Learning Curves and Bootstrap Estimates for Inference with Gaussian Processes: A Statistical Mechanics Study

Dörthe Malzahn$^{(1)}$, Manfred Opper$^{(2)}$

$^{(1)}$ *Informatics and Mathematical Modelling, Technical University of Denmark,*
Richard-Petersens-Plads Building 321, DK-2800 Lyngby, Denmark

$^{(2)}$ *School of Engineering and Applied Science / NCRG,*
Aston University, Birmingham B4 7ET, United Kingdom

(Dated: May 27, 2002)

Abstract

We employ the replica method of statistical physics to study the average case performance of learning systems. The new feature of our theory is that general distributions of data can be treated, which enables applications to real data. For a class of Bayesian prediction models which are based on Gaussian processes, we discuss Bootstrap estimates for learning curves.

PACS numbers: 84.35.+i,02.50.-r,05.20.-y,87.18.Sn

I. INTRODUCTION

Analyzing the ability of adaptive systems, such as neural networks to learn a rule from examples, has been a fascinating topic in Statistical Physics for many years [1, 2]. Tools from the physics of disordered systems, especially the replica method [3], were found to be well designed for computing the learning performance of model systems. This is measured by the average *generalization error*, i.e. the error on data not previously seen by the system. Many interesting models and learning scenarios have been studied in such a way [1, 2]. However, usually strong idealizations about the probability distributions of data had to be made in order to allow for *exact results* valid in the "thermodynamic limit" of a high dimensional data space. Hence, the question remains, whether the sophisticated methods of statistical physics could provide more than *qualitative* insights into the performance of a learning method. Could one get practically useful *quantitative* results for concrete problems?

In this paper, we would like to demonstrate such a possibility. Based on the replica theory and a variational technique, we develop an approximate theory for the computation of generalization errors, which avoids previous simplifications. Rather than making artificial assumptions on the data distribution, which, in practice, is usually unknown, we apply our method to the *empirical distribution* of a given set of training data. In such a way we are able to approximate simple *Bootstrap* estimates for error measures which might be helpful to assess the performance of learning algorithms in real applications.

In the following, we will formulate our theory for the so-called Gaussian process models, which provide a flexible, widely applicable concept and have attracted considerable attention in the machine learning community [4–7].

II. GAUSSIAN PROCESS MODELS

Gaussian process (GP) models are a non-parametric Bayesian approach to supervised learning. Their goal is to learn a mapping $f(x)$ from inputs $x \in R^d$ to outputs y based on a set D of m example data pairs (x_i, y_i), $i = 1, \ldots, m$. The performance of any function f on the training data is measured by a training energy $E[f; D] = \sum_{i=1}^m h(f(x_i), y_i)$ for some loss function h. In a probabilistic formulation, the "Boltzmann factor" $e^{-E[f;D]} \propto \prod_{i=1}^m P(y_i|f(x_i))$ is proportional to the *likelihood* of the training data based on the assumption that f is the true generator of data outputs. In a Bayesian approach, one incorporates prior knowledge about the plausibility of different functions f by assigning *a priori statistical weights* to them. This helps against overfitting when the class of model functions is rich. It avoids e.g. solutions which perfectly fit though noisy data, but become to "wiggly" to predict well on novel inputs. For parametric models (e.g., when functions are restricted to be linear), this *prior* probability is usually induced by a probability over the parameters by which the functions are defined. In contrast, for GP models, one defines directly a Gaussian measure $\mu[f]$ over the space of functions f. This has the advantage that the space of functions with nonzero prior probability is usually infinite dimensional allowing for an unbounded complexity in modeling.

Using Bayes' theorem, the final, *posterior* information about the statistical weight of a function f of being responsible for the observed training examples D is given by the Gibbs distribution

$$\mu_m[f] = \mu[f]\; e^{-E[f;D]}/Z_m \;, \tag{1}$$

where Z_m is a normalizing partition function. Eq. (1) can be used to make predictions $\hat{f}(x|D)$ on novel inputs x. We will restrict ourselves to predictors which are given by the posterior mean $\hat{f}(x|D) \doteq \langle f(x) \rangle$, where angle brackets denote averages over (1).

The Gaussian measure $\mu[f]$ (assuming a zero mean), is fully specified by the *correlation kernel* $K(x, x') \doteq \int d\mu[f]\; \{f(x)f(x')\}$ which must be supplied by the user. It encodes the a priori assumptions about the typical variability of functions f with the input x. A frequently used kernel is the *radial basis function* (RBF) kernel $K(x, x') \propto \exp[-||x - x'||^2/l^2]$.

III. REPLICA ANALYSIS

We study the average learning performance of Bayesian inference using GP models where the average is over different drawings of training data sets $D = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ and all examples (x_i, y_i) are generated independently from the same distribution $p(y, x) = p(y|x)p(x)$. In the following, we denote data averages by square brackets $[\ldots]_D$ and compute the averaged free energy $F = [-\ln Z_m]_D$ which serves as a generating functional for useful data averaged observables. Using the replica approach [3], we get $F = -\lim_{n\to 0} \frac{\partial \ln[(Z_m)^n]_D}{\partial n}$ with

$$[(Z_m)^n]_D = \int \prod_{a=1}^{n} d\mu[f_a] \prod_{i=1}^{m} \left[\exp\left\{ -\sum_{a=1}^{n} h(f_a(x), y) \right\} \right]_{y,x} \; . \tag{2}$$

Eq. (2) uses the definition of the partition function Z_m as normalizing factor of Eq. (1) and represents the n-fold product of Z_m by introducing an index $a = 1, \ldots, n$ to the n integration variables f. We exploited the statistical independence of the data (remember that $E[f; D] = \sum_{i=1}^{m} h(f(x_i), y_i)$) where $[\ldots]_{y,x}$ denotes an average with respect to the data density $p(y, x)$.

To facilitate subsequent analytical calculations, we use a ”grand canonical” formulation

$$\Xi_n(\nu) \doteq \sum_{m=0}^{\infty} \frac{e^{\nu m}}{m!} [(Z_m)^n]_D = \int \prod_{a=1}^{n} d\mu[f_a] \; e^{-H} \; . \tag{3}$$

This has the advantage that the grand canonical partition function $\Xi_n(\nu)$ of the n times replicated system can be expressed in terms of a Hamiltonian $H = [\mathcal{H}(\{f_a\}, x)]_x$ which is the average of a *purely local* Hamiltonian density

$$\mathcal{H}(\{f_a\}, x) = -e^{\nu} \left[\exp\left\{ -\sum_{a=1}^{n} h(f_a(x), y) \right\} \right]_{y|x} \tag{4}$$

where the expectations $[\ldots]_x$, $[\ldots]_{y|x}$ are taken with respect to the input density $p(x)$ and the conditional output density $p(y|x)$, respectively. Eq. (3) represents a ”poissonized” version of our model where the number of examples m fluctuates around a fixed average value e^{ν}. For large ν, we can approximate the sum in Eq. (3) by its dominating term and identify m with the expected value $m = \frac{\partial \ln \Xi_n(\nu)}{\partial \nu}$. In the limit $n \to 0$, this gives the simple relation $\nu = \ln m$ and we recover the original (canonical) free energy as $F = -\lim_{n\to 0} \frac{\partial \ln \Xi_n(\ln m)}{\partial n}$.

For a variety of neural network models, it has been shown that partition functions can be calculated exactly and in closed form in the thermodynamic limit of infinite input dimensionality when data distributions $p(x, y)$ are highly symmetric. Our main interest however in this paper is the problem of *realistic* data distributions which may often be highly structured. Therefore, we do not aim at discussing artificial solvable cases, but resort to a variational approximation of the free energy which can be applied to more realistic scenarios. We replace the Hamiltonian $H = [\mathcal{H}(\{f_a\}, x)]_x$ by a trial replica Hamiltonian $H_0 = [\mathcal{H}_0(\{f_a\}, x)]_x$ which minimizes the variational bound [8]

$$-\ln \Xi_n(\nu) \le -\ln \int \prod_{a=1}^{n} d\mu[f_a] \; e^{-H_0} + \langle H - H_0 \rangle_0 \; . \tag{5}$$

The brackets $\langle \ldots \rangle_0$ denote an average with respect to the distribution $\prod_{a=1}^{n} \mu[f_a] e^{-H_0}$. For Gaussian measures $\mu[f_a]$, a local trial Hamiltonian $H_0 = [\mathcal{H}_0(\{f_a\}, x)]_x$ of the form

$$\mathcal{H}_0(\{f_a\}, x) = \sum_{a \le b} \hat{Q}_{ab}(x) f_a(x) f_b(x) + \sum_{a} \hat{R}_a(x) f_a(x) \tag{6}$$

is an appropriate choice. The resulting Gaussian approximation is expected to become asymptotically exact for training energies $h(f(x), y)$ that are smooth functions of $f(x)$, when the Gibbs distribution (1) becomes increasingly concentrated around its mean for large m. The most important feature of Eq. (6) is the explicit dependence of the variational parameters $\hat{Q}_{ab}(x)$ and $\hat{R}_a(x)$ on the input x. We will see later that the variationally optimal set of functions $\hat{Q}_{ab}(x)$, $\hat{R}_a(x)$ are given by specific averages $[\ldots]_{y|x}$ over the conditional output density $p(y|x)$.

The variational free energy (5) is an explicit functional of the local moments

$$R_a(x) \doteq \langle f_a(x)\rangle_0 \qquad Q_{ab}(x,x) \doteq \langle f_a(x) f_b(x)\rangle_0 \tag{7}$$

which replace the simpler order parameters of previous replica calculations for learning problems [1] by *order parameter fields*. (Note that $Q_{ab}(x,x)$ is a special case of the general two point function $Q_{ab}(x,x')$.) Straightforward variation of (5) yields $\frac{\delta\langle\mathcal{H}-\mathcal{H}_0\rangle_0}{\delta p} = 0$ for the optimal set $p = \hat{R}_a(x)$, $\hat{Q}_{ab}(x,x)$ of variational parameters where the variation δ acts only on the Gaussian measure $\langle\cdots\rangle_0$. The latter is fully characterized by its moments (7) and we obtain the variational equations

$$\frac{d\langle\mathcal{H}\rangle_0}{dR_a(x)} = \hat{R}_a(x) \qquad \frac{d\langle\mathcal{H}\rangle_0}{dQ_{ab}(x,x)} = \hat{Q}_{ab}(x)\,. \tag{8}$$

We solve Eq. (8) under the assumption of replica symmetry. This amounts to setting $\hat{R}_a(x) = \hat{R}(x)$, $\hat{Q}_{aa}(x,x') = \hat{Q}_0(x,x')$ for all a as well as $\hat{Q}_{ab}(x,x') = \hat{Q}(x,x')$ for all $a \neq b$. The order parameters Eq. (7) inherit this symmetry and we obtain three order parameter fields $R(x)$, $Q_0(x,x')$ and $Q(x,x')$. They have simple physical interpretations in terms of averages over example data sets D. To outline the general argument, let us briefly summarize the main steps of our theory. So far, we replaced (2) by (3) arguing that both formulations become identical for sufficiently large sample sizes m and, as a final step, approximated (3) by a variationally optimized Gaussian density (6). Reversing the argument yields

$$\lim_{n\to 0}\langle f_a(x)\rangle_0 \;\approx\; \lim_{n\to 0}\left[(Z_m)^{n-1}\int d\mu[f_a]e^{-\sum_{i=1}^m h(f_a(x_i),y_i)} f_a(x)\right]_D = [\langle f(x)\rangle]_D \tag{9}$$

where we used the definition of Z_m and the input x is statistically independent with respect to all inputs in the training sets D over which we average. The replica limit $n \to 0$ restores the normalizing factor $(Z_m)^{-1}$ to the posterior Gibbs distribution (1) and we recover the expression for the trained model $\langle f(x)\rangle$ as mean of (1). We can now interpret $R(x)$ as theoretical estimate of the mean prediction of the trained model at a test input x, $R(x) \approx [\langle f(x)\rangle]_D$. By a similar line of reasoning we see that $Q_0(x,x') \approx [\langle f(x)f(x')\rangle]_D$ (where $\langle\cdots\rangle$ denotes an average over the posterior (1)) and $Q(x,x') \approx [\langle f(x)\rangle\langle f(x')\rangle]_D$, i.e., we can interpret $G(x,x') \doteq Q_0(x,x') - Q(x,x')$ as theoretical estimate for the average posterior correlation function and $V(x,x') \doteq Q(x,x') - R(x)R(x')$ as theoretical estimate for the covariance of the trained model under the data average. Using simple properties of Gaussian measures, we obtain from (6), (7)

$$\begin{aligned} R(x) &= -[G(x,x')\hat{R}(x')]_{x'} && (10) \\ V(x,x') &= -[G(x,x'')G(x',x'')\hat{Q}(x'')]_{x''}\ , && (11) \end{aligned}$$

and the posterior correlation function $G(x,x')$ is expressed as the inverse operator

$$G = \left(K^{-1} + u\right)^{-1} \qquad (12)$$

where K is the kernel integral operator and $u(x,x') \doteq p(x)\Delta\hat{Q}(x)\delta(x-x')$. Finally, $\Delta\hat{Q}(x) = (\hat{Q}_0(x) - \hat{Q}(x))$. The order parameter equations (10)-(12) must be solved together with the variational equations (8).

We will now specialize to GP models for regression. The model has training energy

$$h(f;y) = \frac{1}{2\sigma^2}(y - f(x))^2\ , \qquad (13)$$

where σ^2 accounts for the noise in the output labels. The variational equations (8) become

$$\begin{aligned} \Delta\hat{Q}(x) &= \frac{m}{(\sigma^2 + G(x,x))} && (14) \\ \hat{R}(x) &= -[y]_{y|x}\Delta\hat{Q}(x) && (15) \\ \hat{Q}(x) &= -[(R(x)-y)^2 + V(x,x)]_{y|x}\frac{\Delta\hat{Q}^2(x)}{m} && (16) \end{aligned}$$

with $\Delta\hat{Q}(x) = (\hat{Q}_0(x) - \hat{Q}(x))$.

IV. LEARNING CURVES

Learning curves display data averaged measures of the generalization properties of the trained model $\langle f(x)\rangle$ as a function of the number m of data points in the training sets D. GP models are Bayesian and therefore provide an intrinsic measure of "prediction uncertainty" which is given by the average variance $[G(x,x)]_x \approx [\langle f^2(x)\rangle - \langle f(x)\rangle^2]_{x,D}$ of the posterior distribution (1). Hence, a simple example of a learning curve can be obtained by computing the average value $[G(x,x)]_x$ as a function of m. We note that for the model (13), the posterior variance is found to be *independent* of the output labels.

We can employ our theory to calculate the average prediction performance $\varepsilon = [[g(\langle f(x)\rangle; x, y)]_D]_{x,y}$ of the trained model $\langle f(x)\rangle$ on test points (x,y) using more complex measures g. This can be accomplished by data averaging the Taylor expansion of g. Inserting the definition of the trained model $\langle f(x)\rangle$ as posterior mean into a polynomial $[(\langle f(x)\rangle - y)^r]_D$ generalizes Eq. (9) to $[(\langle f(x)\rangle - y)^r]_D \approx \lim_{n\to 0}\langle \prod_{a=1}^r (f_a(x) - y)\rangle_0$ where we used the Gaussian approximation (6) to compute the replica average. The standard performance measure for regression models is the average square generalization error $\varepsilon = [[(\langle f(x)\rangle - y)^2]_D]_{x,y}$

$$\varepsilon \approx \lim_{n\to 0}[\langle (f_1(x) - y)(f_2(x) - y)\rangle_0]_{x,y} = [(R(x) - y)^2 + V(x,x)]_{x,y} \tag{17}$$

which we obtain as expression of the the order parameter functions $R(x)$ and $V(x,x)$. Note however, that Eq. (17) can be computed explicitly only if the exact distribution $p(x,y)$ of the data was given.

V. APPLICATION TO THE EMPIRICAL DISTRIBUTION

For a concrete data set, the underlying distribution is not known. Hence, we propose to use our replica approach on *the empirical distribution* instead. This amounts to setting

$$p(x,y) = \frac{1}{N}\sum_{i=1}^{N} \delta(x - x_i)\, \delta(y - y_i)\ , \tag{18}$$

where N is the total number of examples (x_i, y_i) available in the data set. Applying our replica theory to the empirical distribution (18) allows us to predict *Bootstrap averages*, which correspond to training the GP algorithm on randomly drawn (with replacement!) *subsets* D of size m out of all N example data. The generalization error is defined as the average error on all N points $\varepsilon = \frac{1}{N}\sum_{i=1}^{N}\varepsilon(i)$. Note that the sample size m is a free parameter for this procedure.

What do we expect to gain from such computations? First, we can simply test the validity of our variational replica theory on real data by comparing the theory with an actual experiment of resampling and training on subsets. More important, Bootstrap techniques can give practically useful predictions for the performance of learning algorithms for unseen test data [9]. The replica approach could then give an efficient technique to approximate the outcome of real, but more time consuming resampling experiments. Of course, the generalization error computed from our *simple* definition of a Bootstrap will usually underestimate the "true" generalization error on a hold out test set, when $m \geq N$, because a substantial amount (on average $N(1 - e^{-\frac{m}{N}})$) of the available data points have been used for training. However, more clever Bootstrap techniques can be designed to cope with this problem [9].

Our replica results, which were derived for arbitrary distributions are easily translated to the empirical distribution (18). Functions of inputs x become N dimensional vectors, integrals are replaced by sums and operators are simply matrices. Eq. (12) becomes the matrix equation

$$G = (I + Ku)^{-1} K \tag{19}$$

for matrix $G_{ij} \doteq G(x_i, x_j)$, kernel matrix $K_{ij} \doteq K(x_i, x_j)$ and the diagonal matrix $u_{ii} = \frac{\Delta \hat{Q}(x_i)}{N}$. $u_{ij} = 0$ for all $i \neq j$. Eq. (19) and (14) are coupled and can be easily solved by iteration. Details are given in the appendix. The average posterior variance is obtained by taking the trace over the matrix G and dividing by N. Eq. (10), (11) together with Eq. (15), (16) allow us to compute the local bias $\vec{R} = (R(x_1), \ldots, R(x_N))$ and the local variance $\vec{V} = (V(x_1, x_1), \ldots, V(x_N, x_N))$ of the predictor from which the generalization error can be computed.

VI. NUMERICAL EXPERIMENTS

We have compared the replica results with simulations of the GP algorithm on two regression benchmark data sets [10]. For the Boston data set, the task is to predict house prices in the Boston Massachusetts area from census data (input dimension $d = 13$). For the Abalone data set, the task is to predict the age of Abalone from physical measurements (input dimension $d = 10$ where we translated the gender encoding into binary inputs M/F/I = 100/010/001). The GP algorithm used a RBF kernel $K(x, x') = \exp[-\sum_i^d (x_i - x_i')^2 / l^2 v_i]$ which assumes a characteristic correlation length $l^2 v_i$ for the i-th input component. A good learning performance was obtained with $l^2 = 10$ (Abalone data), $l^2 = 147.1$ (Boston data) where the scaling v_i was set to the component-wise variance of the inputs (Abalone data) or the square root of this variance (Boston data). The latter realizes effectively different characteristic scales and was found to yield a better generalization performance on the more structured Boston data set. The noise was set to $\sigma^2 = 0.1$ (Abalone data), $\sigma^2 = 0.01$ (Boston data). Figures 1,2 show learning curves for the posterior variance (left) and mean square generalization error

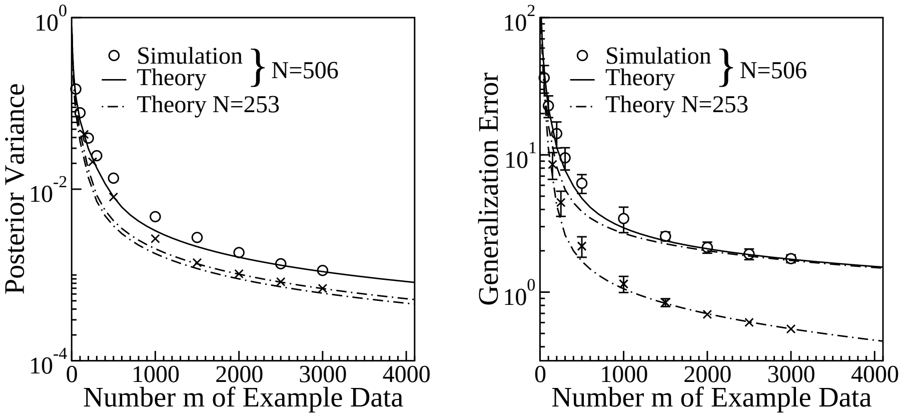

FIG. 1: Learning curves for Boston housing data (506 data). *Simulation: Symbols.* Training data are drawn with replacement from a pool of N data. The predictor was tested on these N data. Circle: Whole set, $N = 506$. Cross: First half of set, $N = 253$. *Theory: Lines.* Learning curves were estimated using all data (solid line) as well as only the first or second half of the data set (two dot-dashed lines).

$[\varepsilon(x, y)]_{(x,y)} \approx \frac{1}{N} \sum_{i=1}^{N} \left((R(x_i) - y_i)^2 + V(x_i, x_i)\right)$ (right). The lines are theoretical results, whereas the symbols are obtained from simulations. For the solid lines in Fig. 1, all available data (i.e. $N = 506$ examples) of the *Boston housing* data set are used to compute theoretical learning curves. They are in good agreement with simulations (circles) obtained by training on a randomly sampled (with replacement) subset of m data and testing on all N examples. Note, that we have highly oversampled the rather small data set, which can be understood as an attempt to *extrapolate* to the “true” learning curve which would be based on the complete knowledge of the distribution. To investigate the effect of the restricted size of the full dataset on the learning curves, we have repeated the experiment twice on full sets of $N = 253$ data points using the first and second half of the Boston dataset, respectively. Again, replica theory and simulations are in fairly good agreement. However, the generalization error ε (Fig. 1, right) (dot-dashed lines) differs substantially between the two sets. Theory and simulations (crosses and lower dot-dashed line) based on the first half of the data set present a poor forecast of the learning performance measured on the bigger data set (circles and solid line). On the other hand, the second half of the data set (upper dot-dashed line) seems to represent the statistics of the full data set much better. The regression model can learn some sets of outputs with significantly smaller square errors than others. This has a strong effect on the simple Bootstrap estimate of generalization error due to the optimistic bias caused by having many common output labels

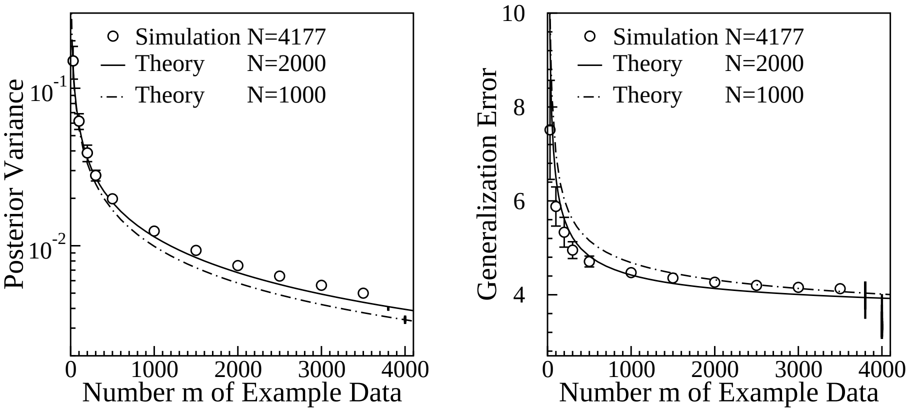

FIG. 2: Learning curves for Abalone data (4177 data). *Symbols*: Simulation using the whole data set (see Fig. 1). *Lines*: Theory. Learning curves as estimated from random subsets of 2000 data (solid lines and bar at $m = 3800$) and 1000 data (dot-dashed lines and bar at $m = 4000$). Bars mark the range covered by 20 repetitions on random subsets of the respective size.

for training and testing. In general, things look better for the posterior variance (Fig. 1, left). This is possibly due to the fact, that (for Gaussian regression) the posterior variance does not depend on the output labels y_i but only on the input data x_i. Both halves of the data set seem to supply quite similar information about the input statistics (dot-dashed lines). The learning curves also resemble more the one obtained from the bigger dataset.

We have conducted similar experiments for the much larger *Abalone* set [10]. In Fig. 2, simulations (circles) were performed with respect to all available data ($N_0 = 4177$ points), whereas theoretical learning curves (lines) were calculated based on different fractions ($N = 1/4N_0$ (dot-dashed line), $N = 1/2N_0$ (solid line)) of all available data. The respective subsets were chosen at random from the whole data set *without* replacement. To show how wide results scatter, we display the range covered by 20 consecutive runs by bars at $m = 4000$ (using $N = 1000$ data) and at $m = 3800$ (using $N = 2000$ data). Results for the posterior variance (Fig. 2, left) are highly consistent for the different runs, the bars are barely visible. Working with smaller data sets leads to a weak underestimation of the posterior variance. The Abalone data set seems to contain a relatively small subset of data (less than one fourth) which is responsible for large generalization errors. These difficult examples have a fair chance to be underrepresented when we choose 1000 data at random but are quite well represented by draws of 2000 data. Learning curves (Fig. 2, right) on the basis of 1000 data points can already be regarded as very informative. It is important to note that all displayed theoretical learning curves have been obtained computationally much faster than their respective simulated learning curves.

VII. CONCLUSION

Learning curves monitor the average learning performance of an adaptive system with respect to the amount of training examples. Using the tools of statistical mechanics in combination with a variational method, we have shown how to compute theoretical learning curves for Gaussian process models on random data whereby the underlying data distribution is a free parameter in the theory. This makes our theory suitable for applications on *real data*. We compare our theoretical predictions with simulations and demonstrate that our theory makes reliable predictions for Bootstrap-like learning experiments. The theory is much faster to compute than the actual simulated learning curve. Hence, we can use the method for a fast approximate exploitation of Bootstrap ensembles. So far, the theory is developed only for rather simple Bootstrap estimates. However, the encouraging preliminary results motivate the development of a replica approach for more complex Bootstrap estimates.

The central ideas of our theory can be applied to the analysis of a broad range of learning algorithms where one might replace the Variational Gaussian approximation by alternative approximation methods. One particular strength of the theory is that it avoids the need for explicit analytical formulas for the trained model by deriving its properties from a suitably constructed posterior distribution.

Acknowledgement: This work was supported by EPSRC grant GR/M81601.

VIII. APPENDIX: COMPUTING THE POSTERIOR COVARIANCE G

We first derive an asymptotic result for the diagonal elements $G(x, x)$, valid in the limit of large training data sets $m \to \infty$. In this limit, the "potential" u (which grows with m, see Eq. (14)) in the operator $G = \left(K^{-1} + u\right)^{-1}$, Eq. (12), becomes dominating. Motivated by similar treatments of such a "quasi classical" limit in quantum mechanics, we neglect the non-commutativity of the operators u and K^{-1}. This approximation becomes exact if u is a constant.

An integral representation of the inverse of an operator leads then to the approximation

$$
\begin{aligned}
G(x,x) &\approx \int\limits_0^\infty d\beta\; e^{-\beta u(x)} \langle x | e^{-\beta K^{-1}} | x \rangle \\
&= \sum_k \frac{|\langle x|\phi_k\rangle|^2}{\lambda_k^{-1} + u(x)} \,. \qquad (20)
\end{aligned}
$$

ϕ_k are the eigenfunctions of the operator K^{-1} with their respective eigenvalues λ_k^{-1} and $u(x) \doteq p(x)\Delta\hat{Q}(x)$. When working with the empirical distribution (18), ϕ_k and λ_k must be replaced by the eigenvectors and eigenvalues of the empirical kernel matrix K which is calculated from the set of input data.

Comparing with the variational equation (14), we see that we have obtained a nonlinear equation for the diagonal elements $G(x,x)$ which can be solved independently for each x by a simple one-dimensional root finding procedure. All off-diagonal elements and the final nonasymptotic values of the diagonal elements can be obtained by iterating Eq. (19) with (14) using (20) as an initialization. The procedure is found to be fast and stable, and, when m is large, requires only very few iterations of Eq. (19) to achieve convergence.

[1] Engel, A.; Van den Broeck, C. *Statistical Mechanics of Learning*. Cambridge University Press, 2001.

[2] Nishimori, H. *Statistical Physics of Spin Glasses and Information Processing*. Oxford Science Publications, 2001.

[3] Mézard, M.; Parisi, G.; Virasoro, M. A. *Spin Glass Theory and Beyond*. Lecture Notes in Physics **9**. World Scientific, 1987.

[4] Wahba, G. *Splines Models for Observational Data*. Series in Applied Mathematics **59**. SIAM: Philadelphia, 1990.

[5] Neal, R. *Bayesian Learning for Neural Networks*. Lecture Notes in Statistics **118**. Springer, 1996.

[6] Bialek, W.; Callan, C. G.; Strong, S. P. *Phys. Rev. Lett.* **77**. pp 4693-4697, 1996.

[7] Williams, C. K. I.; Rasmussen, C. E. In: *Advances in Neural Information Processing Systems* **8**, pp 514-520, Touretzky, D. S.; Mozer, M. C.; Hasselmo, M. E., Ed.; MIT Press, 1996.

[8] Feynman, R. P.; Hibbs, A. R. *Quantum mechanics and path integrals*. Mc Graw-Hill Inc., 1965.

[9] Efron, B.; Tibshirani, R. J. *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability 57. Chapman & Hall, 1993.

[10] The datasets can be downloaded from `http://www1.ics.uci.edu/~mlearn/MLSummary.html`.