Prediction of Data with help of the Gaussian Process Method

R. Preuss, U. von Toussaint
Max-Planck-Institute for Plasma Physics
EURATOM Association
85748 Garching, Germany

March 20, 2015

Abstract

The simulation of plasma-wall interactions of fusion plasmas is extremely costly in computer power and time – the running time for a single parameter setting is easily in the order of weeks or months. We propose to exploit the already gathered results in order to predict the outcome for parametric studies within the high dimensional parameter space. For this we utilize the Gaussian process method within the Bayesian framework. Uncertainties of the predictions are provided which point the way to parameter settings of further (expensive) simulations.

Keywords: Parametric studies, Gaussian process, uncertainty prediction
PACS: 02.50.-r, 52.65.-y

1 Introduction

The problem of predicting function values in a multi-dimensional space supported by given data is a regression problem for a non-trivial function of unknown shape. Given n input data vectors $\boldsymbol{x}_i$ of dimension N_{dim} (with matrix $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n)$) and corresponding target data $\boldsymbol{y} = (y_1, ..., y_n)^T$ blurred by Gaussian noise of variance σ_d^2 the quested quantity is the target value f_* at test input vector $\boldsymbol{x}_*$. The later would be generated by a function $f(\boldsymbol{x})$

$$y = f(\boldsymbol{x}) + \epsilon \quad , \tag{1}$$

where $\langle \epsilon \rangle = 0$ and $\langle \epsilon^2 \rangle = \sigma_d^2$. For being completely ignorant about a model describing function our Ansatz is to employ the Gaussian process method, with which any uniformly continuous function may be represented. As a statistical process it is fully defined by its covariance function and called Gaussian, because any collection of random variables produced by this process has a Gaussian distribution.

The Gaussian process method defines a distribution over functions. One can think of the analysis as taking place in a space of functions (function-space view) which is conceptually different to the familiar view of solving the regression problem of, for instance, the standard linear model (SLM)

$$f^{\mathrm{SLM}}(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{w} \quad , \tag{2}$$

in the space of the weights $\boldsymbol{w}$ (weight-space view). At this point it is instructive to restate the results for the later: the predictive distribution depending on mean $\bar{f}_*$ and variance for a test input data point $\boldsymbol{x}_*$ is given by

$$p(f_*^{\mathrm{SLM}} | \boldsymbol{X}, \boldsymbol{y}, X_*) \propto \mathcal{N}\left(\bar{f}_*^{\mathrm{SLM}}, \mathrm{var}(f_*^{\mathrm{SLM}})\right) \quad , \tag{3}$$

with

$$\bar{f}_*^{\mathrm{SLM}} = \frac{1}{\sigma_d^2} \boldsymbol{x}_*^T \left[\sigma_d^{-2} \boldsymbol{X}\boldsymbol{X}^T + \Sigma_p^{-1}\right]^{-1} \boldsymbol{X}\boldsymbol{y} \quad , \tag{4}$$

$$\mathrm{var}(f_*^{\mathrm{SLM}}) = \boldsymbol{x}_*^T \left[\sigma_d^{-2} \boldsymbol{X}\boldsymbol{X}^T + \Sigma_p^{-1}\right]^{-1} \boldsymbol{x}_* \quad . \tag{5}$$

Σ_p is the covariance in a Gaussian prior on the weights. In the next chapter these results will be transferred to the function-space view of the Gaussian process method.

The Gaussian process method has been appreciated much in the fields of neural networks and machine learning [1, 2, 3]. Throughout this paper, we follow in notation the book of Rasmussen & Williams [4]. Most of the presented analysis may be found there, except for small amendments.

2 Prediction of function values

As stated above the defining quantity of the Gaussian process method is the covariance function. Its choice is decisive for the inference we want to apply. It is the place where we incorporate all the properties which we would like our (hidden) problem describing function to have in order to influence the result. For example, the neighbourhood of two input data vectors $\boldsymbol{x}_p$ and $\boldsymbol{x}_q$ should be of relevance for the smoothness of the result. This shall be expressed by a length scale λ which represents the long range dependence of the two vectors. For the covariance function itself we employ a Gaussian type exponent with the negative squared value of the distance between two vectors $\boldsymbol{x}_p$ and $\boldsymbol{x}_q$

$$k(\boldsymbol{x}_p, \boldsymbol{x}_q) = \sigma_f^2 \exp\left\{-\frac{1}{2}\left|\frac{\boldsymbol{x}_p - \boldsymbol{x}_q}{\lambda}\right|^2\right\} \quad . \tag{6}$$

σ_f^2 is the signal variance and apriori set to one, if we are ignorant about this value. To avoid lengthy formulae, we abbreviate the covariance matrix of the input data as $(\boldsymbol{K})_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and the vector of covariances between test point and input data as $(\boldsymbol{k}_*)_i = k(\boldsymbol{x}_*, \boldsymbol{x}_i)$.

Moreover, we consider the degree of information which the data possesses by an overall variance σ_n^2 accounting that the data are noisy and – more detailed – $(\sigma_d)_i$ for the uncertainty estimation of a single data point y_i provided by the experimentalist. It can be shown [4] that in analogy to Eq. (3) for given λ, σ_f and σ_n the probability distribution for a single function value f_* is

$$p(f_*|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}_*) \propto \mathcal{N}\left(\bar{f}_*, \mathrm{var}(f_*)\right) \quad , \tag{7}$$

with mean and variance

$$\bar{f}_* = \boldsymbol{k}_*^T \left(\boldsymbol{K} + \sigma_n^2 \boldsymbol{\Delta}\right)^{-1} \boldsymbol{y} \quad , \tag{8}$$

$$\mathrm{var}(f_*) = \boldsymbol{k}(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}_*^T \left(\boldsymbol{K} + \sigma_n^2 \boldsymbol{\Delta}\right)^{-1} \boldsymbol{k}_* \quad . \tag{9}$$

$\boldsymbol{\Delta}$ is a matrix with the variances $\boldsymbol{\sigma}_d^2$ of the input data on its diagonal and zero otherwise. If no uncertainties of the input data are provided, $\boldsymbol{\Delta}$ is set to the identity matrix.

3 Marginalizing the hyper-parameters

The hyper-parameters $\boldsymbol{\theta} = (\lambda, \sigma_f, \sigma_n)^T$ determine the result of the Gaussian process method. Since we do not know a priori, which setting is useful, we

marginalize over them later on in order to get the target values $\boldsymbol{f}_*$ for test inputs $\boldsymbol{X}_*$. Their expectation values are

$$\langle \boldsymbol{\theta} \rangle = \frac{\int \mathrm{d}\boldsymbol{\theta}\ \boldsymbol{\theta} p(\boldsymbol{\theta}|\boldsymbol{y})}{\int \mathrm{d}\boldsymbol{\theta}\ p(\boldsymbol{\theta}|\boldsymbol{y})} = \frac{\int \mathrm{d}\boldsymbol{\theta}\ \boldsymbol{\theta} p(\boldsymbol{y}|\boldsymbol{\theta}) p(\boldsymbol{\theta})}{\int \mathrm{d}\boldsymbol{\theta}\ p(\boldsymbol{y}|\boldsymbol{\theta}) p(\boldsymbol{\theta})} \quad . \tag{10}$$

Gaussian priors are employed for the hyper-parameters with mean and variance one but constrained to be positive,

$$p(\theta_i) \sim \mathcal{N}(1,1) \ \ \forall \ \ \theta_i \geq 0 \quad \text{and} \quad p(\theta_i) = 0 \ \ \text{otherwise} \quad . \tag{11}$$

The marginal likelihood $p(\boldsymbol{y}|\boldsymbol{\theta})$ is obtained by

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \int \mathrm{d}\boldsymbol{f}\ p(\boldsymbol{y}|\boldsymbol{f}, \boldsymbol{\theta}) p(\boldsymbol{f}|\boldsymbol{\theta}) \quad . \tag{12}$$

As we deal with the Gaussian process the probability functions are of Gaussian type, with the likelihood as $p(\boldsymbol{y}|\boldsymbol{f}, \boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{f}, \sigma_n \boldsymbol{\Delta})$ and the prior for $\boldsymbol{f}$ as $p(\boldsymbol{f}|\boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K})$ [4]. Thus the integration in Eq. (12) yields

$$\log p(\boldsymbol{y}|\boldsymbol{\theta}) \sim -\frac{1}{2} \boldsymbol{y}^T \left[\boldsymbol{K}(\boldsymbol{\theta}) + \sigma_n^2 \boldsymbol{\Delta} \right]^{-1} \boldsymbol{y} - \frac{1}{2} \log \left| \boldsymbol{K}(\boldsymbol{\theta}) + \sigma_n^2 \boldsymbol{\Delta} \right| \quad . \tag{13}$$

The expectation value for the target f_* at test input $\boldsymbol{x}_*$ employs the marginal likelihood and priors for the hyper-parameters from above

$$\langle \boldsymbol{f}_* \rangle = \int \mathrm{d}\boldsymbol{\theta}\ \bar{f}_* \frac{p(\boldsymbol{y}|\boldsymbol{\theta}) p(\boldsymbol{\theta})}{\int \mathrm{d}\boldsymbol{\theta}'\ p(\boldsymbol{y}|\boldsymbol{\theta}') p(\boldsymbol{\theta}')} \quad , \tag{14}$$

where the fraction contains the sampling density in Markov chain Monte Carlo.

4 Simulation of one-dimensional data sets

In order to examine the dependence of the result on the hyper-parameters we first have a look at the situation in one dimension. While the variances of the signal and the noise will mainly effect the accuracy of the result, the most interesting hyper-parameter is the length scale λ. In Fig. 1(a-d) we set σ_f=1 and σ_n=0.1 and have a look at the maximum aposteriori prediction of Eq. (7). For a decent number of data (n=21, right panel) the model is perfectly reproduced with λ=0.1 (d), while for a large value of $\lambda = 1$ – emphasizing the long range behavior – the result becomes far to smooth (b). The later can be seen on the left in Fig. 1(a), too, but in absence of further knowledge (i.e. more data) the variance between the few input points (n=5) becomes large (c). The result for marginalizing over all hyper-parameters is

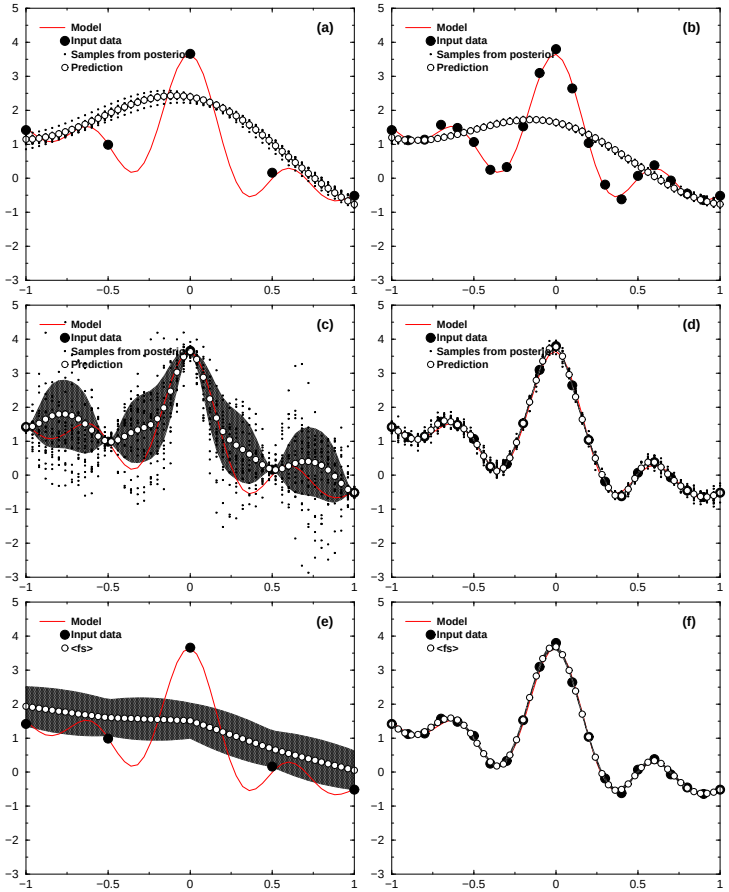

Figure 1: One dimensional example: left panel n=5, right panel n=21. Maximum aposteriori prediction with σ_f=1, σ_n=0.1 for λ=1 (top) and λ=0.1 (middle). Bottom: Expectation values of the target function from MCMC-calculation.

n	$\langle\lambda\rangle$	$\langle\sigma_f\rangle$	$\langle\sigma_n\rangle$
5	1.11 ± 0.83	0.94 ± 0.65	1.40 ± 0.52
21	0.191 ± 0.030	1.34 ± 0.39	0.106 ± 0.041

Table 1: MCMC expectation values.

shown by Fig. 1(e) and (f). With n=5 input data it is not possible to infer the $\sin(x)/x$-model and an uninformative line with large uncertainty region is the most honest result (e). The same conclusion can be drawn examining the expectation values of the hyper-parameters in the table below of Fig. 1. For the n=5 case the standard deviations are in the same order as the expectation values themselves, thus making them not very reliable. However, for n=21, the expectation value of the length scale $\langle\lambda\rangle$ becomes pretty sharp and the noise $\langle\sigma_n\rangle$ resembles closely the originally used value of 10% to blur the data. Not much may be learned from the signal variance $\langle\sigma_f\rangle$ but that our apriori estimation of one seems to be a good choice.

5 Multi-modality of marginal likelihood

We saw in the chapter above that the target function may be just an uninformative line through data space, regarding deviations from the data as tolerable within uncertainty range, or following each data point assuming high accuracy – just according to the setting of the hyper-parameters λ and σ_n. If both alternatives are possible explanations to the same data set, the marginal likelihood Eq. (13) will show multi-modality. In this case the analytic prediction of mean and variance from Eqs. (8, 9) for certain λ and σ_n will fail to represent the complete solution. This can be seen in Fig. 2 for four input data values with standard deviation one. Though the result for λ=1 and σ_n=1 on the top right (b) has highest probability a straight (uninformative) line (a) corresponding to λ=0.2 and σ_n=0.45 contributes significantly to the marginal likelihood (smaller hump in center of Fig. 2(e)). Since the marginalization integral takes into account all contributions from the marginal likelihood, the target function shown in Fig. 2(c) overcomes this problem. However, one should bear in mind that for sparse and unreliable data the handy maximum aposteriori result which simply employs the expectation values of the hyper-parameters (table below Fig. 2) in Eqs. (8, 9) in order to generate target values for test input data without redoing the (MCMC) integration of Eq. (14) is not sufficient (see Fig. 2(d)).

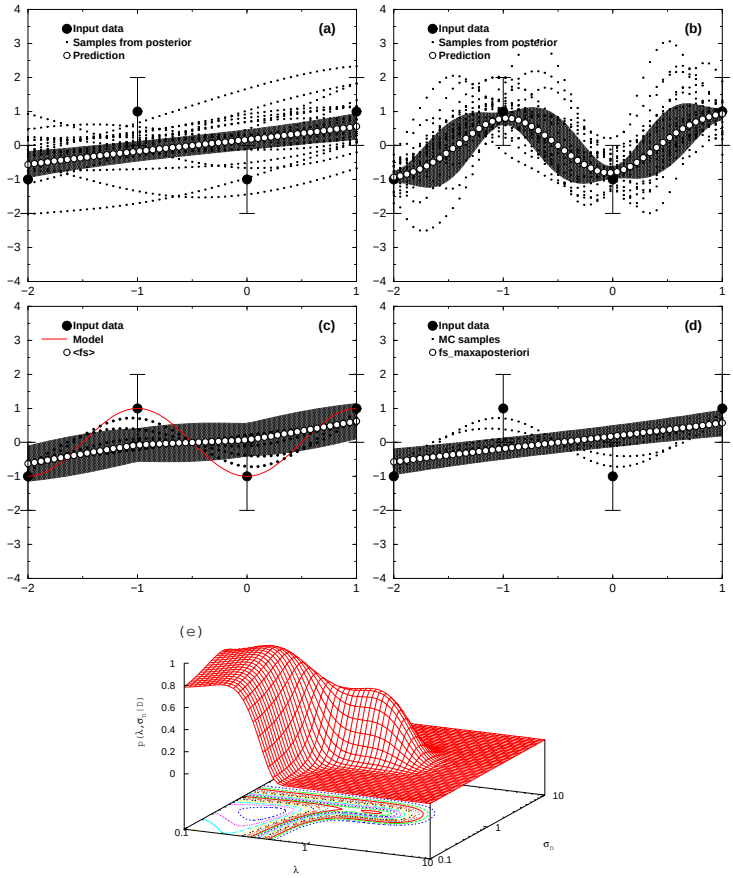

Figure 2: One dimensional example with four input data. Upper panel: target function prediction of Eq. (7) with σ_f=1 for λ=1, σ_n=1 (a) and λ=0.2, σ_n=0.45 (b). (c) target function from MCMC marginalization of the hyper-parameters; (d) maximum aposteriori result with hyper-parameter expectation values of table below; (e) marginal likelihood for the hyper-parameters λ and σ_n.

$\langle\lambda\rangle$	$\langle\sigma_f\rangle$	$\langle\sigma_n\rangle$
1.20 ± 0.83	0.85 ± 0.61	1.18 ± 0.51

Table 2: MCMC expectation values.

6 Simulation of two-dimensional data set

To expand our investigations to the two-dimensional case we examine the results for 5×5 and 21×21 input data points (see Fig. 3). Like in the one-dimensional case for too little data points (or information) the expectation value for the target function is a flat, uninformative hyper-plane with considerably uncertainty. The latter may be inferred likewise from the large variance of $\langle \sigma_n \rangle$ assigning the input data points negligible relevance except for the overall slope (last column in table below Fig. 3). For more data (21×21) the situation turns completely and the expectation target function is an even better representation of the original $\sin|(\boldsymbol{x})|/|\boldsymbol{x}|$-model than the artificially blurred input data itself. Again the blurring factor of 10% is revealed nicely by the analysis (last column/line in the table). Additionally, the difference that the amount of data makes may be seen in the peaky structure of the posterior distribution for λ and σ_n compared to the 5×5-case (lower panel in Fig. 3).

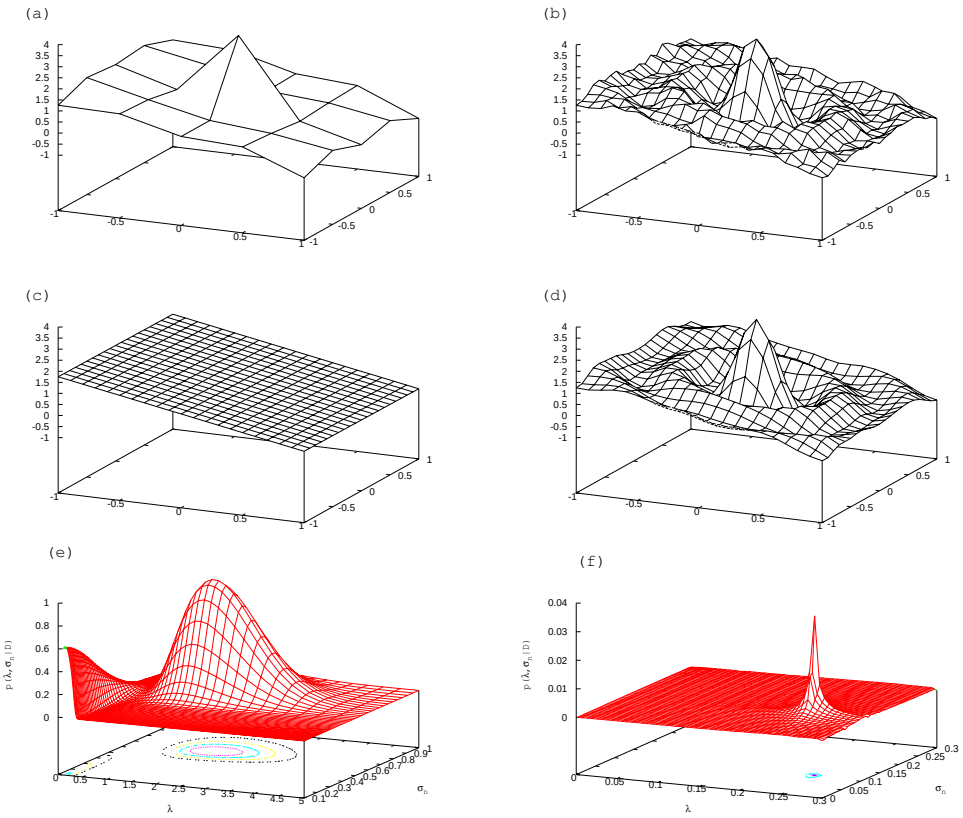

Figure 3: Two dimensional example – lines between lattice points are guide to the eye: left panel n=5×5, right panel n=21×21. Top: input data; middle target expectation values; bottom: (unnormalized) marginal likelihood for λ and σ_n.

n	$\langle\lambda\rangle$	$\langle\sigma_f\rangle$	$\langle\sigma_n\rangle$
5×5	0.576 ± 0.045	0.84 ± 0.19	1.29 ± 0.79
21×21	0.198 ± 0.011	0.592 ± 0.074	0.1023 ± 0.0043

Table 3: MCMC expectation values.

7 Real world example with fusion plasma data

The power of the Gaussian process method is its straightforward applicability in any number of dimensions for input data or target function. This becomes of special use in spaces of various (fusion plasma) input and output parameters, if the number of source data is already sufficient for reliable inferences. A particular computationally expensive area is that of predicting the outcome of particle transport and plasma-wall interaction in the scrape-off layer in fusion plasma experiments. Here the theoretically acquired results are obtained by the interplay of two sophisticated codes either describing the plasma solving a fluid equation or the transport of neutrals by a Monte-Carlo method. The run for a single parameter setting is in the order of weeks, sometimes even several months on the fastest many-core computers available. A data base of 1500 parameter settings will be the platform we intend to start from to make inferences about outcomes within the ranges of the acquired data. To keep it instructive we restrict ourselves in this paper to the two-dimensional space for the input data (the core densities of deuterium *na:core:D:ave* and helium *na:core:He:ave*) and one-dimensional target function (maximum electron density, outboard divertor: *nemxap:ave*). Further restrictions on the data set caused by physics considerations (e.g. density constraints) leave a number of 76 input data vectors (see Fig. 4a). The expectation value of the target function, as well as the result from the maximum aposteriori distribution with the expectation values of the hyper-parameters shows those areas in input space where further (expensive) computations should take place to enforce the reliability of the outcome (Fig. 4(b) and (c)).

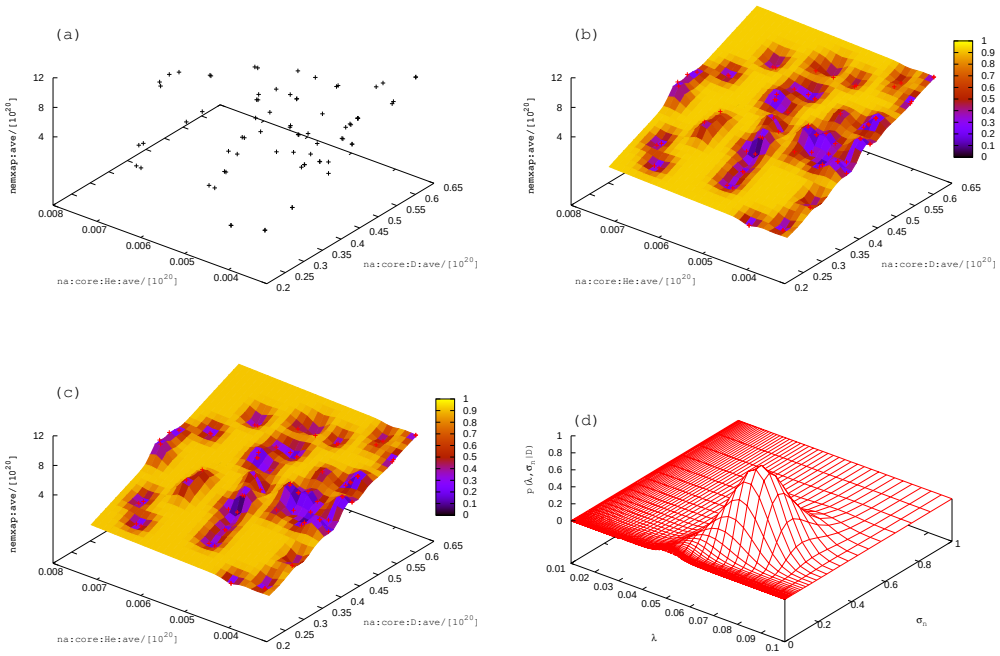

Figure 4: Scrape-off layer plasma simulation: Predictive mean for target 'nemxap:ave' and two dimensional input from 'na:core:D:ave' and 'na:core:He:ave'. 'ELM' is 'False' and only deuterium core densities above $10^{19}/\mathrm{m}^3$ and helium core densities above $3.1 * 10^{17}/\mathrm{m}^3$. are considered making a data pool of 76 entries. (a) input data; (b) 31x31 target expectation values; (c) 31x31 target maximum aposteriori; (d) marginal likelihood for λ and σ_n. For violet (darker) points one is pretty sure about the prediction. Further experiments should take place for parameter settings at yellow (lighter) areas.

References

[1] D. Barber, *Bayesian Reasoning and Machine Learning*, Cambridge University Press, 2012.

[2] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1996.

[3] D. MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 2003.

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