Bachelor thesis
Computing Science

Radboud University

Gaussian Processes versus Autoregressive Wild Bootstrap: Climate Application

Author:
Jaap Dijkstra
s4793048

First supervisor/assessor:
Professor T.M. Heskes
tom.heskes@ru.nl

Second supervisor/assessor:
Dr. Yuliya Shapovalova
y.shapovalova@cs.ru.nl

January 17, 2020

Abstract

In this thesis we compare two nonparametric regression methods that construct confidence bands around a function. The ability of the two methods to estimate future values of a time series is compared in a simulation study when given data containing autocorrelation, heteroscedasticity, and missing data. We found that the autoregressive wild bootstrap (AWB) method outperforms the Gaussian process regression (GPR) method when data contains both autocorrelation and heteroscedasticiy, and no portions of the data were missing. The GPR method either performs equally well or outperforms the AWB method when data contains autocorrelation, large portions of missing data, and no heteroscedasticity. Additionally, we use the GPR method on a real life ethane emissions time series in which large portions of the data are missing.

Contents

Chapter 1

Introduction

Regression is a statistical method that uses data to investigate the relation between independent variables and a dependent variable. Additionally, it can be used as a prediction tool that estimates the value of the dependent variable for future values of the independent variable. When the relationships between variables can not be expressed by a linear function, linear regression will not give us an accurate depiction of the system. Instead of assuming a certain shape, nonparametric regression attempts to learn the entire function without assuming what shape it is.

In this thesis we are mainly interested in the application of nonparametric regression to time series: a series of observations indexed by time. Time series are used for example in climatology, economics, signal processing, and pattern recognition. Time series data comes with some inherent challenges: autocorrelation refers to the phenomenon where error terms at some time point are influenced by the error terms of previous time points. Furthermore, data shows heteroscedasticity when the degree of variance in the data changes over time. Lastly, portions of the data might be missing due to recording device failures. In the case of weather observations for example, portions of the data are often missing due to cloud coverage.

Friedrich et al. (2020) proposed the autoregressive wild bootstrap (AWB) method, which constructs confidence bands around a function. Confidence bands represent a region of functions in which it is believed the actual function exists with some level of confidence. They tested the performance of the method on data containing autocorrelation, heteroscedasticity, and data sets where large portions of data were missing.

An alternative Bayesian approach to nonparametric regression is Gaussian process regression (GPR). This approach has been used for example by Kaufman et al. (2010) and Murray-Smith and Girard (2001). In this paper, we study how well this method handles autocorrelation, heteroscedasticity, and missing data in the context of climate applications. We do a simulation study in which we will apply both the AWB and GPR method to time series containing autocorrelation, heteroscedasticity, and missing data. We will construct confidence bands that extend past the existing observations and compare the ability of the two methods to accurately estimate future values of the function. Additionally, we will apply the GPR method to a real life ethane emissions data set in which 70% of the data is missing.

Chapter 2 describes the AWB method and the GPR method. In Chapter 3 we do a simulation study where we compare the two methods. In Chapter 4 we apply the GPR method to a real life climate time series, and Chapter 5 concludes.

Chapter 2

Methods

In Section 2.1 we discuss the autoregressive wild bootstrap (AWB) method proposed by Friedrich et al. (2020). Section 2.2 discusses Gaussian Process Regression (GPR). Several other nonparametric regression methods exist, which we do not consider in this thesis: decision tree learning can be used to create regression trees. A method based on this is investigated in Chaudhuri et al. (2002). Observations about an object defined on the branches of the tree lead to the object's target value in the form of a real number, defined on the leaves of the tree. Furthermore, local polynomial regression creates a model by fitting multiple smaller models to subsets of the data, building up a larger function. The basic method is discussed in Fan et al. (1993), and a version considering autocorrelation in Xiao et al. (2003).

2.1 Autoregressive wild bootstrap

Friedrich et al. (2020) proposed the autoregressive wild bootstrap (AWB) method that constructs confidence bands around a function. In Section 2.1.1 we explain the basic bootstrapping technique and discuss confidence bands. Section 2.1.2 describes how the AWB method addresses autocorrelation and heteroscedasticity. Finally, Section 2.1.3 describes the construction of confidence bands and how it handles missing data.

2.1.1 Bootstrapping and confidence bands

The foundation of the AWB method is the bootstrapping technique. It is a resampling method that relies on sampling with replacement. As an example, assume we only have a single sample available. Values from this sample are drawn at random, possibly more than once, and are put in a new sample called a bootstrap sample. The bootstrap sample has the same size as the original sample, but is very likely to be different. Some values might have been picked more than once, while others have not been picked at all. This process is typically repeated a large number of times, resulting in a collection of many bootstrap samples. The method is used in statistics to determine the value of some parameter of a population, such as its variance or prediction error. It is also often used to create error bars, which are bar charts indicating the error of

some measurement.

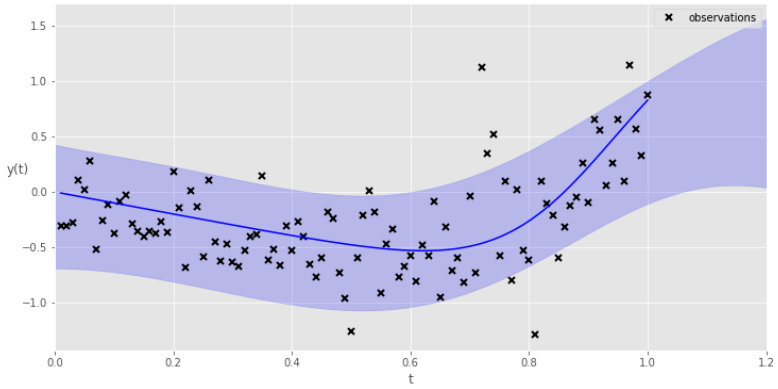
In cases where we can not simply draw multiple samples from the population, the one sample is all the information we have to learn from. Here bootstrapping is particularly useful. An example situation is the measurement of temperature in a particular location over time. We can not turn back time and measure the temperatures again to draw another, different sample.

When estimating any statistic, arguably the belief we have in the accuracy of the estimation is of utmost interest. Confidence intervals provide a measure of certainty regarding the statistic we are estimating. For instance, when estimating a single real-valued number, a confidence interval describes the range in which we believe this number to exist with some level of confidence. In the case of functions, confidence bands are used to express our beliefs about the function. Figure 2.1 shows an example confidence band for the following system:

$$y_t = m(t) + \epsilon(t)$$

The black marks are the noisy observations y_t and the solid blue line is the underlying function $m(t)$. The blue shaded area is the confidence band. It represents the space of possible functions in which it is believed, with some level of confidence, the underlying function $m(t)$ exists. As the confidence band is extended to future time points (past $t = 1.0$), it gradually grows wider, as it becomes less confident in its prediction.

Figure 2.1: Confidence band



2.1.2 Autocorrelation and heteroscedasticity

The AWB method aims to construct confidence bands around a function describing time series data. The method addresses two inherent challenges for this type of data, namely autocorrelation and heteroscedasticity. With autocorrelation, the error term at a certain time point is influenced by the error terms of previous time points. When the degree of variance in the data changes over time, this is called heteroscedasticity. Many statistical tests assume that there is no autocorrelation and heteroscedasticity in the data. For example,

the ordinary least squares (OLS) method used in linear regression assumes the errors are independent and identically distributed over the entire time period. As a result, these tests become invalid when data contains autocorrelation or heteroscedasticity.

As stated in Section 2.1.1, bootstrapping involves the creation of many bootstrap samples that are derived from the original sample. In the context of nonparametric regression of time series data, the original sample $\mathbf{y}$ refers to the vector of observations indexed by time, i.e. $\mathbf{y} = [(t_1, y_1), (t_2, y_2), ..., (t_n, y_n)]$. Throughout this thesis we will use bold notation to refer to a vector, e.g. $\mathbf{y}$ is the vector of observations. The bootstrapping technique creates a new vector called a bootstrap sample $\mathbf{y}^*$ using observations $\mathbf{y}$ as follows, for each time point t:

$$y_t^* = \hat{m}\left(\frac{t}{n}\right) + \xi_t^*,$$

where ξ_t^* is an independent, identically distributed (i.i.d) bootstrap error sampled from a normal distribution with mean 0 and variance 1, denoted as $\xi_t^* \sim \mathcal{N}(0, 1)$. The bootstrap error can be considered as the noise in the bootstrap sample. Our bootstrap sample $\mathbf{y}^*$ will thus look as follows:

$$\mathbf{y}^* = [(t_1, y_1^*), (t_2, y_2^*), ..., (t_n, y_n^*)]$$

The term $\hat{m}\left(\frac{t}{n}\right)$ is the Nadaraya (1964), Watson (1964) estimator. With nonparametric regression we are trying to estimate the underlying function $m(t)$ of a noisy function indexed by time: $y_t = m(t) + \epsilon(t)$, where $m(t)$ is not restricted to linear functions. The Nadaraya-Watson estimator estimates the value $m(t)$ for each time point t by looking at surrounding observations $\mathbf{y}$. We will denote this vector of estimates for all time points as $\hat{\mathbf{m}}$. This vector is often referred to as an estimator. For time point t, $\hat{m}(\frac{t}{n})$ is defined as:

$$\hat{m}(\tau) = \frac{\sum_{t=1}^{n} K\left(\frac{t/n-\tau}{h}\right) y_t}{\sum_{t=1}^{n} K\left(\frac{t/n-\tau}{h}\right)}, \qquad \tau = \frac{t}{n} \in (0, 1] \tag{1}$$

where h is the bandwidth parameter, and K is some kernel function. Both need to be chosen and kept the same for the estimation of all time points. The bandwidth is a real number that defines how far along the x-axis from τ we should take points into account for the estimation. The implementation of the kernel defines how the weights that are put on the values of the observations are distributed. The *uniform* kernel for example places equal weight on all observations in the interval, and estimates the value $\hat{m}(\tau)$ simply as the average of the y_t values that reside in the specified interval. Using the uniform kernel, $\hat{m}(\tau)$ becomes:

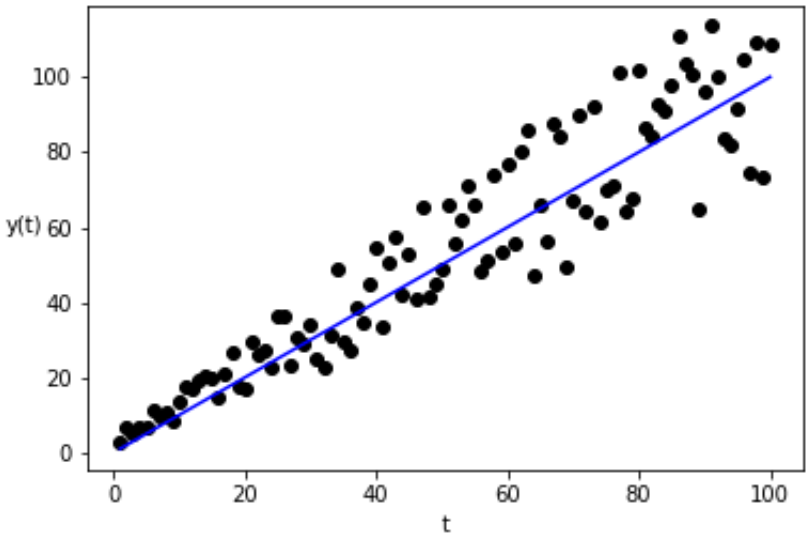
$$\hat{m}(\tau) = \frac{\sum_{i=1}^{n} \mathbf{1}(|t_i - \tau \cdot n| \leq h) y_t}{\sum_{i=1}^{n} \mathbf{1}(|t_i - \tau \cdot n| \leq h)}, \qquad \tau \in (0, 1)$$

The numerator sums up the y_t values that correspond to the time points within the interval. The denominator counts the number of observations that exist in the interval. The resulting estimate $\hat{m}(\tau)$ is the average of the values around time point τ. Other kernels, such as the Epanechnikov kernel chosen by Friedrich

et al. (2020), place decreasing weight on y_t values as we move further away from τ.

So far we have seen how the regular bootstrapping technique creates bootstrap samples in the context of nonparametric regression. The wild bootstrap, proposed by Wu et al. (1986), is an extension to the regular bootstrapping technique that is suited especially well when dealing with heteroscedastic data. When data is heteroscedastic, the conditional variance can change over time. That is, the noise in the data is dependent on the time point t. This results in points being spread out differently along the y-axis over time. An example of heteroscedastic data is shown in Figure 2.2.

Figure 2.2: Heteroscedastic data



When there is more variance at a certain time point t, there is a higher chance that the actual value y_t is further away from the estimation $\hat{m}(t)$. To compensate for this difference in estimation accuracy between low variance and high variance, the wild bootstrap finds the residual $\hat{z}_t = y_t - \hat{m}(t)$ at each time point and multiplies it with the generated bootstrap error ξ_t^*. The wild bootstrap thus creates a bootstrap sample $\mathbf{y}^*$ as follows, for each timepoint t:

$$y_t^* = \hat{m}\left(\frac{t}{n}\right) + \hat{z}_t * \xi_t^*,$$

where $\hat{z}_t$ is the residual, and ξ_t^* is the generated bootstrap error.

Note however that the regular bootstrap and wild bootstrap generate the bootstrap errors $\boldsymbol{\xi}^*$ as i.i.d random variables. Consequently, any dependence between different timepoints in the data is inherently destroyed. Instead, to account for autocorrelation in time series data, the autoregressive wild bootstrap method generates the bootstrap errors as follows:

$$\xi_t^* = \gamma * \xi_{t-1}^* + v_t^*, \tag{2}$$

where $\xi_0^* \sim \mathcal{N}(0,1)$ and $v_t^* \sim \mathcal{N}(0,(1-\gamma^2))$.

From the term $\gamma * \xi_{t-1}^*$ in (2) we can see that the bootstrap errors are constructed in such a way that their value in part consists of the value of the bootstrap error at the previous time point, and implicitly also all other previous time points. Here γ is the autoregressive parameter of the AWB method, which is a number between zero and one.

The term v_t^* is sampled from a normal distribution with mean zero and variance $(1-\gamma^2)$. If γ is set higher, generally we would expect there to be more autocorrelation, which means past values have a higher influence. By setting the variance of the normal distribution of v_t^* to $(1-\gamma^2)$, the value of v_t^* will be smaller (positive or negative) for higher γ. As a result, the balance between the influence of the two terms in equation (2) is shifted more towards the autoregressive term $\gamma * \xi_{t-1}$ when γ is set higher.

The AWB method thus handles heteroscedasticity by multiplying the bootstrap error at each timepoint with the residual, and handles autocorrelation by constructing the bootstrap errors in such a way that they are dependent on the bootstrap errors of previous time points.

2.1.3 AWB confidence bands

Next we will discuss how the autoregressive wild bootstrap constructs confidence bands around a function, and how it handles missing data in the process.

In the AWB method the original Nadaraya-Watson estimator described in equation (1) is adjusted to account for missing data. The process **D** is introduced, which equals 1 if value y_t has been observed, and returns 0 if it is missing:

$$D_t = \begin{cases} 1 & \text{if } y_t \text{ is observed} \\ 0 & \text{if } y_t \text{ is missing} \end{cases}$$

With the addition of D_t, $\hat{m}(\tau)$ becomes:

$$\hat{m}(\tau) = \frac{\sum_{t=1}^{n} K\left(\frac{t/n-\tau}{h}\right) D_t y_t}{\sum_{t=1}^{n} K\left(\frac{t/n-\tau}{h}\right) D_t} \tag{3}$$

We can see that both the numerator and denominator now only take into account the time points for which an observation was made.

The method starts off by creating estimator $\tilde{\mathbf{m}}$ using the Nadaraya-Watson estimator in equation (3) with observations $\mathbf{y}$ and bandwidth $\tilde{h} = 2h^{\frac{5}{9}}$ Bühlmann et al. (1998). Next, the residuals $\hat{\mathbf{z}}$ are obtained by computing the difference between observations $\mathbf{y}$ and the estimator $\tilde{\mathbf{m}}$. For each time point t:

$$\hat{z}_t = y_t - \tilde{m}\left(\frac{t}{n}\right)$$

A bootstrap sample $\mathbf{y}^*$ is then constructed as follows: for each time point t,

$$y_t^* = D_t * \left[\tilde{m}\left(\frac{t}{n}\right) + \xi_t^* * \hat{z}_t\right]$$

The process $\mathbf{D}$ causes the bootstrap value y_t^* to be 0 if there was no observed value for time point t. This means the structure of the original sample, including the missing data structure, is maintained when creating the bootstrap samples. While the values for missing observations will be zero in the bootstrap sample, these will not be taken into account in the estimation due to the implementation of the Nadaraya-Watson estimator in equation (3).

Now that we have a bootstrap sample $\mathbf{y}^*$, we find bootstrap estimator $\hat{\mathbf{m}}^*$ using equation (3) again at each time point, this time with the bootstrap sample $\mathbf{y}^*$ instead of observations $\mathbf{y}$. We now have an estimator $\hat{\mathbf{m}}^*$ that describes this particular bootstrap sample. This process is typically repeated many times, and we end up with a collection of many bootstrap estimators each describing a different bootstrap sample. Friedrich et al. (2020) suggest to create $B = 999$ bootstrap samples and estimators. These estimators together provide us with the tools we need to create the confidence bands.

In the construction of the confidence bands, a statistic of interest is the residual value of each of the bootstrap estimators. The residual value of a bootstrap estimator $\hat{\mathbf{m}}^*$ for a certain time point τ is $\hat{m}^*(\tau) - \tilde{m}(\tau)$. We now have a single estimator $\tilde{\mathbf{m}}$, and 999 bootstrap estimators $\hat{\mathbf{m}}^*$, each describing one of 999 bootstrap samples. Using estimator $\tilde{m}(\tau)$ and our collection of 999 bootstrap estimators, we can find α-quantiles of this statistic.

Quantiles are normally used as cut points in a distribution that divides the space into intervals with equal probabilities. In this case, an α-quantile defines a cut point such that $\alpha\%$ of bootstrap residuals fall below this cut point, and (1 - $\alpha\%$) fall above this cut point.

Let $\tilde{m}(\tau)$ be the estimator used with $\mathbf{y}$, using bandwidth $\tilde{h}$. The α-quantile of the bootstrap statistic $\hat{m}^*(\tau) - \tilde{m}(\tau)$ for a time point τ is defined as:

$$\hat{q}_\alpha(\tau) = \inf\{u \in \mathbb{R} : \mathbb{P}^*[\hat{m}^*(\tau) - \tilde{m}(\tau) \leq u] \geq \alpha\}, \tag{6}$$

which means $\hat{q}_a(\tau)$ is the smallest real number u, such that the fraction of bootstrap residuals $\hat{m}^*(\tau) - \tilde{m}(\tau)$ that is smaller than or equal to u, is larger than α. In practice, $\hat{q}_a(\tau)$ is the residual of some bootstrap estimator $\hat{\mathbf{m}}^*$ such that $\alpha\%$ of all bootstrap estimators have an equal or smaller residual value. Intuitively, if we were to judge the accuracy of an estimator only on its residual, then we could say $\hat{q}_\alpha(\tau)$ is the residual of the $(\alpha * B)$-th 'best' bootstrap estimator $\hat{m}^*(\tau)$ out of all B estimators. For example, if $\alpha = 0.05$ and $B = 100$, then $\hat{q}_{0.05}$ is the residual of the 5-th best bootstrap estimator.

Using equation (6), pointwise confidence intervals are constructed as follows, for each timepoint $\tau \in (0, 1]$:

$$I(\tau) = [\hat{m}(\tau) - \hat{q}_{1-\alpha/2}(\tau), \quad \hat{m}(\tau) - \hat{q}_{\alpha/2}(\tau)]$$

To further illustrate the meaning of the α-quantiles, if we take $\alpha = 0.05$, we find $\hat{q}_{\alpha/2} = q_{0.025}$ and $\hat{q}_{1-\alpha/2} = q_{0.975}$. The interval $[q_{0.025}, q_{0.975}]$ then contains the residuals of 95% of bootstrap estimators $\hat{\mathbf{m}}^*$, where 2.5% fall below the interval, and another 2,5% fall above it.

With pointwise confidence intervals we look at the points of a function individually. The band consists of intervals for each time point such that its corresponding value is believed to exist within its respective interval. When we are

looking at a time series containing autocorrelation, we are more interested in simultaneous confidence bands. As opposed to looking at each time point individually, with simultaneous confidence bands the individual observations are considered as having occurred as one connected process. With simultaneous confidence bands we state that the function as a whole exists within this band with some level of certainty. Because of this larger implication, simultaneous confidence bands are often wider than its pointwise counterpart. Simultaneous confidence bands are constructed using the procedure proposed by Bühlmann et al. (1998). First find α_s:

$$\alpha_s = \arg\min_{\alpha_p \in [1/B, \alpha]} |\mathbb{P}^*[\hat{q}_{\alpha_p/2} \leq \text{residual} \leq \hat{q}_{1-\alpha_p/2}(\tau), \forall \tau \in G] - (1-\alpha)|.$$

where residual $= \hat{m}^*(\tau) - \tilde{m}(\tau)$.

The resulting α_s is the value $\alpha_p \in [1/B, \alpha]$ for which the ratio of bootstrap estimators $\hat{m}^*(\tau)$ whose residual is inside the interval $[\hat{q}_{\alpha_p/2}, \hat{q}_{1-\alpha_p/2}]$ is closest to (1 - α), the target confidence level. G is a subset of the set of total time points. Having found α_s, the quantiles $\hat{q}_{\alpha_s/2}$ and $\hat{q}_{1-\alpha_s/2}$ along with $\hat{m}(\tau)$ are used to create the simultaneous confidence bands for the underlying function $m(\tau)$. For each τ:

$$I(\tau) = [\hat{m}(\tau) - \hat{q}_{1-\alpha_s/2}, \quad \hat{m}(\tau) - \hat{q}_{\alpha_s/2}]$$

With the resulting confidence band we state that the function m(τ) is believed to exist within this band with (1 - α) certainty.

2.2 Gaussian Process regression

An alternative approach to nonparametric regression called Gaussian process regression performs Bayesian inference over the function space. The method uses a Gaussian process to define a prior distribution over this function space. In Section 2.2.1 we discuss the Gaussian distribution and Gaussian processes, which form the foundation of Gaussian process regression. In Section 2.2.2 we discuss the role of kernels in GPR. Finally, in Section 2.2.3 we see how the GPR method constructs confidence bands around a function.

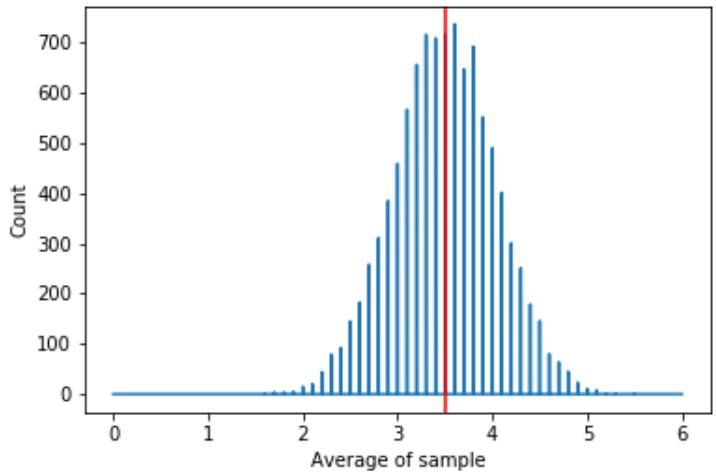
2.2.1 Gaussian distribution and Gaussian processes

The Gaussian or normal distribution is a very common continuous probability distribution. The Gaussian distribution is useful because of the Central Limit Theorem. The theorem states that averages of independently drawn random variables become normally distributed when the number of samples is sufficiently large. For example, when we throw three fair dices repeatedly, the average of the three throws will become normally distributed if we throw often enough. Figure 2.3 on the next page shows this. Note that this is the case even though a single dice throw on its own is not normally distributed. The normal distribution occurs very often in nature. For example: income, height, IQ, and shoe size are all normally distributed.

More generally, when multiple factors have either some positive or negative influence on some outcome, it will rarely be the case that all factors have a

positive outcome or all factors have a negative outcome. More likely there will be a mix of positive and negative influences. This results in more outcomes where the result is close to the average, and less outcomes that are far away from this average. Then if we consider for instance the temperature value at a given time to be the result of many different influences, we can consider this value as being a single sample of a normally distributed random variable.

Figure 2.3: Normal distribution



The multivariate normal distribution is the extension of the normal distribution to multiple dimensions. It consists of multiple random variables, where each individual random variable represents a dimension and is normally distributed. Additionally, every linear combination of these random variables is also joint normally distributed. Joint normally distributed intuitively means that combining the two random variables results in a new single random variable that is normally distributed as well.

A Gaussian process is a collection of random variables indexed by time (e.g. a time series) or space, such that every subcollection of those random variables has a multivariate normal distribution. If we think of individual values of a function as normally distributed random variables, then Gaussian processes can be used to describe a distribution over a function space, where each dimension represents the possible values a measurement at a certain time point can take on. Gaussian Process Regression (GPR) uses Gaussian processes to define the prior probability distribution over the function space.

2.2.2 Kernels

Nonparametric regression aims to construct confidence bands around the underlying function $f(t)$ of the following system:

$$y_t = f(t) + \epsilon(t)$$

Here we have changed the notation of the underlying function $m(t)$ from to $f(t)$, which is a more commonly used notation in GPR literature.

If we define $\mathbf{f}$ as the vector of function $f(t)$ applied to each time point:

$$\mathbf{f} \equiv [f(t_1), f(t_2), ..., f(t_n)]^T,$$

then GPR assumes $\mathbf{f}$ is a sample of the Gaussian process prior distribution:

$$\mathbf{f} \sim \mathcal{GP}(\mu, K)$$

This distribution is defined by a mean vector μ and a covariance matrix K. The mean vector μ is often taken to be a zero vector for simplicity, but can be specified as a function itself as well. Note that setting the mean of the prior distribution to a zero vector does not imply that the mean of the posterior distribution is only allowed to be a zero vector as well.

Because the prior distribution does not yet take into account the actual observations, test time points $\mathbf{T}_*$ need to be chosen randomly for which we can define the prior distribution. The entries of the covariance matrix K of this distribution is defined by the evaluation of some kernel function $\mathcal{K}$ on each pair of test time points. Matrix K is illustrated in Figure 2.4.

Figure 2.4: Covariance matrix K

	t_{*1}	t_{*2}	...	t_{*n}
t_{*1}	$\mathcal{K}(t_{*1}, t_{*1})$	$\mathcal{K}(t_{*1}, t_{*2})$	...	$\mathcal{K}(t_{*1}, t_{*n})$
t_{*2}	$\mathcal{K}(t_{*2}, t_{*1}$	$\mathcal{K}(t_{*2}, t_{*2})$	...	$\mathcal{K}(t_{*2}, t_{*n})$
...	...	...	...	...
t_{*n}	$\mathcal{K}(t_{*n}, t_{*1})$	$\mathcal{K}(t_{*n}, t_{*2})$	...	$\mathcal{K}(t_{*n}, t_{*n})$

The use of kernel function is slightly different in GPR than in the autoregressive wild bootstrap. The AWB method uses a kernel to estimate $m(t)$ by computing a weighted average of the y_t values in some interval around t. In GPR, a kernel is used as a covariance function that takes as input two values and returns their similarity. For example, the linear kernel specifies the covariance between two time points t and t' as:

$$\mathcal{K}_{Lin}(t, t') = \sigma^2 * t * t',$$

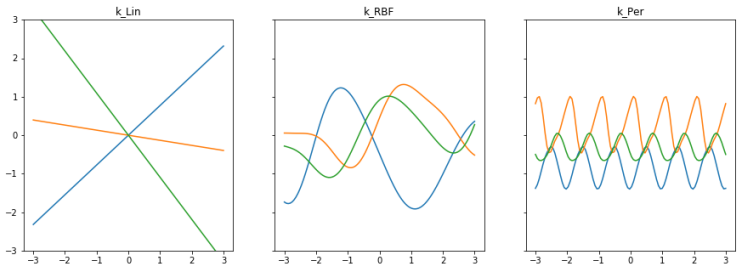
where σ^2 is the variance parameter of the kernel.

As the mean vector is often taken to be a zero vector, the selection of the kernel $\mathcal{K}$ plays a crucial part in Gaussian process regression. The choice of kernel defines the covariance matrix of the prior distribution. To see the effect of the choice of kernel on the prior distribution, we draw three samples from each prior distribution resulting from a different kernel choice. The samples are displayed in Figure 2.5 on the next page. A sample can be obtained from the prior distribution by choosing a number of test input points $\mathbf{T}_*$, obtain the covariance matrix $K(\mathbf{T}_*, \mathbf{T}_*)$ by using our covariance function on all pairs, and generate vector $\mathbf{f}_*$:

$$\mathbf{f}_* \sim \mathcal{N}(\mu, K(\mathbf{T}_*, \mathbf{T}_*))$$

where μ is the mean vector. The resulting vector $\mathbf{f}_*$ is a sample of the Gaussian process prior distribution.

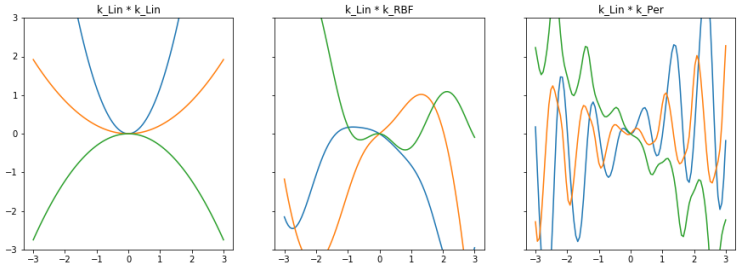
Figure 2.5



A useful characteristic of kernels is that they can be combined into new kernels by multiplying and adding existing kernels together. Three example kernel combinations are shown in Figure 2.6. For instance, multiplying a linear kernel with another linear kernel results in a quadratic kernel. A linear kernel combined with a periodic kernel results in samples that show periodicity and growing amplitude as we move away from the origin.

As the prior distribution is the initial basis we start our process off with, being able to combine kernels thus allows us to create custom kernels that might give us a better starting point. Defining the prior distribution by choosing a kernel is essentially providing the GPR method with an educated guess to take into consideration.

Figure 2.6



2.2.3 GPR confidence bands

After having chosen the kernel and finding the prior distribution, the next step is the incorporation of our observations. Let us first assume we have noise-free observations. This implies $\mathbf{y} = \mathbf{f}$, as noise is not part of the system. Then the joint distribution of our observations $\mathbf{f}$, which we will call training outputs, and the test outputs $\mathbf{f}_*$ sampled from the prior distribution is:

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(\mathbf{T}, \mathbf{T}) & K(\mathbf{T}, \mathbf{T}_*) \\ K(\mathbf{T}_*, \mathbf{T}) & K(\mathbf{T}_*, \mathbf{T}_*) \end{bmatrix}\right)$$

Here $\mathbf{T}$ is the set of training time points and $\mathbf{T}_*$ is the set of test time points. $K(\mathbf{T}, \mathbf{T}_*)$ denotes the covariance matrix of size n x n_*, where n is the number of observations or training points $\mathbf{T}$, and n_* is the number of test points $\mathbf{T}_*$. The covariance matrix of this joint distribution is thus one matrix consisting of four submatrices. Our objective now is to restrict this joint prior distribution to contain only those functions that match the observed measurements. This corresponds to *conditioning* the joint prior distribution on the observations, yielding the conditional or posterior distribution $\mathbf{f}_*|\mathbf{T}_*, \mathbf{T}, \mathbf{f}$:

$$\begin{aligned}\mathbf{f}_*|\mathbf{T}_*, \mathbf{T}, \mathbf{f} &\sim \mathcal{N}(\mu, K), \quad \text{where:} \\ \mu &= K(\mathbf{T}_*, \mathbf{T})K(\mathbf{T}, \mathbf{T})^{-1}\mathbf{f}, \\ K &= K(\mathbf{T}_*, \mathbf{T}_*) - K(\mathbf{T}_*, \mathbf{T})K(\mathbf{T}, \mathbf{T})^{-1}K(\mathbf{T}, \mathbf{T}_*).\end{aligned}$$

In Appendix A we discuss how the equations for μ and K were derived.

Now we move on to the more realistic case in which we only have noisy observations $y_t = f(t) + \epsilon(t)$, where $\epsilon(t)$ is some i.i.d error with variance σ_n^2. The new covariance matrix for these noisy observations $\mathbf{y}$ is:

$$Cov(\mathbf{y}) = K(\mathbf{T}, \mathbf{T}) + \sigma_n^2 I$$

where I is the identity matrix, in which diagonal entries are 1 and all other entries are 0. Adding the term $\sigma_n^2 I$ ensures that the noise is considered, by adding the variance σ_n^2 to the diagonal of the covariance matrix $K(\mathbf{T}, \mathbf{T})$.

Taking into account the noise, we obtain the new joint distribution of observations $\mathbf{y}$ and test outputs $\mathbf{f}_*$:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(\mathbf{T}, \mathbf{T}) + \sigma_n^2 I & K(\mathbf{T}, \mathbf{T}_*) \\ K(\mathbf{T}_*, \mathbf{T}) & K(\mathbf{T}_*, \mathbf{T}_*) \end{bmatrix}\right)$$

The new posterior distribution is then found:

$$\begin{aligned}\mathbf{f}_*|\mathbf{T}_*, \mathbf{T}, \mathbf{y} &\sim N(\bar{\mathbf{f}}_*, \text{cov}(\mathbf{f}_*)), \quad \text{where} \\ \bar{\mathbf{f}}_* &= K(\mathbf{T}_*, \mathbf{T})[K(\mathbf{T}, \mathbf{T}) + \sigma_n^2 I]^{-1}\mathbf{y}, \\ \text{cov}(\mathbf{f}_*) &= K(\mathbf{T}_*, \mathbf{T}_*) - K(\mathbf{T}_*, \mathbf{T})[K(\mathbf{T}, \mathbf{T}) + \sigma_n^2 I]^{-1}K(\mathbf{T}, \mathbf{T}_*).\end{aligned}$$

A practical implementation of Gaussian process regression by Williams and Rasmussen (2006) is given in Algorithm 1 on the next page. For improved ease of reading, $K = K(\mathbf{T}, \mathbf{T})$ and $\mathbf{k}_*$ is the vector of size n consisting of the covariances between test point t_* and the n training points $\mathbf{T}$.

As the inversion of matrices is an expensive computation, the method uses Cholesky decomposition to find $K[\mathbf{T}, \mathbf{T} + \sigma_n^2 I]^{-1}$. In steps two and four, '\' is the backslash operator, which solves a system of linear equations.

Steps three to five are repeated for each chosen test point x_*, resulting in the mean vector $\bar{\mathbf{f}}_*$, variance vector $\mathbb{V}[\mathbf{f}_*]$, and the log-marginal likelihood. The log-marginal likelihood is a value that describes how well the obtained model is fitting the data given the chosen kernel and its parameters.

Algorithm 1 Gaussian process regression, Williams and Rasmussen (2006)

1: $L \leftarrow chol(K + \sigma_n^2 I)$
2: $\alpha \leftarrow L^T \backslash (L \backslash \mathbf{y})$
3: $\bar{\mathbf{f}}_* \leftarrow \mathbf{k}_*^T \alpha$
4: $\mathrm{v} \leftarrow L \backslash \mathbf{k}_*$
5: $\mathbb{V}[f_*] \leftarrow k(t_*, t_*) - \mathrm{v}^T \mathrm{v}$
6: $\log p(\mathbf{y}|\mathbf{T}) \leftarrow -\frac{1}{2}\mathbf{y}^T \alpha - \sum_i \log L_{ii} - \frac{n}{2}\log 2\pi$
7: **return**: $\bar{\mathbf{f}}_*$ (mean), $\mathbb{V}[\mathbf{f}_*]$ (variance), $\log p(\mathbf{y}|\mathbf{T})$ (log marginal likelihood)

If a normal distribution defines the distribution over a random variable, then 95% of the area under the curve contains the values in the range $[\mu - 1.96 * \sigma, \quad \mu + 1.96 * \sigma]$, where μ is the mean and σ the standard deviation. Therefore, the bounds of the 95% confidence bands around function $\mathbf{f}$ can be constructed by adding and subtracting $1.96 * \sqrt{\mathbb{V}[\mathbf{f}_*]}$ from the mean vector $\bar{\mathbf{f}}_*$ at each time point:

$$\text{lower bound} = \bar{\mathbf{f}}_* - 1.96 * \sqrt{\mathbb{V}[\mathbf{f}_*]}$$
$$\text{upper bound} = \bar{\mathbf{f}}_* + 1.96 * \sqrt{\mathbb{V}[\mathbf{f}_*]}$$

The region within these bounds then represents the function space in which it is believed the actual function $\mathbf{f}$ to exist with 95% certainty.

Chapter 3

Simulation study

In order to compare the two methods, we do a simulation study. In Section 3.1 we discuss the setup for this simulation study. In Section 3.2 we attempt to reproduce the results of Friedrich et al. (2020) for simultaneous coverage of the AWB method in their simulation study. Finally, in Section 3.3 we use both the AWB and GPR method to construct prediction bands, and compare their accuracy.

3.1 Simulation setup

We follow the simulation setup of Friedrich et al. (2020), where the observations $\mathbf{y}$ are generated as:

$$y_t = m\left(\frac{t}{n}\right) + \sigma_t u_t,$$

where $m\left(\frac{t}{n}\right)$ is the underlying function we wish to construct confidence bands around, and $\sigma_t u_t$ represents the noise. The noise consists of heteroscedasticity σ_t, and autoregressive error term u_t. The function $m(\tau)$ is defined as follows:

$$m(\tau) = \beta_1 \tau + \beta_2 \cdot \tau \cdot G(\tau, \lambda, c),$$

in which $G(\tau, \lambda, c)$ is the transition function defined as:

$$G(\tau, \lambda, c) = (1 + \exp\{-\lambda(\tau - c)\})^{-1}$$

The function describes a shifting mean model. Parameter c defines when in time the shift happens, and λ the smoothness of the shift. Figure 3.1 on the next page shows the function $m(\tau)$ for β_1 = -1, β_2 = 2.5, λ = 10, and c = 0.9. The chosen values result in a downward function during the first three quarters of the process shifting to an upward trend in the final quarter. This particular trend is interesting as it mimics the pattern expected for atmospheric ethane time series.

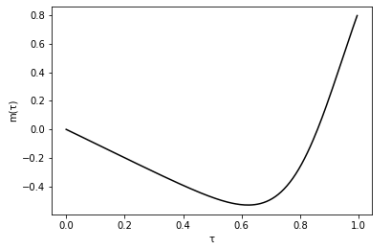


Figure 3.1: $m(\tau)$

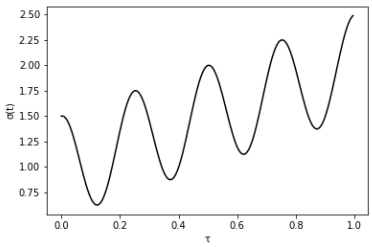


Figure 3.2: $\sigma(\tau)$

The heteroscedasticity process σ is defined as:

$$\sigma(\tau) = \sigma_0 + (\sigma_1 - \sigma_0)\tau + a\cos(2\pi k\tau)$$

Figure 3.2 shows the heteroscedasticity process for $\sigma_0 = 1$, $\sigma_1 = 2$, $a = 0.5$, and $k = 4$. Parameter a controls the amplitude of the waves, while k determines the amount of waves that occur. Increasing a results in higher amplitude, and higher k results in more waves.

The autoregressive error term u_t is defined as:

$$u_t = \phi u_{t-1} + \psi\epsilon_{t-1} + \epsilon_t,$$

where

$$\epsilon_t \sim \mathcal{N}\left(0, \frac{(1-\phi^2)/4}{1+\psi^2-2\phi\psi}\right).$$

The parameter ϕ is the autoregressive (AR) parameter, which determines the influence of previous values of u_t on the current value of u_t. Parameter ψ is the moving average (MA) parameter of the data generating model, which determines the influence of previous values of ϵ_t on the current value of u_t. The degree of autocorrelation in the data can thus be varied by altering the AR and MA parameters.

3.2 AWB simultaneous coverage

A statistic that gives insights into the accuracy of a regression result is the coverage probability. Intuitively this statistic tells us how well the method was able to fit a confidence band to the underlying function of the observed data. Friedrich et al. (2020) report the coverage probabilities for pointwise and simultaneous confidence bands of the AWB method. In this section we try to reproduce their results for the simultaneous confidence bands.

In the case of simultaneous confidence bands, the simultaneous coverage probability is defined as the fraction of simulations in which for all $t \in \mathbf{G}$, where $\mathbf{G}$ is a subset of total time points $\mathbf{T}$, it holds that $m(t)$ is within the simultaneous confidence band. $\mathbf{G}$ is a subset of the $n = 200$ time points $\mathbf{T}$, defined as:

$$\mathbf{G} = U_1(h) + U_2(h) + U_3(h) + U_4(h),$$
$$\text{where} \quad U_i(h) = \{(i/5) - h + j/100 \quad | \quad j \in \mathbb{N}, 0 \leq j \leq (200 \cdot h)\},$$

and h is the bandwidth parameter. A higher bandwidth leads to a larger set $U_i(h)$ and consequently a larger set $\mathbf{G}$.

It is important to note that we do not measure the simultaneous coverage of the confidence bands created by the GPR method in this section. This is because the method discussed above is very much in line with the frequentist way of looking at probability, where probabilities represent long run frequencies. The Bayesian approach considers probability in the perhaps more intuitive sense, simply as the probability of an event happening. Consequently, applying the same evaluation method as used for the AWB on the GPR would not be in line with its underlying intuition.

We will refer to the process of selecting the AR, MA, k, and a parameters and generating the data for those specifications as the Data Generating Process (DGP). For each specification of the DGP we run 100 simulations. If all $t \in \mathbf{G}$ were within the interval, we assign that simulation the value one. Otherwise, we assign it the value zero. The simultaneous coverage is then found by adding up the ones and zeroes, and dividing it by the number of simulations.

Table 3.1 on the next page shows the simultaneous coverage probabilities we found for several different specifications of the DGP, along with the average median interval lengths in brackets. We generated data of size $n = 200$. The degree of heteroscedasticity in the data was kept the same for all specifications at $k = 4$ and $a = 0.5$. Autocorrelation was varied by changing the AR and MA parameters of the DGP. Six different scenarios have been tested. $AR_{0.2}$ means that the AR parameter is set to 0.2, and the MA parameter is set to 0. Vice versa, $MA_{0.2}$ means that AR is set to 0 and MA to 0.2.

In addition to varying the parameters of the DGP, two parameters of the AWB method were varied. For each of the different setups, three different bandwidths h were used in the AWB method, and three different γ values. The kernel used in all specifications is the Epanechnikov kernel, defined as:

$$K(u) = \frac{3}{4}(1-u)^2 \quad \text{for} \quad |u| \leq 1,$$

where u is the distance between two time points. The Epanechnikov kernel places higher importance on values closer to time point t than values that are further away.

The values we found generally match the values found by Friedrich et al. (2020). When our median width had some difference with theirs, the corresponding coverages differed proportional to the difference in width. This makes sense, as a confidence band with a smaller width will generally result in lower coverage, and wider bands will result in higher coverage. For each specification we ran 100 simulations while Friedrich et al. (2020) ran 5000 simulations. We ran 100 simulations for each specification because we had a time constraint. Running significantly more simulations would require parallellization, which is outside the scope of this thesis.

Table 3.1: **AWB Simultaneous coverage and average median interval length (in brackets) for $k = 4$ and $a = 0.5$**

h	DGP	$\gamma = 0.2$		$\gamma = 0.4$		$\gamma = 0.6$	
0.02	0	0.98	(0.563)	0.94	(0.496)	0.85	(0.410)
	$AR_{0.2}$	0.91	(0.526)	0.84	(0.510)	0.81	(0.447)
	$AR_{0.5}$	0.78	(0.478)	0.76	(0.456)	0.57	(0.423)
	$AR_{-0.5}$	1.00	(0.418)	0.99	(0.356)	0.94	(0.285)
	$MA_{0.2}$	0.95	(0.533)	0.90	(0.508)	0.79	(0.446)
	$MA_{0.5}$	0.92	(0.486)	0.84	(0.479)	0.82	(0.406)
0.04	0	0.98	(0.407)	0.86	(0.386)	0.84	(0.332)
	$AR_{0.2}$	0.76	(0.405)	0.82	(0.397)	0.70	(0.347)
	$AR_{0.5}$	0.58	(0.385)	0.63	(0.385)	0.54	(0.354)
	$AR_{-0.5}$	0.99	(0.331)	1.00	(0.281)	0.98	(0.229)
	$MA_{0.2}$	0.92	(0.407)	0.83	(0.401)	0.74	(0.359)
	$MA_{0.5}$	0.75	(0.385)	0.79	(0.378)	0.67	(0.347)
0.06	0	0.92	(0.354)	0.86	(0.338)	0.79	(0.294)
	$AR_{0.2}$	0.84	(0.357)	0.76	(0.352)	0.78	(0.314)
	$AR_{0.5}$	0.44	(0.327)	0.57	(0.339)	0.48	(0.324)
	$AR_{-0.5}$	1.00	(0.283)	1.00	(0.254)	0.96	(0.216)
	$MA_{0.2}$	0.85	(0.355)	0.77	(0.346)	0.66	(0.316)
	$MA_{0.5}$	0.76	(0.336)	0.78	(0.337)	0.71	(0.313)

Next we measure the coverage probabilities in the case where approximately 70% of the data is missing. For ease of comparison, we want to have 200 existing observations for this experiment as well. With 70% missing data, this means we will have 200 existing observations distributed over 666 time points. For this experiment, no heteroscedasticity is present in the data. As before, we vary the degree of autocorrelation in six different ways. Table 3.2 on the next page shows the simultaneous coverage and average median interval lengths for these different setups, using only bandwidth $h = 0.06$. A Markov chain is implemented for the process $\mathbf{D}$ with the following transition probabilities:

$$\begin{array}{c} \\ D_{t-1} = 0 \\ D_{t-1} = 1 \end{array} \begin{array}{c} \begin{array}{cc} D_t = 0 & D_t = 1 \end{array} \\ \begin{pmatrix} 0.80 & 0.20 \\ 0.45 & 0.55 \end{pmatrix} \end{array}$$

This Markov chain leads to a missingness pattern that is representative of the one found in the ethane emissions dataset at the Jungfraujoch station in the Swiss Alps.

While the median interval lengths match the ones found by Friedrich et al. (2020), the coverages we found are consistently lower than theirs. This could be due to the difference in amount of simulations ran for each setup.

Table 3.2: **AWB Simultaneous coverage and average median interval length (in brackets) with 70% missing data with 100 simulations**

h	DGP	$\gamma = 0.2$		$\gamma = 0.4$		$\gamma = 0.6$	
0.06	0	0.87	(0.257)	0.80	(0.239)	0.71	(0.209)
	$AR_{0.2}$	0.75	(0.252)	0.72	(0.240)	0.56	(0.217)
	$AR_{0.5}$	0.61	(0.235)	0.54	(0.225)	0.54	(0.211)
	$AR_{-0.5}$	0.88	(0.210)	0.86	(0.191)	0.74	(0.162)
	$MA_{0.2}$	0.78	(0.252)	0.70	(0.245)	0.64	(0.211)
	$MA_{0.5}$	0.65	(0.236)	0.70	(0.233)	0.50	(0.208)

3.3 GPR vs AWB prediction

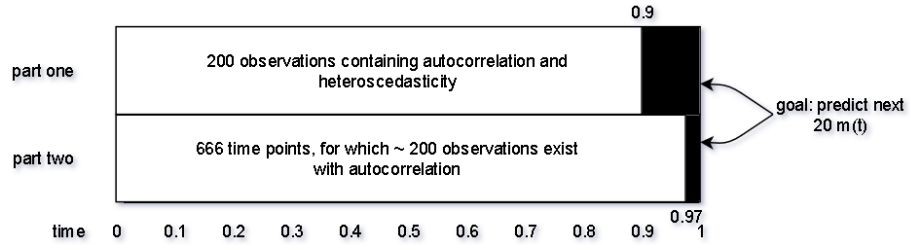
To give us insight in the prediction accuracy of the two methods, we use both methods to construct confidence bands for future time points. Similar to the previous section, we divide the experiment up into two parts. Figure 3.3 on the next page describes the setup of our experiment.

In part one we look at data that contains autocorrelation, heteroscedasticity, and no missing data. Thus we have 200 observations, and wish to estimate the $m(t)$ values of the next 20 time points by extending the confidence band to these future time points.

In part two we look at data that contains autocorrelation, missing data, and no heteroscedasticity. We stick to a target data set size of $n = 200$, and again construct confidence bands that extend to the next 20 time points. The Markov chain defined in Section 3.2 results in approximately 70% missing data, meaning that if we want to have 200 existing observations, we need to have an original data set of 666 data points. As a result, in both parts of the experiment we will have approximately 200 observations to work with, and wish to create a confidence band that extends to the next 20 time points.

Note that we need to know the actual 'correct' $m(t)$ values of the next 20 time points, i.e. the $m(t)$ values that made up the observations $\mathbf{y}$. Only if we know these we can say with absolute certainty whether the methods have predicted future $m(t)$ values correctly. Therefore, in part one of our method we generate 220 observations along with their underlying 220 $m(t)$ values. Of these 220 observations only the first 200 are used by the two methods. In part two we generate 686 observations along with their $m(t)$ values, where 70% of the observations in the first 666 time points will be missing. The last 20 existing observations of the 686 time points are not used by the methods.

Figure 3.3: Prediction experiment setup



3.3.1 GPR prediction bands

We will now discuss the steps necessary to create confidence bands that extend to future time points with the GPR method.

In part one of the experiment we have generated 220 observations corresponding to time points between $\tau = 0$ and $\tau = 1$. Because the method only uses the first 200, only the time points from $\tau = 0$ to $\tau = 0.9$ will have existing observations. The goal is to estimate the values of the next 20 time points, which are between $\tau = 0.9$ and $\tau = 1$. Recall that in the first step of the GPR method, the creation of the covariance matrix, we need to choose a set of test time points $\mathbf{T}^*$. If we choose 220 time points from $\tau = 0$ to $\tau = 1$, then approximately 20 will exist between $\tau = 0.9$ and $\tau = 1$. This $\mathbf{T}^*$ will then give us a confidence band extending past the existing 200 observations. We then find the coverage by computing the fraction of $m(t)$ values from 201 to 220 that are inside the confidence bands.

In part two there are only existing observations for time points from $\tau = 0$ to $\tau \approx 0.97\,(\frac{666}{686})$. In this scenario we choose 686 time points $\mathbf{T}^*$ from $\tau = 0$ to $\tau = 1$ and construct the confidence bands. Approximately 20 time points will be within $\tau \approx 0.97$ and $\tau = 1$. We now find the prediction coverage by computing the fraction of $m(t)$ values from 667 to 686 that are inside the confidence bands.

For the GPR method, a kernel function $\mathcal{K}$ needs to be chosen. With our choice of kernel, we define the covariance matrix of our prior distribution. The flexibility of the GPR approach becomes apparent when deciding on a kernel. When plotting the generated data, certain characteristics are clearly visible without yet looking at the specific values of the observations.

Figure 3.4: Example data

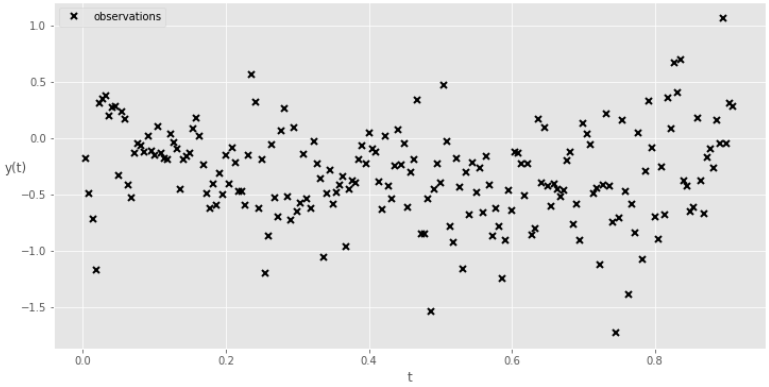


Figure 3.4 shows data generated with parameters $k = 4$, $a = 0.5$, $AR = 0$, $MA = 0.2$. A certain degree of seasonality is visible, and there is lots of noise present in the data. We can find our kernel by choosing separate kernels to handle some of the individual characteristics specifically, and combine them using multiplication and addition.

The basis of our kernel will be the Radial Basis Function (RBF) kernel, also known as the Squared Exponential (SE) kernel. It is defined as:

$$\mathcal{K}_{RBF}(t,t') = \sigma^2 \exp\left(-\frac{(t-t')^2}{2l^2}\right),$$

where σ^2 is the output variance. Every kernel contains this parameter. It describes the average distance away from the mean function. Parameter l is the lengthscale, which describes the length of a 'wiggle' in the function. Figure 2.5 in Section 2.2.2 shows possible samples from a prior distribution resulting from the RBF kernel. The RBF kernel is a commonly used kernel for Gaussian Processes, because it can be used to model a wide variety of functions.

To handle the periodicity in the data, we use a combination of the periodic kernel $\mathcal{K}_{PER}$ with the $\mathcal{K}_{RBF}$ kernel. Multiplying the periodic kernel with the RBF kernel allows the function to be periodic, but does not restrict the function to be periodic in the exact same manner for the entire range of the function. The 'waves' are allowed to vary in amplitude and length over time. This combination of the two kernels is called the Locally Periodic Kernel and is defined as:

$$\begin{aligned}\mathcal{K}_{LocalPer}(t,t') &= \mathcal{K}_{Per}(t,t') * \mathcal{K}_{RBF}(t,t') \\ &= \sigma^2 \exp\left(-\frac{2sin^2(\pi|t-t'|/p)}{l^2}\right) \exp\left(-\frac{(t-t')^2}{2l^2}\right),\end{aligned}$$

where σ^2 is again the variance, p is the distance between two repetitions, and l is the lengthscale parameter.

To take into account the noise into the data, we use a white noise kernel. It is defined as:

$$\mathcal{K}_{White}(t,t') = \delta(t,t') * \sigma^2,$$

where $\delta(t, t')$ is the Kronecker delta, which is 1 if $t = t'$, and 0 otherwise. σ^2 is the variance. We find our final kernel by adding the three kernels specified so far together:

$$\mathcal{K}_{Final} = \mathcal{K}_{RBF} + \mathcal{K}_{LocalPer} + \mathcal{K}_{White}$$

This kernel has a total of six parameters: two coming from $\mathcal{K}_{RBF}$, three from $\mathcal{K}_{LocalPer}$ and one from $\mathcal{K}_{White}$.

The practical implementation of Gaussian Process Regression given in Algorithm 1 Williams and Rasmussen (2006) returns the mean vector, variance vector, and log-marginal likelihood. The log-marginal likelihood tells us how well the model is fitting the data given the chosen parameters. To find appropriate values for these parameters, we maximize the marginal likelihood. This also means that it is not absolutely necessary to base the kernel decision on the plotted data. One can simply try out multiple kernels, and maximize the log-marginal likelihood for each of the kernels. The final kernel can then be chosen simply by taking the kernel with the highest log-marginal likelihood, i.e. the kernel resulting in a model that best fits the observed data.

Once we have found suitable parameter values, we use the mean vector and variance vector to construct the confidence bands as described at the end of Section 2.2.3. We use the python package GPflow from Matthews et al. (2017) for the GPR method.

3.3.2 AWB prediction bands

Now we will discuss how confidence bands can be created using the AWB method.

If we only give the existing 200 observations to the AWB method, it would create a confidence band for these time points only. In order to extend this band to future time points, we need to supply 220 time points to the method. We can ensure the method does not take into account values y_{201} to y_{220} by setting the last 20 values of the missing data process $\mathbf{D}$ to zero: $\mathbf{D}_{201}, \ldots, \mathbf{D}_{220} = 0$. This way, the AWB will construct confidence bands for all 220 time points, and will not take into account the last 20 y_t values in the process. Similarly, for part two of the experiment, we supply the method with all 686 time points, and set $\mathbf{D}_{667}, \ldots, \mathbf{D}_{686}$ to zero.

Note that a sufficiently high bandwidth needs to be chosen in order to successfully create the prediction bands. In the Nadaraya-Watson estimator described in equation (3) in Section 2.1.3, the bandwidth parameter of the estimator specifies the range around t from which we should take into account the y_t values for the estimation. If the bandwidth is chosen such that this range is too small and consequently there are no existing observations inside it, no estimation can be made for this t, as the denominator of the estimator will be zero.

Table 3.3 on page 25 shows the prediction coverage and median interval length of the created confidence band for both methods in part one of our experiment. As in the previous section, we have six different specifications of the degree of autoregression in the data generating process (DGP). The heteroscedasticity is kept the same throughout all specifications at $k = 4$ and $a = 0.5$. For the AWB method we use two bandwidths, $h = 0.10$ and $h = 0.12$. We vary the γ parameter in the same way as before, using $\gamma = 0.2, 0.4$, and

0.6. For both methods we run 25 simulations for each specification. We only do 25 simulations because we have a time constraint. In order to do significantly more simulations, it would be required to use parallellization. This is however outside the scope of this thesis.

The AWB method with bandwidth $h = 0.10$ outperforms the GPR method throughout all specifications of the DGP and all choices of γ in part one of our experiment. The GPR method creates considerably smaller confidence bands in all scenarios, resulting in lower coverage. This shows the GPR method has severe difficulties handling data that is both autocorrelated and heteroscedastic in comparison to the AWB method. When the DGP is set to have no autocorrelation, the GPR method also seems to have problems. This signifies that heteroscedasticity on its own in time series is difficult for the GPR method.

Table 3.4 on page 26 shows the prediction coverage for both methods for part two of our experiment, in which data contains autocorrelation and large portions of missing data. For all specifications of the DGP, the GPR method performs either equally well as the AWB, or outperforms it. Additionally, the width of the band produced with GPR is smaller, meaning the GPR is able to do a more confident prediction.

Table 3.3: **AWB vs. GPR Prediction coverage and median interval length (in brackets) for $k = 4$ and $a = 0.5$**

h	DGP	$\gamma = 0.2$ AWB	$\gamma = 0.4$ AWB	$\gamma = 0.6$ AWB	GPR
0.10	0	0.846	0.916	0.89	0.666
		(1.351)	(1.521)	(1.464)	(0.847)
	$AR_{0.2}$	0.774	0.856	0.864	0.608
		(1.373)	(1.528)	(1.475)	(0.850)
	$AR_{0.5}$	0.772	0.992	0.89	0.742
		(0.898)	(1.470)	(1.611)	(1.074)
	$AR_{-0.5}$	0.898	0.858	0.988	0.514
		(1.327)	(1.478)	(1.570)	(0.761)
	$MA_{0.2}$	0.822	0.858	0.824	0.598
		(1.440)	(1.478)	(1.421)	(0.847)
	$MA_{0.5}$	0.668	0.83	0.876	0.662
		(1.213)	(1.357)	(1.556)	(0.888)
0.12	0	0.746	0.734	0.814	0.666
		(1.107)	(1.116)	(1.296)	(0.847)
	$AR_{0.2}$	0.502	0.812	0.784	0.608
		(0.980)	(1.267)	(1.312)	(0.850)
	$AR_{0.5}$	0.79	0.726	0.754	0.742
		(1.085)	(1.123)	(1.245)	(1.074)
	$AR_{-0.5}$	0.802	0.894	0.954	0.514
		(1.013)	(1.157)	(1.197)	(0.761)
	$MA_{0.2}$	0.668	0.714	0.812	0.598
		(1.094)	(1.128)	(1.181)	(0.847)
	$MA_{0.5}$	0.746	0.762	0.572	0.662
		(1.057)	(1.190)	(1.032)	(0.888)

Table 3.4: **AWB vs. GPR Prediction coverage and median interval length (in brackets) for 70% missing data**

h	DGP	$\gamma = 0.2$ AWB	$\gamma = 0.4$ AWB	$\gamma = 0.6$ AWB	GPR
0.10	0	1.0	0.948	1.0	1.0
		(1.023)	(0.994)	(1.042)	(0.912)
	$AR_{0.2}$	0.988	0.964	0.96	1.0
		(0.951)	(1.110)	(1.045)	(0.905)
	$AR_{0.5}$	0.942	0.926	0.99	0.992
		(1.047)	(1.045)	(1.011)	(0.795)
	$AR_{-0.5}$	1.0	1.0	0.96	1.0
		(0.928)	(1.036)	(1.011)	(0.796)
	$MA_{0.2}$	0.96	1.0	0.96	1.0
		(0.974)	(1.151)	(1.009)	(0.915)
	$MA_{0.5}$	0.984	0.948	0.938	1.0
		(1.122)	(1.018)	(0.995)	(0.823)
0.12	0	0.768	0.91	0.886	1.0
		(0.816)	(0.921)	(0.910)	(0.912)
	$AR_{0.2}$	0.722	0.834	0.712	1.0
		(0.838)	(0.862)	(0.879)	(0.905)
	$AR_{0.5}$	0.87	0.832	0.944	0.992
		(0.819)	(0.908)	(0.934)	(0.795)
	$AR_{-0.5}$	0.894	0.954	0.892	1.0
		(0.891)	(0.956)	(0.847)	(0.796)
	$MA_{0.2}$	0.96	0.808	0.744	1.0
		(0.858)	(0.922)	(0.907)	(0.915)
	$MA_{0.5}$	0.878	0.76	0.834	1.0
		(0.819)	(0.839)	(0.915)	(0.823)

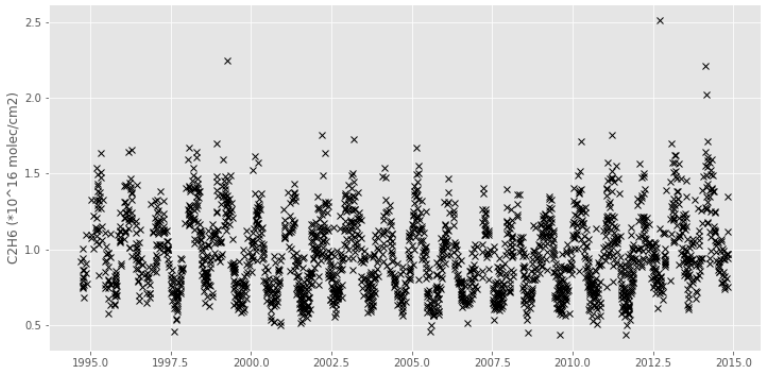
Chapter 4

Real life application

In this chapter we apply the Gaussian process regression method to a real life ethane emissions time series. Ethane emissions are a regularly used measure of pollution in the atmosphere. The data for this time series was derived by Franco et al. (2015) from observations made at the Sphinx Observatory at the Jungfraujoch station in the Swiss Alps between September 1994 and August 2014. There are a total of 2260 points over the approximately 20 year time period, meaning the average number of observations per year is 112.6. This further underlines the importance of using a nonparametric regression method that handles large portions of missing data well.

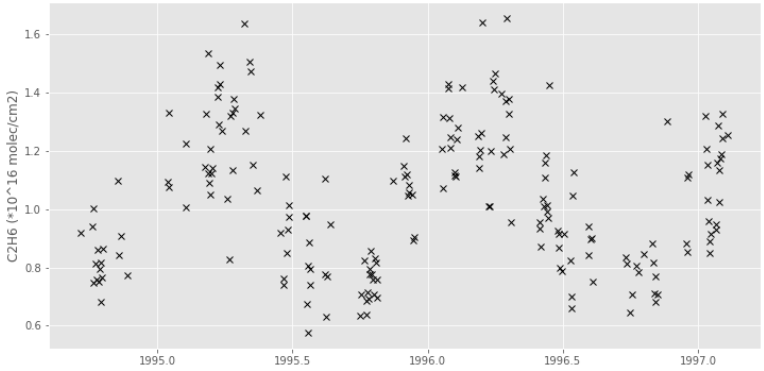
A kernel needs to be chosen for the GPR method. We look at the observed data and make our decision based off of the characteristics we see. Figure 4.1 shows the entire time series, containing all observations from September 1994 to August 2014. Immediately a degree of periodicity is visible in the data.

Figure 4.1



As the density of the data points is quite high, we zoom in on a subset of the data to get a closer look. Figure 4.2 on the next page shows the first 200 observations of the time series, which were made between September 1994 and February 1997.

Figure 4.2



The seasonal pattern becomes even more clear. However, by looking at the complete data set, we can see that the amplitude of the waves differs over time. Additionally, it might very well be the case that the length of the waves changes as well. Therefore, we select the Locally Periodic kernel, which allows the waves to vary over time. The locally periodic kernel is the multiplication of the periodic kernel and the RBF kernel, defined as:

$$\begin{aligned}\mathcal{K}_{LocalPer}(t,t') &= \mathcal{K}_{Per}(t,t') * \mathcal{K}_{RBF}(t,t') \\ &= \sigma^2 \exp\left(-\frac{2sin^2(\pi|t-t'|/p)}{l^2}\right) \exp\left(-\frac{(t-t')^2}{2l^2}\right),\end{aligned}$$

It is also clear that there is lots of noise in the data, as drawing a single smooth function through the points is very much impossible. For this we add the white noise kernel:

$$\mathcal{K}_{White}(t,t') = \delta(t,t') * \sigma^2,$$

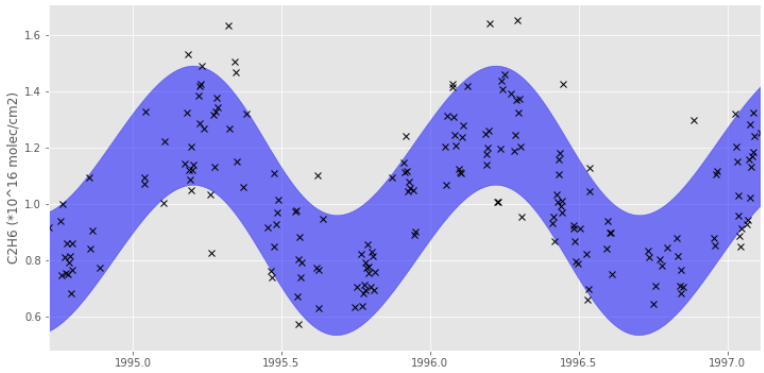
We take as basis for our kernel the RBF kernel, and end up with the same kernel as used in our simulation study:

$$\mathcal{K}_{Final} = \mathcal{K}_{RBF} + \mathcal{K}_{LocalPer} + \mathcal{K}_{White}$$

Note that even though we have based our kernel choice on the characteristics of the plotted data, this is not strictly necessary for the GPR method. A kernel can be chosen simply by trying out multiple kernels and maximizing their respective log-marginal likelihoods, and choosing the one with the highest log-marginal likelihood.

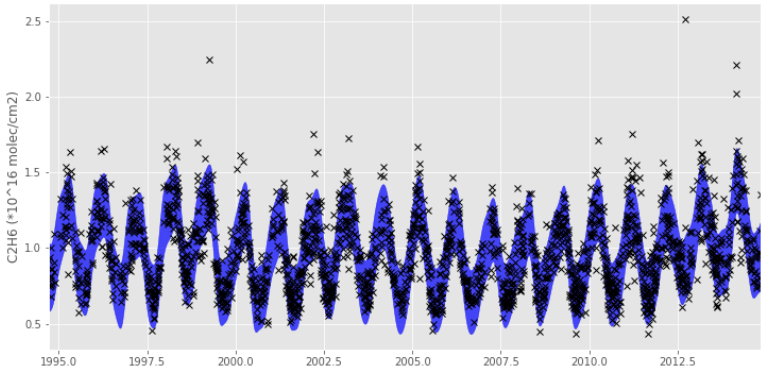
We again use the python package GPflow Matthews et al. (2017) to construct the confidence band. For the subset of 200 observations the confidence band is shown in Figure 4.3 on the next page.

Figure 4.3



When applying the method to the entire time series, a few random restarts were used. The obtained result is shown in Figure 4.4.

Figure 4.4



Franco et al. (2015) have estimated a linear function for this time series. They found a decreasing linear trend until 2009, followed by an increasing linear trend. Friedrich et al. (2020) used the AWB method on the ethane emissions time series, finding the same broken trend, but additionally finding two local peaks in 1998 and 2002-2003. The use of a nonparametric regression method such as the AWB has allowed for the detection of these local peaks, as the function is not restricted to be linear. They argue that the local peaks could be explained by boreal forest fires which took place during both these peaks. In Figure 4.4 we can see that the GPR method detects these local peaks in 1998 and 2002-2003 as well, as seen by the sudden increase in both the lower and upper bound of the confidence band, immediately followed by a decrease again.

Chapter 5

Discussion

In this thesis we compared the ability of two nonparametric regression methods to construct confidence bands around a function describing a time series. We evaluated the prediction accuracy of the AWB and GPR method in a simulation study. The AWB method outperforms the GPR method when autocorrelation and heteroscedasticity are present in the data, and there is no missing data. The main problem for the GPR method seems to lie in the heteroscedasticity, as when autocorrelation is removed, the issue remains. Therefore, when heteroscedasticity is present in the data, the AWB method should be preferred over the standard GPR approach. The GPR method performs equally well or outperforms the AWB when data does not contain heteroscedasticity, but does contain autocorrelation and large portions of missing data.

Several papers have proposed methods based on the standard GPR that do not assume the noise to be independent of the signal, i.e. do not assume the data to be homoscedastic. Le et al. (2005) estimate the variance locally by making the noise itself a random variable, which is estimated along with the mean of the posterior distribution. Similarly, Lázaro-Gredilla et al. (2013) place a Gaussian process prior distribution on the noise. Hong et al. (2018) use a weighting strategy, where weights are put on individual data points or subsets of the time series, depending on their noise level.

We have compared the prediction ability of the two methods by looking at the fraction of underlying function values of the system that were predicted correctly in a simulation study. Other methods that compare coverage between frequentist and Bayesian confidence bands can be considered. Yang et al. (2017) propose a framework for assigning frequentist coverage probabilities to Bayesian confidence bands, allowing for direct comparison between the two methods.

We applied the GPR method to a real life ethane emissions time series. The method found a decreasing trend until around 2009, followed by an increasing trend. This finding is in line with Franco et al. (2015). Additionally, the GPR method found two local peaks during 1998 and 2002-2003 also found by Friedrich et al. (2020), which demonstrate the advantage nonparametric methods have over parametric methods.

A benefit GPR offers over the AWB method lies in the initialization of the methods. While there are data driven methods for the choice of the bandwidth parameter in the AWB method, a completely satisfactory way to select the best bandwidth does not yet exist. Additionally, the selection of the autoregressive

parameter is open-ended. The initialization of the GPR method is done by choosing a kernel, and maximizing the log-marginal likelihood with respect to the parameters. The kernel choice can be made based on observed characteristics in the data, or simply trying out multiple kernels and comparing the maximized log-marginal likelihood of the respective methods.

The GPR method also has a computational advantage over the AWB method, as the AWB method relies on running many Monte Carlo simulations.

Chapter 6

Acknowledgments

I would like to thank my supervisor Yuliya Shapovalova for her continued support throughout the project, and many insightful discussions particularly regarding the Gaussian process regression method. I would also like to thank Marina Friedrich for providing supplementary code for the AWB method and for providing the ethane emissions data set used in the real life application.

Bibliography

Bühlmann, P. et al. (1998). Sieve bootstrap for smoothing in nonstationary time series. *The Annals of Statistics*, 26(1):48–83.

Chaudhuri, P., Loh, W.-Y., et al. (2002). Nonparametric estimation of conditional quantiles using quantile regression trees. *Bernoulli*, 8(5):561–576.

Fan, J., Gasser, T., Gijbels, I., Brockman, M., and Engel, J. (1993). Local polynomial fitting: a standard for nonparametric regression. Technical report, North Carolina State University. Dept. of Statistics.

Franco, B., Bader, W., Toon, G., Bray, C., Perrin, A., Fischer, E., Sudo, K., Boone, C., Bovy, B., Lejeune, B., et al. (2015). Retrieval of ethane from ground-based FTIR solar spectra using improved spectroscopy: Recent burden increase above Jungfraujoch. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 160:36–49.

Friedrich, M., Smeekes, S., and Urbain, J.-P. (2020). Autoregressive wild bootstrap inference for nonparametric trends. *Journal of Econometrics*, 214(1):81–109.

Hong, X., Ding, Y., Ren, L., Chen, L., and Huang, B. (2018). A weighted heteroscedastic Gaussian Process Modelling via particle swarm optimization. *Chemometrics and Intelligent Laboratory Systems*, 172:129–138.

Huang, T.-K. et al. (2006). A technical introduction to Gaussian process regression.

Kaufman, C. G., Sain, S. R., et al. (2010). Bayesian functional ANOVA modeling using Gaussian process prior distributions. *Bayesian Analysis*, 5(1):123–149.

Lázaro-Gredilla, M., Titsias, M. K., Verrelst, J., and Camps-Valls, G. (2013). Retrieval of biophysical parameters with heteroscedastic Gaussian processes. *IEEE Geoscience and Remote Sensing Letters*, 11(4):838–842.

Le, Q. V., Smola, A. J., and Canu, S. (2005). Heteroscedastic Gaussian process regression. In *Proceedings of the 22nd international conference on Machine learning*, pages 489–496. ACM.

Matthews, D. G., Alexander, G., Van Der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z., and Hensman, J. (2017). Gpflow: A Gaussian process library using TensorFlow. *The Journal of Machine Learning Research*, 18(1):1299–1304.

Murray-Smith, R. and Girard, A. (2001). Gaussian process priors with ARMA noise models. In *Irish Signals and Systems Conference, Maynooth*, pages 147–152. Citeseer.

Nadaraya, E. A. (1964). On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142.

Watson, G. S. (1964). Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 359–372.

Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

Wu, C.-F. J. et al. (1986). Jackknife, bootstrap and other resampling methods in regression analysis. *the Annals of Statistics*, 14(4):1261–1295.

Xiao, Z., Linton, O. B., Carroll, R. J., and Mammen, E. (2003). More efficient local polynomial estimation in nonparametric regression with autocorrelated errors. *Journal of the American Statistical Association*, 98(464):980–992.

Yang, Y., Bhattacharya, A., and Pati, D. (2017). Frequentist coverage and sup-norm convergence rate in Gaussian process regression. *arXiv preprint arXiv:1708.04753*.

Appendix A

GPR derivation

In Section 2.2.3 the posterior distribution is defined as:

$$\begin{aligned}\mathbf{f}_*|\mathbf{T},\mathbf{y},\mathbf{T}_* &\sim N(\bar{\mathbf{f}}_*, \text{cov}(\mathbf{f}_*)), \quad \text{where} \\ \bar{\mathbf{f}}_* &= K(\mathbf{T}_*,\mathbf{T})[K(\mathbf{T},\mathbf{T})+\sigma_n^2 I]^{-1}\mathbf{y}, \\ \text{cov}(\mathbf{f}_*) &= K(\mathbf{T}_*,\mathbf{T}_*) - K(\mathbf{T}_*,\mathbf{T})[K(\mathbf{T},\mathbf{T})+\sigma_n^2 I]^{-1}K(\mathbf{T},\mathbf{T}_*).\end{aligned}$$

In *A Technical Introduction to Gaussian Process Regression* Huang et al. (2006) these formulas are derived. We follow their steps, and add the intermediate steps left out in an attempt to get a more complete understanding of how the formulas are derived. We change our previous notation for individual time points t and the set of all time points $\mathbf{T}$, to x_t and $\mathbf{X}$ respectively to match with the notation used by Huang et al. (2006).

Sample S is defined as $S = \{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$. Here x_t denotes time point t and y_t is the corresponding observed value.

The relationship between x_t and y_t is as follows:

$$y_t = f(x_t) + \epsilon(x_t).$$

Function $f(x_t)$ maps the input value x_t to the target value, which is corrupted by noise $\epsilon(x_t)$, resulting in the final noisy observation y_t.

Gaussian process regression assumes that the vector resulting from applying function f on all timepoints is a sample from the normal distribution with a zero mean vector μ and covariance matrix K:

$$\mathbf{f} \equiv [f(x_1), f(x_2), ..., f(x_n)]^T \sim \mathcal{N}(\mu, K) \tag{1}$$

K is the covariance matrix, in which the entries consist of the pairwise eveluation of a kernel $\mathcal{K}$ on the test time points. Furthermore:

$$(\mathbf{y}|\mathbf{f}) = ([y_1, y_2, ...y_n]^T|f) \sim \mathcal{N}(\mathbf{f}, \sigma^2 I) \tag{2}$$

which states the noise comes from an independent joint Gaussian distribution with mean zero. Equation (2) implies that $\mathbf{y}$ is conditionally independent of $\{x_1, x_2, ..., x_n\}$ given $\mathbf{f}$, as knowing the values of vector $\mathbf{f}$ already tells us everything we need to know about $\mathbf{y}$.

The goal of GPR is to obtain the posterior probability distribution, which gives us the probability distribution of $f(\mathbf{x}^*)$ for a new time point $\mathbf{x}^*$ given the sample S: $P(f(\mathbf{x}^*)|\mathbf{x}^*, S)$. For convenience, we denote f($\mathbf{x}^*$) as f^*. We get:

$$\begin{aligned} P(f^*|\mathbf{x}^*, S) &= \int P(f^*, \mathbf{f}|\mathbf{x}^*, S)d\mathbf{f} \\ &= \int P(f^*|\mathbf{f}, \mathbf{x}^*, S)P(\mathbf{f}|\mathbf{x}^*, S)d\mathbf{f} \end{aligned} \tag{3}$$

In A.1 we find $P(f^*|\mathbf{f}, \mathbf{x}^*, S)$, in A.2 we find $P(\mathbf{f}|\mathbf{x}^*, S)$. Finally, in A.3 we derive $P(f^*|\mathbf{x}^*, S)$.

A.1 $P(f^*|\mathbf{f}, \mathbf{x}^*, S)$

Define $\mathbf{k} \equiv [\mathcal{K}(x^*, x_1), \mathcal{K}(x^*, x_2), ..., \mathcal{K}(x^*, x_n)]^T$, which is the vector of applying some kernel function $\mathcal{K}$ on a single test point x^* with every training point x. Then the joint distribution of $[\mathbf{f} f^*]^T$ is:

$$\begin{bmatrix} \mathbf{f} \\ f^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K & \mathbf{k} \\ \mathbf{k}^T & \mathcal{K}(x^*, x^*) \end{bmatrix}\right) \tag{4}$$

Where K is the covariance matrix of the training points $K(\mathbf{X}, \mathbf{X})$. Since the conditions on $\mathbf{x}^*$ and S are already embedded in the covariance matrix in (4), $P(f^*|\mathbf{f}, \mathbf{x}^*, S)$ is equivalent to $P(f^*|\mathbf{f})$, giving us:

$$P(f^*|\mathbf{f}) = \frac{P(f^*, \mathbf{f})}{P(\mathbf{f})} \tag{5}$$

Let

$$\begin{bmatrix} A & \mathbf{b} \\ \mathbf{b}^T & c \end{bmatrix} \equiv \begin{bmatrix} K & \mathbf{k} \\ \mathbf{k}^T & \mathcal{K}(x^*, x^*) \end{bmatrix}^{-1} \tag{6}$$

The probability density of a normally distributed random variable X is defined as:

$$P(X) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(X-\mu)^2}{2\sigma^2}}$$

Where σ^2 is the variance of X, and μ the mean.
We can say $P(X)$ is proportional to:

$$\begin{aligned} P(X) &\propto e^{-\frac{(x-\mu)^2}{2\sigma^2}} \\ &= e^{-\frac{1}{2}(x-\mu)^2{\sigma^2}^{-1}} \\ &= e^{-\frac{1}{2}(x-\mu)\sigma^{-2}(x-\mu)} \end{aligned} \tag{P}$$

Then, from (5) and (P) we get:

$$
\begin{aligned}
P(f^*|\mathbf{f}) \propto \quad & \frac{\exp\left(-\frac{1}{2}\begin{bmatrix}\mathbf{f}\\ f^*\end{bmatrix}^T\begin{bmatrix}A & \mathbf{b}\\ \mathbf{b}^T & c\end{bmatrix}\begin{bmatrix}\mathbf{f}\\ f^*\end{bmatrix}\right)}{\exp\left(-\frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right)} \\
= \quad & \exp\left(-\frac{1}{2}\begin{bmatrix}\mathbf{f}\\ f^*\end{bmatrix}^T\begin{bmatrix}A & \mathbf{b}\\ \mathbf{b}^T & c\end{bmatrix}\begin{bmatrix}\mathbf{f}\\ f^*\end{bmatrix} - \left(-\frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right)\right) \\
= \quad & \exp\left(-\frac{1}{2}\begin{bmatrix}\mathbf{f}\\ f^*\end{bmatrix}^T\begin{bmatrix}A & \mathbf{b}\\ \mathbf{b}^T & c\end{bmatrix}\begin{bmatrix}\mathbf{f}\\ f^*\end{bmatrix} + \frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right) \\
= \quad & \exp\left(-\frac{1}{2}\left[\mathbf{f}^T A + f^*\mathbf{b}^T, \mathbf{f}\mathbf{b}^T + f^* c\right]\begin{bmatrix}\mathbf{f}\\ f^*\end{bmatrix} + \frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right) \\
= \quad & \exp\left(-\frac{1}{2}\left(\mathbf{f}(\mathbf{f}^T A + f^*\mathbf{b}^T) + f^*(\mathbf{f}\mathbf{b}^T + f^* c\right) + \frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right). \\
= \quad & \exp\left(-\frac{1}{2}\left(c(f^*)^2 + 2(\mathbf{b}^T\mathbf{f})f^* + \mathbf{f}^T A\mathbf{f}\right) + \frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right) \\
= \quad & \exp\left(-\frac{1}{2}c\left(f^* + \frac{\mathbf{b}^T\mathbf{f}}{c}\right)^2 + \frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right) \\
\propto \quad & \exp\left(-\frac{1}{2c^{-1}}\left(f^* + \frac{\mathbf{b}^T\mathbf{f}}{c}\right)^2\right). \qquad (7)
\end{aligned}
$$

From (6) we get:

$$
\begin{bmatrix}A & \mathbf{b}\\ \mathbf{b}^T & c\end{bmatrix} * \begin{bmatrix}K & \mathbf{k}\\ \mathbf{k}^T & \mathcal{K}(x^*, x^*)\end{bmatrix}^{-1} = I
$$

$$
\begin{bmatrix}A & \mathbf{b}\\ \mathbf{b}^T & c\end{bmatrix} * \begin{bmatrix}K & \mathbf{k}\\ \mathbf{k}^T & \mathcal{K}(x^*, x^*)\end{bmatrix}^{-1} = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}
$$

Which gives us:

$$
K\mathbf{b} + c\mathbf{k} = 0 \quad \text{and} \quad \mathbf{k}^T\mathbf{b} + c\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) = 1.
$$

We can rewrite this to:

$$
\begin{aligned}
K\mathbf{b} &= -c\mathbf{k} \\
\mathbf{b} &= K^{-1} * -c\mathbf{k} \qquad \text{(HelpB)}
\end{aligned}
$$

and,

$$
\begin{aligned}
c\mathbf{k} &= -K\mathbf{b} \\
c &= -\mathbf{k}^{-1}K\mathbf{b} \qquad \text{(HelpC)}
\end{aligned}
$$

and with that,

$$\begin{aligned} k^T\mathbf{b} + c\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) &= 1 \\ c\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) &= 1 - \mathbf{k}^T\mathbf{b} \\ c &= \frac{1 - \mathbf{k}^T\mathbf{b}}{\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)} \end{aligned}$$

Nog we plug in (HelpB) and get:

$$\begin{aligned} c &= \frac{1 + \mathbf{k}^T K^{-1}\mathbf{k}c}{\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)} \\ \frac{1 + \mathbf{k}^T K^{-1}\mathbf{k}c}{c} &= \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) \\ \frac{1}{c} + \mathbf{k}^T K^{-1}\mathbf{k} &= \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) \\ \frac{1}{c} &= \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k} \\ c &= \frac{1}{\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k}} \end{aligned} \tag{9}$$

We do the same for b:

$$\begin{aligned} \mathbf{k}^T\mathbf{b} + c\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) &= 1 \\ \mathbf{b} &= \frac{1 - c\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)}{\mathbf{k}^T} \end{aligned}$$

Now we plug in (HelpC) and get:

$$\begin{aligned} \mathbf{b} &= \frac{1 + \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)\mathbf{k}^{-1}K\mathbf{b}}{\mathbf{k}^T} \\ \frac{1 + \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)\mathbf{k}^{-1}K\mathbf{b}}{\mathbf{b}} &= \mathbf{k}^T \\ \frac{1}{\mathbf{b}} + \frac{\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)}{\mathbf{k}K^{-1}} &= \mathbf{k}^T \\ \frac{1}{\mathbf{b}} = \mathbf{k}^T - \frac{\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)}{\mathbf{k}K^{-1}} &= \frac{\mathbf{k}^T\mathbf{k}K^{-1}}{\mathbf{k}K^{-1}} - \frac{\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)}{\mathbf{k}K^{-1}} \\ &= \frac{\mathbf{k}^T\mathbf{k}K^{-1} - \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)}{\mathbf{k}K^{-1}} \\ \mathbf{b} &= \frac{\mathbf{k}K^{-1}}{\mathbf{k}^T K^{-1}\mathbf{k} - \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*)} \\ &= -\frac{K^{-1}\mathbf{k}}{\mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) + \mathbf{k}^T K^{-1}\mathbf{k}} \end{aligned} \tag{8}$$

We can plug in (8) and (9) into (7) and get:

$$
\begin{aligned}
P(f^*|f) &\propto \exp\left(-\frac{1}{2c^{-1}}\left(f^* + \frac{\mathbf{b}^T\mathbf{f}}{c}\right)^2\right) \\
&= \exp\left(-\frac{1}{2(\mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k})}\left(f^* + \mathbf{b}^T\mathbf{f}(\mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k})\right)^2\right) \\
&= \exp\left(-\frac{1}{2(\mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k})}\left(f^* - \frac{K^{-1}\mathbf{k}^T\mathbf{f}(\mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k})}{\mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k}}\right)^2\right) \\
&= \exp\left(-\frac{1}{2(\mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k})}\left(f^* - \frac{K^{-1}\mathbf{k}^T\mathbf{f}}{1}\right)^2\right) \\
&= \exp\left(-\frac{(f^* - \mathbf{k}^T K^{-1}\mathbf{f})^2}{2(\mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k})}\right)
\end{aligned}
$$

Which is again of the form:

$$e^{-\frac{(x-\mu)^2}{2\sigma}}$$

And we can thus find $(f^*|f)$ as:

$$(f^*|\mathbf{f}) \sim \mathcal{N}(\mathbf{k}^T K^{-1}\mathbf{f}, \quad \mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k}). \tag{10}$$

A.2 $P(\mathbf{f}|\mathbf{x}^*, S)$

$\mathbf{f}$ does not depend on $\mathbf{x}^*$, so all we need to do is find $P(\mathbf{f}|S)$.

$$
\begin{aligned}
P(\mathbf{f}|S) &\propto P(S|\mathbf{f})P(\mathbf{f}) && \text{(Bayes rule)} \\
&= P(\mathbf{y},\mathbf{x}|\mathbf{f})P(\mathbf{f}) \\
&= P(\mathbf{y}|\mathbf{x},\mathbf{f})P(\mathbf{x}|\mathbf{f})P(\mathbf{f}) && (11) \\
&\propto P(\mathbf{y}|\mathbf{f})P(\mathbf{f}) && (12)
\end{aligned}
$$

In (11) we use that $P(\mathbf{y}|\mathbf{x},\mathbf{f}) = P(\mathbf{y}|\mathbf{f})$, as $\mathbf{y}$ is independent of $\mathbf{x}$ given $\mathbf{f}$. Additionally, $P(\mathbf{x}|\mathbf{f}) = P(\mathbf{x})$, as $\mathbf{x}$ is not dependent on $\mathbf{f}$. $P(\mathbf{x})$ is assumed to have a uniform distribution.
Combining (1), (2), (12) and letting:

$$
\begin{aligned}
\Sigma &= (K^{-1} + \sigma^{-2}I)^{-1} \\
\mathbf{u} &= \sigma^{-2}\Sigma\mathbf{y}
\end{aligned}
$$

We find

$$
\begin{aligned}
P(\mathbf{f}|S) &\propto \exp\left(-\frac{(\mathbf{y}-\mathbf{f})^T(\mathbf{y}-\mathbf{f})}{2\sigma^2}\right) * \exp\left(-\frac{\mathbf{f}^T K^{-1}\mathbf{f}}{2}\right) \\
&= \exp\left(-\frac{(\mathbf{y}-\mathbf{f})^T(\mathbf{y}-\mathbf{f})}{2\sigma^2} - \frac{\mathbf{f}^T K^{-1}\mathbf{f}}{2}\right) \\
&= \exp\left(-\frac{\sigma^{-2}(\mathbf{y}^T\mathbf{y} - \mathbf{y}^T\mathbf{f} - \mathbf{f}^t\mathbf{y} + \mathbf{f}^t\mathbf{f})}{2} - \frac{\mathbf{f}^T K^{-1}\mathbf{f}}{2}\right) \\
&= \exp\left(-\frac{\sigma^{-2}\mathbf{y}^T\mathbf{y} - \sigma^{-2}\mathbf{y}^T\mathbf{f} - \sigma^{-2}\mathbf{f}^T\mathbf{y} + \sigma^{-2}\mathbf{f}^T\mathbf{f} - \mathbf{f}^T K^{-1}\mathbf{f}}{2}\right) \\
&\propto \exp\left(-\frac{-\sigma^{-2}\mathbf{y}^T\mathbf{f} - \sigma^{2}\mathbf{f}^T\mathbf{y} + \sigma^{2}\mathbf{f}^T\mathbf{f} - \mathbf{f}^T K^{-1}\mathbf{f}}{2}\right) \\
&\propto \exp\left(-\frac{-2\sigma^{-2}\mathbf{y}^T\mathbf{f} + \sigma^{2}\mathbf{f}^T\mathbf{f} - \mathbf{f}^T K^{-1}\mathbf{f}}{2}\right) \\
&= \exp\left(-\frac{\mathbf{f}^T(K^{-1} + \sigma^{-2}I)\mathbf{f} - 2\sigma^{-2}\mathbf{y}^T\mathbf{f}}{2}\right) \\
&= \exp\left(-\frac{\mathbf{f}^T\Sigma^{-1}\mathbf{f}}{2} - \frac{2\sigma^{-2}\mathbf{y}^T\mathbf{f}}{2}\right) \\
&= \exp\left(-\frac{\mathbf{f}^T\Sigma^{-1}\mathbf{f}}{2} - \frac{2\mathbf{u}\mathbf{f}}{2\Sigma}\right) \\
&= \exp\left(-\frac{\mathbf{f}^T\mathbf{f}}{2\Sigma} - \frac{2\mathbf{u}\mathbf{f}}{2\Sigma}\right) \\
&= \exp\left(-\frac{\mathbf{f}^T\mathbf{f} - 2\mathbf{u}\mathbf{f}}{2\Sigma}\right) \\
&\propto \exp\left(-\frac{(\mathbf{f}-\mathbf{u})^2}{2\Sigma}\right) \\
&= \exp\left(-\frac{(\mathbf{f}-\mathbf{u})\Sigma^{-1}(\mathbf{f}-\mathbf{u})}{2}\right)
\end{aligned}
$$

where

$$
\begin{aligned}
\Sigma &= (K^{-1} + \sigma^{-2}I)^{-1} \\
&= (K^{-1} + \sigma^{-2}KK^{-1})^{-1} \\
&= \left((I + \sigma^{-2}K)K^{-1}\right)^{-1} \\
&= \frac{1}{(I + \sigma^{-2}K)K^{-1}} \\
&= \frac{K}{I + \sigma^{-2}K} \\
&= \frac{\sigma^2 K}{\sigma^2 I + K} \\
&= \sigma^2 K(K + \sigma^2 I)^{-1}
\end{aligned}
$$

and

$$\begin{aligned}\mathbf{u} &= \sigma^{-2}\Sigma\mathbf{y} \\ &= \sigma^{-2}(\sigma^2 K(K+\sigma^2 I)^{-1})\mathbf{y} \\ &= K(K+\sigma^2 I)^{-1}\mathbf{y}\end{aligned}$$

Which gives us:

$$(\mathbf{f}|S) \sim \mathcal{N}\left(K(K+\sigma^2 I)^{-1}\mathbf{y}, \quad \sigma^2 K(K+\sigma^2 I)^{-1}\right). \tag{13}$$

A.3 $P(f^*|\mathbf{x}^*, S)$

For ease of reading, we define:

$$\begin{aligned}\mathbf{a} &\equiv K^{-1}\mathbf{k}, \\ \Delta &\equiv \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{k}, \\ \mathbf{b} &\equiv K(K+\sigma^2 I)^{-1}\mathbf{y}, \\ \Sigma &\equiv \sigma^2 K(K+\sigma^2 I)^{-1}.\end{aligned}$$

We use (3), (10) and (13) to find $P(f^*|\mathbf{x}^*, S)$:

$$
\begin{aligned}
\propto & \int \exp\left(-\frac{(\mathbf{f}^* - \mathbf{k}^T K^{-1}\mathbf{f})^2}{2\mathcal{K}(\mathbf{x}^*,\mathbf{x}^*) - \mathbf{k}^T K^{-1}\mathbf{f}}\right) * \exp\left(-\frac{(f - K(K+\sigma^2 I)^{-1}\mathbf{y})^2}{2\sigma^2 K(K+\sigma^2 I)^{-1}}\right) d\mathbf{f} \\
= & \int \exp\left(-\frac{(f^* - \mathbf{a}^T\mathbf{f})^2}{2\Delta}\right) * \exp\left(-\frac{(\mathbf{f}-\mathbf{b})^2}{2\Sigma}\right) d\mathbf{f} \\
= & \int \exp\left(-\frac{(f^* - \mathbf{a}^T\mathbf{f})^2}{2\Delta} - \frac{(\mathbf{f}-\mathbf{b})^T\Sigma^{-1}(\mathbf{f}-\mathbf{b})}{2}\right) d\mathbf{f} \\
= & \int \exp\left(-\frac{(f^*)^2 - 2f^*\mathbf{a}^T\mathbf{f} + \mathbf{a}^T\mathbf{f}\mathbf{f}^T\mathbf{a}}{2\Delta} - \frac{1}{2}(\mathbf{f}-\mathbf{b})^T\Sigma^{-1}(\mathbf{f}-\mathbf{b})\right) d\mathbf{f} \\
= & \int \exp\left(-\frac{(f^*)^2}{2\Delta} - \frac{1}{2}\left((\mathbf{f}-\mathbf{b}^T)\Sigma^{-1}(\mathbf{f}-\mathbf{b}) - \frac{2f^*\mathbf{a}^T\mathbf{f}}{\Delta} + \mathbf{f}^T\frac{\mathbf{a}\mathbf{a}^T}{\Delta}\mathbf{f}\right)\right) d\mathbf{f} \\
= & \int \exp\left(-\frac{(f^*)^2}{2\Delta} - \frac{1}{2}\left(\mathbf{f}^T\Sigma^{-1}\mathbf{f} - \mathbf{f}^T\Sigma^{-1}\mathbf{b} - \mathbf{b}^T\Sigma^{-1}\mathbf{f} + \mathbf{b}^T\Sigma^{-1}\mathbf{b} - \frac{2f^*\mathbf{a}^T\mathbf{f}}{\Delta} + \mathbf{f}^T\frac{\mathbf{a}\mathbf{a}^T}{\Delta}\mathbf{f}\right)\right) d\mathbf{f} \\
\propto & \int \exp\left(-\frac{(f^*)^2}{2\Delta} - \frac{1}{2}\left(\mathbf{f}^T\Sigma^{-1}\mathbf{f} - 2\Sigma^{-1}\mathbf{b}^T\mathbf{f} - \frac{2f^*\mathbf{a}^T\mathbf{f}}{\Delta} + \mathbf{f}^T\frac{\mathbf{a}\mathbf{a}^T}{\Delta}\mathbf{f}\right)\right) d\mathbf{f} \\
= & \int \exp\left(-\frac{(f^*)^2}{2\Delta} - \frac{1}{2}\left(\mathbf{f}^T(\Sigma^{-1} + \frac{\mathbf{a}\mathbf{a}^T}{\Delta})\mathbf{f} - 2(\Sigma^{-1}\mathbf{b} + \frac{f^*\mathbf{a}}{\Delta})^T\mathbf{f}\right)\right) d\mathbf{f} \\
= & \exp\left(\frac{(f^*)^2}{2\Delta} + \frac{1}{2}\left(\frac{f^*\mathbf{a}}{\Delta} + \Sigma^{-1}\mathbf{b}\right)^T\left(\Sigma^{-1} + \frac{\mathbf{a}\mathbf{a}^T}{\Delta}\right)^{-1}\left(\frac{f^*\mathbf{a}}{\Delta} + \Sigma^{-1}\mathbf{b}\right)\right) \\
= & \exp\left(\begin{array}{l} -\frac{(f^*)^2}{2\Delta} + \frac{1}{2}(\frac{f^*\mathbf{a}^T}{\Delta}\Sigma^{-1}\frac{f^*\mathbf{a}}{\Delta} + \frac{f^*\mathbf{a}^T}{\Delta}(\frac{\mathbf{a}\mathbf{a}^T}{\Delta})^{-1}\frac{f^*\mathbf{a}}{\Delta} + 2\Sigma^{-1}\mathbf{b}^T(\Sigma^{-1})^{-1}\frac{f^*\mathbf{a}}{\Delta} + \\ 2\Sigma^{-1}\mathbf{b}^T(\frac{\mathbf{a}\mathbf{a}^T}{\Delta})^{-1}\frac{f^*\mathbf{a}^T}{\Delta} + \Sigma^{-1}\mathbf{b}^T\Sigma^{-1}\mathbf{b}\Sigma^{-1^{-1}} + \Sigma^{-1}\mathbf{b}^T\Sigma^{-1}\mathbf{b}\left(\frac{\mathbf{a}\mathbf{a}^T}{\Delta}\right)^{-1}) \end{array}\right) \\
\propto & \exp\left(\begin{array}{l} -\frac{(f^*)^2}{2\Delta} + \frac{1}{2}(\frac{f^*\mathbf{a}^T}{\Delta}\Sigma^{-1}\frac{f^*\mathbf{a}}{\Delta} + \frac{f^*\mathbf{a}^T}{\Delta}(\frac{\mathbf{a}\mathbf{a}^T}{\Delta})^{-1}\frac{f^*\mathbf{a}}{\Delta} + 2\Sigma^{-1}\mathbf{b}^T(\Sigma^{-1})^{-1}\frac{f^*\mathbf{a}}{\Delta} + \\ 2\Sigma^{-1}\mathbf{b}^T(\frac{\mathbf{a}\mathbf{a}^T}{\Delta})^{-1}\frac{f^*\mathbf{a}^T}{\Delta}) \end{array}\right) \\
= & \exp\left(-\frac{(f^*)^2}{2\Delta} + \frac{1}{2}\left(\frac{\mathbf{a}^T(\Delta\Sigma^{-1})^{-1}\mathbf{a}}{\Delta} + \frac{(f^*)^2\mathbf{a}\mathbf{a}^T*(\mathbf{a}\mathbf{a}^T)^{-1}}{\Delta} + 2\mathbf{a}^T(\Delta\Sigma^{-1})^{-1}\Sigma^{-1}\mathbf{b}f^* + 2\mathbf{a}^T(\mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}f^*\right)\right) \\
= & \exp\left(-\frac{1}{2}\left(\frac{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}{\Delta}(f^*)^2 - 2\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}f^*\right)\right). \\
= & \exp\left(-\frac{1}{2}\left(\frac{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}{\Delta}(f^*)^2 - 2\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}f^*\right)\right) \\
= & \exp\left(-\frac{1}{2}\left(\frac{(1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a})(f^*)^2}{\Delta} - \frac{2\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{b}f^*}{\Sigma}\right)\right) \\
= & \exp\left(-\frac{1}{2}\left(\frac{\Sigma(1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a})(f^*)^2 - 2\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{b}f^*\Delta}{\Sigma\Delta}\right)\right) \\
= & \exp\left(-\frac{1}{2}\left(\frac{(1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a})(f^*)^2 - 2\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}f^*\Delta}{\Delta}\right)\right) \\
= & \exp\left(-\frac{1}{2}\left(\frac{(f^*)^2 - 2\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}f^*\Delta(1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a})^{-1}}{\Delta(1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a})^{-1}}\right)\right) \\
= & \exp\left(-\frac{1}{2}\left(\frac{(f^*)^2 - 2\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}f^*\Delta(1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a})^{-1}}{\frac{\Delta}{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}}\right)\right) \\
= & \exp\left(-\frac{1}{2}\left(\frac{(f^*)^2 - 2\frac{\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}f^*\Delta}{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}}{\frac{\Delta}{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}}\right)\right) \\
\propto & \exp\left(-\frac{1}{2}\left(\frac{(f^* - \Delta\mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b})^2}{\frac{\Delta}{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}}\right)\right)
\end{aligned}
$$

We find

$$(f^*|\mathbf{x}^*, S) \sim \mathcal{N}(\mu^*, (\sigma^*)^2)$$

where

$$\mu^* = \frac{\Delta \mathbf{a}^T (\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}}{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}$$
$$(\sigma^*)^2 = \frac{\Delta}{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}$$

Using the *Sherman-Morrison-Woodbury formula*:

$$(A + UV^T)^{-1} = A^{-1} - A^{-1}U(I + V^T A^{-1} U)^{-1} V^T A^{-1}, \tag{14}$$

we find

$$\begin{aligned}
(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1} &= \frac{\Sigma}{\Delta} - \frac{\Sigma}{\Delta}\mathbf{a}\left(1 + \frac{\mathbf{a}^T\Sigma\mathbf{a}}{\Delta}\right)^{-1}\mathbf{a}^T\frac{\Sigma}{\Delta} \\
&= \frac{\Sigma}{\Delta} - \frac{\Sigma}{\Delta}\mathbf{a}\left(\frac{\Delta + \mathbf{a}^T\Sigma\mathbf{a}}{\Delta}\right)^{-1}\mathbf{a}^T\frac{\Sigma}{\Delta} \\
&= \frac{\Sigma}{\Delta} - \frac{\Sigma}{\Delta}\mathbf{a}\left(\frac{\Delta}{\Delta + \mathbf{a}^T\Sigma\mathbf{a}}\right)\mathbf{a}^T\frac{\Sigma}{\Delta} \\
&= \frac{\Sigma}{\Delta} - \frac{\Sigma\mathbf{a}\Delta\mathbf{a}^T\Sigma}{\Delta^2(\delta + \mathbf{a}^T\Sigma\mathbf{a})} \\
&= \frac{\Sigma}{\Delta} - \frac{\Sigma\mathbf{a}\mathbf{a}^T\Sigma}{\Delta(\Delta + \mathbf{a}\mathbf{a}^T)\Sigma} \\
&= \frac{1}{\Delta}\left(\Sigma - \frac{\Sigma\mathbf{a}\mathbf{a}^T\Sigma}{\Delta + \mathbf{a}^T\Sigma\mathbf{a}}\right)
\end{aligned}$$

substituting this in μ^*:

$$\mu^* = \frac{\Delta \mathbf{a}^T (\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\Sigma^{-1}\mathbf{b}f^*}{1 - \mathbf{a}^T(\Delta\Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1}\mathbf{a}}$$

gives

$$\begin{aligned}
\mu^* &= \frac{\mathbf{a}^T \left(\Sigma - \frac{\Sigma \mathbf{a}\mathbf{a}^T \Sigma}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}\right) \Sigma^{-1} \mathbf{b}}{1 - \frac{\mathbf{a}^T}{\Delta}\left(\Sigma - \frac{\Sigma \mathbf{a}\mathbf{a}^T \Sigma}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}\right)\mathbf{a}} \\
&= \mathbf{a}^T \mathbf{b} \frac{\frac{1}{\Sigma} - \left(\Sigma - \frac{\Sigma \mathbf{a}\mathbf{a}^T \Sigma}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}\right)}{1 - \frac{\mathbf{a}^T}{\Delta}\left(\Sigma - \frac{\Sigma \mathbf{a}\mathbf{a}^T \Sigma}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}\right)\mathbf{a}} \\
&= \mathbf{a}^T \mathbf{b} \frac{1 - \frac{\mathbf{a}\mathbf{a}^T \Sigma}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}}{1 - \frac{\mathbf{a}^T}{\Delta}\left(\Sigma - \frac{\Sigma \mathbf{a}\mathbf{a}^T \Sigma}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}\right)\mathbf{a}} \\
&= \mathbf{a}^T \mathbf{b} \frac{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}}{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}}{\Delta} + \frac{(\mathbf{a}^T \Sigma \mathbf{a}^2)}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} \\
&= \mathbf{a}^T \mathbf{b} \frac{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}}{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})} + \frac{(\mathbf{a}^T \Sigma \mathbf{a}^2)}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} \\
&= \mathbf{a}^T \mathbf{b} \frac{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}}{1 - \frac{(\mathbf{a}^T \Sigma \mathbf{a})\Delta}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} \\
&= \mathbf{a}^T \mathbf{b} \frac{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}}{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}} = \mathbf{a}^T \mathbf{b} \\
&= K^{-1} \mathbf{k}^T K (K + \sigma^2 I)^{-1} \mathbf{y} \\
&= \mathbf{k}^T (K + \sigma^2 I)^{-1} \mathbf{y} \\
&= \mathbf{k}^T (K + \sigma^2 I)^{-1} \mathbf{y}
\end{aligned}$$

and

$$(\sigma^*)^2 = \frac{\Delta}{1 - \mathbf{a}^T (\Delta \Sigma^{-1} + \mathbf{a}\mathbf{a}^T)^{-1} \mathbf{a}}$$

becomes

$$
\begin{aligned}
(\sigma^*)^2 &= \frac{\Delta}{1 - \mathbf{a}^T \left(\frac{1}{\Delta} \left(\Sigma - \frac{\Sigma \mathbf{a} \mathbf{a}^T \Sigma}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}} \right) \right) \mathbf{a}} \\
&= \frac{\Delta}{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}}{\Delta} + \frac{\mathbf{a}^T \Sigma \mathbf{a} \mathbf{a}^T \Sigma \mathbf{a}}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} \\
&= \frac{\Delta}{1 - \frac{\mathbf{a}^T \Sigma \mathbf{a}}{\Delta} + \frac{(\mathbf{a}^T \Sigma \mathbf{a})^2}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} \\
&= \frac{\Delta}{\frac{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a}) - \mathbf{a}^T \Sigma \mathbf{a}(\Delta + \mathbf{a}^T \Sigma \mathbf{a}) + (\mathbf{a}^T \Sigma \mathbf{a})^2}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} \\
&= \frac{\Delta}{\frac{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a}) - \Delta(\mathbf{a}^T \Sigma \mathbf{a})}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} = \frac{\Delta}{\frac{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a} - \mathbf{a}^T \Sigma \mathbf{a})}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} \\
&= \frac{\Delta}{\frac{\Delta^2}{\Delta(\Delta + \mathbf{a}^T \Sigma \mathbf{a})}} = \frac{\Delta}{\frac{\Delta}{\Delta + \mathbf{a}^T \Sigma \mathbf{a}}} \\
&= \frac{\Delta}{\Delta * (\Delta + \mathbf{a}^T \Sigma \mathbf{a})^{-1}} = \Delta + \mathbf{a}^T \Sigma \mathbf{a} \\
&= \Delta + K^{-1} \mathbf{k}^T \sigma^2 K (K + \sigma^2 I)^{-1} K^{-1} \mathbf{k} \\
&= \Delta + \frac{\sigma^2 K \mathbf{k}^T \mathbf{k}}{K^2 (K + \sigma^2 I)} = \Delta + \frac{\sigma^2 \mathbf{k}^T \mathbf{k}}{K(K + \sigma^2 I)} \\
&= \Delta + \frac{\sigma^2 \mathbf{k}^T \mathbf{k}}{KK + \sigma^2 K} \\
&= \Delta + \sigma^2 \mathbf{k}^T (\sigma^2 K + KK)^{-1} \mathbf{k} \qquad (15) \\
&= \Delta + \sigma^2 \mathbf{k}^T (\sigma^{-2} K^{-1} - \sigma^{-2} K^{-1} K (I + K \sigma^{-2} K^{-1} K)^{-1} K \sigma^{-2} K^{-1}) \mathbf{k} \qquad (16) \\
&= \Delta + \sigma^2 \mathbf{k}^T (\sigma^{-2} K^{-1} - \sigma^{-2} (I + \sigma^{-2} K)^{-1} \sigma^{-2}) \mathbf{k} \\
&= \Delta + \sigma^2 \mathbf{k}^T (\sigma^{-2} K^{-1} - \sigma^{-4} (K \sigma^{-2} + I)^{-1}) \mathbf{k} \\
&= \Delta + \sigma^2 \mathbf{k}^T (\sigma^{-2} K^{-1} - (K^{-1} \sigma^{-2} + \sigma^{-4} I)) \mathbf{k} \\
&= \Delta + (\mathbf{k}^T K^{-1} - \mathbf{k}^T (K + \sigma^2 I)^{-1}) \mathbf{k} \\
&= \Delta + \mathbf{k}^T K^{-1} \mathbf{k} - \mathbf{k}^T (K + \sigma^2 I)^{-1} \mathbf{k} \\
&= \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}^T K^{-1} \mathbf{k} + \mathbf{k}^T K^{-1} \mathbf{k} - \mathbf{k}^T (K + \sigma^2 I)^{-1} \mathbf{k} \\
&= \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}^T (K + \sigma^2 I)^{-1} \mathbf{k}
\end{aligned}
$$

Finally, we have the predictive distribution $(f^* | \mathbf{x}^*, S)$:

$$
\begin{aligned}
(f^* | \mathbf{x}^*, S) &\sim \mathcal{N}\big(\mu^*, (\sigma^*)^2\big) \\
&\sim \mathcal{N}\Big(\mathbf{k}^T (K + \sigma^2 I)^{-1} \mathbf{y}, \quad \mathcal{K}(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}^T (K + \sigma^2 I)^{-1} \mathbf{k}\Big).
\end{aligned}
$$