Accurate Solubility Prediction with Error Bars for Electrolytes: A Machine Learning Approach

J. Chem. Inf. Model., 47, 2, 407 - 424, 2007

DRAFT paper, find the original at

http://dx.doi.org/10.1021/ci600205g

Anton Schwaighofer[1†], Timon Schroeter[‡ †], Sebastian Mika[¶], Julian Laub[†], Antonius ter Laak[‖], Detlev Sülzle[‖], Ursula Ganzer[‖], Nikolaus Heinrich[‖], Klaus-Robert Müller[‡ †]

[†] Fraunhofer FIRST, Kekuléstraße 7, 12489 Berlin, Germany
[‡] Technical University of Berlin, Computer Science, Franklinstraße 28/29, 10587 Berlin, Germany
[¶] idalab GmbH, Sophienstraße 24, 10178 Berlin, Germany
[‖] Research Laboratories of Schering AG, Müllerstraße 178, 13342 Berlin, Germany

March 30, 2007

Abstract

Accurate in-silico models for predicting aqueous solubility are needed in drug design and discovery, and many other areas of chemical research. We present a statistical modelling of aqueous solubility based on measured data, using a Gaussian Process nonlinear regression model (GPsol). We compare our results with those of 14 scientific studies and 6 commercial tools. This shows that the developed model achieves much higher accuracy than available commercial tools for the prediction of solubility of electrolytes. On top of the high accuracy, the proposed machine learning model also provides error bars for each individual prediction.

1 Introduction

Aqueous solubility is of paramount importance to many areas of chemical research, such as medicinal chemistry, pharmacokinetics, formulation, agrochemistry[1] and environmental applications. In the drug design process, 50% of the failures[2] are due to an unfavorable ADMET profile (Absorption, Distribution, Metabolism, Excretion & Toxicity), often resulting from poor aqueous solubility.

A lot of research has thus been devoted to developing in-silico models to predict aqueous solubility directly from a compound's structure.[3–17] Yet, one can not expect global models to be sufficiently accurate.[18] Many physical factors with separate mechanisms are involved in the phase transition from solid to

[1]Corresponding author e-mail: anton@first.fraunhofer.de

solvated molecules. The aqueous solubility of electrolytes at a specific pH is especially hard[19,20] to predict—yet, many drugs are electrolytes.

In this paper we introduce a modern machine learning tool, the Gaussian Process model, and use it to develop accurate models for predicting water solubility at pH 7 (GPsol). This is done following standard machine learning protocols: Based on a "training" data set of solubility measurements for about 4,000 electrolytes and non-electrolytes, the goal is to build a learning machine that uses the statistical fine structure of the molecular descriptor space to predict solubility for *unseen* compounds. In this manner the "learning" (statistical inference) procedure allows to generalize from a given set of measurements to unseen data ("out-of-sample data"[21]).

A particular focus of our approach is to provide meaningful error bars, that is, that each prediction of the model is equipped with an individual confidence estimate. Our analysis shows that the trained model achieves a high performance gain over existing methods. Furthermore, the confidence estimates provide reliable estimates for the deviation of predicted solubility from true solubility.

1.1 Background: Machine Learning

Machine learning subsumes a family of algorithmic techniques with a solid statistical foundation that aim to find reliable predictions by inferring from a limited set of measurements. In computational chemistry, the data could be a compound for which we seek, e.g., a predictor for the property "water solubility", or whether the compound is a drug-like molecule.[22] A large variety of techniques have been developed in the machine learning and statistics communities to account for different prediction tasks and application areas.[21,23,24]

Machine learning techniques, for example Neural Networks[25,26] have already been used in computational chemistry.[13,27] In the last years, however, Neural Networks have been used somewhat less in engineering and science. Instead there has been an upsurge of interest in Support Vector Machines[21,28–31] (SVMs) for various domains, due to their ease in handling complex problems.

It is a key requirement for predictive models in this context to provide confidence estimates, such that users can assess whether they can trust the predictions made by the model. With that in mind, SVMs are not ideal for modelling solubility, since they can not provide theoretically well founded confidence estimates.[2] Instead, a Bayesian modelling approach is more suited. As one of the seminal works, Neal[33] has shown excellent results with a Bayesian approach to neural network learning.

The limiting case of Bayesian neural networks are the Gaussian Process (GP) models,[34,35] nonlinear regression models that are straight-forward to use and readily provide confidence estimates.[3] The Bayesian framework provides criteria (see Sec. 2.7.4) to automatically choose the "amount of nonlinearity", thereby circumventing problems such as the choice of architecture in neural network

[2] Platt[32] shows a simple heuristic for the case of SVM classification, yet no such heuristic is known for the case of support vector regression.

[3] The advantage comes however at the cost of higher computational complexity.

models. The authors have demonstrated a number of successful applications of GP models to different domains,[36–38] see also Rasmussen and Williams[34] for further references.

Previous work[39–41] has shown the applicability of Gaussian Process models to problems in computational chemistry, mainly for predicting log P from relatively small data sets. Yet, the full potential of this class of models has seemingly not been recognized. We believe that the principled approach to providing confidence estimates, combined with the ease of use, are the key advantages when applying Gaussian Process models to problems in computational chemistry.

1.2 Background: Water Solubility

Among the physico-chemical characteristics of chemical compounds, water solubility is a key parameter in drug discovery. Many processes affecting the systemic availability of a drug such as intestinal absorption, transport through the blood and tissue distribution are linked to the dissolved neutral fraction of administered drug material that is amenable to pass through biological barriers and/or distribute among body tissues before binding to the target protein. But even in less complex in vitro systems such as cell-free or cell-based biological assays used in high-throughput screening (HTS), insufficient solubility of the test compound may be a limiting factor to the biological response.

Meanwhile, progress in instrumentation and miniaturization allows for medium throughput measurements of thermodynamic solubilities under constant and well-defined conditions even for larger sets of compounds. This valuable source of experimental data can be used to establish tools for solubility predictions which are primarily applied to virtual compounds in synthesis planning, e.g. as one important criterion to decide on the most promising subset of structures for chemical synthesis from fully-enumerated combinatorial libraries with millions of structures. Solubility predictions are also used to decide for compounds with reasonable bio-pharmaceutical properties in compound purchasing campaigns. Given this scenario of applications, the primary goal of models for solubility prediction is speed—with a sufficiently high degree of accuracy to support the decisions mentioned above.

Predicting aqueous solubility directly from a compound's structure is a difficult exercise because so many physical factors with separate mechanism are involved in the phase transition from a solid form to a solvated solute molecule. In particular, the pH dependency of electrolytes in water, the temperature dependency of solubility and the effect of polymorphs in the solid state are challenging tasks for any predictive model aimed to be based on first physical principles.[4]

Despite these general difficulties, many attempts have been made to estimate water solubility applying more heuristic approaches, i.e., accepting the trade-

[4]Klamt et al.[42] published a model mostly based on quantum mechanical calculations. The problem of finding the correct crystal structure is avoided by including a QSAR-model for solid state properties. This type of model can be used for small numbers of compounds, but is computationally too demanding for HTS or library design (a single solubility prediction takes 2 hours on a 1 GHz CPU)

off between computational demand and general applicability of the predictive model. With a reasonable set of experimental high-quality data, regression methods provide an attractive and pragmatic approach for solubility models. These methods are based on a fragmental representation of molecular structure by combinations of molecular descriptors with different degrees of complexity (1D, 2D).

A number of different regression methods has been used for this purpose, neural networks being a particularly popular approach.[3–8,10–13,17] Linear methods were typically used as baseline methods when studying more complex methods.[3,7,10] Also, results obtained with support vector regression have been reported.[15,17,43,44]

So far, existing models typically have two shortcomings. Firstly, most of them are only applicable to molecules in their neutral form (i.e., they predict intrinsic solubilities, see Sec. 2.2). Yet, a lot of current and future drugs are electrolytes, existing in ionized forms at physiologic pH. Secondly, to our knowledge, none of the currently available commercial tools can provide confidence intervals for each individual prediction. From the available literature, we could only identify one single paper[45] where this issue is taken into account.[5]

Our work addresses both of these issues. To be able to predict the aqueous solubility of electrolytes at physiologic pH, we included a large number of high quality measurements of solubility of electrolytic drugs and drug-like compounds in buffered solutions. In order to produce accurate predictions with individual confidence intervals, we employ a Bayesian modelling approach, as discussed in Sec. 1.1. In the Bayesian framework, uncertainty is an integral part of the modelling assumptions, thus confidence estimates can be given with a solid statistical foundation.

2 Methods and Data

2.1 Methodology overview

For each molecule, the 3D structure of one conformation in its neutral form is predicted using the program Corina.[46] From this 3D structure, 1,664 Dragon[47] descriptors are generated. Based on solubility measurements and molecular descriptors of a large set of compounds, a Gaussian Process (GPsol) model is fitted to infer the relationship between the descriptors and the solubility for a set of training data. When applying this model to a previously unseen compound, descriptors are calculated as described above and passed on to the trained Gaussian Process model, which in turn produces an estimate of the solubility together with a confidence estimate (error bar).

[5] Göller et al.[45] uses a heuristic that is based on the spread of predictions made by a number of differently trained neural networks. This heuristic can potentially lead to seriously underestimating the prediction error, in particular in regions far from the training data.

2.2 Different kinds of solubility

Aqueous solubility is defined as the maximum amount of compound dissolved in water under equilibrium conditions. For electrolytes this property is strongly pH dependent, so we need more precise definitions:

Buffer Solubility The solubility in a buffer solution at a specific pH is called buffer solubility or apparent solubility. It can be estimated using, e.g., the Henderson-Hasselbalch equation or combinations of multiple such equations. For drug-like molecules these corrections have been shown to be unreliable.[48]

Intrinsic solubility This is the solubility of a compound in its neutral form. Electrolytes can exist in different ionic forms, depending the the pH. Intrinsic solubility is only reached at a pH where almost 100 % of the compound is present in its neutral form.

Pure Solubility The pure solubility of a compound (sometimes also called native solubility) can be observed by adding the compound to pure (unbuffered) water. The pH of the solution typically changes during this process.

Kinetic Solubility Kinetic Solubility is the concentration when an induced precipitate first appears in a solution. This value is often much higher that the intrinsic solubility. Furthermore, it is more difficult to obtain reproducible results.[49,50]

2.3 Data preparation

Subsequently, we describe data sets of solubility measurements, obtained from different sources. Compounds that were originally present in more than one set were only retained in one set, such that all data sets are pairwise disjoint.

2.3.1 Data Set 1: Physprop and Beilstein

We extracted 34,314 measurements of aqueous solubility for 23,516 unique compounds from the Physprop[51] and Beilstein[52] databases. Restricting the range of temperatures to $15 \ldots 45°C$, excluding salts, and measurements for electrolytes at unknown pH values leaves 5,652 measurements for 3,307 individual compounds. All of these compounds are neutral.

2.3.2 Data Set 2: Flask

From an in-house database at Schering AG we extracted high quality flask measurements in the pH range between 7.0 and 7.4. Measurements indicating ranges or bounds were eliminated. Again, we restricted the range of temperatures to $15 \ldots 45°C$ and excluded salts, leaving 688 high quality flask measurements for 632 individual compounds. 549 of these compounds are electrolytes.

2.3.3 Data Set 3: Huuskonen

This is the well known dataset originally extracted from the Aquasol[53] and Physprop[51] databases by Huuskonen.[3] Ran et al.,[54] Tetko et al.[4] and Gasteiger et al.[5] later revised this set. We used the latest version containing measurements for 1311 compounds[6] availabe from `www.vcclab.org` (January 2006). This dataset has been used by numerous researchers[3–5,7,8,10,12–17,43,54] and is considered a benchmark set. The dataset contains measurements of pure solubilities,[55] i.e., the solubility observed when adding the compound to unbuffered water.

2.3.4 Data Set 4: Flask Blind Test

For a second validation stage, Schering AG collected 536 flask measurements of drug candidates, using the same filters as for data set 3, Sec. 2.3.2. The set of compounds is disjoint from other data used in this study, and was never used for training in any form.

2.4 Training and Validation Setups

Based on the data sets described in Sec. 2.3, we employ different setups for model building. These setups, respectively the models built in these setups, all have different aims:

- In the "Flask" setup, we combine data that makes the model predictive for buffer solubility. This setup is the basis for the GPsol model that has been implemented for production use at Schering AG.
- In all other setups, we obtain models that are predictive for pure solubility or pure solubility of neutral compounds. These models are built for the purpose of comparing the performance of our modelling approach with the performance obtained by other methods or commercial tools.

The setups are described subsequently, a graphical summary is given in Table 1. In all cases, it was ensured that no molecules were used both for model training and evaluation, that is, that there were no duplicate molecules in training and test validation set.

Setup "Flask" Our ultimate modelling goal is to predict the solubility of biologically relevant compounds. A large fraction of these compounds are electrolytes (i.e., they can exist in solution in a number of ionic forms). The model should be predictive for the *buffer solubility* in the pH-range of 7.0 through 7.4. Neutral compounds can also be used here, since their buffer solubility is simply independent of the pH value. Thus, in the Flask setup, we combine the data set

[6] Due to problems with generating Dragon descriptors for this set, we could effectively only use 1290 of these compounds.

Physprop/Beilstein (solubility of neutral compounds), the subset of 704 neutral compounds from the Huuskonen data set and the Flask data set (buffer solubility at known pH 7.0 . . . 7.4).

After model validation, all data from the "Flask" setup were used to build a final GPsol model that is now implemented in a PCADMET toolbox at Schering AG.

Setup "Flask Blind Test" The final GPsol model, trained on data in the "Flask" setup, was evaluated by Schering AG on the Flask blind test data set. The experimental values were not disclosed to the modellers at Fraunhofer FIRST, in order to verify the predicted performance of the GPsol model.

Setup "Huuskonen" In order to obtain a model that predicts *pure solubility*, we combine the data sets Physprop/Beilstein (pure solubility of neutral compounds) and Huuskonen (pure solubility of neutral compounds and electrolytes).

Models built in the "Huuskonen" setup are used solely to compare the Gaussian Process methodology with the performance of commercial tools when predicting pure solubility.

Setup "Huuskonen Only" In the "Huuskonen Only" setup, we only use the Huuskonen data set, without adding other literature data.

Models built in the "Huuskonen Only" setup are used to compare the GP methodology with other methodologies reported in the literature, see Sec. 4.4.

Setup "Huuskonen Neutral" Predicting the solubility of neutral compounds is considered easier than predicting the solubility of electrolytes. To compare the performance of different tools on that task (see Sec. A.2), we combine data sets Physprop/Beilstein (solubility of neutral compounds, independent of pH) and the subset of the Huuskonen data that corresponds to neutral compounds.

Training and Validation Procedure In the setups "Flask", "Huuskonen", and "Huuskonen neutral", we compile training and validation set as follows: Training data are the Physprop/Beilstein data, plus half of the "Flask" (resp. "Huuskonen" and "Huuskonen neutral") data. After training the model, it is evaluated on the other half of "Flask" (resp. "Huuskonen" and "Huuskonen only") data. This is repeated with the two halves of the validation set exchanged. The overall procedure is then repeated 10 times with a different random split. Thus, in each of the 10 runs, model predictions for the full "Flask" (resp. "Huuskonen" and "Huuskonen neutral") validation set are generated, where each prediction is an out-of-sample prediction, made by a model that has not seen the particular compound in its training data.

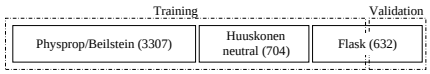
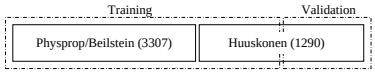
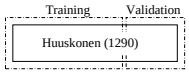
Setup	Prediction	Data
Flask	buffer sol, Sec. 3	
Flask blind test	buffer sol, Sec. 3.1.1	
Huuskonen	pure sol, Sec. 3	
Huuskonen neutral	pure sol, Sec. A.2	
Huuskonen only	pure sol, Sec. 4.4	

Table 1: Summary of the different setups that are used for performance evaluation. See Sec. 2.4 for a description, and Sec. 2.3 for details on the individual data sets

The setup "Huuskonen only" is evaluated in 3-fold cross-validation.[7] This means that the data is randomly split into three parts. Two of these are used to train a model, which is in turn evaluated on the third part of the data. This is repeated three times, such that each of the three parts is used as the test set once. Again, this is repeated 10 times with different random splits.

When computing error rates, as listed in Sec. 3.1, we average over the error rates of all 10 runs. For plotting model predictions versus measured value (such as in Figure 2), we only use out-of-sample predictions from one randomly chosen run.

2.5 Molecular descriptors

Initially, we used the full set of 1,664 Dragon descriptors. These include, among others, constitutional descriptors, topological descriptors, walk and path counts, eigenvalue-based indices, functional group counts and atom-centered fragments. A full list of these descriptors including references can be found online.[5,6] After a first modelling stage using all descriptors, it turned out that the computationally most expensive descriptors (Dragon blocks 5 and 13) can be omitted without significantly impacting the models performance. A prediction for $\log \mathrm{D}$ at pH 7, obtained from a model we previously trained on 20,000 $\log \mathrm{D}$ measurements, was found to slightly[8] increase the performance of our solubility model and was therefore included as a descriptor. An evaluation of the importance of individual

[7] 3-fold was chosen since it gives training sets of around 860 compounds. This matches well with most previous work, where models were trained on around $800 - -880$ compounds, see Table 4. Thus, we believe that the reported model performance is comparable with previous results.

[8] By including $\log \mathrm{D}$ information, the percentage of compounds with error ≤ 1 log unit increased by 1% at maximum, depending on the setup used

descriptors can be found in section 4.3.

2.6 Multiple Measurements

Due to merging data from different sources (see Sec. 2.3), many compounds have a number of associated solubility measurements. Multiple measurements could, in principle, be used directly for model fitting. We decided to first apply a preprocessing step, where a consensus measurement is found for each compound that has multiple measurements. This is necessary since the set of measured values is noisy and often prone to gross outliers (the range spanned by the measurements can be as large as 8 orders of magnitude).

Determining such a consensus value is an unsupervised learning problem, that can be tackled by the following decision rule: We consider the histogram of measurement values on a logarithmic scale, $\log S_W$. Such a histogram is characterized by two antagonist quantities, the spread of values (y-spread) and the spread of the bin heights (z-spread).

Several cases arise regularly: For small y-spreads (all measured values are similar), taking the median value is the most sensible choice. On the other hand, large y-spread with large z-spread hints at outliers. In such a case, we use the median of the values in the higher of the two bins as the consensus value. The worst case is given by two far apart bins of equal height (high y-spread and zero z-spread). In this case we omit the compound altogether, since we have equally strong evidence for the conflicting measurements.

We found that 0.5 log units is a suitable value for the threshold between small and large spreads. Also, it has been noted that the measurement error for solubility values from different sources is typically 0.5 log units.[18]

2.7 Gaussian Process Models

2.7.1 Overview

Gaussian Process (GP) models are techniques from the field of Bayesian statistics. O'Hagan[57] presented one of the seminal work on GPs, a recent book[34] presents an in-depth introduction.

Before going into detail, we first give a short overview of the procedure that underlies Bayesian learning with Gaussian Processes. This overview can be summarized with the three graphs shown in Figure 1. In GP modelling, we consider a family of functions that could potentially model the dependence of solubility (function output, denoted by y) from the descriptor (function input, denoted by $\mathbf{x}$). This space of functions is described by a *Gaussian Process prior*. 25 such functions, drawn at random from the prior, are shown in Figure 1(left). The prior captures, for example, the inherent variability of solubility as a function of the descriptor.

This prior belief is then updated in the light of new information, that is, the solubility measurements at hand. In Figure 1(middle), the available measurements are illustrated by three crosses. Principles of statistical inference are used

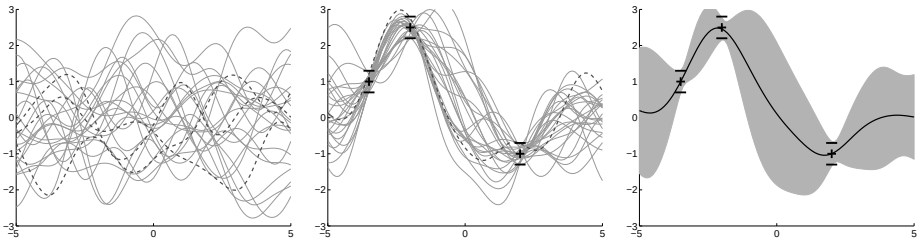


Figure 1: Modelling with Gaussian Process priors. *Left:* 25 samples from a Gaussian Process prior over functions, each plotted as $y = f(x)$. For illustration, we only consider functions for one-dimensional input x. *Middle:* After observing 3 data points (crosses), we only believe in functions from the prior that pass through a "tunnel" near the data depicted by the parallel lines above and below the crosses. These functions are in fact samples from the "posterior" distribution, and we have highlighted one of them (dashed line). *Right:* Summarizing representation of our beliefs about the plausible true functions, obtained from the 25 samples from the posterior shown the middle pane. For each input we compute the mean of these functions (black line) and the standard deviation. The shaded area encompasses ± 2 standard deviations around the mean.

to identify the most likely posterior function, that is, the most likely solubility function as a combination of prior assumptions and observed data (shown in the right panel of Figure 1). The formulation with a prior function class is essential in order to derive error bars for each prediction. Note also that the uncertainty increases on points that are far from the measurements.

2.7.2 Key ideas

The main assumption of a Gaussian Process model is that solubility can be described by an (unknown) function f that takes a vector of molecular descriptors as input, and outputs the solubility. We denote by $\mathbf{x}$ a vector of descriptors, which we assume to have length d. The solubility of a compound described by its descriptor vector $\mathbf{x}$ can thus be written as $f(\mathbf{x})$. It is assumed that f is inherently random.[9]

The Gaussian Process model is built from solubility measurements for a set of n compounds. For each of these n compounds, we have a descriptor vector, $\mathbf{x}_1 \dots \mathbf{x}_n$, (each of length d), together with a solubility measurement, $y_1, \dots y_n$. We additionally account for the fact that these measurements are not accurate, and assume that the n measured values are related to actual solubility by

$$y_i = f(\mathbf{x}_i) + \epsilon, \tag{1}$$

[9]The notation here is chosen to allow an easy understanding of the material, thus dropping, e.g., a clear distinction between random variables and their outcome.

where ϵ is Gaussian measurement noise[10] with mean 0 and standard deviation σ.

The name "Gaussian Process" stems from the assumption that f is a random function, where functional values $f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)$ for any finite set of n points form a Gaussian distribution.[11] This implies that we can describe the process also by considering pairs of compounds $\mathbf{x}$ and $\mathbf{x}'$. The covariance for the pair is given by evaluating the covariance function,

$$\mathrm{cov}(f(\mathbf{x}), f(\mathbf{x}')) = k(\mathbf{x}, \mathbf{x}'), \tag{2}$$

similar to kernel functions in Support Vector Machines.[21,29] Note, that all assumptions about the family of functions f are encoded in the covariance function k. Each of the possible functions f can be seen as one realization of an "infinite dimensional Gaussian distribution".

Let us now return to the problem of estimating f from a data set of n compounds with solubility measurements $y_1, \ldots, y_n$, as described above in Eq. (1). Omitting some details here (see Sec. B), it turns out that the prediction of a Gaussian Process model has a particularly simple form. The predicted function (solubility) for a new compound $\mathbf{x}_*$ follows a Gaussian distribution with mean $\bar{f}(\mathbf{x}_*)$,

$$\bar{f}(\mathbf{x}_*) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_*, \mathbf{x}_i). \tag{3}$$

Coefficients α_i are found by solving a system of linear equations,

$$\begin{pmatrix} k(\mathbf{x}_1, \mathbf{x}_1) + \sigma^2 & k(\mathbf{x}_1, \mathbf{x}_2) & \ldots & k(\mathbf{x}_1, \mathbf{x}_n) \\ k(\mathbf{x}_2, \mathbf{x}_1) & k(\mathbf{x}_2, \mathbf{x}_2) + \sigma^2 & \ldots & k(\mathbf{x}_2, \mathbf{x}_n) \\ \vdots & \vdots & & \vdots \\ k(\mathbf{x}_n, \mathbf{x}_1) & k(\mathbf{x}_n, \mathbf{x}_2) & \ldots & k(\mathbf{x}_n, \mathbf{x}_n) + \sigma^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \tag{4}$$

In matrix notation, this is the linear system $(K + \sigma^2 I)\alpha = \mathbf{y}$, with I denoting the unit matrix. In this framework, we can also derive that the predicted solubility has a standard deviation (error bar) of

$$\mathrm{std}\, f(\mathbf{x}_*) = \sqrt{k(\mathbf{x}_*, \mathbf{x}_*) - \sum_{i=1}^{n} \sum_{j=1}^{n} k(\mathbf{x}_*, \mathbf{x}_i) k(\mathbf{x}_*, \mathbf{x}_j) L_{ij}} \tag{5}$$

where L_{ij} are the elements of the matrix $L = (K + \sigma^2 I)^{-1}$.

[10] In the typical Gaussian Process model, all measurements share the same measurement noise. We will relax this condition in Sec. 2.7.5, to account for compounds where several agreeing measurements are available.

[11] For simplicity, we assume that the functional values have zero mean. In practice, this can be achieved easily by simply shifting the data before model fitting.

2.7.3 Relations to Other Machine Learning Methods

Gaussian Process models show close relations to a number of methods that have been considered in the machine learning community.[34] Here, we only focus on two particular models, that have already been used successfully in computational chemistry.

Support Vector Machines SVM Gaussian Process models share with the widely known support vector machines the concept of a kernel (covariance) function. Support vector machines (SVM) implicitly map the object to be classified, $\mathbf{x}$, to a high-dimensional feature space $\phi(\mathbf{x})$. Classification is then performed by linear separation in the feature space, with certain constraints that allow this problem to be solved in an efficient manner. Similarly, support vector regression[21] can be described as linear regression in the feature space. Gaussian Process models can as well be seen as linear regression in the feature space that is implicitly spanned by the covariance (kernel) function.[34] The difference lies in the choice of the loss function: SVM regression has an insensitivity threshold, that amounts to ignoring small prediction errors. Large prediction errors contribute linearly to the loss. GP models assume Gaussian noise, equivalent to square loss.

Note, however, that SVMs are completely lacking the concept of uncertainty. SVMs have a unique solution that is optimal under certain conditions.[21,28] Unfortunately, these assumptions are violated in some practical applications.

Neural Networks Radial Basis Function networks with a certain choice of prior distribution for the weights yield the same predictions as a Gaussian Process model.[34] More interestingly, it can be shown that a two-layer neural network with an increasing number of hidden units converges to a Gaussian Process model with a particular covariance function.[33]

2.7.4 Using GP Models

For predicting water solubility, we use a covariance function of the form

$$k(\mathbf{x}, \mathbf{x}') = \left(1 + \sum_{i=1}^{d} w_i (x_i - x_i')^2\right)^{-\nu} \tag{6}$$

(the "rational quadratic" covariance function[34]). $k(\mathbf{x}, \mathbf{x}')$ describes the "similarity" (covariance) of solubility for two compounds, given by their descriptor vectors $\mathbf{x}$ and $\mathbf{x}'$. The contribution of each descriptor to the overall similarity is weighted by a factor $w_i > 0$ that effectively describes the importance of descriptor i for the task of predicting water solubility.

Clearly, we can not set the weights w_i and the parameter ν a priori. Thus, we extend the GP framework by considering a superfamily of Gaussian Process priors, each prior encoded by a covariance function with specific settings for w_i. We guide the search through the superfamily by maximizing a Bayesian criterion

called the evidence (marginal likelihood). For n molecules $\mathbf{x}_1, \ldots, \mathbf{x}_n$ with associated measurements $y_1, \ldots, y_n$, this criterion is obtained by "integrating out" everything we don't know, namely all the true functional values $f(\mathbf{x}_i)$. Using vector notation for $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))$ and $\mathbf{y} = (y_1, \ldots, y_n)$, we obtain

$$\mathcal{L} = p(\mathbf{y} \,|\, \mathbf{x}_1, \ldots, \mathbf{x}_n, \theta) = \int p(\mathbf{y}|\mathbf{f}, \theta)\, p(\mathbf{f} \,|\, \mathbf{x}_1, \ldots, \mathbf{x}_n, \theta)\, \mathrm{d}\mathbf{f}. \tag{7}$$

This turns out to be

$$\mathcal{L} = -\frac{1}{2} \log \det(K_\theta + \sigma^2 I) - \frac{1}{2} \mathbf{y}^\top (K_\theta + \sigma^2 I)^{-1} \mathbf{y} - \frac{n}{2} \log 2\pi \tag{8}$$

Here, det denotes the determinant of a matrix, and $^\top$ is vector transpose. We use K_θ to explicitly denote the dependence of the covariance matrix K on a set of parameters θ of the covariance function.[12] Gradient ascent methods[13] can now be used to maximize $\mathcal{L}$ with respect to covariance function parameters θ and the measurement variance σ^2. References[34,35] present further details, and a discussion of problems such as multi-modality.

2.7.5 Noise groups

After inspection of the data and outlier removal (see Sec. 2.6), it turned out that for a number of compounds, multiple agreeing measurements were available. Clearly, if two or more agreeing measurements are available, one would put more confidence in the data for this compound. Following this intuition, we grouped compounds according to the number of available measurements (if they were consistent), and used a different standard deviation for the measurement noise ϵ, Eq. (1):

- Single measurement available (2,532 compounds): Noise std σ_1
- Two measurements available (1,160 compounds): Noise std σ_2
- ≥ 3 measurements available (242 compounds): Noise std σ_3

Similar to the procedure outlined above, the effective values for σ_1, σ_2 and σ_3 are estimated from the data by gradient descent on the evidence, Eq. (8).[14]

2.8 Evaluating Error Bars

Once the model is fitted, we assess its performance on a set of validation data (see Table 1), with compounds $\mathbf{x}_{*1} \ldots \mathbf{x}_{*M}$ and according measured solubility values $y_{*1}, \ldots, y_{*M}$. The Gaussian Process model outputs for each of these compounds a probabilistic prediction, that is, a predicted mean $\bar{f}(\mathbf{x}_{*i})$ and standard deviation std $f(\mathbf{x}_{*i})$.

[12] In the case of Eq. (6), $\theta = \{\nu, w_1, \ldots, w_d\}$ for a total of d descriptors.

[13] In our actual implementation, we use the Broyden-Fletcher-Goldfarb-Shanno method.[58]

[14] An alternative approach would be to put multiple measurements directly in the GP model, yet this would increase the computational cost of model fitting.

To evaluate the predicted confidences, we proceed as follows[15]: The model assumptions, Eq. (1) and Eq. (2), imply that the predictive distribution is again Gaussian. By employing the cumulative density function of the Gaussian, we can easily compute confidence intervals of the form "True value is within the interval $\bar{f}(\mathbf{x}_{*i}) \pm v$ with confidence c per-cent". We can subsequently count the percentage of points in the validation set where the true value lies within this interval. Ideally, for c per-cent of the validation data, the true solubility value should fall into the predicted c per-cent confidence interval. This check is made for several values for c.

3 Results

3.1 Accuracy

We analyzed the performance of GPsol for the two separate tasks of predicting buffer solubility and predicting pure solubility.

- Performance for predicting buffer solubility at pH 7.0 through 7.4 is listed in Table 2 (top). This is the "Flask" setup, see Sec. 2.4 for a description of the involved data sets. We also list the performance of the best two commercial tools for predicting buffer solubility: SimulationsPlus Admet Predictor[59] and ACD/Labs v9.0.[60]

- Prediction performance for pure solubility is evaluated in the Huuskonen setup, see Table 2 (bottom). We compare the performance with the EPI Suite[61] and SimulationsPlus Admet Predictor.[59]

The best commercial tools were chosen on the basis of an extensive evaluation that is given in Sec. A.2. Accuracy is measured using different criteria: Percentage of compounds for which the prediction error is below 1 (on log scale, meaning that the solubility is correct within one order of magnitude), mean absolute error MAE, and root mean square error RMSE.[16]

As one would expect, the two setups "Flask" and "Huuskonen" represent different levels of difficulty. GPsol achieves a performance that is comparable with the best commercial tools on the Huuskonen data set.[17] Yet, on the difficult task of predicting the solubility of drug-like electrolytic compounds, the performance of the GP model is clearly superior to that of the commercial tools. This also shows up clearly when plotting measured versus predicted[18] solubil-

[15] It has been suggested to use numeric criteria, such as log probability of the predictive distribution, for this purpose. Our experience suggests that these criteria can be misleading, they thus have not been used.

[16] $\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\mathrm{pred}_i - \mathrm{measured}_i|$, $\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathrm{pred}_i - \mathrm{measured}_i)^2}$

[17] It remains unknown to us whether the Huuskonen data has been used in the process of training the commercial tools we had evaluated. Thus, the evaluation we present here might be biased in favor of the commercial tools.

[18] Note that, as described in Sec. 2.4, the predictions in Figure 2 are always made by a Gaussian Process model that has not seen any training data for the compound it is being evaluated on.

Flask	Percent within ±1	RMSE	Mean abs. error
S+ ph7	57	1.34	1.03
ACDv9	64	1.16	0.90
GPsol	82	0.77	0.60

Huuskonen	Percent within ±1	RMSE	Mean abs. error
EPI	84	0.75	0.54
S+ native	93	0.56	0.43
GPsol	91	0.61	0.43

Table 2: Accuracy achieved by GPsol and the best two commercial tools. Top: Performance when predicting buffer solubility of electrolytes in the Flask setup, see Sec. 2.4. Bottom: Predicting pure solubility in the Huuskonen setup. Performance is listed in terms of "% predictions correct within one log unit", root mean square error RMSE and mean absolute error MAE. "S+ ph7" refers to SimulationsPlus AdmetPredictor,[59] "ACDv9" is ACD/Labs v9.0,[60] "EPI" is Wskowwin v1.41 in the EPI Suite.[61] See Sec. A.2 for an extended table, and tool details

ity, see Figure 2. On the Flask data, GPsol predictions are correct up to one log unit for about 82% of the compounds, whereas the best commercial tools, ACD/Labs v9.0[60] reaches only 64%. We believe that the performance in the Flask setup is most relevant for practitioners, since it contains mostly drug-like molecules.

A full list of results for all evaluated commercial tools and all performance measures can be found in Sec. A.2 in the appendix.

3.1.1 Second Validation

The final GPsol model, trained on all data in the Flask setup, went through an additional validation stage. GPsol predictions were compared against flask measurements for 536 drug candidates in recent projects at Schering AG. This additional validation was performed at Schering AG in a "blind test" scenario, without revealing the experimental results to the modellers at Fraunhofer FIRST.

Results of this validation are summarized in Figure 3. We can observe a small drop in performance of the GPsol model when compared to the results in Table 2. Still, there is a significant performance gain over the best commercial tool, ACD/Labs[60] v9.0.

3.2 Analysis of Error Bars

As one of its most distinctive features, GPsol is able to predict a standard deviation (error bar) for each of the water solubility predictions it makes. As described in Sec. 2.8, we evaluate the error bars by counting the percentage of points that fall into the c per-cent confidence interval, for several values of

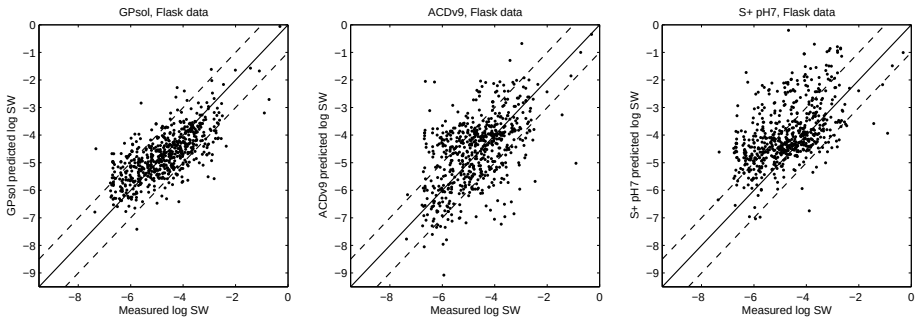
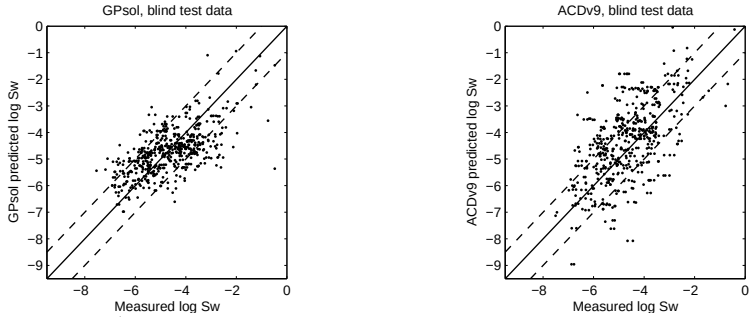


Figure 2: Predicted buffer solubility (y-axis) versus measured buffer solubility (x-axis) in the Flask setup. We compare GPsol (left), and predictions of buffer solubility at pH 7.4 from the two best commercial tools, ACD/Labs[60] v9.0 (middle) and the AdmetPredictor v1.2.3 of SimulationsPlus[59] (right). The dashed lines indicate the region where the prediction is within the range ± 1 from the measured value. Similar plots for other commercial tools can be found in Figure 2 in the appendix.



Blind test	Percent within ±1	RMSE	Mean abs. error
ACDv9	58	1.24	0.98
GPsol	75	0.92	0.73

Figure 3: Predicted buffer solubility (y-axis) versus measured buffer solubility (x-axis) on an independent validation set of 536 recent drug candidates ("blind test"). We compare GPsol (left graph), and predictions of buffer solubility at pH 7.4 from ACD/Labs v9.0[60] (right). The dashed lines indicate the region where the prediction is within the range ± 1 from the measured value

c. c per-cent of the points should optimally fall into the c per-cent confidence interval.

This analysis of error bars is shown in Figure 4. Optimal error bars would give the curve given in dashed in Figure 4. One can see that the predicted error bars do give a good match to the optimal curve. We can conclude that the predicted error bars can serve as reliable estimates of the model's confidence in its prediction. Still there are some compounds that fall outside the predicted confidence intervals at $c = 99\%$, that is, compounds for which the GP model was over-confident about its predictions. An analysis of these compounds is given in Sec. 4.1.

As a general tendency, one can notice that the error bars in the Huuskonen setup are much tighter than those in the Flask setup. The region in the chemical space that is spanned by the neutral compounds in the Huuskonen data set is also populated by data from the Physprop/Beilstein data set. The more data is available, the tighter error bars usually become. On the other hand, only few data is available in the region spanned by the Flask data (electrolytes). Despite the high prediction performance of the GPsol model, the model recognizes that its training data is only loosely distributed, and accordingly predicts larger error bars. Thus more data for buffer solubility will be a key to further improvements of the model.

4 Discussion

4.1 Outliers

Figure 4 showed a graphical evaluation of the quality of errorbars. These plots show that the predicted standard deviation, Eq. (5), does faithfully represent the expected deviation from the true solubility value. Still, there are a few compounds where the true solubility is outside the predicted 99% confidence interval. In this section, we will analyze these compounds in more detail. It turned out that the two major reasons for these mis-predictions are low data quality (e.g., contradictory measurements), and inherent limitations caused by the molecular descriptors (two compounds with different solubility, but almost identical descriptors).

Recall from Sec. 2.4 that our evaluation methodology generates 10 out-of-sample predictions for each compound in the data sets Flask and Huuskonen. The subsequent discussion focusses on compounds where the measured $\log S_W$ is outside the interval "predicted mean $\pm$ 3 standard deviations" ($\bar{f}(\mathbf{x}_*) \pm 3\,\mathrm{std}\, f(\mathbf{x}_*)$, corresponding to a 99.7% confidence interval) in more than 5 (out of the 10) cases.

Table 3 lists these compounds for the Huuskonen set[19], along with measured $\log S_W$ and predicted solubility $\bar{f}(\mathbf{x}_*)$.[20] Additionally, we list the three com-

[19] A discussion of outliers on the Flask data can not be provided, since these are in-house drug candidate molecules.

[20] $\bar{f}(\mathbf{x}_*)$ is given for the first of the 10 random splits where an error is made

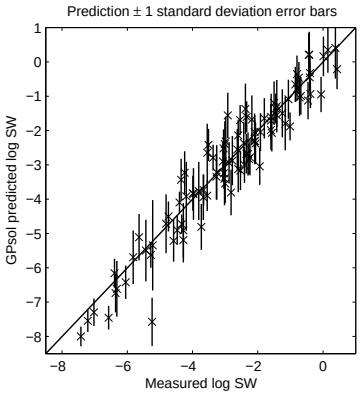
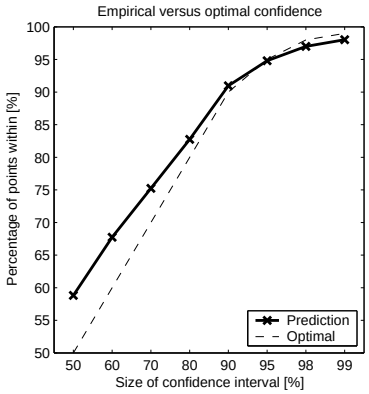
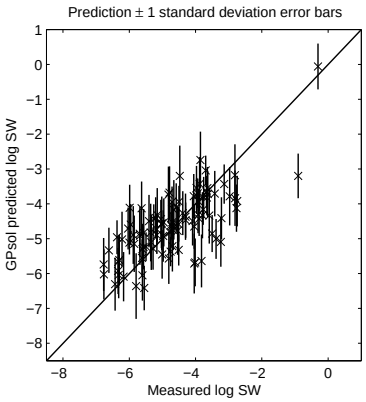
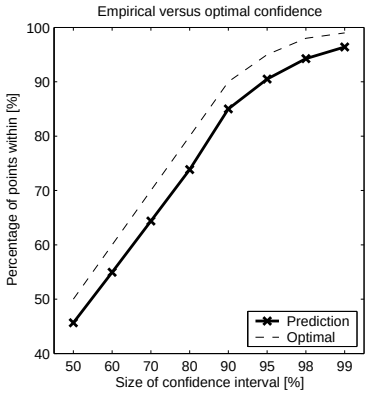
Setup	Error bars	Confidence intervals
Huuskonen		
Flask		

Figure 4: Evaluation of error bars that are generated by GPsol in the Huuskonen and the Flask setup. In the left column, we plot model prediction (indicated by $\times$) on the y-axis versus measured value on the x-axis, plus a vertical line ± 1 standard deviation, corresponding to a 68% confidence interval. To avoid over-cluttered plots, this is only done for a random subset of 100 compounds. In the right column, we plot the percentage of points that fall into the c per-cent confidence interval (x-axis) versus c (y-axis). See Sec. 3.2 for a discussion of the plots

pounds in the model's training data that are considered most similar to the test compound in terms of solubility, i.e., the compounds with highest value for the covariance function, Eq. (6).

From the set of mis-predictions, we could identify a subset where the solubilities listed by Huuskonen[3] are contradictory to measurements obtained from other data sources. In some cases, GPsol supports the alternative measurement. For some of the polychlorinated biphenyls listed in Table 3, we had one or two results indicating very low solubility ($\log S_W = -8$ to insoluble) and three or more results indicating a solubility that is several orders of magnitude higher (including the measurement from[3]). Suspecting transcription errors, we checked some of the original publications.[62,63] The authors indeed reported the high values that are included in the Physprop[51] and Beilstein[52] databases. At this point, it is unclear which measurement is correct.

In other cases, the "nearest neighbors" that are identified by Eq. (6) are misleading. A particular example are the hydroxypyridines listed in Table 3. Based on the availabe descriptors, *ortho*, *meta* and *para* are considered as almost identical. Yet, *meta*-hydroxypyridine has a solubility of $S_W \approx 0.35$mol/l while the *ortho*- and *para*-compound have a very high solubility of $S_W \approx 10$mol/l. This shows that the molecular descriptors used for predictive models set a strict limit to the achievable accuracy of any type of model.

In the last group of cases, the model prediction was simply wrong, and no clear reason for this mis-prediction could be identified.

Open points for further developments of the technology are the use of different noise models and descriptors. It has turned out that for some forms of data, the assumption of a Gaussian distribution for measurement noise is not fully met, and that performance can be improved by switching to a more heavy-tailed noise model.[64]

Table 3: Compounds from the Huuskonen data set that are mispredicted by GPsol. The table lists the compound and the nearest neighbors in terms of solubility, based on Eq. (6). We also list the experimental values for $\log S_W$ (given in mol/l) that were used for model building, obtained as outlined in Sec. 2.6. $\bar{f}$ denotes the model prediction for the respective compound, Eq. (3). See Sec. 4.1 for further discussions

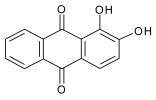
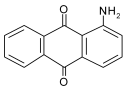
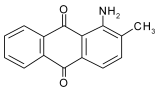
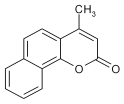
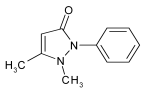
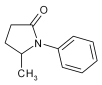
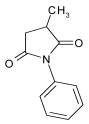
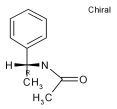
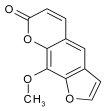
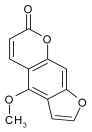
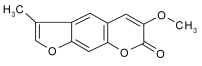
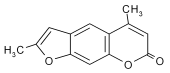
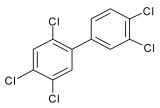
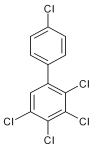
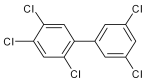
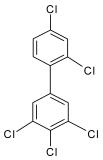
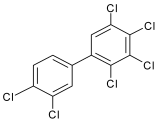
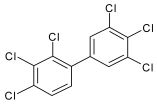
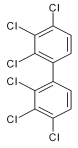
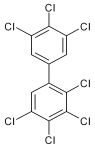
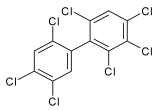
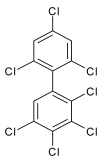
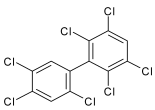
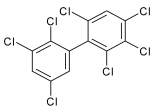
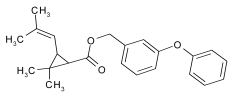
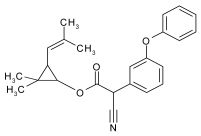
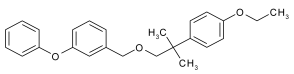
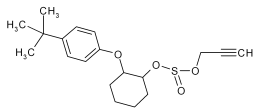
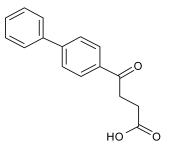
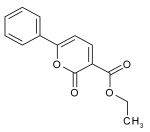
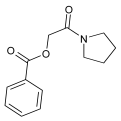
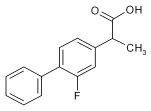
Test compound	Neighbor 1	Neighbor 2	Neighbor 3
			
predicted $\bar{f} = -5.2$ $\log S_W = -2.8$	$\log S_W = -5.9$	$\log S_W = -5.9$	$\log S_W = -2.6$
Comments: Alternative measurement for test compound: $\log S_W = -5.6$ at 25°C (from[52]) Measurements at higher temperatures[52] indicate that $\log S_W = -5.6$ is the more likely value			
			
predicted $\bar{f} = -1.38$ $\log S_W = 0.39$	$\log S_W = -0.85$	$\log S_W = -1.8$	$\log S_W = -1.1$
Comments: Alternative measurements for test compound: $\log S_W = -0.27$ (from[65]), $\log S_W = -0.56$ at 25°C (from[53])			
			
predicted $\bar{f} = 0.91$ $\log S_W = -0.46$	$\log S_W = 1.02$	$\log S_W = 1.02$	$\log S_W = 0$
Comments: Misleading neighbors: Based on the availabe descriptors, *ortho*, *meta* and *para* are considered as almost identical.			
			
predicted $\bar{f} = -6.5$ $\log S_W = -3.7$	$\log S_W = -7$	$\log S_W = -5$	$\log S_W = 5.2$
Comments: Alternative measurement[66] for test compound: $\log S_W = -6.91$. Reference[66] lists the solubility of neighbor 1 as $\log S_W = -7$			

Table 3: (continued)

Test compound	Neighbor 1	Neighbor 2	Neighbor 3
			
predicted $\bar{f} = -5.5$ $\log S_W = -7.4$	$\log S_W = -1.95$	$\log S_W = -7.5$	$\log S_W = -2.4$
Comments: Neighbor 1 was used in model building with $\log S_W = -1.95$. Original data were 5 measurements $\log S_W = \{-7.3; -1.8; -2.1; -2.3; -2.6\}$. Outlier detection had discarded $\log S_W = -7.3$ as an outlier. Similarly, neighbor 3 had 5 measurements $\log S_W = \{-2.32; -2.56; -2.81; -3.04\}$ and "insoluble"[21]. Again, the smallest value was treated as an outlier			
			
predicted $\bar{f} = -4.7$ $\log S_W = -7.82$	$\log S_W = -3$	$\log S_W = -8$	$\log S_W = -3.6$
Comments: Neighbor 1 was used in model building with $\log S_W = -3$. Originally 5 measurements, four indicating $\log S_W \approx -3$, and one indicating $S_W = 0$ (unsoluble). Outlier detection had discarded $S_W = 0$ as an outlier. Similarly, neighbor 3 was modelled with $\log S_W = -3.63$. Measurements were $\log S_W = \{-3.46; -3.80; -3.95; -4.14; -8.72\}$ and one indicating $S_W = 0$ (unsoluble)			
			
predicted $\bar{f} = -8.7$ $\log S_W = -7.9$	$\log S_W = -8.7$	$\log S_W = -7.9$	$\log S_W = -8.7$
Comments: The prediction error is not particularly large here (0.8 log units). Yet, all neighbors are very close in descriptor space and have similar solubility values. This makes GPsol output a small error bar, Eq. (5).			

[21] I.e. solubility below limit of detection.

Table 3: (continued)

Test compound	Neighbor 1	Neighbor 2	Neighbor 3
			
predicted $\bar{f} = -7.6$ $\log S_W = -5.2$	$\log S_W = -7.6$	$\log S_W = -8.6$	$\log S_W = -5.8$
Comments: Alternative measurement[51] for test compound: $\log S_W = -7.56$, this matches with model prediction			
			
predicted $\bar{f} = -2.9$ $\log S_W = -5.1$	$\log S_W = -2.7$	$\log S_W = 0.8$	$\log S_W = -4.5$
Comments: Misleading neighbors			

4.2 Noise parameters

In Sec. 2.7.5, it was outlined that different parameters for the measurement noise were used, depending on the number of available measurements. The values σ_1, σ_2 and σ_3 for the measurement noise for compounds with 1, 2, or ≥ 3 measurements were not set to fixed values. Rather, the value was estimated from the available data by gradient descent on the marginal likelihood, Eq. (8).

After model fitting, we are now ready to examine the learned parameters. Compounds with single measurements turned out to have $\sigma_1 = 0.46$. This matches well with previous experience that the measurement noise for solubility measurements can be estimated to be around 0.5 log units.[18] For compounds with two measurements, the noise standard deviation was estimated to be $\sigma_2 = 0.15$, agreeing with the intuition that two agreeing measurements entail lower uncertainty. Compounds with ≥ 3 measurements had $\sigma_3 = 0.026$, meaning that they are considered as even more trustworthy by GPsol. Still, we had expected the value σ_2 to be smaller. When re-examining the data, we noted that some of double measurements were exactly equal up to the last digit, most likely caused by one source having copied from other sources. Such duplicate measurements can clearly not lower the measurement uncertainty.

4.3 Relevant Descriptors

As one of their most pronounced features, Gaussian Process models allow to assign weights to each descriptor that enters the model as input. The similarity of water solubility for two compounds, Eq. (6), takes into account that the ith descriptor contributes to the similarity with weight w_i. Similar to the noise parameters (see Sec. 4.2), these weights are chosen automatically when maximizing marginal likelihood, Eq. (8). After model fitting, the assigned weights can be inspected, in order to get an impression of the relevance of individual descriptors.

Among the 30 descriptors with highest weight, a set with a clear link to solubility could be identified:

- Number of hydroxy groups (nROH, Dragon block 17)
- Number of carboxylic acid groups(C-040, Dragon block 18)
- Number of keto groups (O-058, Dragon block 18)
- LogD at ph 7, see section 2.5
- Total polar surface area (TPSA(NO), Dragon block 20)
- Number of nitrogen atoms (nN, Dragon block 1)
- Number of oxygen atoms (nO, Dragon block 1)

A number of other descriptors were given high weight, and it seems plausible that they influence solubility. These are: number of 10-membered rings (nR10,

Dragon 1), sum of topological distances between N and O (T(N..O), Dragon 2), different forms of autocorrelation weighted by electronegativity and polarizability (GATS1e, GATS2e, and MATS1p, Dragon 6), measures for the global charge transfer in the molecule (JGI9, Dragon 9), quadrupole moments (QXXe and QXXp, Dragon 9), radius of gyration (RGyr, Dragon 12)[22] , and a number of different fragment counts (C-026, C-028, C-029, C-034, N-075, and O-059, Dragon 18).

As one of the reviewers pointed out, the descriptors used in this study include the influence of crystal packing only implicitly. Future descriptor generators that can provide accurate information on molecular packing interactions might help to improve the presented models.

4.4 Comparison With Other Approaches

The dataset by Huuskonen[3] has been used by numerous researchers since it was first published in 2000. All results known to us are summarized in Table 4. If multiple methods of evaluation were employed, we chose the method that is most comparable with those in other studies. If a reference contained different results, e.g., reflecting different model parameters, we only included the best result in Table 4.

The results listed contain squared correlation coefficient, r^2, and root mean square error RMSE as performance criteria. r^2 is listed since many references only report this criterion. Yet, we find that r^2 is not well suited as a performance criterion for regression. r^2 is scale invariant and invariant to systematic errors. E.g., if $(1, 2, 3, 4)$ is the a set of measured $\log S_W$ values, and $(2, 4, 6, 8)$ is the set of according predictions, we obtain $r^2 = 1$, indicating perfect prediction.

For the many approaches from literature we cite in Table 4, one should bear in mind that very diverse methodologies with respect to evaluation and choice of model parameters were employed. For example, the performance on the test set should never be used to guide the choice of model parameter. Also, results on a single training/test split are prone to random effects that only occur in this very split, and can thus not give reliable estimates of the generalization error on unseen data. All these aspects make a direct comparison of the results in Table 4 quite difficult.

The RMSE criterion is essentially the standard deviation of the residuals after model fitting. Considering that the measurement errors for solubility are believed to be at least 0.5 log units,[18] it is unclear how far the results on the Huuskonen data can be improved upon without over-fitting.

Note also, that none of the models employed in the listed references[3–17] can provide individual measures of confidence (error bars) for each individual prediction.

[22]Many measures of size are contained in the DRAGON descriptors. It is not clear to us why in particular this measure of size has been given high weight

Reference	Method	Split	n_{train}	n_{test}	r^2	RMSE
Huuskonen[3]	MLR	single[3]	884	413	0.88	0.71
	ANN	single[3]	884	413	0.92	0.60
Tetko et al.[4]	MLR	single[3]	879	412	0.85	0.81
	ANN	single[3]	879	412	0.90	0.66
Liu et al.[8]	ANN	single	1033	258		0.87
Ran et al.[54]	GSE	subset[3]	0	380		0.76
Bruneau[10]	ANN	subset[3]	1560	673		0.82
Engkvist et al.[12]	ANN	crossval.	1160	130	0.95	
Gasteiger et al.[7]	MLR	max overlap	797	496	0.82	
	ANN	max overlap	797	496	0.92	
Gasteiger et al.[5]	MLR	max overlap	741	552	0.89	
	ANN	max overlap	741	552	0.94	
Lind et al.[43]	SVM	single[3]	884	412	0.89	0.68
Gasteiger et al.[13]	ANN	subset[3]	2083	799	0.94	
Hou et al.[14]	MLR	single[3]	887	412	0.90	
Fröhlich et al.[15]	SVM	crossval.	1135	162	0.90	
Clark[16]	PLS	subset[3]	2427	1297	0.84	
Rapp[17]	SVM	single	1,016	253	0.92	
	ANN	single	1,016	253	0.91	
This study, "Huuskonen only"	GP	crossval.	860	430	0.93	0.55
This study, "Huuskonen"	GP	cv subset[3]	3,952	645	0.91	0.55

Table 4: A comparison of previous results on the data provided by Huuskonen.[3] Column "Method" lists the model used (*MLR* Multiple Linear Regression, *ANN* Artificial Neural Network, *SVM* Support Vector Machine, *GSE* General Solubility Equation, *PLS*, Partial Least Squares, *GP* Gaussian Process). Column "split" indicates how the original data[3] was separated into training (size n_{train}) and test set (size n_{test}). Some researchers used the same single split as described in the original reference[3] (indicated by "single[3]"). Others use a different single random split ("single"), cross-validation ("crossval"), or trained on external data and then evaluated on differently sized subsets of the Huuskonen[3] data ("subset[3]"). One group chose splits that maximize the overlap between training and test set ("max overlap"). Performance criteria are squared correlation coefficient, r^2 and root mean square error RMSE. See Sec. 4.4 for a discussion of the results, and of problems with using r^2 as a performance criterion.

5 Summary

Over the last decade, the prediction of water solubility has shown a tremendous impetus on many areas of chemical research. In this work, we presented a novel method for predicting buffer solubility by means of a nonlinear regression model, namely a Gaussian Process model (GPsol). Training data were measurements for around 4,000 literature compounds and in-house drug candidates. For the difficult task of predicting the buffer solubility of drug candidates, the developed GPsol model achieves excellent accuracy and high performance gains over available commercial tools. We could verify the high performance in cross-validation, as well as in an additional "blind test" on data from recent projects.

Note that global models for aqueous solubility are typically unable to predict the solubility of in-house compounds (drug candidates) with satisfactory accuracy. Using machine learning approaches, such as the presented Gaussian Process methodology, previously acquired in-house data can be employed to create *tailored models* of buffer solubility (or other properties of interest) that are particularly accurate in the investigated region of the chemical space. Our final GPsol model was then implemented in C# (suitable for Linux and Windows operating systems) and can produce one prediction per second on a single 2 GHz Pentium CPU.

Error bars are of great value to any form of black box model built without explicit knowledge of the mechanisms that underly water solubility. As a particular advantage of the proposed method, GPsol is able to estimate the degree of certainty for each of its predictions. This can be used to assess the validity of the prediction, and whether the model is queried within its range of expertise. Our evaluation has shown that the predicted error bars can be interpreted as a true measure of confidence. GPsol is the first model of water solubility that outputs error bars with a solid statistical foundation.

We finally conjecture that the progress in solubility prediction is far from coming to an end. Certainly, the collection of well-defined, high quality solubility measurements of electrolytes and drug-like compounds is a key issue in this endeavor. Further progress can be made due to the increasing availability of more detailed and accurate molecular descriptors, and in particular also by exploiting the benefits of novel machine learning methods. Drug design and discovery permanently explores new regions of the chemical space. Machine learning can provide the tools to keep track with the latest developments, and guide further explorations.

Acknowledgements

The authors gratefully acknowledge partial support from the PASCAL Network of Excellence (EU #506778), DFG grants JA 379/13-2 and MU 987/2-1. We thank Vincent Schütz and Carsten Jahn for maintaining the PCADMET database, and two anonymous reviewers for detailed suggestions that helped to improve the paper.

References

1. Clarke, E. D.; Delaney, J. S. Physical and Molecular Properties of Agrochemicals: An Analysis of Screen Inputs, Hits, Leads, and Products. *Chimia* **2003,** *57,* 731-734.

2. Hou, T.; Xu, X. ADME Evaluation in Drug Discovery. 3. Modeling Blood-Brain Barrier Partitioning Using Simple Molecular Descriptors. *J. Chem. Inf. Comput. Sci.* **2003,** *43,* 2137–2152.

3. Huuskonen, J. Estimation of Aqueous Solubility for a Diverse Set of Organic Compounds Based on Molecular Topology. *J. Chem. Inf. Comput. Sci.* **2000,** *40,* 773-777.

4. Tetko, I. V.; Tanchuk, V. Y.; Kasheva, T. N.; Villa, A. E. Estimation of Aqueous Solubility of Chemical Compounds Using E-State Indices. *J. Chem. Inf. Comput. Sci.* **2001,** *41,* 1488–1493.

5. Yan, A.; Gasteiger, J. Prediction of Aqueous Solubility of Organic Compounds by Topological Descriptors. *QSAR Comb. Sci.* **2003,** *22,* 821-829.

6. Wegner, J. K.; Zell, A. Prediction of Aqueous Solubility and Partition Coefficient Optimized by a Genetic Algorithm Based Descriptor Selection. *J. Chem. Inf. Comput. Sci.* **2003,** *43,* 1077-1084.

7. Yan, A.; Gasteiger, J. Prediction of Aqueous Solubility of Organic Compounds Based on a 3D Structure Representation. *J. Chem. Inf. Comput. Sci.* **2003,** *43,* 429-434.

8. Liu, R.; Sun, H.; So, S.-S. Development of Quantitative Structure - Property Relationship Models for Early ADME Evaluation in Drug Discovery. 1. Aqueous Solubility. *J. Chem. Inf. Comput. Sci.* **2001,** *41,* 1633-1639.

9. Delaney, J. S. ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. *J. Chem. Inf. Comput. Sci.* **2004,** *44,* 1000 -1005.

10. Bruneau, P. Search for Predictive Generic Model of Aqueous Solubility Using Bayesian Neural Nets. *J. Chem. Inf. Comput. Sci.* **2001,** *41,* 1605-1616.

11. Yaffe, D.; Cohen, Y.; Espinosa, G.; Arenas, A.; Giralt, F. A Fuzzy ARTMAP Based on Quantitative Structure-Property Relationships (QSPRs) for Predicting Aqueous Solubility of Organic Compounds. *J. Chem. Inf. Comput. Sci.* **2001,** *41,* 1177–1207.

12. Engkvist, O.; Wrede, P. High-Throughput, In Silico Prediction of Aqueous Solubility Based on One- and Two-Dimensional Descriptors. *J. Chem. Inf. Comput. Sci.* **2002,** *42,* 1247-1249.

13. Yan, A.; Gasteiger, J.; Krug, M.; Anzali, S. Linear and nonlinear functions on modeling of aqueous solubility of organic compounds by two structure representation methods. *J. Comput.-Aided Mol. Des.* **2004,** *18,* 75-87.

14. Hou, T. J.; Xia, K.; Zhang, W.; Xu, X. J. ADME Evaluation in Drug Discovery. 4. Prediction of Aqueous Solubility Based on Atom Contribution Approach. *J. Chem. Inf. Comput. Sci.* **2004,** *44,* 266-275.

15. Fröhlich, H.; Wegner, J. K.; Zell, A. Towards Optimal Descriptor Subset Selection with Support Vector Machines in Classification and Regression. *QSAR Comb. Sci.* **2004,** *23,* 311-318.

16. Clark, M. Generalized Fragment-Substructure Based Property Prediction Method. *J. Chem. Inf. Model.* **2005,** *45,* 30-38.

17. Rapp, F.-R. PhD thesis, Fakultät für Informations- und Kognitionswissenschaften, Eberhard-Karls-Universität Tübingen, Germany, 2005.

18. Balakin, K. V.; Savchuk, N. P.; Tetko, I. V. In Silico Approaches to Prediction of Aqueous and DMSO Solubility of Drug-Like Compounds: Trends, Problems and Solutions. *Curr. Med. Chem.* **2006,** *13,* 223-241.

19. Johnson, S. R.; Zheng, W. Recent Progress in the Computational Prediction of Aqueous Solubility and Absorption. *The AAPS Journal* **2006,** *8,* E27–E40.

20. Delaney, J. S. Predicting aqueous solubility from structure. *Drug Discovery Today* **2005,** *10,* 289–295.

21. Schölkopf, B.; Smola, A. J. *Learning with Kernels;* MIT Press, 2002.

22. Müller, K.-R.; Rätsch, G.; Sonnenburg, S.; Mika, S.; Grimm, M.; Heinrich, N. Classifying 'Drug-likeness' with Kernel-Based Learning Methods. *J. Chem. Inf. Model* **2005,** *45,* 249-253.

23. Duda, R. O.; Hart, P. E.; Stork, D. G. *Pattern Classification;* John Wiley & Sons: New York, 2 ed.; 2000.

24. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning;* Springer Verlag, 2001.

25. Bishop, C. M. *Neural Networks for Pattern Recognition;* Oxford University Press, 1995.

26. Orr, G.; Müller, K.-R., Eds.; *Neural Networks: Tricks of the Trade;* volume 1524 Springer LNCS, 1998.

27. Gasteiger, J.; Engel, T., Eds.; *Chemoinformatics: A Textbook;* Wiley-VCH, 2003.

28. Vapnik, V. N. *The Nature of Statistical Learning Theory;* Springer Verlag, 1995.

29. Müller, K.-R.; Mika, S.; Rätsch, G.; Tsuda, K.; Schölkopf, B. An Introduction to Kernel-Based Learning Algorithms. *IEEE Transactions on Neural Networks* **2001,** *12,* 181–201.

30. Merkwirth, C.; Mauser, H.; Schulz-Gasch, T.; Roche, O.; Stahl, M.; Lengauer, T. Ensemble Methods for Classification in Cheminformatics. *J. Chem. Inf. Comput. Sci.* **2004,** *44,* 1971-1978.

31. Byvatov, E.; Fechner, U.; Sadowski, J.; Schneider, G. Comparison of Support Vector Machine and Artificial Neural Network Systems for Drug/Nondrug Classification. *J. Chem. Inf. Comput. Sci.* **2003,** *43,* 1882–1889.

32. Platt, J. Probabilistic outputs for Support Vector Machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*; Smola, A. J.; Bartlett, P.; Schölkopf, B.; Schuurmans, D., Eds.; MIT Press, 1999; pp 61–74.

33. Neal, R. M. *Bayesian Learning for Neural Networks;* Springer Verlag, 1996.

34. Rasmussen, C. E.; Williams, C. K. *Gaussian Processes for Machine Learning;* MIT Press, 2005.

35. Neal, R. M. Regression and Classification Using Gaussian Process Priors. In *Bayesian Statistics 6*; Bernardo, J. M.; Berger, J.; Dawid, A.; Smith, A., Eds.; Oxford University Press, 1998; Vol. 6, pp 475–501.

36. Schwaighofer, A.; Grigoraş, M.; Tresp, V.; Hoffmann, C. GPPS: A Gaussian Process Positioning System for Cellular Networks. In *Advances in Neural Information Processing Systems 16*; Thrun, S.; Saul, L.; Schölkopf, B., Eds.; MIT Press, 2004; pp 579–586.

37. Schwaighofer, A.; Tresp, V.; Mayer, P.; Krause, A.; Beuthan, J.; Rost, H.; Metzger, G.; Müller, G. A.; Scheel, A. K. Classification of Rheumatoid Joint Inflammation Based on Laser Imaging. *IEEE Transactions on Biomedical Engineering* **2003,** *50,* 375–382.

38. Yu, K.; Tresp, V.; Schwaighofer, A. Learning Gaussian Processes from Multiple Tasks. In *Machine Learning: Proceedings of the 22nd International Conference (ICML 2005)*; Cohen, W. W.; Moore, A., Eds.; ACM Press, 2005; pp 1012–1019.

39. Burden, F. R. Quantitative Structure-Activity Relationship Studies Using Gaussian Processes. *J. Chem. Inf. Comput. Sci.* **2000,** *41,* 830–835.

40. Enot, D.; Gautier, R.; Le Marouille, J. Gaussian process: an efficient technique to solve quantitative structure-property relationship problems. *SAR QSAR Environ. Res.* **2001,** *12,* 461–469.

41. Tino, P.; Nabney, I.; Williams, B. S.; Lösel, J.; Sun, Y. Non-linear Prediction of Quantitative Structure-Activity Relationships. *J. Chem. Inf. Comput. Sci.* **2004,** *44,* 1647–1653.

42. Klamt, A.; Eckert, F.; Hornig, M.; Beck, M. E.; Bürger, T. Prediction of Aqueous Solubility of Drugs and Pesticides with COSMO-RS. *J. Comput. Chem.* **2002,** *23,* 275–281.

43. Lind, P.; Maltseva, T. Support Vector Machines for the Estimation of Aqueous Solubility. *J. Chem. Inf. Comput. Sci.* **2003,** *43,* 1855 -1859.

44. Equbits LLC, Estimation of Aqueous Solubility; Case Study 11/13/2003, Equbits LLC: Livermore, CA, USA, 2003. `http://www.equbits.com/casestudies/EqubitsAqueousSolubilityCaseStudyV6.pdf` (accessed 14 May 2006).

45. Göller, A. H.; Hennemann, M.; Keldenich, J.; Clark, T. In Silico Prediction of Buffer Solubility Based on Quantum-Mechanical and HQSAR- and Topology-Based Descriptors. *J. Chem. Inf. Model.* **2006,** *46,* 648-658.

46. Sadowski, J.; Schwab, C.; Gasteiger, J. *Corina v3.1;* Molecular Networks GmbH Computerchemie: Erlangen, Germany,.

47. Todeschini, R.; Consonni, V.; Mauri, A.; Pavan, M. *DRAGON v1.2;* Talete SRL: Milano, Italy,.

48. Bergström, C. A. S.; Luthmanb, K.; Artursson, P. Accuracy of calculated pH-dependent aqueous drug solubility. *Eur. J. Pharm. Sci.* **2004,** *22,* 387-398.

49. Stouch, T.; Kenyon, J.; Johnson, S.; Chen, X.; Doweyko, A.; Li, Y. In silico ADME/Tox: why models fail. *J. Comput.-Aided Mol. Des.* **2003,** *17,* 83-92.

50. Kariv, I.; Rourick, R.; Kassel, D.; Chung, T. Improvement of "Hit-to-Lead" Optimization by Integration of in Vitro HTS Experimental Models for Early Determination of Pharmacokinetic Properties. *Comb. Chem. High Throughput Screening* **2002,** *5,* 459-472.

51. *Physical/Chemical Property Database (PHYSPROP);* Syracuse Research Corporation, Environmental Science Center: Syracuse, NY, USA,.

52. *Beilstein CrossFire Database;* MDL Information Systems: San Ramon, CA, USA,.

53. Yalkowsky, S.; Dannelfelser, R. *The Arizona Database of Aqueous Solubility;* College of Pharmacy, University of Arizona: Tuscon, AZ, USA,.

54. Ran, Y.; Jain, N.; Yalkowsky, S. H. Prediction of Aqueous Solubility of Organic Compounds by the General Solubility Equation. *J. Chem. Inf. Comput. Sci.* **2001,** *41,* 1208-1217.

55. Livingstone, D. J.; Martyn,; Ford,; Huuskonenc, J. J.; Salt, D. W. Simultaneous prediction of aqueous solubility and octanol/water partition coefficient based on descriptors derived from molecular structure. *J. Comput.-Aided Mol. Des.* **2001,** *15,* 741-752.

56. Todeschini, R.; Consonni, V.; Mauri, A.; Pavan, M. DRAGON For Windows and Linux 2006, `http://www.talete.mi.it/help/dragon_help/` (accessed 14 May 2006).

57. O'Hagan, A. Curve Fitting and Optimal Design for Prediction. *Journal of the Royal Statistical Society, Series B: Methodological* **1978,** *40,* 1–42.

58. Zhu, C.; Byrd, R. H.; Nocedal, J. L-BFGS-B: Algorithm 778: L-BFGS-B, FORTRAN routines for large scale bound constrained optimization. *ACM Transactions on Mathematical Software* **1997,** *23,* 550–560.

59. *AdmetPredictor v1.2.3;* Simulations Plus, Inc.: Lancaster, CA, USA,.

60. *ACD/Solubility Batch v9.0;* Advanced Chemistry Development, Inc. (ACD/Labs): Toronto, Canada,.

61. *EPI Suite Wskowwin v1.41;* US Environmental Protection Agency: Washington DC, USA,.

62. Hong, C.-S.; Qiao, H. Generator column determination of aqueous solubilities for non-ortho and mono-ortho substituted polychlorinated biphenyls. *Chemosphere* **1995,** *31,* 4549-4557.

63. Huang, Q.; Hong, C.-S. Aqueous solubilities of non-ortho and mono-ortho PCBs at four temperatures. *Water Res.* **2006,** *36,* 3543-3552.

64. Kuss, M. PhD thesis, Technische Universität Darmstadt, 2006.

65. O'Neil, M. J.; Smith, A.; Heckelman, P. E., Eds.; *The Merck Index;* Merck Publications, 1996.

66. Said, A.; Makki, S.; Muret, P.; Rouland, J.-C.; Toubin, G.; Millet, J. Lipophilicity determination of psoralens used in therapy through solubility and partitioning: Comparison of theoretical and experimental approaches. *J. Pharm. Sci.* **1996,** *85,* 387-392.

A Appendix

A.1 Further Evaluations of GPsol

Figure 5 shows a cumulative histogram of the prediction error made by GPsol. The cumulative histogram is over the absolute error, and thus indicates how many predictions are within $0.5, 1, 1.5, \ldots, 3, 3.5$ orders of magnitude from the measured value. Again, the high performance of GPsol can be seen clearly. For more than 50% of the compounds in the Flask data set, the prediction error is even smaller than 0.5 orders of magnitude.[23] As a baseline, we can assign

[23] In general, the uncertainty of solubility measurements is believed to be about 0.5 orders of magnitude.[18] The exceptionally good performance on this set results from the fact that it was generated under highly standardized conditions (see section 2.3 on the preparation of the datasets). The very low noise parameter learned from the data in this set (see Sec. 2.7.5 and 4.2) is another indicator for the high quality of this data.

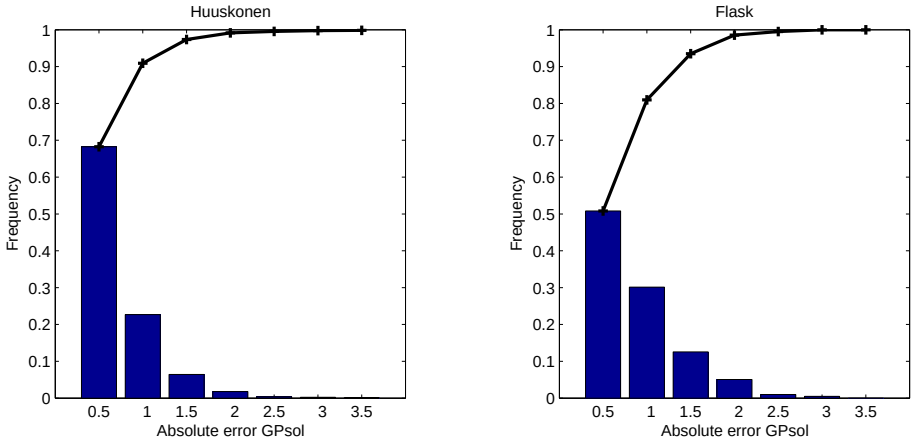


Figure 5: Cumulative histogram of the absolute prediction error made by GPsol, when either predicting pure solubility in the Huuskonen setup (left plot) or when predicting buffer solubility of electrolytes in the Flask setup (right plot). The solid line indicates how many predictions are within $0.5, 1, 1.5, \ldots, 3, 3.5$ orders of magnitude from the measured value

each compound an average solubility value within its respective data set. With this baseline prediction, 60% of the compounds in the Flask data set are correct within 1 log unit (32% within 0.5 log unit). For the Huuskonen data, 39% of the compounds are correct within 1 log unit, 20% within 0.5 log units.

A.2 Comparison With Available Tools

On the data sets "Flask", "Huuskonen" and "Huuskonen neutral", as described in Sec. 2.3, we evaluated six commercial tools. The performance is compared to GPsol trained in the "Flask", "Huuskonen" and "Huuskonen neutral" setups, see Sec. 2.4. The settings used for each tool are listed in table Table 5. The full set of results is compiled in Table 6, with a graphical summary given in Figure 6. Plots showing predicted versus measured solubility are given in Figure 9 for the "Flask" data set, Figure 8 for the "Huuskonen" data set, and Figure 7 for the "Huuskonen neutral" data.

AdmetPredictor of SimulationsPlus[59] was used in six different modes. The models for native (pure) solubility, intrinsic[24] solubility and solubility at a specified pH (which was set to 7.4) all come in two flavors, general and drug-like compounds. Surprisingly, the best predictions for buffer solubility on the Flask dataset (drug candidates at pH 7.0 to 7.4) are produced using the predictor for native (i.e., pure, unbuffered) solubility of non-drug compounds.

[24] We found that the predictions for intrinsic solubilities were in general not as precise as those for native solubility and solubility at pH 7.4. Thus, we omitted results for intrinsic solubility from our evaluation.

Abbrev.	Vendor / Agency	Product	Version	Settings
S+ nat	SimulationsPlus	AdmetPredictor	1.2.3	native
S+ pH7				pH 7.4
S+d nat				native, drugs
S+d pH7				pH 7.4, drugs
PP	SciTegic	PipelinePilot	5.0.1.100	native
QIKP	Schroedinger	QikProp	2.2	
EPI	U.S. Environmental Protection Agency	Wskowwin (from the EPI Suite)	1.41	
ACDv8	ACD/Labs	Solubility Batch	8.0	pH 7.0
ACDv9	ACD/Labs	Solubility Batch	9.0	pH 7.0

Table 5: Settings used for the various commercial tools. The leftmost column is the shorthand name used for the tool in tables and figures

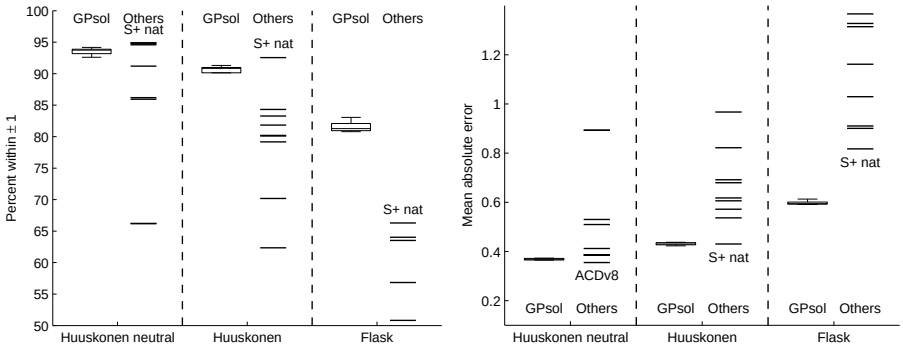


Figure 6: Graphical summary of the results listed in Table 6. The left plot shows accuracy in terms "percentage of points where prediction is correct within one log unit" (the higher, the better), whereas the right plot shows accuracy in terms of mean absolute error (the lower, the better). A horizontal line marks the performance for each commercial tool, the name of the best tool is written out. For the GP model, we show a box plot for the performance obtained in each of the 10 runs, see Sec. 2.4. The "whiskers" extend to minimum and maximum performance over the 10 splits, median performance is shown as the central line. See Table 5 for the list of tools and their settings

Percent within ±1	GPsol	ACDv8	ACDv9	S+ nat	S+ pH7	S+d nat	S+d pH7	EPI	QIKP	PP
Huuskonen neutral	**95**	**95**	**95**	**95**	**95**	66	66	*91*	86	86
Huuskonen	*91*	79	80	**93**	82	70	62	84	83	80
Flask	**82**	64	64	*66*	57	50	43	51	46	42

Mean absolute error	GPsol	ACDv8	ACDv9	S+ nat	S+ pH7	S+d nat	S+d pH7	EPI	QIKP	PP
Huuskonen neutral	*0.37*	**0.36**	0.39	0.39	0.39	0.89	0.89	0.41	0.51	0.53
Huuskonen	*0.43*	0.68	0.69	**0.43**	0.61	0.82	0.97	0.54	0.57	0.62
Flask	**0.60**	0.91	0.90	*0.82*	1.03	1.16	1.37	1.33	1.31	1.52

Root mean squared error	GPsol	ACDv8	ACDv9	S+ nat	S+ pH7	S+d nat	S+d pH7	EPI	QIKP	PP
Huuskonen neutral	0.55	**0.49**	0.51	*0.51*	0.51	1.18	1.18	0.58	0.73	0.70
Huuskonen	*0.61*	1.07	1.08	**0.56**	0.88	1.09	1.27	0.75	0.78	0.81
Flask	**0.77**	1.16	1.16	*1.05*	1.34	1.44	1.68	1.75	1.64	1.91

r^2	GPsol	ACDv8	ACDv9	S+ nat	S+ pH7	S+d nat	S+d pH7	EPI	QIKP	PP
Huuskonen neutral	0.94	**0.96**	0.95	*0.95*	0.95	0.80	0.80	0.94	0.91	0.91
Huuskonen	*0.91*	0.78	0.78	**0.93**	0.83	0.74	0.67	0.87	0.87	0.85
Flask	**0.53**	0.26	0.28	*0.31*	0.27	0.09	0.13	0.13	0.16	0.18

Table 6: Accuracy on different validation sets that are achieved by a number of commercial tools and GPsol. Accuracy is measured as the percentage of compounds where the true value is within 1 log unit from the predicted value (top), mean absolute error (middle), and squared correlation coefficient r^2 (bottom). The best result for each validation set (Flask, Huuskonen and Huuskonen neutral) is marked in bold, the second best result is marked in italics. A graphical summary of the table is given in Figure 6. See Table 5 for the list of tools and their settings

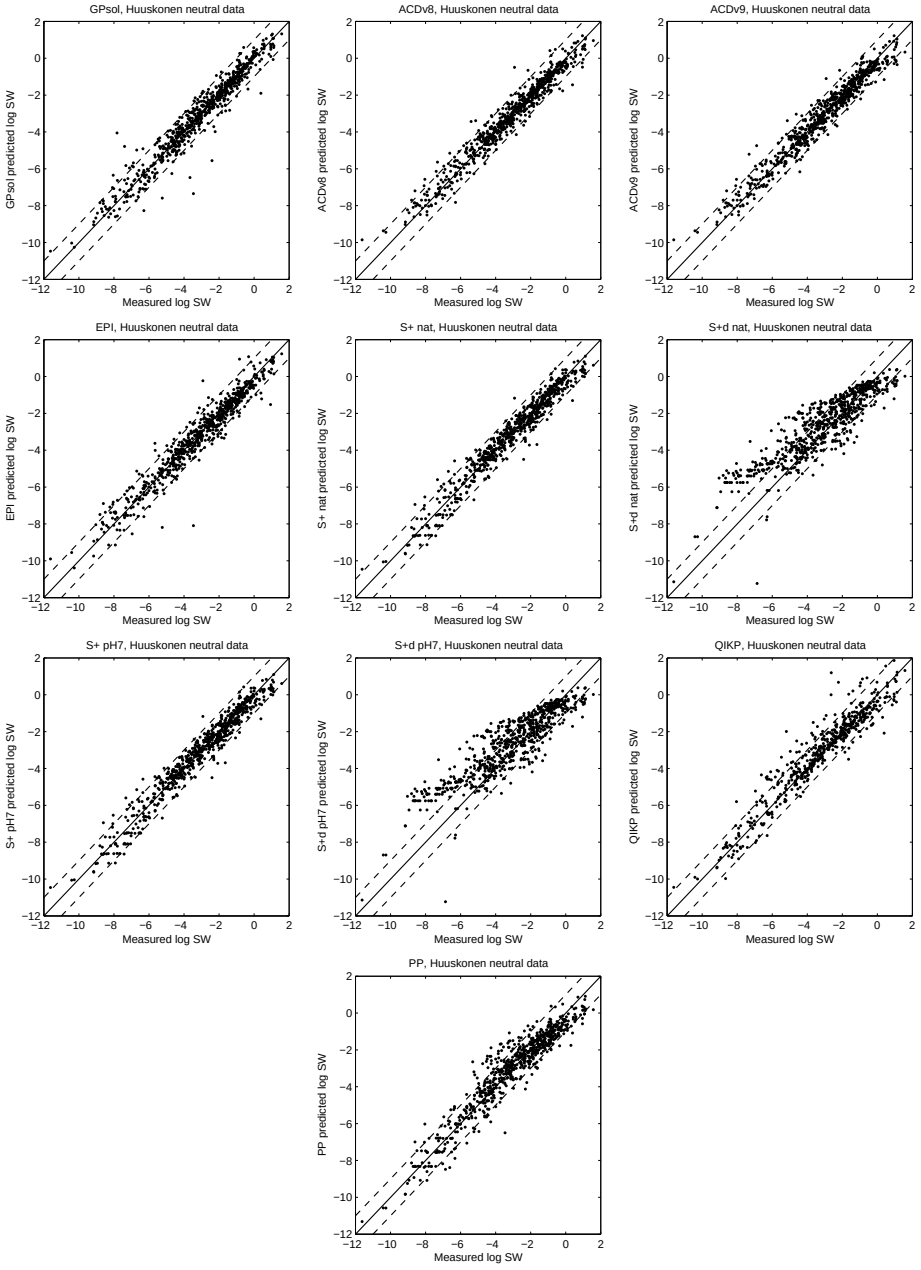


Figure 7: Predicted pure solubility (y-axis) versus measured pure solubility (x-axis) on the Huuskonen neutral data for GPsol and a number of commercial tools. See Table 5 for the list of tools and their settings

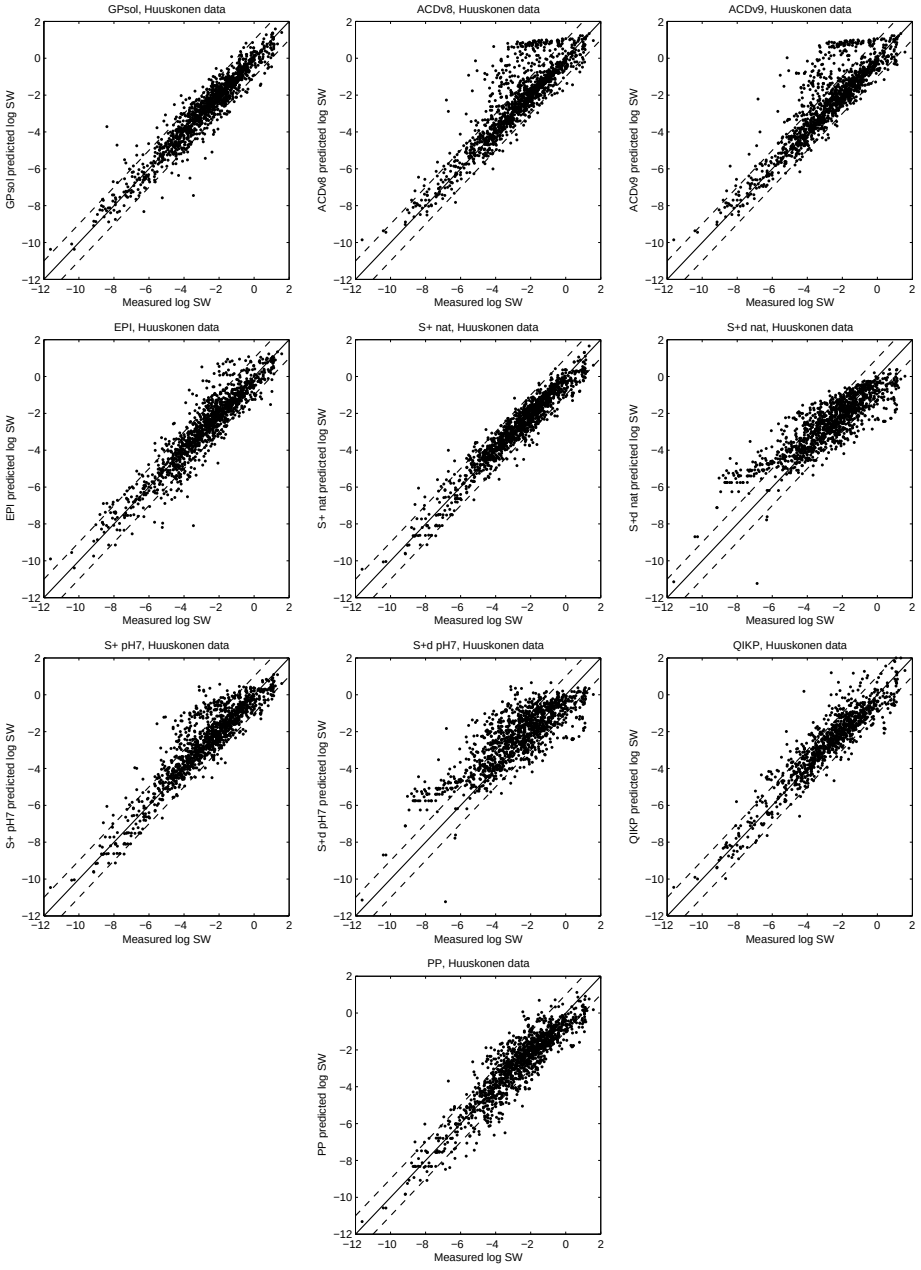


Figure 8: Predicted pure solubility (y-axis) versus measured pure solubility (x-axis) on the Huuskonen data for GPsol and a number of commercial tools. See Table 5 for the list of tools and their settings

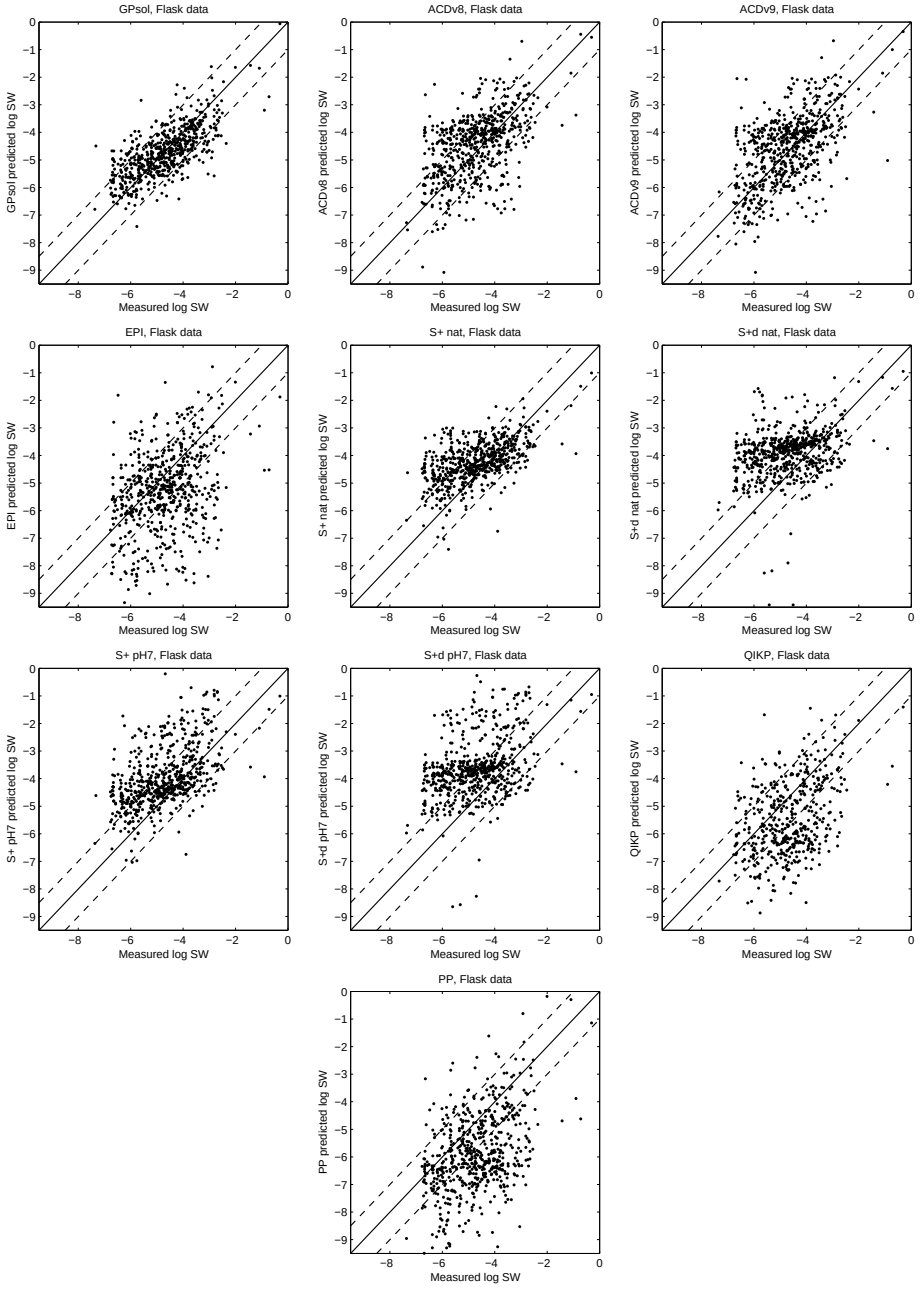


Figure 9: Predicted buffer solubility (y-axis) versus measured buffer solubility (x-axis) on the Flask data for GPsol and a number of commercial tools. See Table 5 for the list of tools and their settings

B Gaussian Process Derivations

In order to derive Eq. (3), we need a simple theorem for conditional distribution of the multivariate normal. Assume a vector $\mathbf{z} = (z_1, \ldots, z_n)$, the elements of which follow a joint multivariate normal distribution with mean vector μ and covariance matrix V, denoted by $\mathbf{z} \sim \mathcal{N}(\mu, V)$. We partition the vector $\mathbf{z} = (\mathbf{z}_o, z_n)$, where $\mathbf{z}_o$ holds all elements excluding the last one. Accordingly, we can partition mean vector and covariance matrix, and write

$$\mathbf{z} = \begin{pmatrix} \mathbf{z}_o \\ z_n \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu_o \\ \mu_n \end{pmatrix}, \begin{pmatrix} V_{oo} & V_{on} \\ V_{on}^\top & V_{nn} \end{pmatrix} \right) \tag{9}$$

If we observe values for all elements $\mathbf{z}_o$ up to the last one, the conditional distribution for z_n, $p(z_n \,|\, \mathbf{z}_o)$, is a univariate normal distribution,

$$z_n \,|\, \mathbf{z}_o \sim \mathcal{N}\left(\mu_n + V_{on}^\top V_{oo}^{-1}(\mathbf{z}_o - \mu_o),\ V_{nn} - V_{on}^\top V_{oo}^{-1} V_{on}\right) \tag{10}$$

With this lemma, we can already derive Eq. (3). Recall from Sec. 2.7.2 the key assumption about GP models, namely that all solubility values $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))$ form a joint Gaussian distribution, $\mathbf{f} \sim \mathcal{N}(0, K)$. For convenience, we assume for the mean $\mu = 0$.[25]. Elements of the covariance matrix K are given by the covariance function, Eq. (2), with $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. Similarly, we can consider the joint distribution of solubility values $\mathbf{f}$ along with the solubility $f(\mathbf{x}_*)$ of the compound $\mathbf{x}_*$ we wish to predict. Again, the Gaussian Process assumption allows us to write

$$\begin{pmatrix} \mathbf{f} \\ f(\mathbf{x}_*) \end{pmatrix} \sim \mathcal{N}\left(0, \begin{pmatrix} K & \mathbf{v} \\ \mathbf{v}^\top & k(\mathbf{x}_*, \mathbf{x}_*) \end{pmatrix}\right) \tag{11}$$

$\mathbf{v}$ denotes the vector of evaluations of the covariance function $\mathbf{v}_i = k(\mathbf{x}_i, \mathbf{x}_*)$.

From that, we can easily derive the joint distribution of measured values $\mathbf{y} = (y_1, \ldots, y_n)$ and predicted value $f(\mathbf{x}_*)$. We need to include the measurement noise, Eq. (1), which amounts to adding σ^2 along the diagonal of the covariance matrix. The joint distribution becomes

$$\begin{pmatrix} \mathbf{y} \\ f(\mathbf{x}_*) \end{pmatrix} \sim \mathcal{N}\left(0, \begin{pmatrix} K + \sigma^2 I & \mathbf{v} \\ \mathbf{v}^\top & k(\mathbf{x}_*, \mathbf{x}_*) \end{pmatrix}\right). \tag{12}$$

With that, we can directly use Eq. (10) and compute the conditional distribution of $f(\mathbf{x}_*)$ after having observed all values $\mathbf{y} = (y_1, \ldots, y_n)$,

$$f(\mathbf{x}_*) \,|\, \mathbf{y} \sim \mathcal{N}\left(\mathbf{v}^\top (K + \sigma^2 I)^{-1} \mathbf{y},\ k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{v}^\top (K + \sigma^2 I)^{-1} \mathbf{v}\right). \tag{13}$$

Writing out the matrix expressions as sums, we arrive at Eq. (3) (for the mean of the conditional distribution) and Eq. (5) (for the standard deviation of the conditional).

[25] In practice, this can be achieved by simply shifting the data to have zero mean