique". In: *Advances in Space Research* 69.1, pp. 798–813.

Press, William H and Saul A Teukolsky (1990). "Savitzky-Golay smoothing filters". In: *Computers in Physics* 4.6, pp. 669–672.

Roberts, Gareth O and Jeffrey S Rosenthal (2009). "Examples of adaptive MCMC". In: *Journal of computational and graphical statistics* 18.2, pp. 349–367.

Salvatier, John, Thomas V. Wiecki, and Christopher Fonnesbeck (2016). "Probabilistic programming in Python using PyMC3". In: *PeerJ Computer Science* 2, e55. DOI: 10.7717/peerj-cs.55.

Schrijver, Carolus J and George L Siscoe (2009). *Heliophysics: Plasma physics of the local cosmos*. Cambridge University Press.

SILSO World Data Center (2023). "The International Sunspot Number". In: *International Sunspot Number Monthly Bulletin and online catalogue*.

Siscoe, George (2000). "The space-weather enterprise: past, present, and future". In: *Journal of Atmospheric and Solar-Terrestrial Physics* 62.14, pp. 1223–1232.

Wang, Jingxiu and Jie Jiang (2014). "Magnetohydrodynamic process in solar activity". In: *Theoretical and Applied Mechanics Letters* 4.5, p. 052001.

Williams, Christopher KI and Carl Edward Rasmussen (2006). *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA.

A

Mathematical Background

A.1 Savitzky-Golay Filter

The Savitzky-Golay (SG) filter is a signal-processing algorithm used to smooth and differentiate time-series data. The filter uses a weighted moving average to smooth the data, with weights that are chosen to minimise the least-squares error in fitting a polynomial to the data. The filter is particularly useful for removing high-frequency noise from data while preserving the underlying trends and features (Press and Teukolsky, 1990).

For a given window size W and polynomial order k, the filter fits a polynomial of degree k to the data within each window. The polynomial is chosen to minimise the sum of squared errors between the data and the fitted polynomial:

$$\min_{c_0,c_1,\ldots,c_k} \sum_{j=i-\frac{W-1}{2}}^{i+\frac{W-1}{2}} \left(x_j - \sum_{n=0}^{k} c_n j^n \right)^2 \tag{A.1}$$

where $c_0, c_1, ..., c_k$ are the coefficients of the polynomial, and j is the index of the data point within the window.

The solution to this least-squares problem can be expressed in matrix form as:

$$\mathbf{c} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{x} \tag{A.2}$$

where $\mathbf{c}$ is the vector of polynomial coefficients, $\mathbf{x}$ is the vector of data within the window, and $\mathbf{X}$ is the Vandermonde matrix:

$$\mathbf{X} = \begin{bmatrix} 1 & j_0 & j_0^2 & \ldots & j_0^k & 1 & j_1 & j_1^2 & \ldots & j_1^k \\ \vdots & \ddots & \vdots & 1 & j_{W-1} & j_{W-1}^2 & \ldots & j_{W-1}^k & & \end{bmatrix} \tag{A.3}$$

The coefficient $\mathbf{c}$ can be calculated for a given window size W and polynomial order k, and used to calculate the smoothed value y_i at each point i using the following equation (Press and Teukolsky, 1990):

$$y_i = \sum_{j=i-\frac{W-1}{2}}^{i+\frac{W-1}{2}} c_{j-i+\frac{W-1}{2}} x_j \tag{A.4}$$

where $c_{j-i+\frac{W-1}{2}}$ is the weight corresponding to the distance between points i and j within the window.

In general, a larger window size will provide better smoothing of the data, but may also introduce more lag and distortion. Similarly, a higher polynomial order will provide better fitting of the data, but may also introduce more noise and over-fitting. The implementation of this SG filter can be seen here

A.2 Gradient Descent Algorithm

One process of minimising the loss, as necessary for minimising the log-likelihood, is to use a gradient descent algorithm. This is a first-order iterative optimisation algorithm for finding the local minimum of a differentiable function. It's a generative model, and works by minimising the loss function, which for a linear regression example is given by:

$$\frac{1}{N} \sum_{i=0}^{N} \left(y_i - (mx_i + b)\right)^2 \tag{A.5}$$

where N is the number of points in our data set, m is our current gradient guess, and b is our current intercept guess. Note: there is an analytical solution to this particular problem, but an algorithm can be used to calculate it.

To minimise this function the partial derivatives must be made as small as possible, which, with respect to m, is given as:

$$-\frac{2}{N} \sum_{i=0}^{N} x_i \left(y_i - (mx_i + b)\right) \tag{A.6}$$

In order to make m converge on the optimal values, one first needs to calculate what the gradient of m currently is at the current point, before iteratively moving by a scale known as the *learning rate*, which must be small enough such that it can converge, but not too small that it takes too long to do so. This example is demonstrated in this notebook

A.3 Prior Distributions

The probability density function (PDF) of a continuous uniform distribution $\mathcal{U}$ is defined as:

$$p(\theta) = \begin{cases} \frac{1}{b-a}, & a \leq \theta \leq b \\ 0, & \text{otherwise} \end{cases} \tag{A.7}$$

where a and b are the lower and upper bounds of the distribution, respectively. For a continuous Gaussian distribution $\mathcal{N}$ it is defined as:

$$p(\theta) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\theta-\mu)^2}{2\sigma^2}}, \tag{A.8}$$

where μ and σ are the mean and standard deviation of the distribution respectively.

A.4 Evaluation Metrics

To quantify the agreement between our full-cycle forecasts and the true data two metrics were calculated, the root mean squared error (RMSE) and the mean absolute error (MAE). The mean-squared error (MSE) measures the average squared distance between the actual data point and the predicted value and can be used to compare how close each model is to the true values. The RMSE is the square root of the MSE, and more heavily penalises outliers. The RMSE is given by:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}, \tag{A.9}$$

where n is the number of data points, y_i is the actual value, and $\hat{y}_i$ is the predicted value. The MAE takes an average of the absolute errors, and is given by:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{A.10}$$

The mean of the $\mathcal{GP}$ samples was taken to be the 'predicted value'. A bar chart of RMSE and MAE was plotted, with the lowest errors indicating the best model for forecasting.

The Pearson correlation coefficient r for the investigation in Section 6.3 was calculated using the equation:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}, \tag{A.11}$$

where x and y are the corresponding data values, and $\bar{x}$ and $\bar{y}$ are the mean values, and $-1 \leq r \leq 1$. $|r| \approx 1$ indicates a strong correlation, with the sign specifying whether the correlation is positive or negative. $|r| \approx 0$ indicates no statistically significant correlation.

A.5 Kernel Cookbook

Below is a list of some commonly used kernels, all of which were used when experimenting with various kernel combinations. The scaling factor of σ_f^2 is omitted from each equation (Duvenaud, 2014). The kernels are given in the following order: Squared Exponential, Matern ν, Rational Quadratic, Periodic, and Cosine

$$\mathrm{SE}(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2l^2}\right) \tag{A.12}$$

$$\mathrm{M}_\nu(x, x') = \left(\frac{\sqrt{2\nu}}{l}\|x - x'\|\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}}{l}\|x - x'\|\right) \tag{A.13}$$

$$\mathrm{RQ}(x, x') = \left(1 + \frac{\|x - x'\|^2}{2\alpha l^2}\right)^{-\alpha} \tag{A.14}$$

$$\mathrm{Per}(x, x') = \exp\left(-\frac{2\sin^2(\pi\|x - x'\|/p)}{l_P^2}\right) \tag{A.15}$$

$$\mathrm{Cos}(x, x') = \cos\left(\frac{|x - x'|}{l_P}\right) \tag{A.16}$$

B

Supplementary Tables

B.1 Solar Cycle Period

Below are the tables of solar cycle period, calculated by finding the time difference between successive maxima or minima. The successive maxima data has not been confirmed for Cycle 24, as Cycle 25 is still underway.

Table B.1: Solar cycle periods (Maxima to Maxima) using SG signals

Cycle	**Period [Years]**		
	Polyorder 1	Polyorder 2	Polyorder 3
7	8.485	7.586	8.428
8	11.581	11.573	11.438
9	11.234	11.279	10.951
10	10.709	10.817	10.635
11	12.233	12.907	13.493
12	10.432	9.756	9.206
13	12.773	12.531	12.348
14	11.466	11.773	11.732
15	9.419	9.425	10.438
16	10.203	10.709	9.720
17	10.601	9.685	10.020
18	9.667	10.129	10.394
19	10.880	11.458	11.863
20	11.666	11.091	9.973
21	9.739	9.988	9.709
22	10.710	10.710	12.496
23	12.411	12.843	12.135

Table B.2: Solar cycle periods (Minima to Minima) using SG signals

Cycle	**Period [Years]**		
	Polyorder 1	Polyorder 2	Polyorder 3
7	10.962	10.446	9.781
8	10.049	9.288	10.254
9	11.945	12.838	12.438
10	10.951	10.817	10.943
11	11.367	11.731	11.745
12	10.803	10.781	10.921
13	12.306	12.392	11.770
14	11.730	11.647	12.230
15	10.435	9.860	9.216
16	10.057	10.427	10.899
17	10.400	10.238	10.344
18	10.077	10.264	10.141
19	10.561	10.791	10.348
20	11.330	11.150	11.134
21	10.715	9.735	9.808
22	9.777	10.670	10.673
23	12.333	12.095	12.555
24	10.764	10.742	10.849

B.2 Amplitude vs Descending Time

Table B.3: Amplitudes and Descending times three cycles earlier for SG signal polyorder = 1.

Cycle	**Amplitude [S_N]**	**Descending Time (-3 cycles) [Months]**
10	158.819	4.368
11	182.777	5.932
12	103.223	6.296
13	125.100	6.013
14	96.103	6.671
15	127.800	5.241
16	115.426	7.115
17	164.282	6.072
18	191.116	5.041
19	242.520	5.679
20	145.073	5.876
21	201.899	5.352
22	191.124	6.246
23	162.454	6.696
24	95.624	5.745