PSFC/JA-14-22

Improved profile fitting and quantification of uncertainty in experimental measurements of impurity transport coefficients using Gaussian process regression

Chilenski, M.A., Greenwald, M., Marzouk, Y.[1], Howard, N.T., White, A.E., Rice, J.E., Walk, J.R.

[1]Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139, USA

September, 2014

Plasma Science and Fusion Center
Massachusetts Institute of Technology
Cambridge MA 02139 USA

This work was supported by the U.S. Department of Energy, Grant No. DE-FC02-99ER54512.

Improved profile fitting and quantification of uncertainty in experimental measurements of impurity transport coefficients using Gaussian process regression

M A Chilenski[1], M Greenwald[1], Y Marzouk[2], N T Howard[1], A E White[1], J E Rice[1], and J R Walk[1]

[1] Plasma Science and Fusion Center, Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139, USA

[2] Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139, USA

E-mail: markchil@mit.edu

Abstract. The need to fit smooth temperature and density profiles to discrete observations is ubiquitous in plasma physics, but the prevailing techniques for this have many shortcomings that cast doubt on the statistical validity of the results. This issue is amplified in the context of validation of gyrokinetic transport models (Holland et al. 2009, *Phys. Plasmas* **16**, 052301), where the strong sensitivity of the code outputs to input gradients means that inadequacies in the profile fitting technique can easily lead to an incorrect assessment of the degree of agreement with experimental measurements. In order to rectify the shortcomings of standard approaches to profile fitting, we have applied Gaussian process regression (GPR), a powerful nonparametric regression technique, to analyze an Alcator C-Mod L-mode discharge used for past gyrokinetic validation work (Howard et al. 2012, *Nucl. Fusion* **52**, 063002). We show that the GPR techniques can reproduce the previous results while delivering more statistically rigorous fits and uncertainty estimates for both the value and the gradient of plasma profiles with an improved level of automation. We also discuss how the use of GPR can allow for dramatic increases in the rate of convergence of uncertainty propagation for any code that takes experimental profiles as inputs. The new GPR techniques for profile fitting and uncertainty propagation are quite useful and general, and we describe the steps to implementation in detail in this paper. These techniques have the potential to substantially improve the quality of uncertainty estimates on profile fits and the rate of convergence of uncertainty propagation, making them of great interest for wider use in fusion experiments and modeling efforts.

PACS numbers: 02.50.Cw, 02.50.Ey, 02.50.Fz, 02.50.Tt, 02.60.Ed, 02.60.Jh, 02.70.Rr, 02.70.Uu, 07.05.Kf, 52.25.Vy, 52.30.Gz, 52.55.Fa, 52.70.Kz, 52.70.La, 52.65.Pp

1. Introduction

A situation that is ubiquitous in plasma physics and many other fields is that a quantity of interest is computed by a complicated, computationally expensive code whose inputs are not single values but rather entire profiles of quantities given as functions of space, time and possibly other independent variables. In plasma physics, examples of these derived quantities include heat fluxes and particle diffusivities. As many quantities of interest and processes such as transport depend strongly on the gradient of a measured profile, it is critical that the process of taking noisy, discrete observations and turning them into a smooth curve be done in a rigorous, statistically principled way. This is particularly true in the context of validation of gyrokinetic codes [1]: if a statistically meaningful comparison between the code and experiment is to be performed, then the high sensitivity of turbulent transport to profile gradients means that experimental data must be analyzed very carefully to deliver valid uncertainty estimates on the gradient scale lengths, as well as other derived experimental quantities to be compared such as heat and particle fluxes. Furthermore, for the results of the analysis to be complete they must include an estimate of their uncertainty, so it is desirable that the fit be performed in a way that enables the uncertainty to be propagated through the model with a minimal number of code runs. Splines [2, 3], the traditional tool for this profile fitting and sampling task, have a number of shortcomings with respect to these objectives that will be discussed in this paper. We show that improvements in the quality of results, rate of convergence and level of automation of the data analysis workflow can be obtained by instead fitting profiles and producing samples using Gaussian process regression (GPR) [4]. As a profile fitting approach, GPR is very general, and can be applied in any situation where it is necessary to fit a smooth curve to noisy, discrete observations – even if the profile is a function of many independent variables. The sampling workflow presented in this paper is also quite general, and can be applied to any code that takes entire profiles as inputs.

In this paper, GPR is used in an analysis workflow built around the STRAHL code [5, 6, 7, 8] to obtain experimental estimates of impurity transport coefficients D and V from measurements of impurity brightness, electron temperature and electron density profiles. This measurement is of interest as impurity transport is critical in determining the power balance of a confined plasma [9], and acts as an additional channel for comparison when testing transport codes [6, 7].

The rest of this paper is organized as follows: section 2 discusses the very general problem of quantifying uncertainty in code outputs when entire smoothed profiles are required as an input and motivates the need for advanced profile fitting. Section 3 presents the basic principles of GPR. Section 4 shows GPR fits to real n_e and T_e profile data from Alcator C-Mod. Section 5 then uses random samples drawn from these GPR fits to quantify the uncertainty in experimental impurity transport coefficients inferred using the STRAHL code. Section 6 summarizes the work and presents the conclusions reached. Appendix A gives a review of the mathematical properties of splines to help

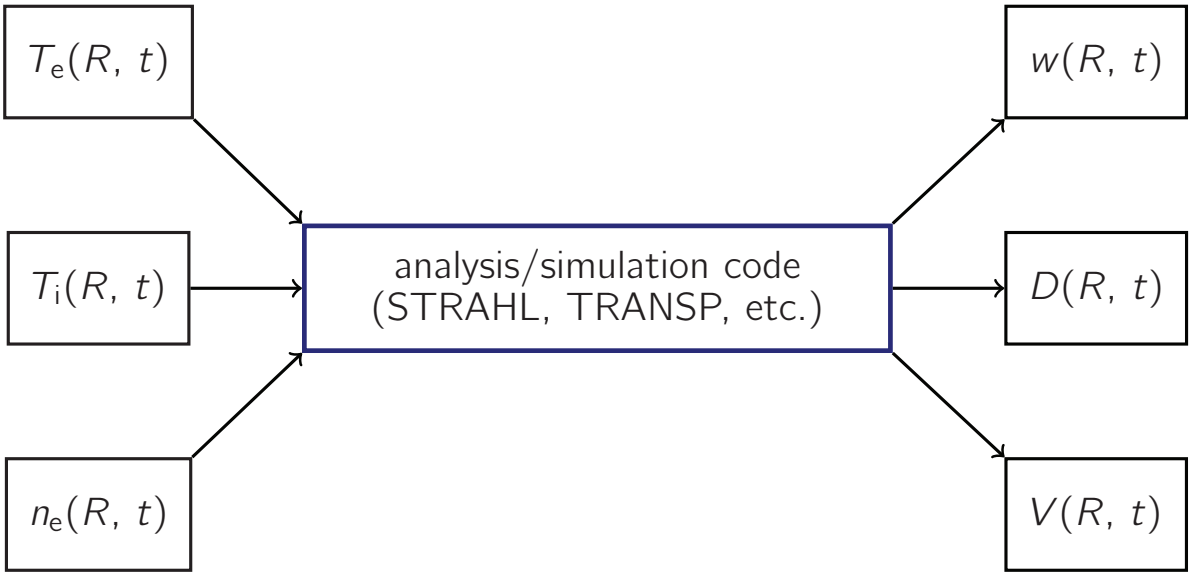


Figure 1: Typical analysis scheme: the analysis code requires complete profiles of quantities that are only measured at discrete points in space and time. The outputs can in general also be functions of space and time. Here, w refers to an arbitrary profile output from the code. Transport coefficients D and V are explicitly specified as outputs as they are the quantities of interest for the analysis in this paper.

set the stage for the advantages GPR offers and Appendix B gives an overview of the remarkably simple mathematics underlying GPR.

2. Uncertainty quantification and the need for advanced profile fitting

2.1. Uncertainty quantification with profile inputs

The situation this paper is concerned with is shown schematically in figure 1: a code takes as inputs one or more profiles and computes one or more output quantities from these profiles. Furthermore, even if the required input is the local value of a gradient, the entire profile must still be analyzed to obtain this result from the discrete experimental measurements of the profile. In order to fully specify the result of the code it is necessary to compute not just a point estimate of the output but also to provide an estimate of the uncertainty in the output and its sensitivity to the input parameters. This task is most often accomplished with techniques such as Monte Carlo sampling: a series of input samples is prepared by randomly perturbing the input profiles according to their respective uncertainty estimates. These samples are then run through the code to produce an ensemble of possible realizations of the outputs. Computing the relevant summary statistics of this ensemble then gives the estimate of the value and its uncertainty. This workflow is shown schematically in figure 2. To carry this workflow out in practice when the inputs are noisy, discrete observations it is necessary to have a fitting procedure that takes the observations and produces an estimate of the underlying smooth curve (and potentially its derivatives) and the accompanying uncertainty in a

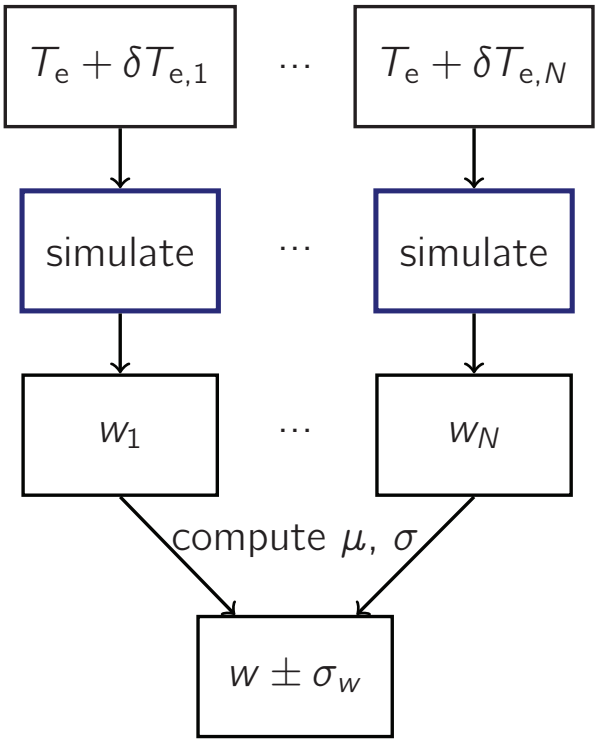


Figure 2: Overview of sampling based ("Monte Carlo") uncertainty propagation. The fitted profile T_e is perturbed N times by random (or otherwise intelligently selected) amounts δT_e. This produces N possible realizations of the output quantity w. The relevant summary statistics are computed from this ensemble to give the final estimate of the quantity and its uncertainty. N must be selected such that these estimates are sufficiently accurate.

way that perturbed samples can be extracted.

Furthermore, models for turbulence-induced transport are highly sensitive to the gradient scale lengths, given here as the normalized (against the minor radius a) inverse scale length for T_e (but which could in general be for T_e, T_i, n_e, etc.):

$$\frac{a}{L_{T_e}} = a\frac{|\nabla T_e|}{T_e} \approx a\frac{\partial T_e/\partial R}{T_e}, \tag{1}$$

where R refers to the mapped midplane major radius. Because this depends on the derivative $\partial/\partial R$, it is inevitably highly sensitive to the fine details of the profile. This sensitivity means it is essential to fit the discrete observations using a mathematically principled procedure, avoiding the temptation to pick the properties of the smoothing curve "by eye."

2.2. Profile fitting with splines

A very common approach at present is to use a spline to fit a smooth curve to experimental data ([10] and the references therein give a mathematical perspective, [11] shows a more recent application including gradient scale lengths). In order to highlight the advantages of the approach employed for this paper, a brief outline of

the mathematical properties of splines is given in Appendix A. Full details and further references can be found in [2, 3]. Splines have the advantage of being thoroughly explored in a large body of literature and routines for performing spline fitting are readily available in most programming languages commonly used for scientific data analysis. There is, however, a number of drawbacks that the Gaussian process approach employed in this paper overcomes.

With splines, selection of how flexible/complex the curve should be is a difficult problem. Lee [12] presents and compares a number of approaches for performing this operation, but the general theme is that this is a rather involved process, with Dierckx [3] admitting that the positioning of knots often becomes a matter of (manual) trial and error. Holland [1] comments on manual choice of spline properties as a potentially substantial source of systematic error in tokamak profile fits. Free-knot splines (see Appendix A) additionally suffer from the so-called "lethargy property" which means that there will be many local minima to contend with when optimizing the knot positions [13, 14]. As will be seen, the approach adopted for this paper selects the properties of the fitted curve using basic statistical procedures.

A further problem arises when attempting to fit data which depend on more than one independent variable. The most common choice when using splines on multivariate data is the tensor product spline [3], but this has the disadvantage of requiring that the knots fill a rectangular grid, which can present problems depending on the nature of the data to be fit. A further problem encountered is that most readily available implementations only support bivariate data. In contrast, the approach used in this paper can work on data of arbitrary dimension with little to no modification.

Confidence intervals for spline fits are discussed widely in the literature, including [15, 16, 17, 18, 19, 20], though the most common software packages fall short of providing an implementation of these extra steps. Literature regarding uncertainties in derivatives of splines appears to be far more scarce, but includes [21, 22, 23]. There has been some work to provide confidence bands on the gradients of plasma profiles estimated using so-called exponential splines in [24, 25, 26]. A simple approach that is widespread in plasma physics is to perform Monte Carlo sampling to obtain uncertainty estimates on the fit and its gradients, such as was done in [11]. In contrast to the mathematical constructions in the preceding references or the brute force application of Monte Carlo sampling, the Gaussian process regression approach used in this paper is based directly on the properties of the multivariate normal distribution, and therefore permits an intuitive interpretation of the variance of the fitted curve and its derivatives.

3. Profile fitting with Gaussian process regression

3.1. Basic details of Gaussian process regression

Gaussian process regression (GPR) is a general-purpose Bayesian nonparametric regression technique [4, 27]. Here, nonparametric refers to the fact that the observations

must be used in order to make a prediction and that a specific functional form is not assumed: the form of the fit is left exceedingly flexible so that the data themselves can give rise to the correct shape in a statistically rigorous manner. As discussed in [2, 4], there is in fact a very deep mathematical connection between GPR and splines – smoothing splines are simply a special case of GPR with a specific choice of prior distribution. The difference is that GPR is cast in a statistical framework that makes interpretation of the fit, its gradients and the associated uncertainties far more straightforward. Furthermore, GPR can be used to yield a low-dimensional representation of the profile uncertainty that enables the use of efficient uncertainty propagation techniques such as sparse quadrature [28] that can dramatically reduce the number of code runs necessary to propagate the uncertainty through a computationally expensive analysis code. Gaussian process regression has been in use in one form or another for many years under the term "kriging" [29], though the term Gaussian process regression is preferred here as it emphasizes the full statistical framework in which the approach is cast. Appendix B presents the full mathematical details of GPR, starting from a basic example of inferring a single value given a single observation. The concepts and equations essential to the following sections are presented here, following the nomenclature of [4] with additional references given as needed.

The essence of Gaussian process regression is that all observations and predictions are related through a multivariate normal distribution with a given mean function $m(\boldsymbol{x})$ and covariance kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j) \equiv \mathrm{cov}(y(\boldsymbol{x}_i), y(\boldsymbol{x}_j))$ where $\boldsymbol{x} \in \mathbb{R}^D$ is a D-dimensional vector corresponding to a single input location. For example, $\boldsymbol{x}$ could consist of the R, Z, ϕ and t values at which a measurement was made, in which case $D = 4$. The mean function can encode any prior knowledge regarding the typical value or underlying shape of the data, but a zero mean prior distribution (i.e., $m(\boldsymbol{x}) = 0$) was found to perform well for the work presented here. The covariance kernel plays a key role in determining the smoothness of the fit – it determines how the correlation between points drops off with distance, as illustrated in figure 3. For a function to be a valid covariance kernel, it must give rise to a symmetric positive semidefinite covariance matrix for all possible inputs. A covariance kernel is said to be *stationary* if it only depends on $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ through the quantity $\boldsymbol{\tau} = \boldsymbol{x}_i - \boldsymbol{x}_j$, and is furthermore said to be *isotropic* if it only depends on $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ through $r = |\boldsymbol{\tau}|$.

A very common and useful choice is the squared-exponential (SE) covariance kernel:

$$k_{SE}(r) = \sigma_f^2 \exp\left(-\frac{r^2}{2\ell^2}\right), \tag{2}$$

where ℓ is the covariance length scale which sets how fast the correlation drops off and σ_f^2 is the signal variance which sets the extent of variation in the fitted curve. The SE covariance kernel is isotropic and encodes the assumption that the underlying curve to be predicted is smooth (specifically, infinitely differentiable) and has a constant length scale throughout its domain. It is very important to note that the covariance length scale ℓ is *not* in any way the same thing as the gradient scale length – even if ℓ is constant throughout the domain, the gradient scale length can still vary.

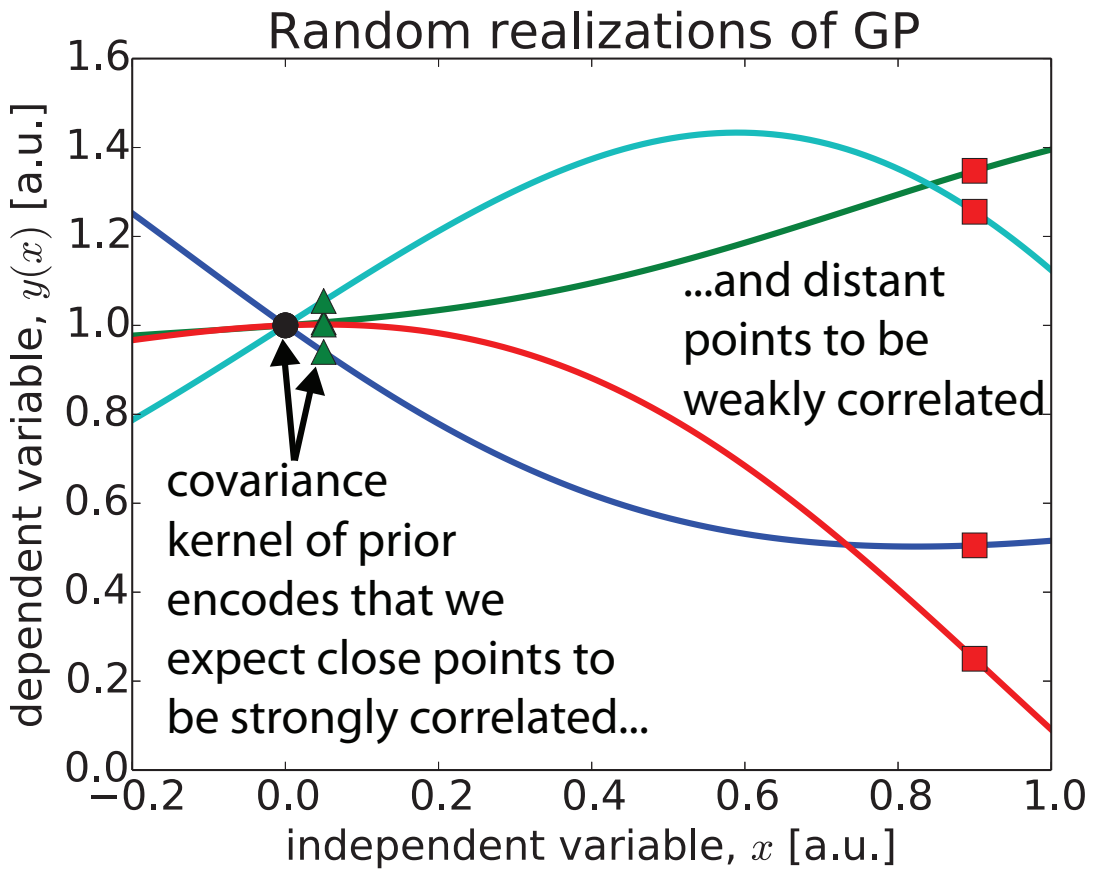


Figure 3: Illustration of the effect of the covariance kernel. Under the assumption that the underlying true curve to be reconstructed is smooth, adjacent points should be very close in value but distant points can differ substantially. The covariance kernel determines how this correlation drops off with distance. Shown are four random draws from a Gaussian process with a squared exponential covariance kernel (2) with $\sigma_f = 1$ and $\ell = 1$, conditioned on the single observation $y = 0$ at $x = 0$. In other words, each curve represents a possible realization of the profile consistent with the observation and the selected covariance kernel. This choice of covariance kernel causes the values at $x = 0.05$ (green triangles) to be close to the observed value $y(0) = 1$ (black circle). But, the values at $x = 0.9$ (red squares) are much less correlated with the observation at $x = 0$ and hence exhibit a much wider spread across the four samples shown.

The objective of profile fitting is to take n observations collected into the vector $\boldsymbol{y}$ at locations that have been gathered into the $D \times n$ matrix X and use them to make n_* predictions of the values of the underlying smooth curve collected into the vector $\boldsymbol{y}_*$ at locations in the $D \times n_*$ matrix X_*. In the plasma context $\boldsymbol{y}$ could be, for instance, the electron temperature T_e measured as a function of radius, in which case X would be a vector of radial locations. The end result of Gaussian process regression is the multivariate normal posterior distribution given in (B.10) and reproduced here:

$$f_{\boldsymbol{y}_*|\boldsymbol{y}}(\boldsymbol{y}_*|\boldsymbol{y}) = \mathcal{N}(\mathsf{K}(\mathsf{X}_*, \mathsf{X})[\mathsf{K}(\mathsf{X}, \mathsf{X}) + \Sigma_n]^{-1}\boldsymbol{y},$$

$$\mathsf{K}(\mathsf{X}_*, \mathsf{X}_*) - \mathsf{K}(\mathsf{X}_*, \mathsf{X})[\mathsf{K}(\mathsf{X}, \mathsf{X}) + \Sigma_n]^{-1}\mathsf{K}(\mathsf{X}, \mathsf{X}_*)), \tag{3}$$

where $f_{\boldsymbol{y}_*|\boldsymbol{y}}$ is the probability density function (PDF) for the predictions $\boldsymbol{y}_*$ at locations X_* conditioned on the observations $\boldsymbol{y}$ at locations X, $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ is the multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ, the notation $\mathsf{K}(\mathsf{A}, \mathsf{B})$ means the result of evaluating the covariance function $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ between all possible pairs of locations in A and B, and Σ_n is the noise covariance matrix. (The term "posterior distribution" refers to the fact that this is the distribution that has been conditioned on the observations, and is in contrast to the "prior distribution" which is the distribution before observations have been included. The prior distribution encodes any prior knowledge regarding the form of the solution. These terms are often shortened to "posterior" and "prior," respectively.) The mean of the distribution given in (3) is then used as the estimate of the profile and the diagonal elements of the covariance matrix represent the uncertainty on the fit.

One of the features that makes GPR very well-suited to plasma profile analysis is that the covariance matrix of (3) can be constructed to include not just the value of its fit but also the gradients of the fit – both for the observations and for the predictions. This means that it is trivial both to add a zero slope constraint at the magnetic axis and to obtain values *and error estimates* for the gradients. Refer to Appendix B.3 for the mathematical details.

Note that the squared exponential covariance kernel given in (2) has two *hyperparameters*, σ_f and ℓ – other choices of covariance kernel may have more. The term hyperparameters is used because these set the properties of the (prior) distribution and do not have anything to do with a parameterization of the data into a specific functional form. It is necessary to use the data to select appropriate values of these hyperparameters. This process is spelled out in full detail in Appendix B.4. There are two approaches explored in this paper: a simple point estimate comes from adopting an empirical Bayes approach and using the maximum a posteriori (MAP) estimate, the set of hyperparameters that is most likely given the observations. To fully capture any uncertainty hidden in the posterior distribution for the hyperparameters given the data, it is necessary to adopt a fully Bayesian approach and marginalize out the hyperparameters using Markov chain Monte Carlo (MCMC) techniques [30, 31, 27]. Marginalization refers to integrating out one or more variables from a joint distribution to yield a marginal distribution for the remaining variables. In the context of marginalizing out the hyperparameters, this has the form

$$f_{\boldsymbol{y}_*|\boldsymbol{y}}(\boldsymbol{y}_*|\boldsymbol{y}) = \int f_{\boldsymbol{y}_*,\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{y}_*, \boldsymbol{\theta}|\boldsymbol{y})\, \mathrm{d}\boldsymbol{\theta}, \tag{4}$$

where $f_{\boldsymbol{y}_*,\boldsymbol{\theta}|\boldsymbol{y}}$ is the joint posterior distribution for the predictions and the hyperparameters. This is developed in more detail in Appendix B.4 and given in a more useful form in (B.19).

3.2. Handling the edge: non-stationary covariance kernels

A stationary covariance kernel such as the squared exponential discussed to this point is limited by the fact that there is one length scale over the entire domain – given the abrupt change that occurs around the last closed flux surface even in L-mode plasmas, this precludes modeling the entire profile. Gibbs [32] obtained the following non-stationary version of the SE covariance kernel:

$$k_G(\boldsymbol{x},\, \boldsymbol{x}') = \sigma_f^2 \left(\frac{2\ell(\boldsymbol{x})\ell(\boldsymbol{x}')}{\ell^2(\boldsymbol{x}) + \ell^2(\boldsymbol{x}')} \right)^{1/2} \exp\left(-\frac{|\boldsymbol{x} - \boldsymbol{x}'|^2}{\ell^2(\boldsymbol{x}) + \ell^2(\boldsymbol{x}')} \right), \tag{5}$$

where $\ell(\boldsymbol{x})$ is now an arbitrary function of $\boldsymbol{x}$ and σ_f^2 is the signal variance as before. It is important to note that the functional form of $\ell(\boldsymbol{x})$ does *not* correspond to the functional form of the profile – it merely sets how fast the profile can vary in space. Letting ℓ be a function of $\boldsymbol{x}$ allows the profile to have regions with slowly varying spatial structure smoothly joined to regions with more rapidly varying spatial structure. In order to model a tokamak profile, we need a function with a core saturation value, a shorter edge saturation value to allow the rapid drop at the edge, and a smooth transition between the two. These requirements motivated the use of a hyperbolic tangent, given here for the univariate case:

$$\ell(x) = \frac{\ell_1 + \ell_2}{2} - \frac{\ell_1 - \ell_2}{2} \tanh \frac{x - x_0}{\ell_w}, \tag{6}$$

where ℓ_1 is the core saturation value, ℓ_2 is the edge saturation value, x_0 is the location of the center of the transition between the two length scales and ℓ_w is the characteristic width of the transition. In light of the popularity of tanh-like functions for fitting pedestal data it is very important to recall that this is *not* in any way forcing the fitted curve to follow a tanh function – it merely dictates the spatial correlation length as described above. This formulation has the advantage that it yields a curve which is infinitely differentiable. Length scale functions consisting of two constant regions joined with either cubic or quintic polynomials were also tested, but were found to not produce fits as satisfactory as those using the hyperbolic tangent. This formulation can easily be extended to include an arbitrary number of breakpoints, for instance adding an extra region to fit a profile exhibiting an internal transport barrier (ITB). Schemes have been devised for efficiently partitioning the domain into regions governed by different models [33, 34, 35], but this level of sophistication was not attempted in the present work.

3.3. Drawing samples for uncertainty propagation

One of the main goals of adopting an improved approach to fit plasma profiles is to be able to produce inputs for an uncertainty propagation technique such as Monte Carlo (or other more efficient techniques like Latin hypercube sampling [36], quasi Monte Carlo [37] or sparse quadrature [28]). Specifically, for many of the codes used to analyze plasma data, what is needed is not a random draw of a single scalar quantity but rather a random realization of the entire profile $\boldsymbol{y}_*$ at the n_* points in X_*. This

is exceptionally straightforward with GPR, as the result (3) is simply a multivariate normal distribution over the values of the profile $\boldsymbol{y}_*$ at the points X_*. There are well-established techniques to efficiently produce random samples from the multivariate normal distribution, or otherwise compute the expectation of a code output given a multivariate normal distribution on the inputs, which are discussed in Appendix B.5.

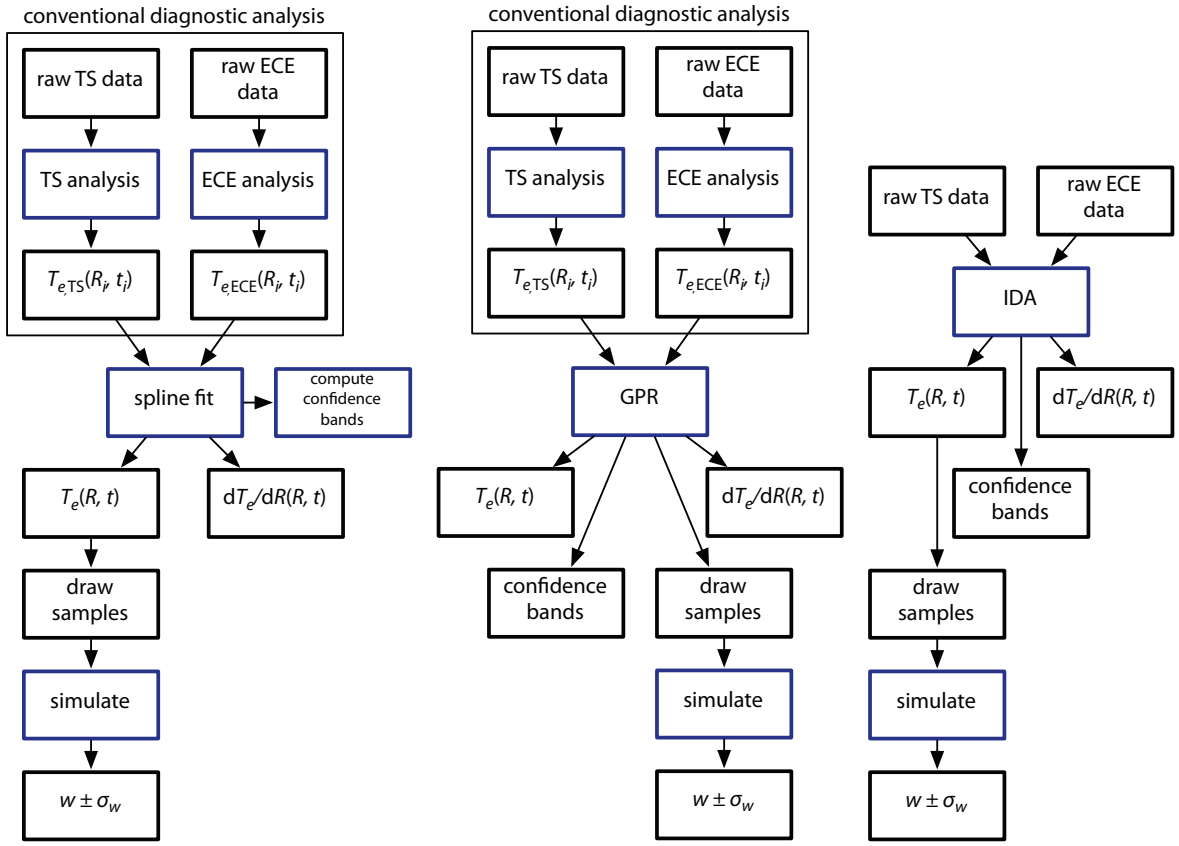
3.4. Gaussian process regression versus Bayesian integrated data analysis

It is worth comparing the present results to the work that has been done on Bayesian integrated data analysis (IDA) to combine multiple data sources into a single smooth profile [38, 39, 40, 41, 42, 43, 44]. This type of integrated analysis approach has in fact been done in a way that incorporates Gaussian processes on MAST [45]. While both techniques share the fact that they are built within a Bayesian statistical framework, they differ substantially in their details and how they fit into an analysis workflow. Essentially, IDA starts from the level of more or less raw data and infers the most likely profile(s) given a number of potentially diverse measurements. The role of GPR in the present work is to simply replace the profile fitting, data fusion and sample generation steps of a traditional analysis workflow, still using the existing procedures for turning the raw data into discrete measurements. In this way, GPR can be more readily deployed in cases where trusted data analysis codes are already in place, though it does not have some of the very powerful capabilities that the more complicated IDA approach offers. Simplified workflows using splines, GPR and IDA are shown in figure 4.

4. Application of GPR to Alcator C-Mod profiles

The rest of this paper will focus on data from an Alcator C-Mod [49] L-mode discharge with $I_p = 800\,\mathrm{kA}$, $B_T = 5.4\,\mathrm{T}$ and $1\,\mathrm{MW}$ of ICRF heating power. In order to avoid H-mode, this discharge was operated in the upper single null configuration such that the ∇B drift was away from the active x-point. Under these conditions, on-axis parameters of $n_{e,0} = 1.5 \times 10^{20}\,\mathrm{m}^{-3}$ and $T_{e,0} = 2.5\,\mathrm{keV}$ were obtained over a steady period around $0.4\,\mathrm{s}$ long. Results from this discharge were previously shown and compared to nonlinear gyrokinetic simulations in [6]. In this section, we reanalyze the background n_e and T_e profiles from this L-mode using Gaussian process regression and then proceed to obtain profiles of the inverse gradient scale lengths with statistically rigorous uncertainty estimates. Having valid estimates of these uncertainties is critical for comparing to gyrokinetic codes, and the Gaussian process framework makes propagating the uncertainty in the profiles through the analysis code to determine the experimental impurity transport coefficients very efficient, as is demonstrated in the next section.

C-Mod has an extensive diagnostic suite which is described in [50]. Two Thomson scattering (TS) systems are used to measure the n_e, T_e profiles in the core and the edge, and three separate electron cyclotron emission (ECE) systems are used to further constrain the core T_e profile. For the discharge analyzed here, Calcium (a non-intrinsic,



(a) spline (b) GPR (c) IDA (profiles only)

Figure 4: Examples of simplified workflows for obtaining some quantity w and its uncertainty from an input T_e profile using the traditional spline-based approach, GPR and IDA. With splines the computation of the fit, determining the uncertainty on the fit and the drawing of perturbed samples are all typically separate operations. Doing the fit with GPR replaces these three operations, but otherwise leaves the workflow intact. Applying IDA to just the analysis of the input profile data replaces the diagnostic analysis steps, but leaves the process of computing the output quantity w itself untouched. It is also possible to perform a fully integrated analysis to get from raw data to the desired output quantities (such as has been done to estimate Z_{eff} [46, 47, 48]), in which case even these steps are absorbed into the IDA step.

non-recycling impurity) was injected four times during the stationary part of the discharge using a multi-pulse laser blow-off impurity injector [51, 8]. The temporal and spatial evolution of the He-like calcium was measured using an x-ray imaging crystal spectrometer [52, 53] and a line-integrated view of the Li-like calcium is measured with an extreme ultraviolet spectrometer [54]. During the stationary period of the discharge the n_e and T_e profiles were fairly constant with the exception of sawtooth oscillations in the T_e profile. This study is concerned primarily with obtaining sawtooth-averaged estimates of transport in the steady state period, so all signals were time averaged over this 0.4 s period. Because of the large error bars and suspected outliers in the edge n_e data, robust estimators were used for the data from the edge Thomson system

Table 1: Hyperpriors used for the hyperparameters of the Gibbs covariance kernel with tanh *length scale function*

Quantity	σ_f	ℓ_1	ℓ_2	ℓ_w	x_0
n_e	$\mathcal{U}(0, 30 \times 10^{20}\,\mathrm{m}^{-3})$	$\mathcal{U}(0, 2)$	$\mathcal{U}(0, l_1)$	$\mathcal{U}(10^{-2}, 0.1)$	$\mathcal{U}(1.0, 1.1)$
T_e	$\mathcal{U}(0, 30\,\mathrm{keV})$	$\mathcal{U}(0, 2)$	$\mathcal{U}(0, l_1)$	$\mathcal{U}(10^{-2}, 0.1)$	$\mathcal{U}(0.98, 1.05)$

($\psi_n > 0.9$). Namely, the median was used as an estimator for the value of the density and the interquartile range was used to estimate the standard deviation according to $\sigma = IQR/(2\Phi^{-1}(0.75))$, where $\Phi^{-1}(z)$ is the inverse cumulative distribution function of the standard normal and IQR is the interquartile range. The other data were summarized with the conventional estimators for the mean and standard deviation to yield a Gaussian representation of the data, consistent with GPR's assumption of normally-distributed noise. Note that while horizontal error bars are shown on the plots to give a representation of the variability in the equilibrium mapping, these uncertainties were *not* included in the analysis. In general, these error bars are smaller than the width of a given data point. This coupled with the shallow slope throughout the core means that uncertainties in the independent variable are only likely to play a significant role in the edge (where the profile gets much steeper), and so should not affect the calculation of core transport in the present paper.

The Gibbs covariance kernel (5) with the hyperbolic tangent length scale function (6) was used to smooth both the temperature and density profiles expressed as functions of normalized poloidal flux $\psi_n = (\psi - \psi_0)/\psi_a$. The hyperpriors (i.e., prior distributions on the hyperparameters) used for the temperature and density profiles are given in table 1. These ranges were chosen both to ensure that the MAP estimation converged to a physically reasonable value as well as to ensure that the MCMC chains did not get stuck in an unphysical region of the parameter space. As mentioned in Appendix B.3, artificial "observations" can be added to the data $\boldsymbol{y}$ to enforce symmetry and other constraints. A zero slope point at $\psi_n = 0$ was used to approximate a symmetry constraint and value and slope constraints were added outside of the approximate location of the limiter at midplane, $\psi_n = 1.1$. These constraints are given in table 2. Note that the constraints at the edge are given with uncertainties – this is an advantage of this formulation in that it allows a constraint to be specified as being approximate (in the sense of having a Gaussian distribution), such that the data can drive the mean higher or lower at that location if necessary.

The MAP estimate was found using the sequential quadratic programming routine in Scipy [55, 56]. The optimizer was started at 24 points randomly distributed in the parameter bounds in order to ensure the global maximum was found. The MAP estimates of the hyperparameters are given in table 3, and are shown as the red curves in figures 5 and 6.

Marginalization over the hyperparameters was carried out using the Python package emcee [31] which implements the affine-invariant ensemble sampler described in [30].

Table 2: Constraints imposed on the profile fits by the addition of artificial "observations." The slope constraint at the magnetic axis was set as a precise value, the edge values outside of the midplane location of the limiter were set with the indicated $\pm 1\sigma$ uncertainty.

Quantity	ψ_n	y	y'
n_e $[10^{20}\,\mathrm{m}^{-3}]$	0		0
	[1.1, 1.2, 1.3, 1.4]	0.00 ± 0.01	0.0 ± 0.1
T_e [keV]	0		0
	[1.1, 1.25, 1.4]	0.000 ± 0.001	0.0 ± 0.1

Table 3: MAP estimate of hyperparameters of the Gibbs covariance kernel

Quantity	σ_f	ℓ_1	ℓ_2	ℓ_w	x_0
n_e	$2.2 \times 10^{20}\,\mathrm{m}^{-3}$	1.1	0.65	0.016	1.0
T_e	0.97 keV	0.37	0.29	0.012	1.0

Table 4: Autocorrelation times for each hyperparameter

Quantity	σ_f	ℓ_1	ℓ_2	ℓ_w	x_0
n_e	33	13	12	24	12
T_e	21	18	19	40	10

There is a brief description of this algorithm in Appendix B.4. An ensemble of 200 "walkers" split between 24 threads was used to draw samples from the posterior distribution for the hyperparameters (B.18). Each walker was started at a point randomly distributed within the hyperparameter bounds. In order to obtain a full picture of the posterior, each walker was run for 1500 samples. A burn-in of 200 samples was found to be more than sufficient for the chains to forget their initial states and become mixed. The average acceptance fraction over all the walkers was 45% for n_e and 50% for T_e, indicating efficient sampling of the posterior. The autocorrelation times for the unthinned traces for each parameter are given in table 4. This yielded far more samples than is necessary to obtain the uncertainty in the profiles, so the chains were thinned by a factor of 500 before computing any profiles, which is substantially longer than the observed autocorrelation times and had the result of eliminating almost all of the correlation between samples. A more efficient run for cases where it is not necessary to get a smooth picture of the hyperparameter space would be able to use far fewer samples.

Given a set of m samples $\{\boldsymbol{\theta}^{(i)}\}$, the marginalized mean profile (as shown as the solid blue line in figures 5 and 6) was computed using the law of iterated expectations:

$$\mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y}] = \mathbb{E}[\mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}]] = \frac{1}{m}\sum_{i=1}^{m}\mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}^{(i)}], \tag{7}$$

where $\mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}^{(i)}]$ is the mean from (B.10) evaluated with the given vector of

hyperparameters $\boldsymbol{\theta}^{(i)}$. The variance in the marginalized estimate of the profile (as shown as the shaded blue envelopes in figures 5 and 6) was computed using the law of total variance:

$$\begin{aligned}\mathrm{var}[\boldsymbol{y}_*|\boldsymbol{y}] &= \mathrm{var}[\mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}]] + \mathbb{E}[\mathrm{var}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}]] \\ &= \frac{1}{m-1}\sum_{i=1}^{m}(\mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}^{(i)}] - \mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y}])^2 + \frac{1}{m}\sum_{i=1}^{m}\mathrm{var}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}^{(i)}], \quad (8)\end{aligned}$$

where $\mathrm{var}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}^{(i)}]$ is computed from the diagonal of the covariance matrix of (B.10) with the given vector of hyperparameters $\boldsymbol{\theta}^{(i)}$. The fitted profiles are shown as the blue curves in figures 5 and 6 and the bivariate and univariate marginal posterior distributions for the hyperparameters are given in figures 7 and 8. All of the univariate marginals ended up peaked relative to the flat priors used, which indicates that the data provide sufficient information to overcome the weak information contained in the prior distribution. Note from the bivariate marginals that several hyperparameters are very strongly correlated (ℓ_1 and ℓ_2 from the n_e fit, for example). This type of distribution is in general rather inefficient to sample from with a traditional Metropolis-Hastings sampler, but the affine-invariant ensemble sampler was able to keep the acceptance rate moderate and autocorrelation times short with no manual adjustment of the proposal distribution. This performance was also helped by the fact that the marginals are unimodal.

The uncertainties in the normalized inverse gradient scale lengths as shown in figures 5 and 6 were computed using the uncertainty propagation equation [57]:

$$\frac{a}{L_y} \approx a\frac{|\partial y/\partial R|}{y} = \frac{a}{y}\left|y'\frac{\partial\psi_n}{\partial R}\right| \quad (9)$$

$$\begin{aligned}\mathrm{var}\left[\frac{a}{L_y}\right] &= \mathrm{var}[y]\left(-\frac{ay'}{y^2}\frac{\partial\psi_n}{\partial R}\right)^2 + \mathrm{var}[y']\left(\frac{a}{y}\frac{\partial\psi_n}{\partial R}\right)^2 \\ &\quad + \mathrm{cov}[y,y']\left(-\frac{ay'}{y^2}\frac{\partial\psi_n}{\partial R}\right)\left(\frac{a}{y}\frac{\partial\psi_n}{\partial R}\right) \\ &\quad + \mathrm{var}[a]\left(\frac{y'}{y}\frac{\partial\psi_n}{\partial R}\right)^2 + \mathrm{var}\left[\frac{\partial\psi_n}{\partial R}\right]\left(\frac{ay'}{y}\right)^2, \quad (10)\end{aligned}$$

where $y' \equiv \partial y/\partial\psi_n$ and it has been assumed that the geometric terms a and $\partial\psi_n/\partial R$ are not correlated with any of the variables involved. The last two terms which involve the uncertainty in the magnetic geometry were evaluated by computing the variance in the equilibrium reconstruction over the flat top, but were found to be negligible compared to the three terms arising from the uncertainty in the fitted profile. The covariance $\mathrm{cov}[y,y']$ is computed for a given set of hyperparameters by using (B.14) when computing the relevant off-diagonal elements of the covariance matrix of (B.10). The marginalized covariance was estimated from the MCMC samples using the law of total covariance:

$$\mathrm{cov}[\boldsymbol{y}_*,\boldsymbol{y}'_*|\boldsymbol{y}] = \mathbb{E}[\mathrm{cov}[\boldsymbol{y}_*,\boldsymbol{y}'_*|\boldsymbol{y},\boldsymbol{\theta}]] + \mathrm{cov}[\mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}],\mathbb{E}[\boldsymbol{y}'_*|\boldsymbol{y},\boldsymbol{\theta}]] \quad (11)$$

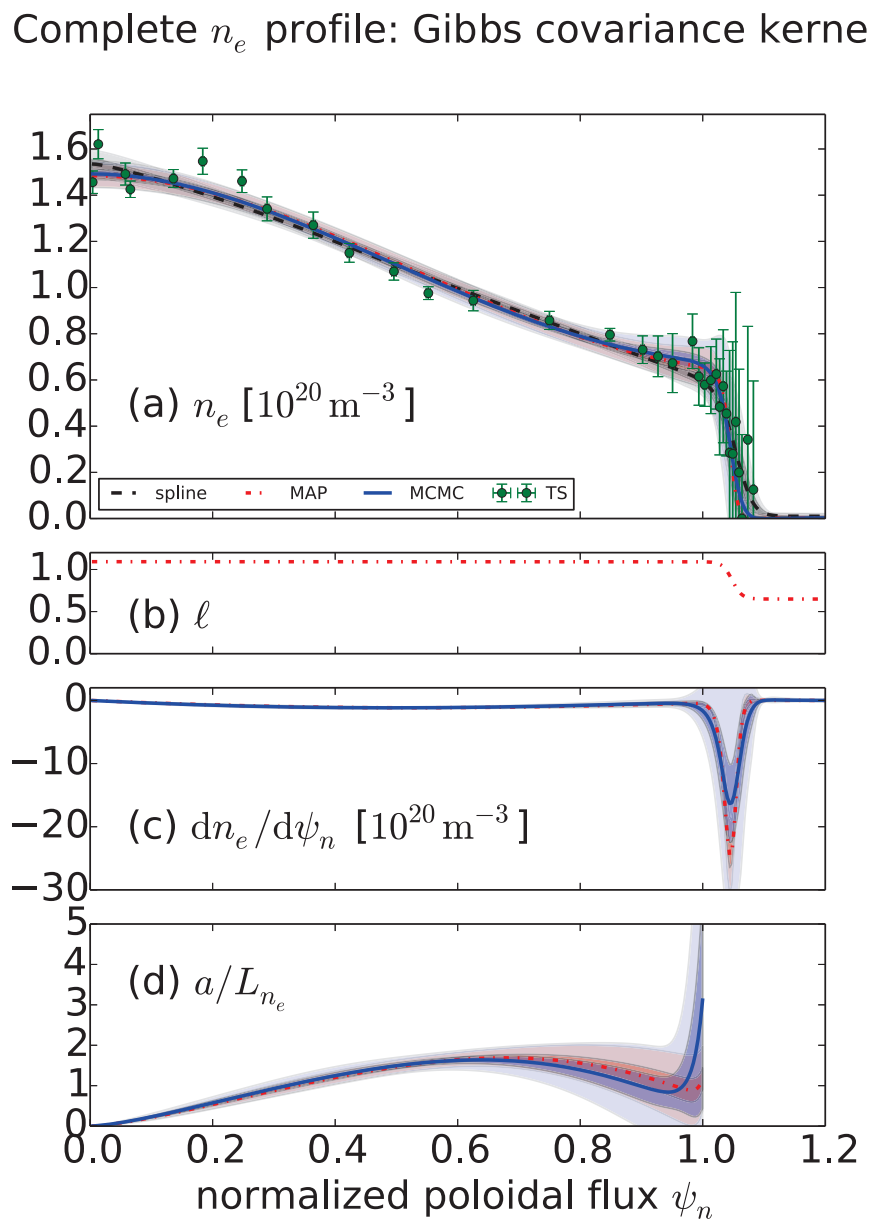


Figure 5: Data and results for the n_e profile. In (a), the TS datapoints are given as green dots. The vertical error bars are $\pm 1\sigma$. Horizontal $\pm 1\sigma$ error bars are, in general, smaller than the width of the points. On the fitted results, the inner dark uncertainty band is $\pm 1\sigma$ and the lighter uncertainty band is $\pm 3\sigma$. The result of the MAP estimation is shown as the red dash-dot curve and the result of the marginalization with MCMC is shown as the solid blue curve. In (a) the spline samples used in the previous work are shown as the black dashed curve; the spline results are not *shown in the other subplots. From top to bottom: (a) the experimental data and fitted profile, (b) length scale from the MAP estimate, (c) the gradient and its uncertainty, and (d) a/L_{n_e} as computed with the uncertainty propagation equation. Note that (d) is cut off at $\psi_n = 1$ because the calculation is not trusted outside of $0 < \psi_n < 1$. All three curves overlay very closely, and the largest discrepancies are near the edge.*

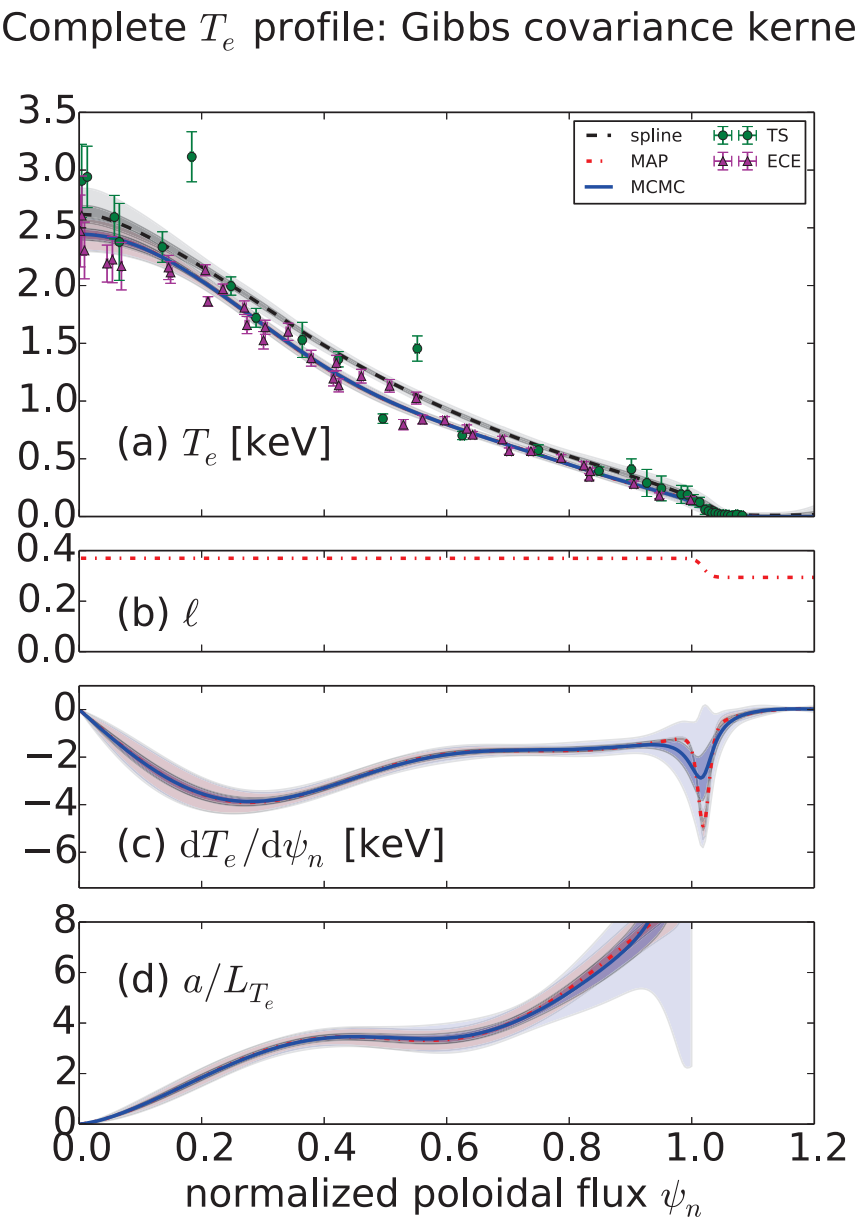


Figure 6: Data and results for the T_e profile, coloring and ordering of subplots is as in figure 5. Measurements from ECE are shown as magenta triangles. There is a much more pronounced discrepancy between the spline and GPR-based fits than was seen with the n_e profile.

While (9) is nonlinear with respect to y and hence the uncertainty propagation equation might not be expected to deliver reliable estimates, it was found to be fairly accurate over $0 < \psi_n < 1$ when compared to a brute force Monte Carlo estimation of a/L_y.

The MAP and marginalized estimates yielded very similar mean curves, but with substantially different uncertainty estimates, particularly on n_e' and a/L_{n_e}. These differences can be seen in figures 5 and 6 and are summarized in table 5, which gives the median relative uncertainties in the quantities of interest over the region $0 < \psi_n < 1$.

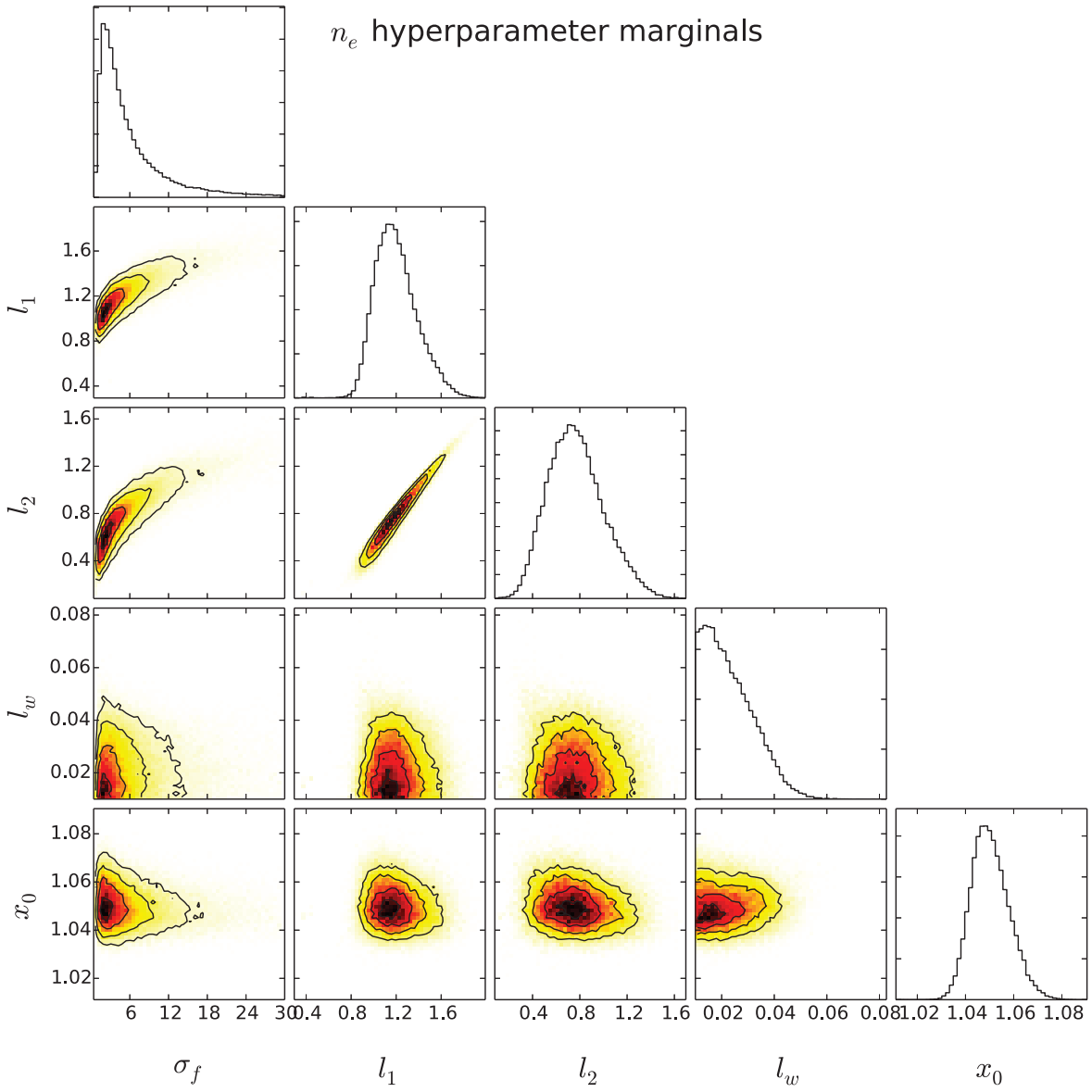


Figure 7: Matrix of univariate (on the diagonal) and bivariate marginal distributions for the hyperparameters of the fit to the n_e profile, as computed with MCMC. These plots are essentially 1- and 2-d projections of the 5-d distribution $f_{\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{\theta}|\boldsymbol{y})$ given in (B.18) for the Gibbs kernel (5) with tanh *length scale warping (6). The univariate marginals all ended up peaked relative to the flat priors used over the ranges shown, indicating that the data provide sufficient information to overcome the weak information of the prior. The bivariate marginals are all unimodal, which helps the MCMC algorithm to sample efficiently. The bivariate marginals yield information on the correlation between hyperparameters: for instance, the tilted and elongated shape of the bivariate marginal distribution between ℓ_1 and σ_f means that if the core length scale is shorter, the signal variance will tend to be smaller. Note that σ_f has units of $10^{20}\,\mathrm{m}^{-3}$ whereas the other hyperparameters are dimensionless.*

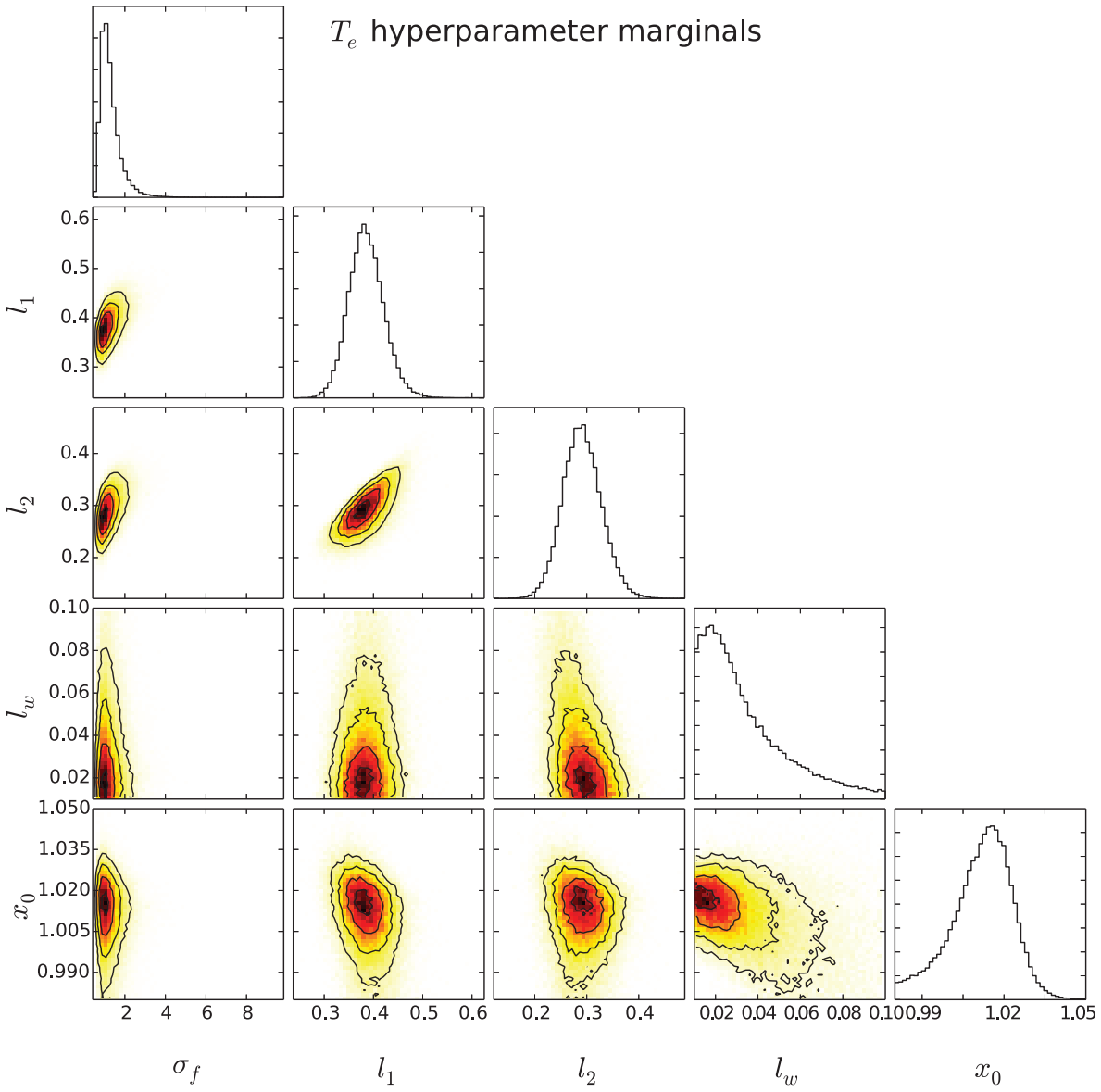


Figure 8: Univariate and bivariate marginal distributions for the hyperparameters of the fit to the T_e profile, as computed with MCMC, presented as in figure 7. Again, the univariate marginals all ended up peaked relative to the flat priors used over the ranges shown, indicating that the data provide sufficient information to overcome the weak information of the prior. Note that σ_f has units of keV *whereas the other hyperparameters are dimensionless.*

The difference in the uncertainties on the gradient between the MAP and MCMC results is very important for applications that are strongly sensitive to gradients: in order to obtain credible estimates of gradients, it is necessary to fully account for any uncertainty in the hyperparameters by marginalizing them out using MCMC. This situation has an analogue with the traditional use of splines: using the MAP estimate is equivalent to simply picking one "best" location for the spline knots and/or smoothing parameter, when this can in fact end up making the curve too restrictive to properly capture the

Table 5: Median relative uncertainties over the region $0 < \psi_n < 1$

Quantity	y	y'	a/L_y
n_e, MAP	1.2%	6.0%	6.0%
n_e, MCMC	1.4%	8.4%	8.3%
T_e, MAP	1.3%	3.7%	4.0%
T_e, MCMC	1.4%	5.4%	5.6%

full uncertainty in the gradients. However, to get an estimate of the uncertainty on just the value of a quantity, it appears to be sufficient to use the much simpler MAP estimate for the hyperparameters. The choice of which level of sophistication to use depends on how sensitive the end use is to gradients; it is preferable to use the computationally cheap MAP approach of handling the hyperparameters when possible.

5. Application of GPR to experimental measurements of impurity transport

This section considers the propagation of the profile uncertainties obtained in the previous section through the analysis workflow used to obtain experimental impurity transport coefficients in Alcator C-Mod. This type of sampling can be extended to any analysis code that needs profile inputs, such as a power balance code used to compute experimental heat fluxes [58]. The approach used to obtain the impurity transport coefficients is described in detail in [6, 7, 8]. The STRAHL code [5] takes as input the n_e and T_e profiles plus guesses for the transport coefficients D and V from the assumed impurity flux $\Gamma_Z = -D\nabla n_Z + V n_Z$ and yields as output the time evolution of the impurity density profile $n_Z(R,t)$. A synthetic diagnostic is used to obtain the line-integrated emissivity from this result which is then compared to the measured time evolution for He-like calcium observed with an x-ray imaging crystal spectrometer and Li-like calcium observed with a single-chord soft x-ray spectrometer. The guesses for D and V are then iterated upon using the MPFIT Levenberg-Marquardt minimizer [59, 60] to find the choices that produce emission time histories that best match the experimental observations. As noted in [6], the results are most sensitive to the uncertainties in the n_e and T_e profiles. Therefore, to quantify the uncertainty in the output D and V profiles, the code is run multiple times with random samples of the n_e and T_e profiles, in the manner discussed in section 2 and shown schematically in figure 2.

The previous work fit the data using splines and obtained random samples by manually re-fitting the data after perturbing the points according to their uncertainties, a process which required considerable manual intervention. The present work improves on this through the use of GPR. The shape of the spline fits has already been shown in figures 5 and 6, and is mostly similar to that of the GPR fits. Sampling from the GPR fit was conducted in two ways. The simplest approach tested is to take the MAP estimate $\hat{\boldsymbol{\theta}}^{MAP}$ for the hyperparameters, then draw samples from $f_{\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}}(\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}^{MAP})$

according to (B.20). Using the eigendecomposition of (B.21), it was found that just 10 eigendirections were sufficient to describe the 400×400 covariance matrix of the profile $\boldsymbol{y}_*$ when evaluating samples at $n_* = 400$ points. The more thorough approach tested is the fully Bayesian hierarchical sampling scheme described at the end of Appendix B.5. In either case, the sampling would sometimes yield samples that exhibited nonphysical behavior such as nonmonotonicity or negativity. Therefore, each of the samples was checked at each of the evaluation points and the sample was thrown out if $y < 0$ or $y' > 0$ at any point within $0 < \psi_n < 1$. In either case, 80 samples that satisfy the constraints were obtained and propagated through STRAHL. It is important to note that, once appropriate hyperpriors have been selected, this entire process proceeds in a completely automated manner – the number of samples run to obtain the accuracy desired from the Monte Carlo study is limited only by how much computer time the user is willing to devote to the STRAHL analysis. This is in contrast with the spline-based approach, where each sample required laborious hand-tuning of the spline parameters to produce an acceptable fit to each set of perturbed data points.

The resulting D and V profiles are given in figure 9. The D profile is very similar between all three techniques, but the fine details of the V profile are different between the GPR-based approaches and the previous spline result. While this difference is not substantially outside of the $\pm 1\sigma$ error bars, it is believed to be a result of the fact that the GPR-based T_e profile has a mean which is, on average, about 12% lower than the mean spline profile. The result is strongly sensitive to T_e, particularly in the region where the curves have the largest disagreement.

It is of interest to note that the MAP and MCMC treatments of the hyperparameters yielded approximately the same results for both the means and the uncertainties of D and V. This can be expected from the small change in uncertainty for the values of n_e and T_e noted in table 5 and the fact that only the value and not the gradient of these background profiles enters the calculation. Therefore, for this case it is possible to use the simpler MAP calculation, which enables the use of advanced sampling strategies such as Latin hypercube sampling [36], quasi-Monte Carlo sampling [37] or sparse quadrature [28] to further improve the rate of convergence, though these have yet to be applied to this problem.

6. Summary and conclusions

The paper has presented the use of Gaussian process regression (GPR) for fitting smooth curves to noisy, discrete observations of plasma profiles and then subsequently propagating the uncertainty in the fitted curve through an analysis code. While the example shown here involved propagation of the uncertainty in the background n_e, T_e profiles through an analysis code to obtain impurity transport coefficients, this approach is extremely general and can deliver benefits in any situation where gradients or profile fits are needed, particularly within the context of gyrokinetic validation. This approach was shown to have considerable advantages over the more traditional use of splines in the

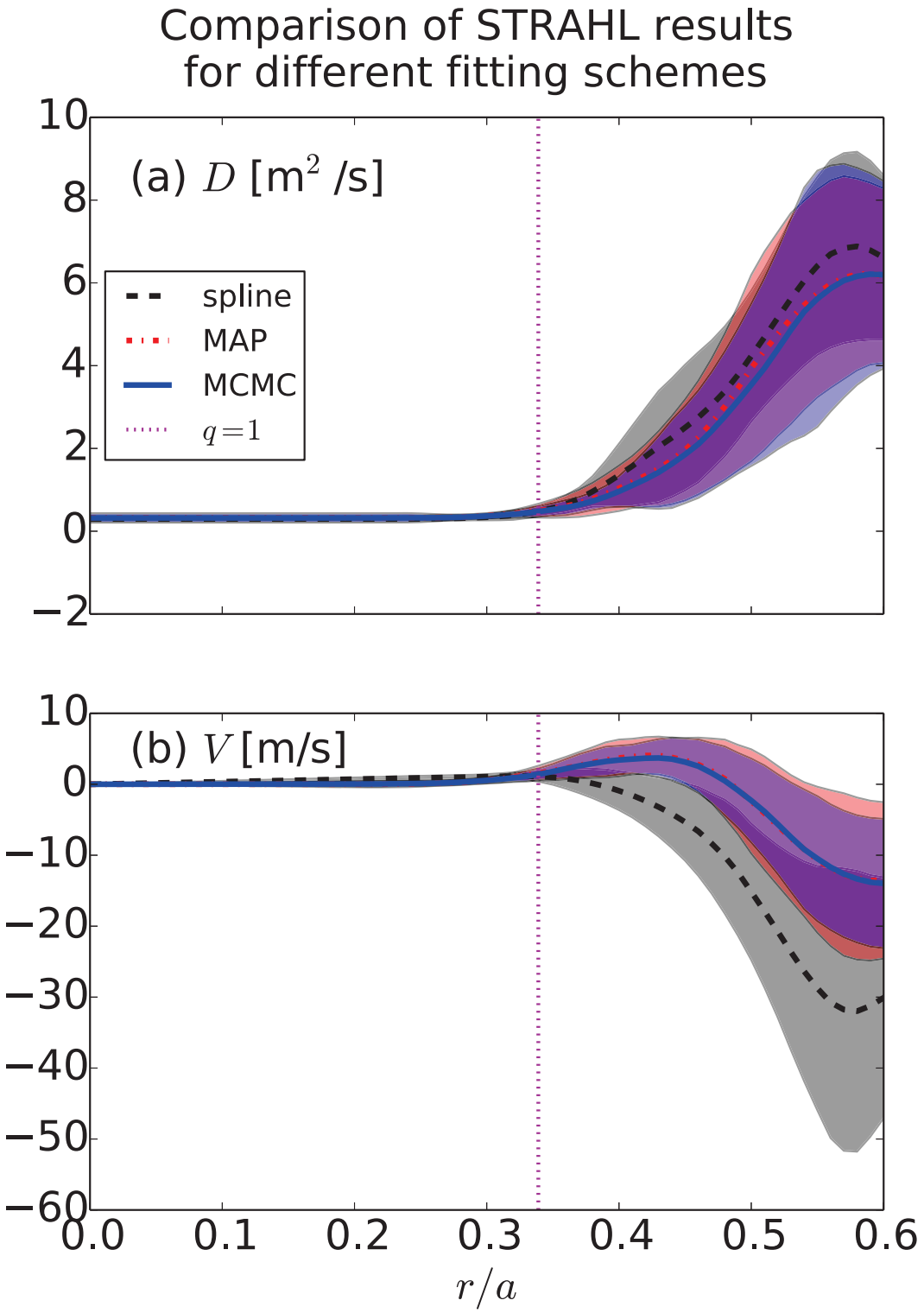


Figure 9: (a) D and (b) V profiles for spline fits (black dashed), sampling from the MAP estimate (red dash-dot) and from hierarchical sampling with MCMC (solid blue). The uncertainty envelopes are $\pm 1\sigma$. The profile is only shown over $0 < r/a < 0.6$ because the results are not trusted outside of this region.

context of fitting profiles and propagating uncertainty through analysis and simulation codes in the following respects:

- The fit proceeds automatically using standard statistical procedures instead of manual hand-tuning.
- This flexible, non-parametric approach does not overly constrain shape of fit.
- It is trivial to apply this approach to multivariate data of arbitrary dimension.
- The method provides an estimate of uncertainty on fitted value *and* gradient without additional work.
- It is straightforward to draw random samples and easy to reduce the dimensionality of the space to be sampled in order to apply advanced techniques to improve the rate of convergence of uncertainty propagation.

Two approaches for handling the hyperparameters that dictate the nature of the fit were compared: the MAP estimator provides a point estimate for the hyperparameters and is faster and simpler to work with, while the use of MCMC to marginalize over the hyperparameters provides the most rigorous accounting of uncertainty hiding in the hyperparameters of the fit. These two approaches give similar results for the uncertainty in the value of the fit, but differ substantially for the uncertainty in the gradient – hence, it is necessary to use the more complicated MCMC-based marginalization when working with processes that are strongly sensitive to gradients. These two approaches were applied to the task of inferring the impurity transport coefficients D and V from experimental data, and yielded results that were comparable to what was obtained previously using splines – but, the new results were obtained in a far more automated manner and demonstrated far more convincing convergence. It was verified that the results for D and V do not depend on the gradients of the background profiles, and hence the use of the simpler MAP estimate is sufficient. Open source software to perform GPR with gradient constraints and predictions has been developed and is available for use by anyone needing to fit smooth curves, estimate uncertainties in gradients and efficiently produce samples for use in uncertainty propagation [61]. Further use of the GPR based fitting and sampling approaches presented here has the potential to improve the quality and trustworthiness of uncertainty estimates on both profile fits and code outputs while simultaneously reducing the time for analysis both by reducing the amount of manual intervention necessary to produce fits and by improving the convergence of uncertainty propagation calculations.

Acknowledgments

This material is based upon work conducted using the Alcator C-Mod tokamak, a DOE Office of Science user facility. This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Fusion Energy Sciences under Award Number DE-FC02-99ER54512. This material is based upon work supported in part by the U.S. Department of Energy Office of Science Graduate Research Fellowship Program

(DOE SCGF), made possible in part by the American Recovery and Reinvestment Act of 2009, administered by ORISE-ORAU under contract number DE-AC05-06OR23100.

Appendix A. Mathematical details of splines

The mathematical details of splines are discussed in detail in [2, 3] and the references therein. These references form the basis for this section, with other references given as needed. A (univariate) spline is a piecewise polynomial of degree k which has continuous derivatives up to order $k-1$. Discontinuities in the k^{th} derivative are allowed to exist at a finite number of locations referred to as knots. An interpolating spline is the curve satisfying these properties that is further required to pass through specified values at each of the knots. But, given noisy data, forcing the curve to go through all of the observations will inevitably result in a curve with too much unphysical structure. There are two general approaches to produce a curve that smoothes rather than interpolates the data. A smoothing spline is the spline of degree $k = 2m - 1$ with knots located at each of the observations that minimizes $\sum_{i=1}^{n}(y_i - f(x_i))^2/n + \lambda \int_a^b (f^{(m)}(x))^2 \, \mathrm{d}x$, where y_i is the observed value at location x_i (where $i = 1, 2, \ldots, n$), $f(x)$ is the spline function and $\lambda > 0$ is called the smoothing parameter. This expression represents a tradeoff between the mean square error (first term) and the complexity of the curve (second term). The smoothing parameter sets the priority of this tradeoff – for small λ complex curves that lie close to the data are preferred, whereas large λ will drive the solution to smoother curves that are allowed to lie farther away from the data points. The other approach is to use a reduced set of knots and minimize the sum of squared residuals, $\sum_{i=1}^{n}(y_i - f(x_i))^2/n$, directly. In this case, the number of knots acts as the smoothing parameter. This type of smoothing can be seen as a sum over basis functions B_j with weights c_j:

$$f(x) = \sum_j c_j B_j(x) \tag{A.1}$$

The B-spline basis functions are a particularly popular choice on account of their favorable computational and mathematical properties [62, 63]. With this approach the knot positions can be used as an additional parameter to help better fit the data, a situation referred to as a free-knot spline [14].

Appendix B. Mathematical explanation of Gaussian process regression

This section follows the development, notation and nomenclature of [4], with other references given as needed. For this work, the Python package gptools [61] was implemented to provide support for GPR with gradient constraints and predictions.

Appendix B.1. An intuitive picture of GPR

Before presenting the full mathematical details, it is useful to consider the $D = 1$ case with one observation and one prediction in order to obtain an intuitive picture of how

GPR works. Given one location x at which a noise-free observation y has been made and one location x_* at which a prediction y_* is to be made, the joint prior probability density function (PDF) is then the bivariate normal

$$\begin{aligned} f_{y,y_*}(y, y_*) &= \mathcal{N}\left(\begin{bmatrix} m(x) \\ m(x_*) \end{bmatrix}, \begin{bmatrix} k(x,x) & k(x,x_*) \\ k(x_*,x) & k(x_*,x_*) \end{bmatrix}\right) \\ &= \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} k(x,x) & k(x,x_*) \\ k(x_*,x) & k(x_*,x_*) \end{bmatrix}\right), \end{aligned} \tag{B.1}$$

where the notation $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ refers to the multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ and in the last step a zero mean function has been used. This is the distribution *before* any observations have been included – it encodes prior assumptions regarding smoothness, bounds, etc. In the context of plasma physics, y could be the electron temperature T_e and x the normalized poloidal flux ψ_n, for example. To make this quantitative, take k to be squared exponential (2) with $\sigma_f = 1$ and $r/\ell = |x - x_*|/\ell = 1$, which gives the joint prior PDF

$$f_{y,y_*}(y, y_*) = \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \mathrm{e}^{-1/2} \\ \mathrm{e}^{-1/2} & 1 \end{bmatrix}\right) \tag{B.2}$$

This is shown along with the marginal prior PDFs

$$f_y(y) = \int_{-\infty}^{\infty} f_{y,y_*}(y, y_*)\,\mathrm{d}y_* = \mathcal{N}(0,1) \tag{B.3}$$

$$f_{y_*}(y_*) = \int_{-\infty}^{\infty} f_{y,y_*}(y, y_*)\,\mathrm{d}y = \mathcal{N}(0,1) \tag{B.4}$$

in figure B1. (As indicated in the previous equations, the marginal PDF for y is the result of integrating the joint distribution over all possible values of y_*. In (B.3), for example, y_* is said to have been marginalized out of the distribution.) The effect of varying r/ℓ is shown in figure B2.

Now consider the situation once a noise-free observation of a specific value for y has been made. The PDF of y_* conditioned on this observation is then

$$\begin{aligned} f_{y_*|y}(y_*|y) &= \frac{f_{y,y_*}(y, y_*)}{f_y(y)} \\ &= \mathcal{N}\left(\frac{k(x_*,x)}{k(x,x)}y, k(x_*,x_*) - \frac{[k(x_*,x)]^2}{k(x,x)}\right) \end{aligned} \tag{B.5}$$

For instance, for $y = 1$ and the parameters used above, this becomes

$$f_{y_*|y}(y_*|y=1) = \mathcal{N}(\mathrm{e}^{-1/2}, 1 - \mathrm{e}^{-1}) \tag{B.6}$$

This is shown as the dashed curve in the top plot of figure B1. As is evident from both the figure and (B.6), the effect of including the information $y = 1$ is to shift the expected value of y_* from $\mathbb{E}(y_*) = 0$ to $\mathbb{E}(y_*|y=1) = \mathrm{e}^{-1/2}$ and to lower the variance from $\mathrm{var}(y_*) = 1$ to $\mathrm{var}(y_*|y=1) = 1 - \mathrm{e}^{-1}$. Hence, the prediction at x_* with $\pm 1\sigma$ uncertainty interval is $y_* = \mathrm{e}^{-1/2} \pm (1 - \mathrm{e}^{-1})^{1/2} = 0.6 \pm 0.8$. As will be seen in subsequent

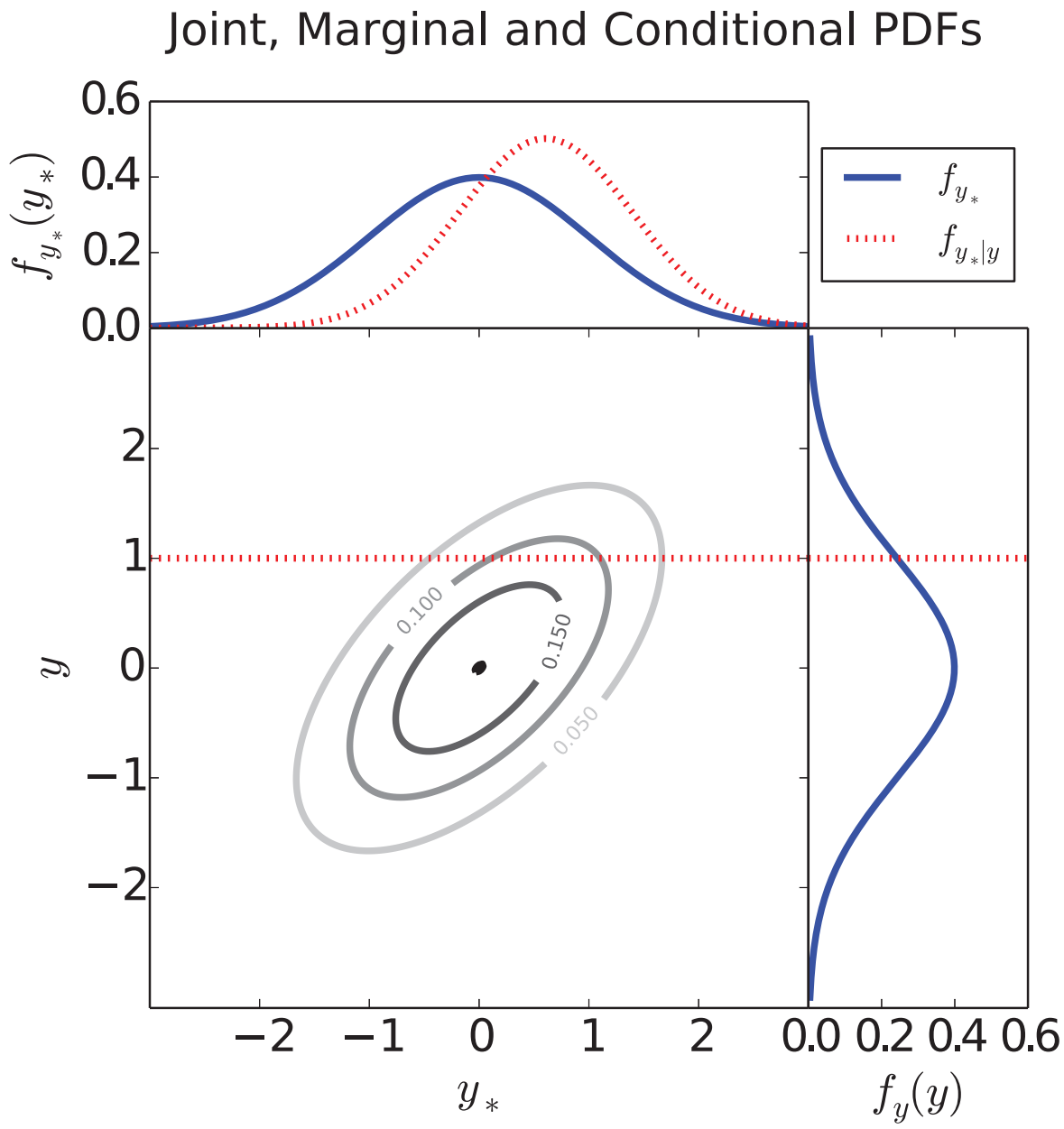


Figure B1: Joint prior probability density function (contours), with the marginal PDFs $f_y(y)$, $f_{y_*}(y_*)$ (solid curves), the conditional PDF $f_{y_*|y}(y_*|y = 1)$ (dashed curve), and the observation $y = 1$ (dashed horizontal line). The covariance matrix was constructed from a squared-exponential covariance kernel with $\sigma_f = 1$ and $r/\ell = 1$. The tilted ellipse shape of the contours is indicative of the correlation between y and y_* – the values for y and y_* are expected to be related (see figure B2). The effect of conditioning on the observation $y = 1$ is to shift the distribution for y_* towards 1 and to make the distribution narrower.

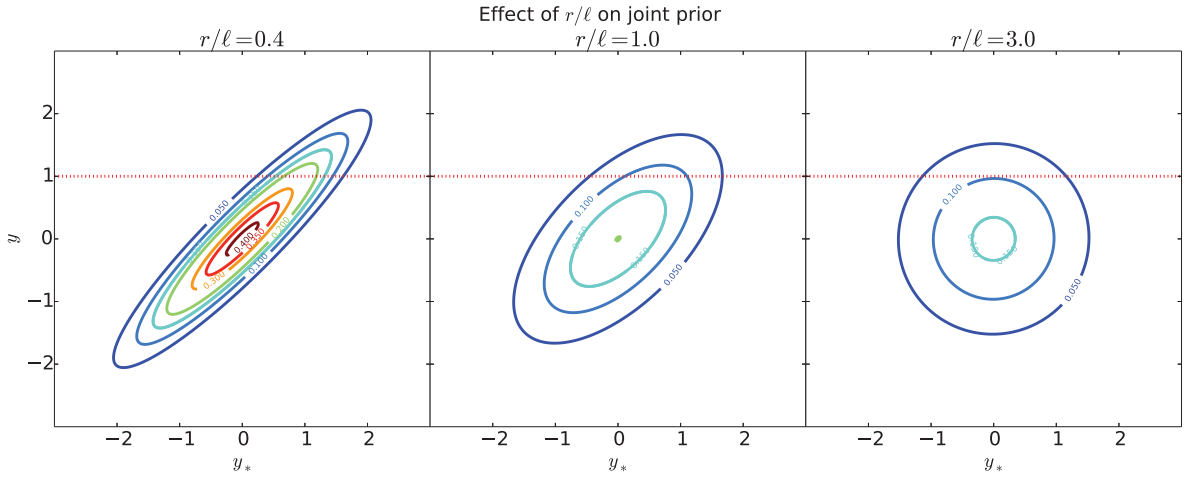


Figure B2: Effect of r/ℓ on the shape of the joint prior PDF for $r/\ell = 0.4$, 1.0 and 3.0. The values were computed using an SE covariance kernel with $\sigma_f = 1$. The smaller r/ℓ is (or equivalently, the closer x is to x_*) the more correlated y and y_* are, thereby causing a more dramatic elongation of the tilted elliptical contours. As r/ℓ increases, y and y_* become less correlated and the contours become circular. Also shown is the observation $y = 1$ as the dashed horizontal line. As r/ℓ increases, the observation is less informative and therefore the conditional PDF will be wider. This can also be thought of in terms of the smoothing effect of ℓ: the larger ℓ is, the smaller r/ℓ will be for any given value of r. Hence, a larger ℓ leads to a smoother curve by driving down the variance over a larger region around each observation.

sections, using more observations would reduce the uncertainty even more, as would be expected.

If instead a noisy observation $z = y + \epsilon$ is made, where the noise ϵ is distributed as a zero mean normal with variance σ_n^2, then a prior between z and y_* is used instead:

$$f_{z,y_*}(z, y_*) = \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} k(x,x) + \sigma_n^2 & k(x,x_*) \\ k(x_*,x) & k(x_*,x_*) \end{bmatrix}\right), \tag{B.7}$$

The analysis is then the same as before, with the one change that in (B.1) through (B.6) $k(x,x)$ is replaced with $k(x,x) + \sigma_n^2$.

Appendix B.2. Full details of GPR

For noisy observations $\boldsymbol{y}$ at n input locations collected in the $D \times n$ matrix $\mathsf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]$, the outputs $\boldsymbol{y}$ have the joint prior PDF

$$f_{\boldsymbol{y}}(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{m}(\mathsf{X}), \mathsf{K}(\mathsf{X},\mathsf{X}) + \Sigma_n), \tag{B.8}$$

where the notation $\boldsymbol{m}(\mathsf{X})$ indicates the n element vector formed by evaluating $m(\boldsymbol{x})$ at each of the columns of X, $\mathsf{K}(\mathsf{X},\mathsf{X})$ indicates the $n \times n$ matrix formed by evaluating $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ between each of the possible pairs of columns in X, and Σ_n is the noise covariance matrix of the observations. In general Σ_n could include correlated noise, but in this application a diagonal matrix was used to model uncorrelated heteroscedastic Gaussian noise. While it is possible to include powerful constraints in

the prior/evaluation of the model itself [64], for this work it was found that the simple zero mean prior as given in [4] was sufficient, such that the joint prior PDF between the observations $\boldsymbol{y}$ and the predictions $\boldsymbol{y}_*$ is

$$f_{\boldsymbol{y},\boldsymbol{y}_*}(\boldsymbol{y},\boldsymbol{y}_*) = \mathcal{N}\left(\mathbf{o}, \begin{bmatrix} \mathsf{K}(\mathsf{X},\,\mathsf{X}) + \Sigma_n & \mathsf{K}(\mathsf{X},\,\mathsf{X}_*) \\ \mathsf{K}(\mathsf{X}_*,\,\mathsf{X}) & \mathsf{K}(\mathsf{X}_*,\,\mathsf{X}_*) \end{bmatrix}\right), \tag{B.9}$$

where X_*, $\boldsymbol{y}_*$ are the locations and values of the predictions, respectively. (Note that $\mathsf{X}_* \in \mathbb{R}^{D\times n_*}$, $\boldsymbol{y}_* \in \mathbb{R}^{n_*}$ where there are n_* points at which a prediction is to be made.)

What is of interest to make predictions is the conditional PDF of $\boldsymbol{y}_*$ given the observations $\boldsymbol{y}$. This is a standard result for the multivariate normal distribution, and is given in a particularly useful form in [4]:

$$\begin{aligned} f_{\boldsymbol{y}_*|\boldsymbol{y}}(\boldsymbol{y}_*|\boldsymbol{y}) = \mathcal{N}(&\mathsf{K}(\mathsf{X}_*,\,\mathsf{X})[\mathsf{K}(\mathsf{X},\,\mathsf{X}) + \Sigma_n]^{-1}\boldsymbol{y}, \\ &\mathsf{K}(\mathsf{X}_*,\,\mathsf{X}_*) - \mathsf{K}(\mathsf{X}_*,\,\mathsf{X})[\mathsf{K}(\mathsf{X},\,\mathsf{X}) + \Sigma_n]^{-1}\mathsf{K}(\mathsf{X},\,\mathsf{X}_*)) \end{aligned} \tag{B.10}$$

The conditional mean then gives the prediction and the diagonal elements of the conditional covariance matrix give the variance in the prediction. As this can be evaluated at any point $\boldsymbol{x}_*$, a Gaussian process is said to represent a distribution over functions. Note that inversion of an $n \times n$ symmetric positive definite matrix is required, which leads to an asymptotic complexity of $\mathcal{O}(n^3)$.

The mean of (B.10) merits further inspection:

$$\overline{\boldsymbol{y}}_* = \mathbb{E}[\boldsymbol{y}_*|\boldsymbol{y}] = \mathsf{K}(\mathsf{X}_*,\,\mathsf{X})[\mathsf{K}(\mathsf{X},\,\mathsf{X}) + \Sigma_n]^{-1}\boldsymbol{y} \tag{B.11}$$

$$\overline{y}_*(\boldsymbol{x}_*) = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x}_i,\,\boldsymbol{x}_*), \tag{B.12}$$

where the weights α_i are linear combinations of the measurements:

$$\boldsymbol{\alpha} = [\mathsf{K}(\mathsf{X},\,\mathsf{X}) + \Sigma_n]^{-1}\boldsymbol{y} \tag{B.13}$$

The conditional mean as a function of $\boldsymbol{x}_*$ is a weighted sum of n copies of the covariance kernel, with each copy centered at an observation. This makes the connection between GPR and splines obvious – if k were an appropriately selected polynomial basis function, this would be equivalent to the spline given in (A.1) with the knots centered at each observation, though with the added benefits alluded to previously in section 2.2 and the additional flexibility of being able to select from a wider variety of basis functions in order to obtain whatever properties might be required for the task at hand.

Appendix B.3. Prediction of gradients and their uncertainties

Another very useful property of Gaussian processes is that there is a very simple relationship between a Gaussian process and its derivatives:

$$\operatorname{cov}\left(y_i,\,\frac{\partial y_j}{\partial x_{dj}}\right) = \frac{\partial k(\boldsymbol{x}_i,\,\boldsymbol{x}_j)}{\partial x_{dj}} \tag{B.14}$$

$$\operatorname{cov}\left(\frac{\partial y_i}{\partial x_{di}},\,\frac{\partial y_j}{\partial x_{dj}}\right) = \frac{\partial^2 k(\boldsymbol{x}_i,\,\boldsymbol{x}_j)}{\partial x_{di}\,\partial x_{dj}}, \tag{B.15}$$

where cov is the covariance and the notation $\partial/\partial x_{dj}$ refers to a derivative with respect to the d^{th} component of the input $\boldsymbol{x}_j$ to the covariance kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Repeated application of these equations allows derivatives of arbitrary order to be included. By constructing the joint distribution between the observed values y, observed values $\partial y/\partial x_d$ (and possibly higher order derivatives), predicted values y_* and predicted values $\partial y_*/\partial x_{*d}$ (and possibly higher order derivatives) it is possible to make a simultaneous prediction of the underlying smooth curve, its derivative(s) *and* the uncertainty in both the value and its derivative(s).

Another application is to use derivative information to approximate symmetry and boundary constraints – in the work presented here, an artificial zero slope "observation" at the magnetic axis was used to approximate a symmetry constraint. While such constraints can be included through transformations on the prior itself [65], this simpler approach was found to perform well in practice.

Appendix B.4. Selection of a covariance kernel and its hyperparameters

The SE covariance kernel given in (2) has two hyperparameters σ_f and ℓ that determine the properties of the fit; other choices of covariance kernel may have more hyperparameters. The term *hyperparameter* is used because we are referring to parameters that determine the prior distribution rather than the shape of the fitted curve directly. It is also instructive to recall at this point that the hyperparameters are *not* the parameters of a parametric model that the data are reduced into: a specific functional form is not assumed, and the observations must be used to make predictions. In other words, given a specific, arbitrary choice for σ_f and ℓ, the conditioned PDF (B.10) will yield a curve that is most consistent with the observations *given that particular choice of hyperparameters*. What now remains is to pick the hyperparameters (and covariance kernel) that are most consistent with the data. Note that this is a different question than asking which hyperparameters fit the data with the smallest residual – with the SE covariance kernel, for example, one could always make the error small by taking ℓ to be very small, but then the model would be fitting the noise. There are several possible approaches to carry out the selection of hyperparameters discussed under the topic of model comparison/selection in [4, 27]. Three levels of sophistication are considered here: maximum likelihood estimation, maximum a posteriori estimation and marginalization over the hyperparameters.

The simplest approach presented here is the maximum likelihood (ML) estimator. The ML estimate is a point estimate for the hyperparameters consisting of the values of the hyperparameters that maximize the probability of the observed data $\boldsymbol{y}$ given the hyperparameters $\boldsymbol{\theta}$ – this is simply the marginal PDF for $\boldsymbol{y}$ as in (B.3) (but now given for the general case) with the dependence on the vector of hyperparameters $\boldsymbol{\theta} \equiv [\sigma_f, \ell, \ldots]$ made explicit:

$$f_{\boldsymbol{y}|\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{m}(\mathsf{X}), \mathsf{K}(\mathsf{X}, \mathsf{X}|\boldsymbol{\theta})\right), \tag{B.16}$$

where the notation $\mathsf{K}(\mathsf{X}, \mathsf{X}|\boldsymbol{\theta})$ refers to the $n \times n$ covariance matrix constructed using the

covariance kernel k with the specific hyperparameters $\boldsymbol{\theta}$. Explicitly, the ML estimate is then $\hat{\boldsymbol{\theta}}^{ML} = \arg\max_{\boldsymbol{\theta}} f_{\boldsymbol{y}|\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{\theta})$. In practice what is used is the natural logarithm of the likelihood:

$$\ln f_{\boldsymbol{y}|\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{1}{2}\boldsymbol{y}^T(\mathsf{K}+\Sigma_n)^{-1}\boldsymbol{y} - \frac{1}{2}\ln|\mathsf{K}+\Sigma_n| - \frac{n}{2}\ln 2\pi \tag{B.17}$$

Each of these terms permits a simple interpretation [4]:

- The first term is the only one that depends on the observations $\boldsymbol{y}$ and is related to how well the model fits the data.
- The second term depends only on the determinant of the covariance matrix, and is related to the complexity of the model.
- The final term only depends on the number of observations n and is a normalization constant that does not depend on the hyperparameters $\boldsymbol{\theta}$ and hence does not affect the optimization.

The next level of sophistication is to include prior information on the hyperparameters in order to obtain the posterior PDF for the hyperparameters. This prior information, encoded in the hyperprior $f_{\boldsymbol{\theta}}(\boldsymbol{\theta})$, can readily be included in (B.17) using Bayes' rule to give the posterior for the hyperparameters:

$$f_{\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{f_{\boldsymbol{y}|\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{\theta})f_{\boldsymbol{\theta}}(\boldsymbol{\theta})}{f_{\boldsymbol{y}}(\boldsymbol{y})} \tag{B.18}$$

The maximum a posteriori (MAP) estimate for the hyperparameters is a point estimate consisting of the most likely values of the hyperparmeters $\boldsymbol{\theta}$ given the observations $\boldsymbol{y}$, or $\hat{\boldsymbol{\theta}}^{MAP} = \arg\max_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{\theta}|\boldsymbol{y})$. Note that the term in the denominator is simply a normalizing constant which is independent of $\boldsymbol{\theta}$, so the end result is that the expression to be maximized over $\boldsymbol{\theta}$ is simply (B.17) with an extra factor $\ln f_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ added in.

It must be noted that both the ML and MAP estimators are point estimates: they select a single value of the hyperparameters given the data and possibly some prior information. The posterior distribution for $\boldsymbol{\theta}$ (B.18) can, however, have substantial variance, leading to uncertainty in the fit that is not captured with a point estimate like ML or MAP gives. What is better is to employ a fully Bayesian approach and marginalize (integrate) the predictive distribution over the hyperparameters:

$$\begin{aligned} f_{\boldsymbol{y}_*|\boldsymbol{y}}(\boldsymbol{y}_*|\boldsymbol{y}) &= \int f_{\boldsymbol{y}_*,\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{y}_*,\boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{\theta} \\ &= \int f_{\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}}(\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta})f_{\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{\theta}, \end{aligned} \tag{B.19}$$

where the second line follows simply from the definition of conditional probability, the term $f_{\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}}(\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta})$ is (B.10) with the conditioning on the hyperparameters $\boldsymbol{\theta}$ made explicit and $f_{\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{\theta}|\boldsymbol{y})$ is as in (B.18). This integration was efficiently carried out in practice using Markov chain Monte Carlo (MCMC) integration [27], specifically the affine invariant ensemble sampler given in [30, 31]. This algorithm uses an ensemble of many "walkers" (typically on the order of several hundred) which in effect perform

a random walk guided by the posterior distribution to yield a collection of samples $\{\boldsymbol{\theta}^{(i)}\}$ of the hyperparameters distributed according to $f_{\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{\theta}|\boldsymbol{y})$ which can then be used to evaluate integrals like (B.19). This formulation can also be used to account for uncertainties in the independent variable X by noting that the result of (B.19) is implicitly conditioned on X and then marginalizing out the values of X, but this was not done in the present work.

Appendix B.5. Drawing samples for uncertainty propagation

Recall from (B.10) that the result of GPR is a multivariate normal distribution over the values of the profile $\boldsymbol{y}_*$ at the points X_*. The standard recipe for producing a random draw $\widetilde{\boldsymbol{y}}_*$ from the n_*-dimensional multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ is to produce through standard means a vector $\boldsymbol{u}$ of n_* independent, standard normal variables (i.e., $\boldsymbol{u} \sim \mathcal{N}(\mathbf{o}, \mathsf{I})$), then find

$$\widetilde{\boldsymbol{y}}_* = \mathsf{A}\boldsymbol{u} + \boldsymbol{\mu}, \tag{B.20}$$

where $\Sigma = \mathsf{A}\mathsf{A}^T$ [66, 4, 67]. A common, computationally efficient choice for how to decompose Σ is the Cholesky decomposition $\Sigma = \mathsf{L}\mathsf{L}^T$, where L is lower triangular. But, for the application of advanced uncertainty propagation methods such as quasi Monte Carlo [37] or sparse quadrature [28], large increases in the convergence rate can be gained by reducing the dimension of the parameter space that must be explored. When using the Cholesky decomposition the dimension of the space to be sampled is equal to the number of points the curve is evaluated at, n_*. Instead, consider the eigendecomposition:

$$\Sigma = \mathsf{Q}\Lambda\mathsf{Q}^{-1} = \mathsf{Q}\Lambda^{1/2}(\mathsf{Q}\Lambda^{1/2})^T, \tag{B.21}$$

where in the last step the fact that Σ is guaranteed to be symmetric and hence have an orthogonal matrix of eigenvectors was used. Hence, we can take $\mathsf{A} = \mathsf{Q}\Lambda^{1/2}$. In practice, the eigenvalues drop off quite rapidly and can therefore be truncated to produce draws while sampling in a space with much lower dimension than the number of points the curve is to be evaluated at.

If MCMC is being used to marginalize over the hyperparameters, then sampling must take place hierarchically [27]: first, a sample $\widetilde{\boldsymbol{\theta}} \sim f_{\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{\theta}|\boldsymbol{y})$ is drawn from (B.18) using MCMC. Then, using (B.20), a sample $\widetilde{\boldsymbol{y}}_*$ is drawn from $f_{\boldsymbol{y}_*|\boldsymbol{y},\boldsymbol{\theta}}(\boldsymbol{y}_*|\boldsymbol{y}, \boldsymbol{\theta} = \widetilde{\boldsymbol{\theta}})$. Performing such sampling repeatedly then gives an ensemble of possible realizations to be used as inputs in the next step of the analysis workflow.

References

[1] C. Holland, A. E. White, G. R. McKee, M. W. Shafer, J. Candy, R. E. Waltz, L. Schmitz, and G. R. Tynan. Implementation and application of two synthetic diagnostics for validating simulations of core tokamak turbulence. *Phys. Plasmas*, 16:052301, 2009.

[2] G. Wahba. *Spline Models for Observational Data.* Society for Industrial and Applied Mathematics, 1990.

[3] P. Dierckx. *Curve and Surface Fitting with Splines.* Oxford University Press, 1993.

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

[5] R. Dux. STRAHL User Manual. Technical Report 10/30, Max-Planck-Institut für Plasmaphysik, September 2006.

[6] N. T. Howard, M. Greenwald, D. R. Mikkelsen, M. L. Reinke, A. E. White, D. Ernst, Y. Podpaly, and J. Candy. Quantitative comparison of experimental impurity transport with nonlinear gyrokinetic simulation in an Alcator C-Mod L-mode plasma. *Nucl. Fusion*, 52:063002, 2012.

[7] N. T. Howard, M. Greenwald, D. R. Mikkelsen, A. E. White, M. L. Reinke, D. Ernst, Y. Podpaly, and J. Candy. Measurement of plasma current dependent changes in impurity transport and comparison with nonlinear gyrokinetic simulation. *Phys. Plasmas*, 19:056110, 2012.

[8] N. T. Howard. *Experimental and Gyrokinetic Studies of Impurity Transport in the Core of Alcator C-Mod Plasmas*. PhD thesis, Massachusetts Institute of Technology, June 2012.

[9] J. Wesson. *Tokamaks*. Oxford University Press, 4 edition, 2011.

[10] O. J. W. F. Kardaun. *Classical Methods of Statistics*. Springer, 2005.

[11] A. E. White, L. Schmitz, W. A. Peebles, T. L. Rhodes, T. A. Carter, G. R. McKee, M. W. Shafer, G. M. Staebler, K. H. Burrell, J. C. DeBoo, and R. Prater. Simultaneous measurement of core electron temperature and density fluctuations during electron cyclotron heating on DIII-D. *Phys. Plasmas*, 17:020701, 2010.

[12] T. C. M. Lee. Smoothing parameter selection for smoothing splines: a simulation study. *Comput. Stat. Data An.*, 42:139–148, 2003.

[13] D. L. B. Jupp. The "lethargy" theorem—a property of approximation by γ-polynomials. *J. Approx. Theory*, 14(3):204–217, 1975.

[14] D. L. B. Jupp. Approximation to data by splines with free knots. *SIAM J. Numer. Anal.*, 15(2):328–343, 1978.

[15] G. Wahba. Bayesian "confidence intervals" for the cross-validated smoothing spline. *J. R. Stat. Soc. B Met.*, 45(1):133–150, 1983.

[16] D. Nychka. Bayesian confidence intervals for smoothing splines. *J. Am. Stat. Assoc.*, 83(404):1134–1143, December 1988.

[17] K.-C. Li. Honest confidence regions for nonparametric regression. *Ann. Stat.*, 17(3):1001–1008, 1989.

[18] Y. Wang and G. Wahba. Bootstrap confidence intervals for smoothing splines and their comparison to Bayesian confidence intervals. *J. Stat. Comput. Sim.*, 51:263–279, 1995.

[19] S. Zhou, X. Shen, and D. A. Wolfe. Local asymptotics for regression splines and confidence regions. *Ann. Stat.*, 26(5):1760–1782, 1998.

[20] W. Mao and L. H. Zhao. Free-knot polynomial splines with confidence intervals. *J. R. Statist. Soc. B*, 65(4):901–919, 2003.

[21] S. Zhou and D. A. Wolfe. On derivative estimation in spline regression. *Stat. Sinica*, 10:93–108, 2000.

[22] G. Qin and M. Tsao. Empirical likelihood based inference for the derivative of the nonparametric regression function. *Bernoulli*, 11(4):715–735, 2005.

[23] G. Cao, J. Wang, L. Wang, and D. Todem. Spline confidence bands for functional derivatives. *J. Stat. Plan. Infer.*, 142(6):1557–1570, June 2012.

[24] V. Dose and R. Fischer. Function estimation employing exponential splines. In Kevin H. Knuth, Ali E. Abbas, Robin D. Morris, and J. Patrick Castle, editors, *Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, volume 803 of *AIP Conference Proceedings*, pages 67–71. AIP Publishing, 2005.

[25] R. Fischer and V. Dose. Flexible and reliable profile estimation using exponential splines. In Ali Mohammad-Djafari, editor, *Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, volume 872 of *AIP Conference Proceedings*, pages 296–303. AIP Publishing, 2006.

[26] R. Fischer, A. Dinklage, and Y. Turkin. Non-parametric profile gradient estimation. In *33rd EPS Conference on Plasma Physics*, volume 30I. ECA, June 2006.

[27] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian Data Analysis*. CRC Press, 3 edition, 2014.

[28] O.P. Le Maître and O.M. Knio. *Spectral Methods for Uncertainty Quantification*. Springer, 2010.

[29] G. Matheron. The intrinsic random functions and their applications. *Adv. Appl. Prob.*, 5:439–438, 1973.

[30] J. Goodman and J. Weare. Ensemble samplers with affine invariance. *Comm. App. Math. and Comp. Sci.*, 5(1):65–80, 2010.

[31] D. Foreman-Mackey, D. W. Hogg, D. Lang, and J. Goodman. emcee: The MCMC hammer. *Publ. Astron. Soc. Pac.*, 125:306–312, 2013.

[32] M. N. Gibbs. *Bayesian Gaussian Processes for Regression and Classification*. PhD thesis, University of Cambridge, 1997.

[33] R. B. Gramacy. *Bayesian Treed Gaussian Process Models*. PhD thesis, University of California Santa Cruz, December 2005.

[34] R. B. Gramacy. tgp: An R package for bayesian nonstationary, semiparametric nonlinear regression and design by treed gaussian process models. *J. Stat. Softw.*, 19(9):1–46, 2007.

[35] R. B. Gramacy and M. Taddy. Categorical inputs, sensitivity analysis, optimization and importance tempering with tgp version 2, an R package for treed gaussian process models. *J. Stat. Softw.*, 33(6):1–48, 2010.

[36] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.

[37] C. Lemieux. *Monte Carlo and Quasi-Monte Carlo Sampling*. Springer, 2009.

[38] A. Dinklage, R. Fischer, G. Kühner, H. Maaßberg, E. Pasch, and J. Svensson. Steps towards an integrated data analysis: Basic concepts and Bayesian analysis of Thomson scattering data. In *29th EPS Conference on Plasma Phys. and Contr. Fusion Monteux*, volume 26B. ECA, 2002.

[39] J. Svensson, A. Dinklage, J. Geiger, and R. Fischer. An integrated data analysis model for the W7-AS stellarator. In *30th EPS Conference on Contr. Fusion and Plasma Phys., St. Petersburg*, volume 27A. ECA, 2003.

[40] R. Fischer and A. Dinklage. Integrated data analysis of fusion diagnostics by means of the Bayesian probability theory. *Rev. Sci. Instrum.*, 75(10):4237–4239, 2004.

[41] R. Fischer, E. Wolfrum, Ch. Fuchs, and ASDEX Upgrade Team. Integrated density profile analysis in ASDEX Upgrade H-modes. In *35th EPS Conference on Plasma Phys. Hersonissos*, volume 32D. ECA, 2008.

[42] R. Fischer, A. Burckhart, N. Hicks, B. Kurzan, E. Wolfrum, and ASDEX Upgrade Team. Multiple diagnostic data analysis of density and temperature profiles in ASDEX Upgrade. In *36th EPS Conference on Plasma Phys. Sofia*, volume 33E. ECA, 2009.

[43] R. Fischer, C. J. Fuchs, B. Kurzan, W. Suttrop, E. Wolfrum, and ASDEX Upgrade Team. Integrated data analysis of profile diagnostics at ASDEX Upgrade. *Fusion Sci. Technol.*, 58(2):675–684, October 2010.

[44] B. Ph. van Milligen, T. Estrada, E. Ascasíbar, D. Tafalla, D. López-Bruna, A. López Fraguas, J. A. Jiménez, I. García-Cortés, A. Dinklage, and R. Fischer. Integrated data analysis at TJ-II: The density profile. *Rev. Sci. Instrum.*, 82:073503, 2011.

[45] G. T. von Nessi and M. J. Hole. Using bayesian analysis and gaussian processes to infer electron temperature and density profiles on the mega-ampere spherical tokamak experiment. *Rev. Sci. Instrum.*, 84:063505, 2013.

[46] S. K. Rathgeber, R. Fischer, S. Fietz, J. Hobirk, A. Kallenbach, H. Meister, T. Pütterich, F. Ryter, G. Tardini, E. Wolfrum, and ASDEX Upgrade Team. Estimation of profiles of the effective ion charge at ASDEX Upgrade with Integrated Data Analysis. *Plasma Phys. Control. Fusion*, 52:095008, 2010.

[47] G. Verdoolaege, R. Fischer, G. Van Oost, and JET-EFDA Contributors. Potential of a Bayesian integrated determination of the ion effective charge via bremsstrahlung and charge exchange

spectroscopy in tokamak plasmas. *IEEE T. Plasma Sci.*, 38(11):3168–3196, November 2010.

[48] G. Verdoolaege, R. Fischer, and G. Van Oost. Integrated analysis and consistency measurement of bremsstrahlung and charge exchange spectroscopy data for the determination of the ion effective charge. *Rev. Sci. Instrum.*, 81:10D703, 2010.

[49] E. S. Marmar and Alcator C-Mod Group. The Alcator C-Mod program. *Fusion Sci. Technol.*, 51(3):261–265, April 2007.

[50] N. P. Basse, A. Dominguez, E. M. Edlund, C. L. Fiore, R. S. Granetz, A. E. Hubbard, J. W. Hughes, I. H. Hutchinson, J. H. Irby, B. LaBombard, L. Lin, Y. Lin, B. Lipschultz, J. E. Liptac, E. S. Marmar, D. A. Mossessian, R. R. Parker, M. Porkolab, J. E. Rice, J. A. Snipes, V. Tang, J. L. Terry, S. M. Wolfe, S. J. Wukitch, K. Zhurovich, R. V. Bravenec, P. E. Phillips, W. L. Rowan, G. J. Kramer, G. Schilling, S. D. Scott, and S. J. Zweben. Diagnostic systems on Alcator C-Mod. *Fusion Sci. Technol.*, 51(3):476–507, April 2007.

[51] N. T. Howard, M. Greenwald, and J. E. Rice. Characterization of impurity confinement on Alcator C-Mod using a multi-pulse laser blow-off system. *Rev. Sci. Instrum.*, 82:033512, 2011.

[52] A. Ince-Cushman, J. E. Rice, M. Bitter, M. L. Reinke, K. W. Hill, M. F. Gu, E. Eikenberry, Ch. Broennimann, S. Scott, Y. Podpaly, S. G. Lee, and E. S. Marmar. Spatially resolved high resolution x-ray spectroscopy for magnetically confined fusion plasmas (invited). *Rev. Sci. Instrum.*, 79:10E302, 2008.

[53] J E Rice, M L Reinke, J M A Ashbourn, C Gao, M M Victoria, M A Chilenski, L Delgado-Aparicio, N T Howard, A E Hubbard, J W Hughes, and J H Irby. X-ray observations of Ca^{19+}, Ca^{18+} and satellites from Alcator C-Mod tokamak plasmas. *J. Phys. B*, 47(7):075701, 2014.

[54] M. L. Reinke, P. Beiersdorfer, N. T. Howard, E. W. Magee, Y. Podpaly, J. E. Rice, and J. L. Terry. Vacuum ultraviolet impurity spectroscopy on the Alcator C-Mod tokamak. *Rev. Sci. Instrum.*, 81:10D736, 2010.

[55] D. Kraft. A software package for sequential quadratic programming. Technical Report DFVLR-FB 88-28, DLR German Aerospace Center, Institute for Flight Mechanics, Koln, Germany, 1988.

[56] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001–.

[57] P. R. Bevington and D. K. Robinson. *Data Reduction and Error Analysis for the Physical Sciences*. McGraw-Hill, 3 edition, 2003.

[58] R. J. Hawryluk. An empirical approach to tokamak transport. In *Physics of Plasmas Close to Thermonuclear Conditions*, volume 1, pages 19–46. Brussels: CEC, 1980.

[59] J. J. Moré. The Levenberg-Marquardt algorithm: Implementation and theory. In G.A. Watson, editor, *Numerical Analysis*, volume 630 of *Lecture Notes in Mathematics*, pages 105–116. Springer Berlin Heidelberg, 1978.

[60] C. B. Markwardt. Non-linear Least-squares Fitting in IDL with MPFIT. In D. A. Bohlender, D. Durand, and P. Dowler, editors, *Astronomical Data Analysis Software and Systems XVIII*, volume 411 of *Astronomical Society of the Pacific Conference Series*, page 251, September 2009.

[61] M. A. Chilenski. gptools: Gaussian processes with arbitrary derivative constraints and predictions. `https://github.com/markchil/gptools`, 2014.

[62] C. de Boor. *A Practical Guide to Splines*. Springer, 2 edition, 2001.

[63] L. L. Schumaker. *Spline Functions: Basic Theory*. Cambridge University Press, 3 edition, 2007.

[64] S. Da Veiga and A. Marrel. Gaussian process modeling with inequality constraints. *Ann. Fac. Sci. Toulouse*, 21(3):529–555, 2012.

[65] R. Murray-Smith and B. A. Pearlmutter. Transformations of Gaussian process priors. In J. Winkler, N. Lawrence, and M. Niranjan, editors, *Deterministic and Statistical Methods in Machine Learning*, volume 3635 of *Lecture Notes in Computer Science*, pages 110–123. Springer, 2005.

[66] Y. L. Tong. *The Multivariate Normal Distribution*. Springer, 1990.

[67] A. Gut. *An Intermediate Course in Probability*. Springer, 2 edition, 2009.