THE UNIVERSITY OF QUEENSLAND

AUSTRALIA

Large Scale Material Science Data Analysis

Eve Bélisle
B. Sc.

A thesis submitted for the degree of Master of Philosophy at The University of Queensland in 2015

School of Information Technology and Electrical Engineering

Abstract

Material Science, the science of studying materials and their properties, involves many aspects such as performing experiments to calculate certain physical properties. Scientists are always looking to utilise the collected experimental data in order to make predictions for new points, where the studied property is unknown. Using a computer model to make these predictions, whether it is via a machine learning or mathematical approach, is the desirable option, since doing actual experiments have proven to be very costly and time consuming. We are therefore looking at utilising the vast quantity of pre-collected data in the literature in order to build models for making future predictions. We already know that the Gaussian process regression interpolation technique gives accurate predictions for some physical properties. However, it is also the slowest of the machine learning algorithms and not suitable for on-line applications. For on-line learning, making quick and accurate predictions is essential. In this research we propose a novel strategy, including batch query processing and co-clustering, to achieve a scalable and efficient Gaussian process regression. This new approach, called the scalable Gaussian process (SGP), allows the use of large databases and makes it suitable for on-line applications. The proposed strategy is applied to a real application involving the prediction of materials properties. Results demonstrate the high accuracy and efficiency of our approach. We test and compare SGP with five different machine learning models on materials properties databases and make recommendations accordingly, also demonstrating that prior knowledge of the problem is essential when choosing a machine learning model.

As one could expect, databases consisting of experimental data are noisy since they rely on human measurements, and also because they are an amalgamation of various independent sources (research papers). Therefore, some conflicting information can be found between the various sources. In our research we also introduce a novel truth discovery approach to reduce the amount of noise and filter the incorrect conflicting information hidden in scientific databases. Our method ranks the multiple data sources by considering the relationships between them, i.e., the amount of conflicting information and the amount of agreement, and as well eliminates the conflicting information. Our previously introduced technique, SGP, is

then applied to the clean dataset to make predictions. We compare the prediction accuracy before and after pruning the databases. With our new approach, we are able to highly improve the accuracy of SGP predictions and provide a more reliable database. Our results also prove the extreme robustness of SGP, as we demonstrate that a relatively high amount of noise is handled very well by this technique.

Declaration by author

This thesis is composed of my original work, and contains no material previously published or written by another person except where due reference has been made in the text. I have clearly stated the contribution by others to jointly-authored works that I have included in my thesis.

I have clearly stated the contribution of others to my thesis as a whole, including statistical assistance, survey design, data analysis, significant technical procedures, professional editorial advice, and any other original research work used or reported in my thesis. The content of my thesis is the result of work I have carried out since the commencement of my research higher degree candidature and does not include a substantial part of work that has been submitted to qualify for the award of any other degree or diploma in any university or other tertiary institution. I have clearly stated which parts of my thesis, if any, have been submitted to qualify for another award.

I acknowledge that an electronic copy of my thesis must be lodged with the University Library and, subject to the policy and procedures of The University of Queensland, the thesis be made available for research and study in accordance with the Copyright Act 1968 unless a period of embargo has been approved by the Dean of the Graduate School.

I acknowledge that copyright of all material contained in my thesis resides with the copyright holder(s) of that material. Where appropriate I have obtained copyright permission from the copyright holder to reproduce material in this thesis.

Eve Bélisle

Publications during candidature

Peer-reviewed papers

1. E. Bélisle, Z. Huang, and A. Gheribi. Scalable gaussian process regression for prediction of material properties. In H. Wang and M. Sharaf, editors, *Databases Theory and Applications*, volume 8506 of *Lecture Notes in Computer Science*, pages 38–49. Springer International Publishing, 2014

2. E. Bélisle, Z. Huang, S. Le Digabel, and A. E. Gheribi. Evaluation of machine learning interpolation techniques for prediction of physical properties. *Computational Materials Science*, 98:170–177, 2015

3. E. Bélisle, Z. Huang, and A. Gheribi. Truth discovery in material science databases. In H. Wang and M. Sharaf, editors, *Proceedings of the 26th edition of the Australiasian Database Conference (Accepted in March 2105)*. Springer International Publishing, 2015

Publications included in this thesis

E. Bélisle, Z. Huang, and A. Gheribi. Scalable gaussian process regression for prediction of material properties. In H. Wang and M. Sharaf, editors, *Databases Theory and Applications*, volume 8506 of *Lecture Notes in Computer Science*, pages 38–49. Springer International Publishing, 2014 - incorporated as Chapter 4.

Contributor	**Statement of contribution**
Eve Bélisle	Designed algorithms (100%) Performed experiments (100%) Wrote the paper (85%)
Zi Huang	Wrote and edited the paper (10%)
Aïmen Gheribi	Wrote and edited the paper (5%)

E. Bélisle, Z. Huang, S. Le Digabel, and A. E. Gheribi. Evaluation of machine learning interpolation techniques for prediction of physical properties. *Computational Materials Science*, 98:170–177, 2015 - incorporated as Chapter 5.

Contributor	**Statement of contribution**
Eve Bélisle	Designed experiments (80%) Wrote the paper (100%)
Zi Huang	Edited the paper
Aïmen Gheribi	Designed experiments (10%)
Sébastien Le Digabel	Designed experiments (10%)

E. Bélisle, Z. Huang, and A. Gheribi. Truth discovery in material science databases. In H. Wang and M. Sharaf, editors, *Proceedings of the 26th edition of the Australiasian Database Conference (Accepted in March 2105)*. Springer International Publishing, 2015 - incorporated as Chapter 6.

Contributor	**Statement of contribution**
Eve Bélisle	Designed algorithms (100%) Designed experiments (90%) Wrote the paper (95%)
Zi Huang	Wrote and edited the paper (5%)
Aïmen Gheribi	Designed experiments (10%)

Contributions by others to the thesis

No contribution by others.

Statement of parts of the thesis submitted to qualify for the award of another degree

None.

Acknowledgements

I would like to thank my supervisors Zi (Helen) Huang and Marcus Gallagher for their help and support during my MPhil.

A special thank you to all the members of the DKE research group at the University of Queensland for their insightful help and support during my research. More specifically, my office room-mates, Vinita Nahar and Sanad Al-Maskari.

I would also like to thank Aïmen Gheribi, Sébastien LeDigabel and Christopher Bale from l'École Polytechnique de Montréal for their assistance and expertise in the area of machine learning and material chemistry.

Many thanks to Paul Voigt, Angus Young, Christian Bouchard and Chris Morecroft for their friendly support and advice during the completion of my MPhil.

This work is partially supported by the ARC grant FT130101530 and DP140103171.

Keywords

Machine learning, data mining, Gaussian process regression, scientific databases, truth discovery.

Australian and New Zealand Standard Research Classifications (ANZSRC)

ANZSRC code: 080604, Database Management, 100%

Fields of Research (FoR) Classification

FoR code: 0806, Information Systems, 100%

Contents

List of Figures

List of Tables

1. Introduction

1.1 Motivation

A knowledge of the physical properties of materials is a very important consideration in materials and process design. Slag properties, such as electrical conductivity, thermal conductivity, density, etc. play a key role in the metal industry [13] in order to design new materials or improve the current processes. Those properties may be hard to measure or estimate from numerical models, not to mention very costly and time consuming. When this is the case, and in order to make substantial savings on research costs, engineers rely on machine learning methods. The idea is to utilise existing measured data to predict properties of new systems, by interpolating and extrapolating properties of known systems.

One of the preoccupations of this research is to have a reliable and accurate machine learning interpolation technique that is fast enough for on-line applications. Our work is motivated by an application on the prediction of materials properties, more specifically, by the optimisation of these predictions, which require a large number of single predictions to be performed sequentially. Each prediction request from users is considered as a query in our work, which is represented as a single vector of real numbers, corresponding to the values of composition for each input component. The result of a query is a predicted value for the studied property. Future work include integration into the FactSage software and the FactOptimal module. FactSage is a software system that was created for treating thermodynamic properties and calculations in chemical metallurgy [4]. It is used today all over the world by more than 400 universities and companies in the domain of material chemistry. It contains various modules allowing users to perform a wide variety of thermochemical calculations [3]. One of the modules, called FactOptimal [22, 24, 23], allows one to find the best set of conditions respecting given constraints. The program uses the NOMAD derivative-free solver [37] to find the best parameters to optimise the chosen properties. For example, given a chemi-

cal system (ex. $a_1\mathrm{C} + a_2\mathrm{Mn} + a_3\mathrm{Si} + a_4\mathrm{Cr}$), one may wish to find the values of chemical compositions (a_i) that would give an equilibrium temperature of around 275°C. To do so, NOMAD tries different combinations of compositions (a_i), obtaining the corresponding value of temperature from FactSage until, hopefully, an optimal solution is found. While performing this optimisation, certain constraints on composition or various properties can be set. The idea to introduce materials properties of a given chemical system as possible constraints or as values to be optimized requires the use of a machine learning tool to predict these properties. Because a large number of predictions are performed during a FactOptimal run, the computational time to make these predictions is of great importance. Furthermore, we wish the chosen model to be usable for on-line learning, as it may be the case that new experimental data is fed dynamically into the learning database.

There exists a variety of machine learning techniques for predicting a function $f(x)$ given x. Polynomial interpolation was one of the first to be developed [41], and is still a very popular method in fields such as digital photography and image re-sampling as well as for scientific data. Gaussian processes (GP) were introduced in the 1940's [49], but it is only in 1978 that they were employed to define prior distributions over functions [47]. More recently, with the introduction and increasing popularity of neural networks with back propagation, Gaussian processes started to be used for supervised machine learning [54] and for regression problems [77]. In the last few years, various attempts have been made to improve known approaches, in particular by the group of Robert B. Gramacy at the University of Chicago, with the introduction of treed Gaussian processes [31] and dynamic trees [69]. In 1996, Radford Neal showed that a Bayesian neural network with a Gaussian prior on individual weights with an infinite number of hidden nodes converges to a GP [45]. The "No Free Lunch theorem" was introduced in 1997 by Wolpert and Macready [78], stating that for every optimization problem, there is no perfect algorithm. For a given problem for which an approach works well, there exists another problem for which the same method fails miserably. One of the aims of this research is to compare different machine learning techniques for predicting properties of different types of material science data.

GP is a well-known and highly reliable regression model in Machine Learning.

Its non-parametric nature makes it flexible and particularly adaptable to various types of data. It has been widely used in scientific data analysis, such as prediction of materials properties, microstructure evolution simulation, prediction in thermo-mechanically processed metals, robot control, etc. It has proven to give very good results for predicting materials properties [1] and is one of the recommended methods. Though GP has proven to be superior to other existing regression models in terms of reliability, it suffers from high computational cost caused by matrix inversion operations in both the learning and regression steps. In some cases, the learning step is only required to be preformed once, as the learned hyperparameters of the model can be repeatedly used for subsequent queries. However, applications such as material property predictions are generally for more than one query. Scientists may upload a large number of chemical compounds with different constraints in order to make predictions. The low efficient regression step in the conventional GP is not capable of dealing with the streaming queries on a large scale. Not only for optimisation, but for a growing number of real-time applications such as robot dynamic control, on-line learning is required. It is extremely time consuming when applying conventional GPs, which makes real-time responses impractical and unsuitable.

In this research, we propose a novel approach to perform the conventional GP efficiently with a three-step strategy. With this so-called scalable Gaussian process (SGP), the size of the training data used for learning and regression is significantly reduced, resulting in a promising efficiency improvement. Meanwhile, the intrinsic information embedded in the training data is kept in the reduced data set, which guarantees a high accuracy of the regression. Our focus is on material science data of molten oxides systems. Real material science applications are studied and we have access to three databases: Martensite start temperature (Ms), electrical conductivity (EC) and molar volume (MV). These datasets are described in details in Section 3.1. Comprehensive experiments on two of these datasets show the outstanding performance of the proposed method compared with the conventional GP. Furthermore, collaborative work testing our SGP using data obtained from gas sensor detection is briefly presented, showing the versatility of our method.

We also perform a comparative study of the predicting power of our new SGP

with five of the most popular and emerging machine learning techniques. We wish to demonstrate how a thorough knowledge of the problem as well as machine-human interactions can improve the quality of the predictions. Stry et al. compared the quadratic and linear interpolation applied to the numerical simulation of crystal growth [68]. They found that a custom quadratic approach developed by them gave more accurate results with smaller computational time. Ghosh and Rudy found an improvement of the relative error of reconstructed versus measured epicardial potentials of Electrocardiographic Imaging when using a quadratic interpolation instead of linear one [27]. Skinner and Broughton published their work on Neural Networks applied to material science, and compared different methods for finding the weights of feed-forward neural networks [60]. In the present work we have added comparisons with more recent techniques: linear and quadratic interpolation, neural network, GPs, and dynamic trees. We also include a comparison with our new strategy, the SGP.

In the field of material science, the databases used to train the models and make predictions on materials properties consist of experimental points collected from the literature. If the databases do not already exist, the work simply involves a bibliographical research, making it far less costly than performing actual experiments. Once a database of experimental points is assembled, one can use a machine learning model to fit the data and predict the desired properties in unknown areas, or simply consult existing data in a desired region.

One issue with databases consisting of experimental points is the human error involved in collecting the measurements. Furthermore, since these databases are assembled from different sources, some conflicting information between sources (authors) can alter the prediction accuracy of the chosen machine learning technique. In this work, on top of introducing our new SGP, we are looking at a way to improve the reliability of databases consisting of experimental points by analysing the conflicting information and attributing a quality measure to each source: the various papers from which the points have been extracted, or authors. We developed a new truth discovery approach to calculate and compare the reliability of sources by using the amount of conflicting information for each source in combination with the amount of non-conflicting similarities with other sources. A level

of reliability can then be attributed and the sources can be ranked, making it possible to choose between two conflicting data points. With this novel approach for analysing the data, a given database can be screened and improved by removing data points believed to be in error.

In order to test our technique, we use one of the databases made available to us, consisting of data points on electrical conductivity (EC). We performed predictions using the previously mentioned Scalable Gaussian process regression (SGP). First, we evaluate the strength of SGP by testing how much conflicting information (noise) can be introduced and supported by this interpolation technique. Then we apply our new truth discovery approach to see how the predictions can be improved. We compare the prediction accuracy before pruning the database using our sources ranking truth discovery technique and after the database has been purged. On top of improving the predictions of machine learning techniques, the filtered database becomes more reliable when consulting existing information. Faced with conflicting data in an existing database, it can be confusing for a human being to decide which source is more reliable than an other. The process can involve time in research and reading and rely on a subjective evaluation. Our approach can therefore automatise this process and improve the quality of existing databases consisting of experimental points.

1.2 Challenges

Because of matrix inversions involved in GP, the computational time is typically n^3, making it very tedious for large datasets. For example, the training phase of a GP for a training set of around 1000 points would take around 1 hour, running on an average desktop computer. This is totally impractical for on-line learning applications, and very challenging when dealing with even larger databases. The solution is therefore to reduce the size of the learning database as much as possible. The main challenge lies in the fact that we want to keep the prediction accuracy by preserving intrinsic information, while compressing and reducing the size of the learning database as much as possible to make the predictions fast enough for on-line learning. Furthermore, special considerations need to be taken into account

because of the nature of the data, as chemical interaction between the components can have an effect on the properties [14].

In the second phase of our work, involving truth discovery, the main challenge resides in making automated decisions as to which source is more reliable than another. Domain experts can manually go trough small databases and evaluate which author, in their opinion, is more reliable than another. But when it comes to a large database, where there can be many conflicting sources, this is impractical. Since it is hard to evaluate manually, it is also difficult to know if our automated technique is making accurate choices when it comes to eliminating data. We certainly do not want to remove important information from the database to filter.

1.3 Contributions

In this research, we propose a novel approach to perform the conventional GP efficiently: the SGP. With our scalable Gaussian process, the size of the training data used for learning and regression is significantly reduced, resulting in a promising efficiency improvement. Meanwhile, the intrinsic information embedded in the training data is kept in the reduced data set, preserving high accuracy of the regression. We also provide an in-depth comparison of commonly used machine learning algorithm, including our new SGP, providing an analysis and recommendations depending on the nature of the data. Finally, we propose a new truth discovery method to improve scientific databases consisting of points collected from the literature. To be more specific, we make the following contributions:

- We propose a three steps method, SGP, making real-time prediction using GP possible. The first step consists of a fast batch query processing algorithm to handle large numbers of queries by grouping them by similar characteristics. In the second step, we analyse the structure of the training data and condense it by removing redundant information and preserving embedded intrinsic information. Finally in the third step, a query-aware training data selection strategy is designed to further enhance the efficiency of the model by taking into account the relationship between the query and the training data.

- We conduct extensive performance studies of SGP on real-life materials datasets, which are large scale from the perspective of machine learning.
- We compare and analyse six machine learning algorithms (linear and quadratic interpolation, neural network, dynamic trees, GP and SGP) on three physical properties databases.
- We propose a new truth discovery approach for scientific databases, utilising the amount of agreements and conflicting information in order to filter the data and remove possible experimental errors.

In the following chapters, we first provide a comprehensive literature review, followed by a description of the databases that were employed for this research. Then, we briefly describe each interpolation technique that was employed in comparing our methods. Then, we present our new scalable Gaussian process, including performance results in terms of computational time and accuracy compared to the traditional GP. Then, we present results in comparison with other models and make recommendations on the use of each method depending on the type of problem. Finally, we present our new truth discovery technique, followed by the conclusion.

2. Literature Review

In this section, we introduce the related work in prediction of materials properties, Gaussian process regression for machine learning using a large amount of data, clustering of high-dimensional data and truth discovery.

Predicting the martensite start temperature (Ms) has been reported by several authors. While some had good predictions using a neural network model [76, 65], others preferred a thermodynamic framework [67] or a purely empirical approach [38, 48]. These methods have been thoroughly investigated by Soumail et al. in 2006 [64]. Their conclusion was that although the thermodynamic approach provides satisfying results, there is a strict limitation in the query points, based on the fundamental assumptions upon which the model was based. They found that the neural network approach performs as well as other methods, however some wild predictions were obtained and they recommended the use of a Bayesian framework. Very accurate predictions were obtained for the prediction of austenite formation (martensite is formed in carbon steels when cooling austenite) using a Bayesian Gaussian process model [2]. However, developing a strategy to make on-line learning possible is highly desirable, as explained in section 1.1. Previous work of Marek Sloński [61] compared feed-forward layered neural network with Gaussian process, testing them on two datasets: high-performance concrete mix proportion and concrete fatigue failure. Their experiments showed the superiority of the Gaussian process in terms of accuracy and computational time. Based on these results, we believed that the GP will perform well with our physical properties datasets and this is why we chose to develop a strategy to adapt this particular model for on-line learning.

An empirical model [43] and a combined model with quantum chemical molecular dynamics and kinetic Monte Carlo method [70] were applied to predict electrical conductivity. Both models are developed specifically for electrical conductivity and would require extensive work to be adapted to predict other physical properties.

All the published work we found on prediction of Ms and electrical conductivity discussed their results in terms of prediction accuracy and no reports were given on the computational time.

The problem of high-dimensionality and large amount of data for Gaussian processes has been studied by E. Snelson et al. [62] and R. Urtasun [72]. They both proposed partitioning the data, which is the approach we adopt in this present study. More recently, a stochastic variational inference approach has been introduced by J. Hensman et al.[33]. A filtering approach based on approximation of eigenvectors, was also developed by J.Q. Shi et al. [58]. Although proven to be efficient, we believe that such approach would include irrelevant data and might miss important information for making accurate predictions as it relies on an initial subset of randomly chosen values. A. Banerjee et al. [5] recently introduced an approach using linear projection of the data points onto a lower-dimensional subspace.

In the area of clustering of high-dimensional and large amount of data, McCallum et al. [40] introduced the idea of using canopies as a cheap approximate distance measure as a first data divider for high-dimensional datasets. Huang et al. [34] introduced an effective co-clustering approach, this method was used for multimedia similarity search and was not fully compatible with databases containing chemical compositions but we are taking inspiration from both ideas.

The topic of truth discovery is not new and has been extensively studied, especially in the domain of social networks and the world wide web, where many conflicting information can be found, and where the duplication of wrong information also becomes a problem. In their paper, Yin et al. [79] discuss the trustworthiness of websites by evaluating the amount of true information contained on the given website. The same authors propose a semi-supervised method for homogeneous network, again applied on web sources [80]. Kleinberg [36] also proposes a test algorithm to evaluate the quality of web pages according to their relationships with other pages. In our work we take inspiration from this approach by considering the amount of similar information linking our various sources together and how much they agree with each other, even though our sources are completely inde-

pendent. Dong et al. [19] discuss truth discovery when accessing various sources of information, when the update history is known. They are evaluating the quality of sources over time and conducting a probability analysis. In another paper [20] they discuss the selection of sources when there is an overwhelming abundance of possible sources. A maximum likelihood approach is used by Wang et al. [74] to filter noisy social sensing data. Zhao et al. developed a probabilistic model for data steams, in order to evaluate sources quality in real time [81], applying their approach to weather forecast data.

None of the previous approaches have been applied on sets of experimental data points. In the field of data mining, Sheng et al. [57] address the problem of noisy labeling of data by carefully selecting a set of points where labelling will be repeated. Dekel et al. [17] are also proposing a way to prune low quality labels in a crowd.

3. Preliminaries

In this chapter we first present the databases that were made available to us, followed by a brief description of each interpolation technique applied in this work.

3.1 Datasets

For this work, we have access to five multidimensional databases in the field of material engineering on the following physical properties: martensite start temperature (Ms), electrical conductivity (EC) and molar volume (MV). The databases all consist of experimental points collected from the literature. For both the MV and EC databases, the materials are insulating oxides, therefore EC refers to the ionic conductivity. MV data is considered smooth, while EC nonsmooth and Ms noisy (several local minima in a small domain). Some physical properties can be measured with reasonable accuracy, therefore there is very little discrepancy between the different data sources. Moreover, certain properties have a quasi linear dependence with the constituents chemical compositions, while others may have a more complex dependence on compositions and can vary exponentially according to the temperature (singularity and local extrema). Measuring the molar volume on liquid oxides at high temperature can lead to a relatively large level of uncertainty and discordance between existing data sources. Despite this fact, we consider the molar volume as **smooth** as most of the dataset has little discrepancy. Furthermore, the theory tells us that it should vary almost linearly and the experimental datasets are in good agreement at equal composition and temperature. Electrical conductivity is also measured at high temperature, leading to a lower level of confidence. This combined with the fact the data is very scattered, and that it has a complex dependence on compositions, and obeys Arrhenius laws (see Section 5.2), we consider EC as being **nonsmooth**. MV and EC are properties dependent on the same variables describing Gibbs free energy under a certain atmospheric pressure. On the other hand, Ms is influenced by kinetic factors such as the cooling rate. These factors are not considered in our dataset and for this

is a reason why Ms is considered **noisy**. Also because of its dependence not only on compositions but also on the different phases within a given steel. Here we are omitting to include certain influential parameters such as the fine austenite grain size [21] and are considering uniquely the initial composition.

3.1.1 Molar volume

The database employed for molar volume predictions has 2,700 data points (n=2,700), with various compositions in mole percent on 10 dimensions (D=10), temperature in Kelvin and an associated molar volume value in cubic centimetres per mole. The experimental points were assembled from a total of 80 sources. See Table 3.1 for an example of data points taken from the molar volume database.

SiO2	**Al2O3**	**MgO**	**CaO**	**Na2O**	**K2O**	**LiO2**	**MnO**	**PbO**	**T(K)**	**MV**
53	0	0	5.1	41.9	0	0	0	0	1573	26.61
56	0	0	0	0	0	0	0	44	1323	25.55
78.56	0	0	0	14.3	0	7.14	0	0	1773	26.5
47.6	5.61	21.29	25.5	0	0	0	0	0	1773	22.93
Example query point:										
55	0.1	0	1	43.9	0	0	0	0	1773	N/A

Table 3.1: Sample data points for the molar volume database. Input compositions are in mole percent and the molar volume in cm^3/mol.

3.1.2 Electrical conductivity

For EC, we have access to three databases. The first dataset consists of approximately 15,700 entries over 29 dimensions (D=29), taken from a total of 121 sources. This is considered to be very large as far as experimental points databases are concerned. As per the MV database, each row has a set of chemical compositions in mol % with an associated logarithmic value of electrical conductivity in Siemens/meter. We have collected this data from the literature, from a total of 121 sources. In addition to the set of chemical compositions, the temperature in Celsius is also provided for each data point. The range of chemical compositions varies between 0 and 100 mol % while the temperature varies between approxi-

mately 90 and 3,000 K. In our calculations, we rescaled the temperature by a factor of 200 to make the data more uniform and thus obtaining better predictions. This database is used for testing our introduced SGP technique, see Chapter 4.

In addition to the full EC database, we are using two reduced datasets (EC Red 1 and EC Red 2). The first one consists of approximately 9,300 data points with compositions in mole percent over 10 dimensions including temperature (T) in Kelvin (D=10) and an associated EC value in Siemens per meter, taken from 97 sources. The second one consists of 5373 data points from 67 different sources over 9 dimensions including temperature. Besides the temperature, the inputs for the first reduced database are chemical compositions from 9 oxide components in mole percent: SiO_2, CaO, K_2O, Li_2O, PbO, Na_2O, MnO, Al_2O_3, MgO. For the second reduced database, the components are: SiO_2, Al_2O_3, MgO, CaO, MnO, PbO, FeO, Fe_2O_3. See Table 3.2 for an example of data points taken from the second reduced database.

SiO_2	**Al_2O_3**	**MgO**	**CaO**	**MnO**	**PbO**	**FeO**	**Fe_2O_3**	**T(K)**	**EC**	**Source**
33.56	0	0	41.96	0	0	24.13	0.35	1573	48.00	1
33.3	0	0	0	0	66.7	0	0	1223	38.50	4
0	25.67	0	40.56	0	0	30.82	2.96	1673	94.00	6
50.00	0	0	25.00	25.00	0	0	0	1873	67.10	7
49.53	16.71	33.76	0	0	0	0	0	1923	18.79	31
25.31	0	18.33	9.01	0	0	46.37	0.99	1593	403.90	46
Example query point:										
54.24	5.32	40.43	0	0	0	0	0	1673	Predicted	N/A

Table 3.2: Sample data points for the electrical conductivity database. The input compositions are in mole percent and the electrical conductivity in Siemens per meter.

3.1.3 Martensite start temperature

Martensite is a crystalline structure formed in the process of cooling carbon steels at high rates (quenching). Controlling the amount of martensite in a given steel is critical as it has an important effect on the physical and mechanical properties of the steel. One of the variables engineers have to take into account is the Martensite Start (Ms) Temperature, which can be predicted by giving the amount of each

chemical component contained in a query steel. The Martensite start temperature (Ms) database consists of approximately 1,100 data points collected from the literature with composition values in weight percent on 14 dimensions (D=14) on 15 columns, where the first 14 columns represent the values in weight percent of 14 chemical components and the last one is the associated Ms temperature value in Kelvin. It covers a wide variety of compositions of steels; the main element, Fe, is not used in the regressions. See Table 3.3 for a full list of components and the composition ranges.

3.1.4 Composition ranges

Table 3.3 gives the range of compositions of each database. The Ms database is available for download on the Thomas Sourmail website [63, 65].

3.2 Theoretical Methods

In this chapter we very briefly introduce each interpolation technique. For more details, refer to the cited authors.

3.2.1 Gaussian Process Regression

In this section we give a brief description of the Gaussian process regression approach for machine learning. A Gaussian process (GP) is a generalized Gaussian probability distribution [49]. A Gaussian process regression computes the posterior distribution based on training data, or prior distribution. It has the advantage of being a non-parametric approach and adaptable to various situations, especially for high dimensional space problems [49]. However, when computing Gaussian process regression, one has to deal with matrices inversions, which leads to a typical computational complexity of n^3 where n is the number of training data points. Consequently, this model may be very slow and not suitable for on-line applications. The Gaussian process regression technique applied in this work is based on the earlier work of Gibbs and MacKay [28].

Let $f = (f_1, f_2, ..., f_n)$ be observed responses for one of the blackbox outputs

	Ms (Wt.%)		**MV (Mol.%)**		**EC (Mol.%)**		**EC Red 1 (Mol.%)**		**EC Red 2 (Mol.%)**	
Element	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**
C	0	2.25								
Mn	0	10.24								
Si	0	3.8								
Cr	0	18								
Ni	0	31.54								
Mo	0	8								
V	0	4.55								
Co	0	16.08								
Al	0	3.01								
W	0	18.6								
Cu	0	3.04								
Nb	0	1.98								
Ti	0	2.52								
B	0	0.006								
N	0	2.65								
Fe	65.09	99.83								
SiO2			0	90	0	100	0	100	0	100
Al2O3			0	90	0	100	0	100	0	50.5295
MgO			0	85.51	0	64.08	0	64.08	0	61.8928
CaO			0	87.91	0	74.79	0	74.79	0	69.45
Na2O			0	60.1	0	62.9277				
K2O			0	50	0	45.8	0	45.8		
Li2O			0	65	0	59.4	0	59.4		
MnO			0	77.17782	0	77.21	0	77.21	0	77.2107
PbO			0	95	0	100	0	100	0	100
CdO					0	50				
Na2O					0	67.92777	0	62.928		
SrO					0	60				
BaO					0	80				
TiO2					0	60				
NiO					0	35				
FeO					0	100			0	98.0146
Fe2O3					0	100			0	54.6376
B2O3					0	100				
V2O5					0	100				
Cr2O3					0	4.559734				
P2O5					0	50				
ZnO					0	50				
BeO					0	30				
Sb2O3					0	100				
Rb2O					0	9.09				
ZrO2					0	9.09				
Cs2O					0	4.23				
Bi2O3					0	4.23				
GeO2					0	4.23				
T(K)			713	3273	364.65	3223.15	398	3223	973	2753

Table 3.3: Ranges of the databases.

at inputs $X = (x_1, x_2, ..., x_n)$ which can be considered as a set of training points in a n dimensional space R^n. The objective is to learn a function $\Gamma(X)$ transforming the input vector into a target function $f(X) = \Gamma(X) + N_G(\mu, \sigma)$ where N_G is a Gaussian noise for which the mean, μ, is assumed to be zero everywhere and the variance is σ_n^2. In this case the covariance function K relates one function value to another one. In this work we consider the Gaussian kernel to define the covariance matrix as in previous work of Gibbs and MacKay[28]:

$$K(X, X') = \sigma_f^2 \exp\left\{-\frac{1}{2}\sum_{j=1}^{n}\frac{(x_i - x'_j)^2}{w_j}\right\} + \sigma_n^2\delta(X, X') \tag{3.1}$$

Where δ is the Kronecker delta function, σ_f^2 denotes the overall variance of the process and w represents the width of the Gaussian kernel, it governs the rate of decay of the special correlation in each input direction, in other words δ is a characteristic euclidean distance above which two points will be uncorrelated. The joint distribution of the observed and predicted function for a special point (i.e. composition in our work) is given by

$$f(X^*) = K_*^T(K + \sigma_n^2 I_w)^{-1} f \tag{3.2}$$

with $K_*^T = K(X, X^*)$. σ_n^2 and I_w are a set of free parameters for a flexible customization of the GP to take into account the specificity of the problem. These two adjustable parameters are called hyperparameters. They are usually automatically optimized using Quasi-Newton methods [56] by maximizing the log marginal likelihood of the model given the data. The choice of the covariance functions and the two hyperparameters is the first step of the GP process often denoted by "model selection".

After the model selection, the second step of a GP consists of performing a model regression performed upon the input functions (training), typically the available or part of the available experimental data. The variance of the predicted function resulting from the regression step is then given by

$$V_f(X^*) = K_{**} - K_*^T(K + \sigma_n^2 I_w)^{-1} K_*^T \tag{3.3}$$

with $K_{**} = K(X^*, X^*)$. From the above equations, one can see that a GP requires operations using a covariance matrix K, represented by covariance functions on all possible combinations of training data point pairs. All training data

points also have to be processed in order to perform the regression and compute K_*^T. From this we can conclude that the computational cost of a GP is heavily dependent on size of the training data and will grow exponentially. For this reason, GPs are not practical for real-time applications.

The standard GP model has two main components: the optimisation of the hyperparameters to be used in the covariance matrix and the actual regression with the query points. Both require matrices inversions, and the computational cost is therefore heavily linked to the size of the training database. Typically, the computational complexity of performing the necessary matrices inversions is proportional to n^3 where n is the number of training data points.

3.2.2 Linear interpolation

Linear interpolation is no doubt one of the simplest method one can employ to fit experimental data. One assigns parameters $b \in \mathbb{R}$ and $c \in \mathbb{R}$ such that $f(x)$ can be predicted using a linear model of the form

$$bx + c. \tag{3.4}$$

For multidimensional problems, normalized areas bound by known data are used in order to interpolate unknown data points [73]. This method has the advantage of being easy to understand, fast and straightforward to implement, but it is an approach specific to a given problem since it is parametric. Because of this approach, while doing on-line learning, parameters have to be recalculated each time new data is added to the learning set. While this is adding to the computational complexity, the most important limitation of the linear interpolation model is that it is a simplistic approach that may be inappropriate for complex problems. Linear interpolation has been used successfully on many varied problems, including pricing and stock market [35, 16], medical science [15] and digital imaging [39].

3.2.3 Quadratic Interpolation

Both linear and quadratic interpolation techniques belong to the polynomial interpolation family. Linear interpolation is limited to a model of the first order while

quadratic interpolation is of the second order. Similarly to the linear interpolation approach, the objective is to find parameters a, b and c such that $f(x)$ can be predicted using a quadratic function of the form

$$\frac{1}{2}ax^T + bx + c \tag{3.5}$$

The data is represented by a quadratic. As with linear interpolation, this is a parametric approach, with the same disadvantages. However, it is also a simple method to implement and predictions do not require a lot of computational time. It has been successfully used in image reconstruction and sampling [18] as well as in astronomy [55].

3.2.4 Neural Network

The Neural Network approach has been extensively employed in recent years in applications such as pattern recognition [53] and material science [59]. Inspired by the nervous system, neural networks are composed of highly interconnected elements, working together to make predictions. It is a very good approach when working with nonlinear functions [66] as it can detect complex relationships between independent variables. However, disadvantages include a large computational time, its empirical nature and a tendency to overfit [71]. As with polynomial interpolation, model parameters have to be carefully chosen and are specific to a problem. For this work we used the Tiberius data mining software [42] version 7.0.7.

3.2.5 Dynamic Trees

The idea with dynamic regression trees or dynaTree, as implemented in the R software package dynaTree [29], is to partition the space with several tree models where each tree corresponds to one partitioning scheme and each leaf of each tree corresponds to a region. Once these trees are determined, predictions are achieved by averaging model values over all trees. The main advantage of such an approach is the use of simple models within each partition [69]. It is a non-parametric approach, and particle learning algorithms make on-line learning possible. Because of the partitioning approach, it may be well suited and modelled for real-world applications where variables can be of totally different nature. However, one of

the disadvantages of such an approach is that the generated trees may be very large and complex. Also, as with any partitioning approach, there is always the risk of too much data approximation. In this work we use two versions of dynamic trees: the constant model (dynaTree CST) and the linear model (dynaTree LIN). The difference lies in space partitioning. Both make use of a full binary tree, the constant model with a fixed number of leaf data points, three, and $2 + D$ for the linear model, D being the dimension of the covariate space.

4. Scalable Gaussian Process Regression

A scalable GP is highly desirable because of the following two issues. Firstly, when given a fixed training data set, the optimisation step only needs to be performed once, since the hyperparameters can be saved and reused. However, to achieve accurate predictions for different kinds of query points, the training set has to contain as much information as possible, which results in a very large scale training data set. For this reason, the existing methods generally suffer from loading the training set with large amounts of data points. Secondly, in many real-time applications, such as robot control, on-line learning and regression are required. Since the computational cost of GP is highly associated with the training data size, in the present study, we aim to design a scalable GP algorithm by reducing the training data size while maintaining the intrinsic information embedded in it. The proposed algorithm has three stages: 1 - Batch query processing, 2 - Training data condensation and 3 - Query-aware training data selection. In the following sections we refer to our strategy as the Scalable GP (SGP).

4.1 Batch query processing

While typical materials optimisation calculations are performed sequentially as they are dependent on the previous result, we will be considering large amounts of input queries in our application. In order to reduce the computational cost on on-line regression for streaming queries, we conduct batch query processing by considering the similarities between query points. According to their different characteristics, the query points are clustered into groups, each of which will be represented by a summarized representative point. The representatives will be passed to the regression model and be used for the training data selection. We apply an agglomerative clustering approach to first group points in pairs of closest points using the Euclidean distance. It then groups pairs together and so on until a target number of points per group is obtained. The function used to measure the Euclidean distance between two points and two groups of points is as follows:

$$\sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \tag{4.1}$$

Where p and q are two points in an Euclidean space of dimension n. This method for clustering data in high-dimensional space has proven to be a simple but efficient one [75].

When comparing two groups of points, the geometrical mean on each dimension is used to calculate the Euclidean distance. Given a data set $\{\overrightarrow{x_1}, \overrightarrow{x_2}, ...\overrightarrow{x_n}\}$, the definition of geometrical mean on dimension d is as follows:

$$\left(\prod_{i=1}^{n} x_{di}\right)^{\frac{1}{n}} \tag{4.2}$$

The geometrical mean of a set is based on the product of the values instead of a sum. This type of mean is particularly useful when attempting to minimise the impact of data with different ranges, which could occur with data for prediction of material, where different scales might be found in the set of points.

4.2 Training data condensation

The second step consists of a pre-filtering of the entire training data. This is done to condense redundant observations, and therefore acts as a first dimensionality reduction step. Inspired by the co-reduction approach introduced by Huang et al. [34], we reorganise the rows by similarity and then combine them together using a reduction function Θ. Our reduction function Θ consists of computing the mean values on each dimension of the two merged rows. A reduction on the number of columns will be achieved in the final selection of data (section 4.3). Here a simple Euclidean distance function between the points is not enough, because we want to avoid a situation where two points would be far on one dimension and identical in every other dimensions. We want the clustered points to be close to each other on every dimension, due to the nature of the data. With chemical compositions, it can be the case that one of the components makes a very big difference on the value of the physical property, as there could be possible interactions with the other components present. For this same reason, we could not fully apply the

co-reduction technique and introduce a column reduction function. For example, let us consider the following three points in a 4 dimensional space:

$$\begin{array}{c} \\ A \\ B \\ C \end{array} \begin{array}{c} \begin{array}{cccc} C & Si & N & Mo \end{array} \\ \begin{pmatrix} 20 & 3 & 0 & 1 \\ 15 & 3 & 4 & 1 \\ 26 & 5 & 0 & 4 \end{pmatrix} \end{array}$$

Using equation 4.1, the Euclidean distance between A and B gives a value of approximately 6.4, while the Euclidean distance between A and C gives a value of 7. According to our previous reasoning, we wish to favour the clustering of A and C because they have actual data in the same dimensions, thus reducing the risk of component interaction affecting the physical property. Therefore, we use the following rule to compare two points p and q:

$$\forall i \in \mathrm{N} : \left(\frac{|q_i - p_i|}{\sum_{j=1}^{N} q_j} < \epsilon \right) \wedge \left(\frac{|q_i - p_i|}{\sum_{j=1}^{N} p_j} < \epsilon \right) \wedge ((p_i = 0) \leftrightarrow (q_i = 0))$$

Where N is the total number of columns (dimensions) and ϵ is an arbitrary condensation constraint, we tested with values of 0.5, 1, and 5%. If the above predicate is true, then the two rows can be merged together applying Θ. The algorithm is executed recursively until no more merges are possible.

4.3 Query-aware training data selection

Before calculating the actual predictions, we perform the final selection of the training points by considering the relation between the representative query points generated from the first stage and the condensed training set created in the second stage. This is done by first calculating the geometrical mean (i.e., Equation 4.2) on each dimension of the batch query. Once the geometrical mean (g) is found, we compare this value to each point (p) in the condensed training set obtained, using a modified Euclidean distance formula:

$$\sqrt{\sum_{i=1}^{n} \left(\frac{g_i}{\sum_{j=1}^{n} g_j} - \frac{p_i}{\sum_{j=1}^{n} p_j} \right)^2} \tag{4.3}$$

In other words, we calculate the Euclidean distance on normalised values. It is because we want to measure the distance using proportions of chemical composi-

tions instead of the actual values. We keep an arbitrary number of results in the final training set, the ones with the closest distance to the geometrical mean (Fig. 4.3). A number of K similar points from the training data set will be selected for each batch query to be considered as its local or specific training data. The parameter K for each batch query is determined by the acceptable predicted error bound. That means the value of K is decided depending on the accuracy of the prediction. As you can see on Fig. 4.3, some points can be present in more than one training set, this will ensure consistency for each batch query.

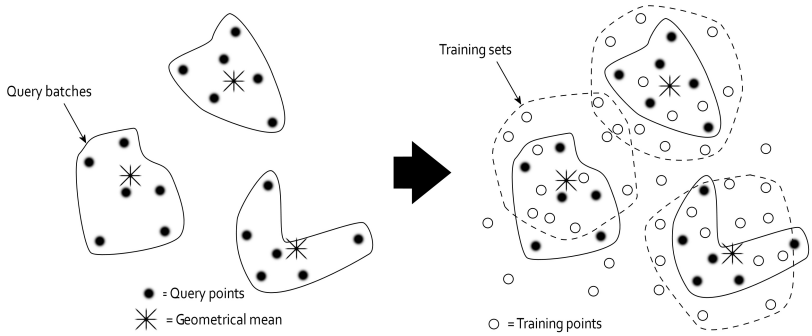


Figure 4.1: Final selection of the training points: K-NN of the geometrical mean

Besides considering the error bound, an optional step is designed to involve the K' nearest-neighbours (K'-NN) from the training data for each target batch query point, using a normal Euclidean distance (Equation 4.1). For data where the training set is very large, this method can be very effective as there is a higher chance of having close data to the query points in the training set. If the set of training points is smaller, we have found that this extra step is not necessary. However, if there is a concern about the data being concentrated in certain areas as illustrated in Figure 4.2, it may be necessary to include this step to ensure that relevant points are included in the training set.

If the target error bound can not be reached using the selected number of training data, the number of training points is increased and the regression is calculated again until the target error bound is reached. The reduced number of training points allows us to do a further clustering of the data, eliminating the dimensions where there is no composition available, therefore reducing the number of columns in the training matrix.

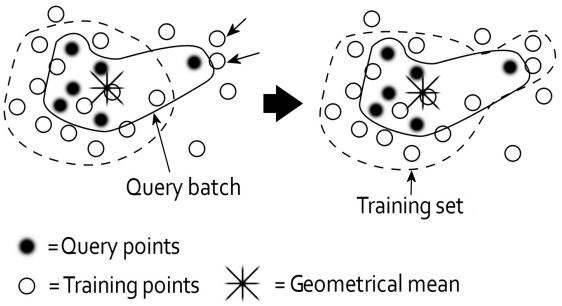


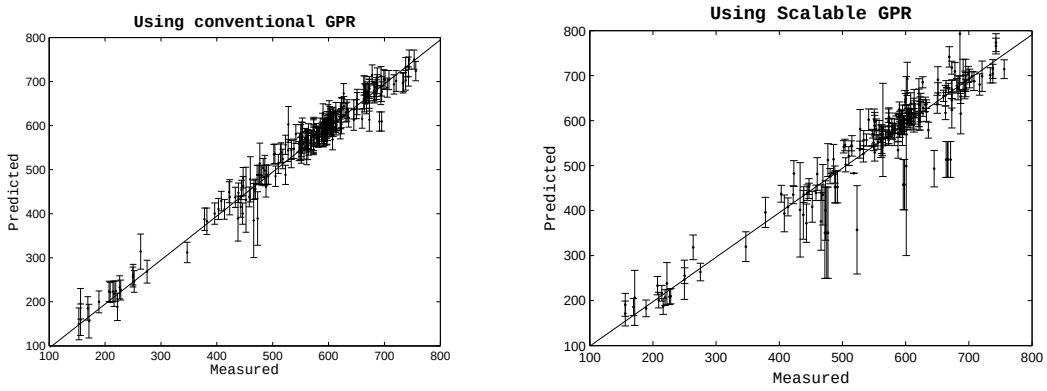
Figure 4.2: Final selection of the training points for each batch of predictions (query batch): the closest points to the geometrical mean are chosen first (left) then the training set is expanded to include the closest points to each point (right).

4.4 Performance study

4.4.1 Application on Prediction of Ms

To evaluate the performance of the proposed Scalable GP, we conduct a series of experiments on Ms temperature predictions. The database used in this study is described in details in Section 3.1.3.

We randomly take 80% of the available points for training and the remaining 20% for testing the predictions. Thus, the numbers of training points and testing points are 870 and 220 respectively. This procedure is repeated 10 times to get the final prediction performance.



(a) (b)

Figure 4.3: Predicted vs Measured Ms Temperature (K)

	AE (%)	**RMS (K)**	**Total training time (sec)**	**Average prediction time per testing point (sec)**	**Average time cost per testing point (sec)**
GP	3.08	21.6	969.40	5.52	9.92
Scalable GP	5.02	42.6	26.5	0.13	0.25

Table 4.1: Conventional GP vs Scalable GP for predicting Ms

Batch	**Size of the training matrix**	**Training time for each batch query (sec)**	**Average prediction time per testing point (sec)**
1	95×13	4.35	0.14
2	93×12	5.29	0.17
3	96×13	3.06	0.10
4	95×11	2.48	0.08
5	108×14	4.99	0.14
6	165×12	7.09	0.14

Table 4.2: Batch query performances for prediction of Ms

Conventional GP: To illustrate the superiority of the proposed Scalable GP, we first test the conventional GP by using the training and testing data described in Section 5.1. The prediction values obtained in our experiment are as consistent as those of Bailer-Jones, Bhadeshia and MacKay [2] (Fig. 4.3(a)). Two performance indicators average error (AE) and root mean square (RMS) are used to evaluate the accuracy of the testing method, which are defined as follows:

$$AE = \frac{1}{N_t} \times \sum_{i=1}^{N_t} \frac{|p_i - a_i|}{a_i} \tag{4.4}$$

$$RMS = \sqrt{\frac{\sum_{i=1}^{N_t} |p_i - a_i|^2}{N_t}} \tag{4.5}$$

Where N_t is the number of testing points, p is the predicted value and a is the actual value.

As reported in Table 4.1, the AE and RMS produced by the conventional GP are 3.08% and 21.6 degrees respectively. The average prediction time in the regression step for each testing point is 5.52 seconds, on an Intel i7 3.4GHz with 16 GB of RAM and the time cost to calculate the hyperparameters in the training step is 969.40 seconds. As we mentioned earlier, on-line training is required in many real-time applications. Taking into account both the training and prediction time costs, GP averagely spends 9.92 seconds on each testing point to make a prediction. The time cost at this scale is certainly impractical.

Scalable GP: Our proposed Scalable GP offers a significant efficiency improvement. Compared with GP, the training time is heavily reduced from 969.4 seconds to 26.5 seconds and the average prediction time in the regression step is reduced from 5.52 seconds to 0.13 seconds. It makes the real-time prediction realistic where the total cost including both training and prediction for an individual query point is 0.25 seconds.

As described in section 4.1, batch query processing is performed in the Scalable GP to achieve efficient query predictions. Here we choose one round of testing as an example to describe the batch query details. As illustrated in Table 4.2, a total of 6 batch queries are created in this testing round. We choose to set a condensation constraint of $\epsilon = 5\%$ which allows us to condense the training data from 870 points to 416 points in the second stage. The final selection of training points for each batch query is then performed using this condensed dataset, as explained in section 4.3. When trying to further condense the data, we observe that using an $\epsilon > 5\%$ leads to too much compression of the data, producing values in every dimensions for too many data points. In addition to a loss of information, this means that further vertical condensation is virtually impossible and therefore there is no further gain on the computational cost. In this example we set a target error of 10% or less, increasing the number of training points and using a lower condensation (ϵ) if not reached. The time cost to optimise the hyperparameters in the training step for each batch is reported in Table 4.2. We can observe that reducing the size of the training matrix is of critical importance to improve the speed of the GP.

It is always a trade off between efficiency and accuracy. To achieve scalable and efficient predictions, the accuracy of the Scalable GP is sacrificed, where the AE and RMS are 5.02% and 42.6 degrees respectively (Table 4.1). However, from a chemistry point of view, an average error of 10-20% or less is considered acceptable for predicting Ms. Thus the Scalable GP delivers a fairly acceptable accuracy (Fig. 4.3(b)) with significant efficiency improvement. Using our Scalable GP, 95% of predictions had an error of 20% or less.

Other comparisons: Besides the conventional GP, we also compare our method with Neural Network and SVM, which are widely used in scientific data prediction. However the Neural Network method takes more than 5 hours on training step for 870 training points and SVM delivers fairly poor predictions with low efficiency. The performances of both methods are not comparable with the Scalable GP in terms of either efficiency or accuracy. In 2011, Słoński also showed the computational cost superiority of the GP compared to Bayesian and standard Neural Network [61].

4.4.2 Application on Prediction of EC

The efficiency and the accuracy of the Scalable GP have been demonstrated in Ms temperature prediction. To further test the scalability of the proposed method, we conduct the second series of experiments on electrical conductivity predictions by involving a much larger scientific dataset. For this study we are using the larger EC database as described in Section 3.1.2. In this group of experiments, the conventional GP is not able to deal with the large scale training dataset due to the extremely expensive computational cost. In the following performance study, we will focus on the scalability our approach and discuss the effect of the batch query processing in the proposed Scalable GP.

Conventional GP: The experiments are conducted on a regular desktop computer, therefore attempting a standard GP using a training dataset with the size

of 2000×29 has proven to be very tedious and extremely slow. Thus we only randomly sample 2000 entries from the original dataset to build up the training data to test GP. With this setting, it costs 9.28 hours for training and 5.67 minutes per prediction in the regression step. Clearly, the conventional GP is not capable to handle real-time applications.

Scalable GP: With the training data condensation described in Section 4.2, the proposed Scalable GP can easily handle the large scale training data by capturing the intrinsic information embedded in and removing the redundant entries. We randomly select 80% entries (i.e., 12,560 entries) from the entire database to build up the initial training data set and use the remaining 20% entries as the testing points pool. Following the training data condensation described in Section 4.2, we condense the size of the training data from 12,560 points to 8,654 points by setting $\epsilon = 0.5\%$, which performs the best compared with $\epsilon = 1\%$ and 5%. We incrementally select 100, 500, 1000, 1500, 2000 and 3000 number of entries from the testing points pool as testing data to show the scalability, efficiency and accuracy of the Scalable GP and also the effect of batch query processing on the performance. As reported in Table 4.3, the performance of the batch query processing is quite stable. With the size increment of the testing data from 100 to 3,000 points, the number of batch queries generated is increased from 25 to 750. The average time to create a batch query was 0.06 seconds. With different numbers of batch queries, the average training time cost, prediction (regression) time cost, and the total time cost for each testing point is very stable. With the error bound of 15%, we can always achieve the real-time prediction response averagely within 0.9 seconds.

Number of testing points	**Number of batches**	**Average training time per testing point**	**Average prediction time per testing point**	**Average total time per testing point**	**AE (%)**
100	25	0.688	0.044	0.732	14.5
500	124	0.885	0.045	0.928	14.9
1000	250	0.966	0.042	1.008	14.8
1500	375	0.741	0.043	0.784	14.2
2000	500	1.01	0.042	1.06	14.8
3000	750	0.715	0.042	0.758	14.7

Table 4.3: Scalable GP for predicting Electrical Conductivity (target error of 15%)

4.4.3 Application on gas sensor data

The SGP approach has also proven to be efficient on another type of data: gas sensor data, or electronic nose. This type of data has high uncertainty due to noise and drift. This problem is divided in two parts. First, a classification model needs to be used to detect which gas is present. Then, another model makes predictions on gas concentration. Our SGP has been tested in step two, using sensor data for six gases over 128 dimensions (sensor values). Interestingly enough, because of the classification step, the traditional GP gives very bad prediction of gas concentrations (over 10,000% average error). This can be explained by the fact that GP attempts to fit a Gaussian curve to the whole set of data. However, in a classification problem, the data distribution is not Gaussian. A function in two dimensions would look more like a stepped line; a smooth interpolation between the steps is bound to give bad results. See Figure 4.4 for an graph illustrating the concentration in function of one dimension of the sensor data. However, using our SGP, this problem is eliminated, since our approach only considers the problem in small areas. In a collaborative paper in progress, SGP has been compared to four other machine learning techniques (SVM, dynaTree, linear regression and Tree Bagger). Results are illustrated in Figures 4.5 and 4.6. Note that results that were too high

are not illustrated in these graphs. Each coloured line represents a category of data (10,30,etc.), in other words, one class from the first classification step. As one can see, SGP gives excellent predictions compared to the other techniques for this type of data, with an overall average error of 14.27%. Furthermore, there were practically no outliers.

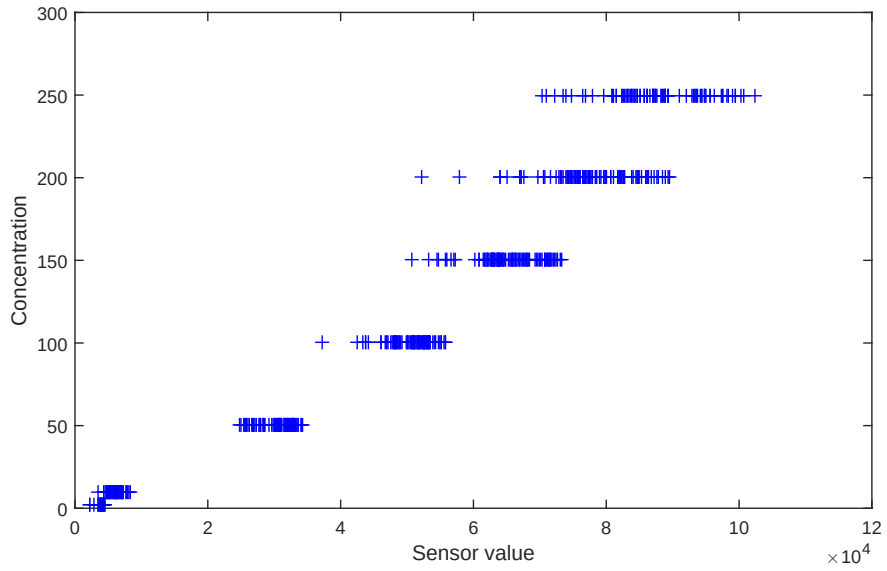


Figure 4.4: Concentration in function of the sensor value, here only one dimension (out of 128) is displayed for the horizontal axis.

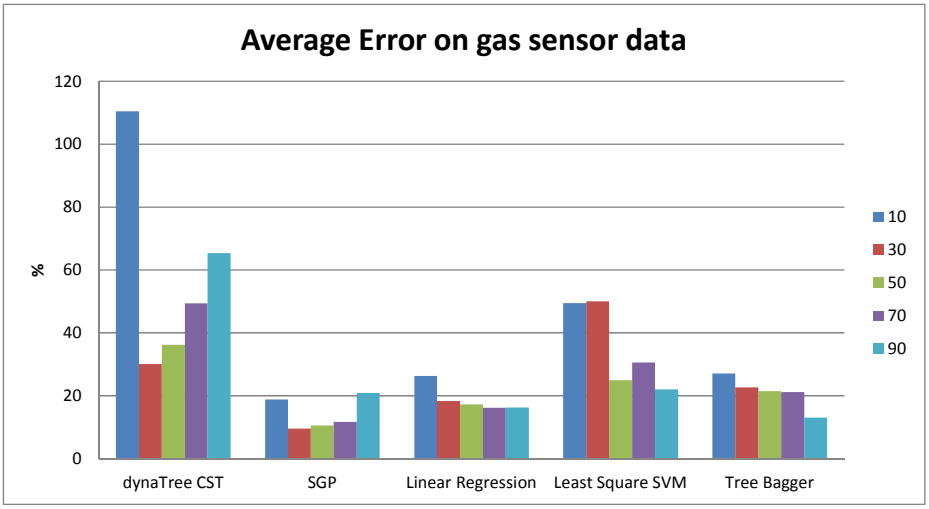


Figure 4.5: Average error on nose sensor data predictions.

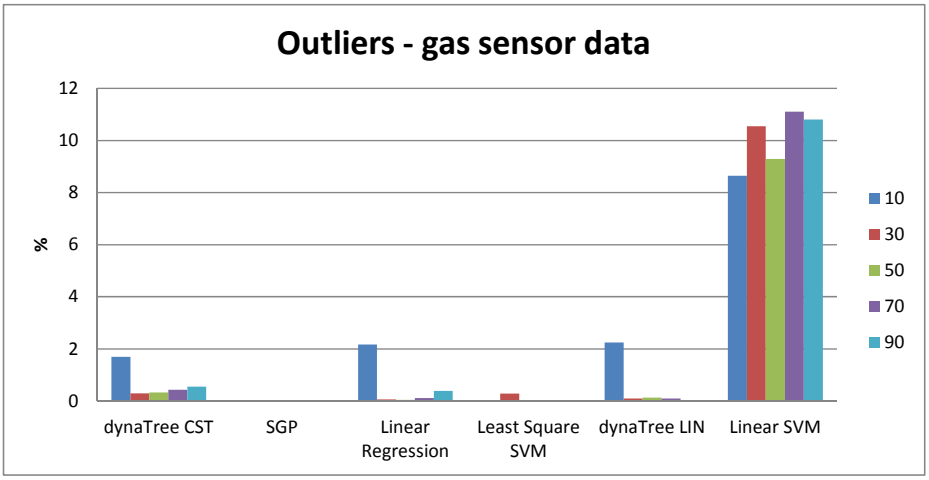


Figure 4.6: Percentage of outliers on nose sensor data predictions.

4.5 Online application

The SGP has been implemented into an web application [6] and is freely available for testing at the following URL:

http://www.crct.polymtl.ca/SGP/run_gp.php

5. Evaluation of Interpolation Techniques

We believe that different machine learning interpolation techniques could be better adapted to each set of data [78], and one of our goals is to compare the techniques, test them on the available datasets and make recommendations accordingly. With machine learning models, one would expect that the larger the training set, the more accurate the predictions. Therefore, our analysis includes verifying this theory and comparing the power of the chosen algorithms by testing different sizes of training sets proportionally with the testing sets.

In this chapter we present the prediction accuracy obtained when training the chosen models, followed by a general discussion. The chosen models are described in Section 3.2, in addition to our SGP approach, as described in Chapter 4. The computational time is discussed in Section 5.4. For each type of dataset treated in this work, we measure the quality of the techniques in terms of root mean square of the relative error (RMSE), given by the following equation:

$$RMSE = \sqrt{\frac{\sum_{x=1}^{X}(Q_o^x - Q_p^x)^2}{X}} \tag{5.1}$$

where Q_o^x is the observed value and Q_p^x is the predicted value for a query point x and X is the total number of query points. We also employ two other predictive accuracy measures: the Nash-Sutcliffe model efficiency coefficient (NS) [44] and the proposed Order efficiency coefficient. The NS coefficient is calculated as follows:

$$NS = 1 - \frac{\sum_{x=1}^{X}(Q_o^x - Q_p^x)^2}{\sum_{x=1}^{X}(Q_o^x - \overline{Q_o})^2} \tag{5.2}$$

It gives an indication of how good the predictions are compared to the mean of the observed values. The Order coefficient is determined by taking all possible combinations on pairs of predictions compared to the actual values. For each pair (i, j), if $(Q_o^i < Q_o^j)$ and $(Q_p^i < Q_p^j)$, or $(Q_o^i > Q_o^j)$ and $(Q_p^i > Q_p^j)$ then it is considered a good prediction, and a counter O is incremented by one. The Order coefficient is then calculated as follows:

$$Order = \frac{O}{X(X-1)/2} \tag{5.3}$$

where X is the total number of predictions. The Order coefficient gives an indication of how accurate the model is at comparing two points. The NS coefficient ranges from $-\infty$ to 1, and the Order coefficient from 0 to 1. In both cases, the closer to 1, the more accurate the predictions. If $NS \simeq 0$, it is an indication that the predictions are as accurate as the mean of the observed data ($\overline{Q_o}$), while $NS < 0$ indicates that the observed mean is a better predictor than the model.

For all three databases, we employ data collected from the literature of tables consisting of chemical composition for martensite start temperature, including the temperature for electrical conductivity and molar volume, for a $n \in [10; 15]$. See Table 5.1 for a simple example training set and query point on molar volume data. Each set of compositions (row) has an associated property that we are predicting. For each data set and technique, we randomly select the training data points from the database and use the remaining data for evaluating the predictions. In the following subsections we present results using from 50% to 90% of training data.

Training:

SiO2	**Al2O3**	**MgO**	**CaO**	**Na2O**	**K2O**	**LiO2**	**MnO**	**PbO**	**T(K)**	**MV**
53	0	0	5.1	41.9	0	0	0	0	1573	26.61
56	0	0	0	0	0	0	0	44	1323	25.55
78.56	0	0	0	14.3	0	7.14	0	0	1773	26.5
47.6	5.61	21.29	25.5	0	0	0	0	0	1773	22.93

Prediction:

60.56	5.08	28.57	0	3.53	2.3	0	0	0	1053	p

Table 5.1: Example of training and prediction query (p) for Molar Volume (MV) data. The input compositions are in mole percent and the molar volume in cm^3/mol.

For each technique, outliers (wild predictions) are excluded from the average RMSE, as we believe that including a few very large numbers would not give an accurate representation and thus the comparison would be distorted. Predictions with an error greater than 200% are considered as outliers. In Section 5.5 we discuss in more detail the percentages of outliers obtained for each tested technique.

5.1 Molar volume data

For this evaluation, we used the MV database as described in Section 3.1.1. The performance of each technique is illustrated in Fig. 5.1 and Table 5.2. All six techniques performed relatively well, maintaining an average RMSE below 10%. However, the GP was the clear winner with an average RMSE below 5% for every test. The linear, quadratic and dynaTree LIN models give very similar results, with an average RMSE of 7 to 9%. As expected, there is a general tendency for an improved accuracy as the proportion of training points increases. The more training data is available, better are the chance of covering the entire space. The dynaTree CST technique gives very good results for 4 datasets out of 5. This behaviour is confirmed by the NS and Order coefficients.

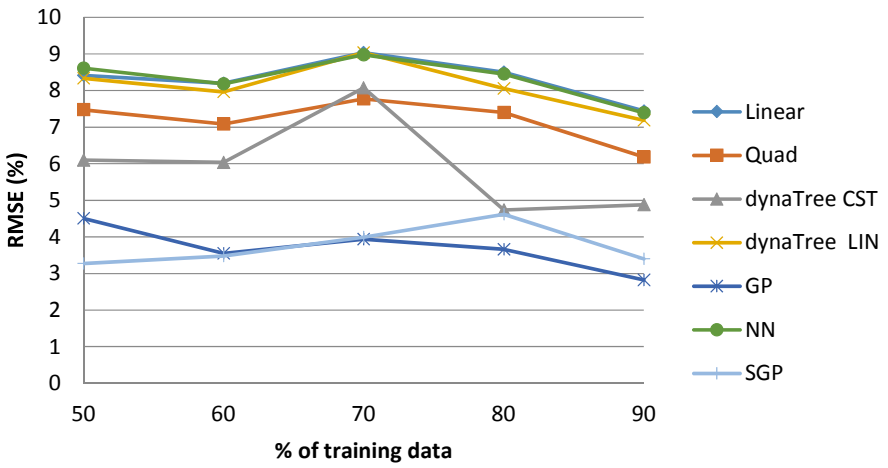


Figure 5.1: Comparison of the RMSE for Molar Volume predictions.

Technique	NS	Order	RMSE
Linear	0.9301	0.9104	8.4144
Quad	0.9501	0.9291	7.4706
dynaTree CST	0.9512	0.9129	6.0979
dynaTree LIN	0.9376	0.9217	8.3299
GP	0.9784	0.9509	4.5036
SGP	0.9783	0.9584	3.2712
NN	0.9166	0.9119	8.6115

Table 5.2: NS, Order and RMSE for Molar volume (50% training points).

5.2 Electrical conductivity data

We test the linear, Quad, dynaTree CST and dynaTree LIN techniques with actual electrical conductivity values *cond* as well as $ln(cond)$ and $ln(T \times cond)$. The database used is described in Section 3.1.2 and referred to as EC Red 1 in Table 3.3. As mentioned earlier, electrical conductivity here refers to the ionic conductivity. Table 5.3 shows that using $log(T \times cond)$ gives the best predictions, therefore we compare the RMSE making predictions on this value. This can be explained because in general, the electrical conductivity (κ) temperature dependence obeys Arrhenius laws, that is: $ln(\kappa) = \alpha + \beta/T$ where β is the activation energy and α is a value of electrical conductivity at a reference temperature. However, for silicate systems, there is a deviation from this law [52]. Consequently, we decided to test all three cases mentioned above to evaluate how the prior knowledge of the problem influences predictions quality. In this case there is a clear improvement on the NS coefficient (27%) while the Order coefficient had only a slight increase (2%). Figure 5.2 shows that the GP technique gave the lowest average RMSE for all the testing sets. The NN and linear interpolation techniques performed quite poorly, especially with only 50% of training data, giving respectively average RMSE of 47% and 28%, both with an NS coefficient of 0.79 compared to 0.98 for GP.

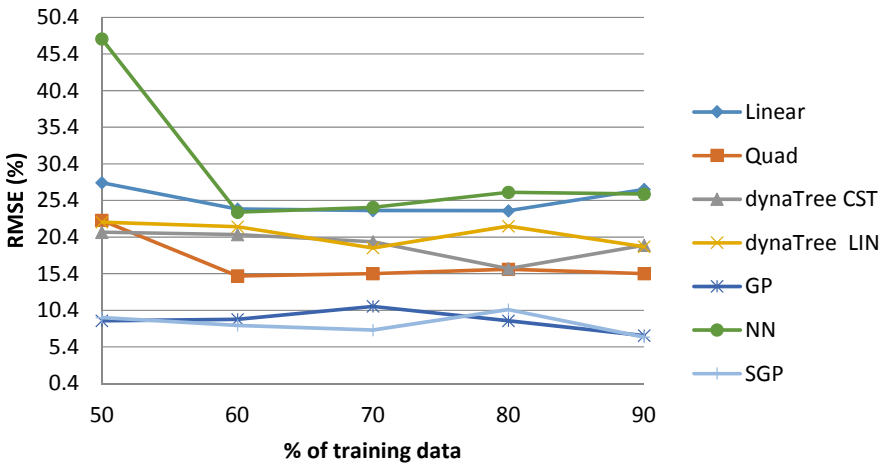


Figure 5.2: Comparison of the RMSE for Electrical Conductivity predictions.

	cond		ln(cond)		ln(T×cond)		
Technique	**NS**	**Order**	**NS**	**Order**	**NS**	**Order**	**RMSE**
Linear	0.5960	0.8419	0.7661	0.8509	0.7853	0.8604	27.8245
Quad	0.7610	0.8675	0.8981	0.8961	0.9117	0.8990	22.6727
dynaTree CST	0.7265	0.8599	0.8847	0.8399	0.8818	0.8442	21.0682
dynaTree LIN	0.6585	0.8397	0.8771	0.8741	0.8909	0.8800	22.4429
GP	N/A	N/A	N/A	N/A	0.9610	0.9193	8.9757
SGP	N/A	N/A	N/A	N/A	0.9872	0.9446	9.4010
NN	N/A	N/A	N/A	N/A	0.7897	0.8630	47.4660

Table 5.3: NS, Order and RMSE for Electrical Conductivity (50% training points). N/A signifies that no data was available for this particular type

5.3 Martensite start temperature data

The database used for this section is described in details in Section 3.1.3. Once again, as illustrated by Figure 5.3 and Table 5.4, GP gave the best predictions, maintaining an average RMSE of 5.85%. Linear interpolation performed remarkably well overall with an average RMSE of 13.6%. Quadratic regression and dynaTree LIN gave good results with a large training set, however performed very poorly with a smaller training set. There is an obvious peak in error for the NN method when 70% training data is used. As explained at the beginning of Chapter 5, outliers have been excluded from the results, considering an arbitrary cut-off value of 200% (i.e. predictions with more than 200% error are not compiled). However, for this particular series of tests, it happens that a considerable amount of results were just under that cut-off value, thus influencing the average RMSE by a large number and causing the unusual peak. For the previous two problems, properties are measured within one chemical phase, therefore, measured values depend only on chemical composition and temperature. However, the value of Ms is dependant on multiple chemical phases, and is influenced by operating factors such as the cooling rate, hence the noisy nature of the data. For this specific problem, we also added an additional measure: the RMSE on the training data at 50% training, in order to show the quality of the regression methods on noisy data. The results are presented in Table 5.5. GP presents the smallest training error with 3.56% while the worse performer is dynaTree LIN with 23.81%.

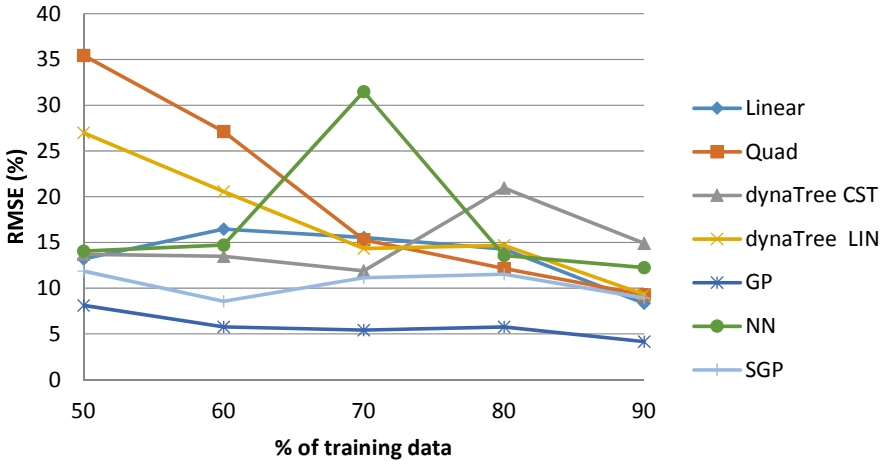


Figure 5.3: Comparison of the RMSE for Martensite start temperature predictions.

Technique	**NS**	**Order**	**RMSE**
Linear	0.8505	0.9000	13.2032
Quad	0.4988	0.8896	35.4405
dynaTree CST	0.7997	0.8517	13.7197
dynaTree LIN	0.5570	0.7917	26.9888
GP	0.8987	0.9120	8.1254
SGP	0.7949	0.8738	11.8719
NN	0.7533	0.8933	14.0708

Table 5.4: NS, Order and RMSE for Martensite start temperature (50% training points).

Technique	**Training RMSE**
Linear	14.3434
Quad	12.9457
dynaTree CST	14.4358
dynaTree LIN	26.3521
GP	3.5598
SGP	6.8557
NN	10.0290

Table 5.5: Training RMSE for Martensite start temperature (50% training points).

5.4 Computational time

We had an average time of 18.8 seconds per prediction point when running a GP regression, on a desktop computer Intel i7 3.4GHz with 16 GB of RAM. The NN was the second slowest with an average of 7.7 seconds per prediction while dynaTree LIN came in third with 4.2 seconds per prediction. The SGP technique produced an average time per prediction of 0.94 seconds. The other three techniques performed under 0.1 seconds, as shown in Table 5.6. The times include both training and prediction.

Technique	**Time (s)**
Linear	4E-6
Quad	3.5E-5
dynaTree CST	0.088
dynaTree LIN	4.22
GP	18.85
SGP	0.94
NN	7.73

Table 5.6: Overall average time per prediction in seconds.

5.5 Discussion

The main preoccupation of an engineer when attempting to model new data is the reliability of the prediction. In terms of predicting accuracy, for all three types of data the GP and SGP (Figure 5.4) are the clear winners in our evaluations. Overall, the GP has a offers slightly better prediction accuracy, but this technique is by far the slowest to run and can be impractical with very large datasets. If time is not a factor, GP seems to be the best choice. However, for on-line applications or any application where computational time is an important factor to consider, one may wish to consider using SGP, which offers a slight setback in accuracy but improves greatly the computational time.

Smooth data Within the three faster techniques, dynaTree CST gave the best performance. Nevertheless, since all models gave acceptable results, one may consider using strictly linear interpolation, as the excess (or deviation from linearity) has proven to be very low and the computational time exceptionally fast.

Nonsmooth data The quadratic interpolation model represents the best choice for this type of data within the faster techniques. With the Electrical Conductivity example, we show that using $ln(T \times cond)$ leads to better predictions. Therefore, this clearly demonstrates that a thorough knowledge of the problem is an important factor influencing the quality of predictive models.

Noisy data For this type of data, prediction accuracy clearly improves as the training set gets larger. As we can see in Figure 5.3, with a large training set (90%), all techniques give acceptable results. Consequently, if the training set is complete enough, the polynomial interpolation models seem to be an interesting choice because of their low computational cost. Some authors have suggested that Ms can be a linear function [25, 26]. However, if this was the case, the linear regression model would give the best predictions. Since the Gaussian approaches are clear winners over the linear approach, it seems apparent that Ms is a much more complex function. Here, using parameters other than the chemical composition as part of the model could improve significantly the predictions accuracy.

There is a clear magnitude difference in the general average relative error obtained by all techniques on all three sets of data. For molar volume, the average error was under 10%, while for electrical conductivity and Martensite start temperature, the average was more around 15 to 20%. This can be explained by the fact that the molar volume is easier to measure than the other two properties, thus minimizing the intrinsic error.

One can argue that the real power of machine learning techniques lies in predictions made with a minimum set of training data. In the real world, it is often the case that engineers have limited experimental points and still wish to make

predictions based on this data set. In this light, if we compare the results with only 50% of training points (Table 5.7), one should avoid neural network for nonsmooth data, and quadratic interpolation for noisy data. With an average RMSE of over 35% on prediction of Ms, Quad clearly overestimates the non-linearity of the function, while it is not the case for the two other types of data. Once again, the most reliable technique is the Gaussian process regression for two of the three cases. SGP and dynaTree CST are good alternatives to GP to reduce the computational time. The training RMSE at 50% training points (Table 5.5) is representative of the results obtained on testing points with the exception of Quad, which has a training error of 12.95% and a prediction error of 35.44%. However, as mentioned at the beginning of the present Chapter, outliers were excluded from the average RMSE, and the same treatment has been done whilst calculating the training error. While most technique produced practically no outliers on training data, 3.3% of outliers were excluded for the Quad technique. The neural network models well the training data despite giving somewhat erratic results on testing points.

	Lin.	**Quad**	**d.Tree CST**	**d.Tree LIN**	**GP**	**SGP**	**NN**
Molar volume	8.41	7.47	6.10	8.33	4.50	**3.27**	**8.61**
Electrical Conductivity	27.82	22.67	21.07	22.44	**8.98**	9.41	**47.47**
Ms	13.20	**35.44**	13.72	26.99	**8.13**	11.87	14.07

Table 5.7: Relative RMSE in percent at 50% training. On each row, lowest RMSE is represented in green and highest in red.

Figure 5.5 illustrates the total percentage of excluded data per technique. From this figure, we can conclude that a smooth data set leads to very few wild predictions, however, for nonsmooth data, human validation is required in order to make sure that these predictions are not considered. Quadratic interpolation gave very few outliers for smooth and nonsmooth data, however it ended up having almost 2% of rejected data for a noisy set of data. In general, SGP was the most reliable technique with less than 0.05% of outliers for each training set, while the neural network model was unreliable especially for nonsmooth data, giving more than 2% wild predictions, and performing erratically for noisy data. The remarkably small number of outliers for SGP is interesting and can be explained by the fact that

this technique is partitioning the data in very small clusters. Rather than approximating a Gaussian function over the entire training set, it is done in very small areas, and this reduces the error where in areas where data would be sparse over the entire space. This is a very important advantage of SGP, and it also explains while in some cases, especially at 50% training data, SGP had a smaller average RMSE than a conventional GP.

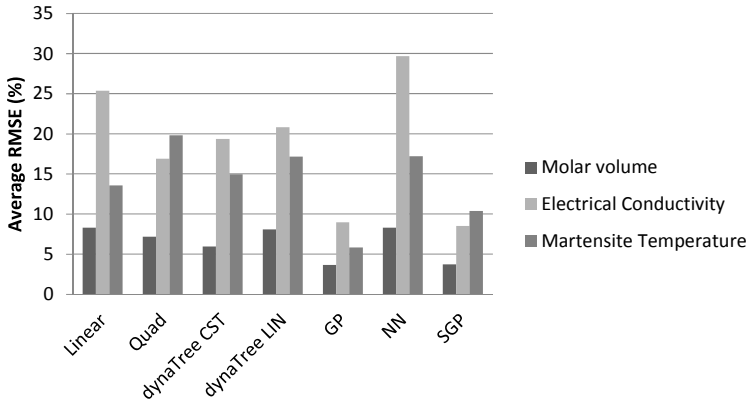


Figure 5.4: Average RMSE obtained for each set of data.

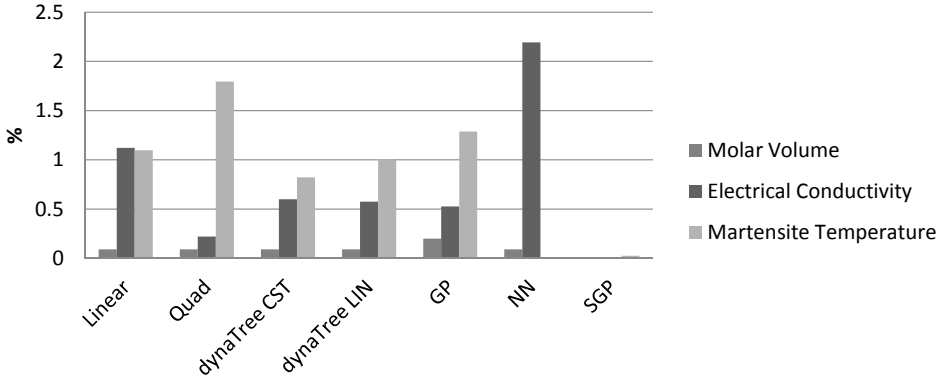


Figure 5.5: Percentage of excluded outliers (RMSE>200%) per set of data.

6. Truth discovery in Material Science databases

Refer to Section 3.1 for a complete descriptions of the databases used in our work. To conduct experiments on truth discovery, we used a reduced database of EC, refereed to as EC Red 2 in Table 3.3.

Here we consider each scientific paper from where the data points have been extracted to be independent sources of information. This database can be fed to a machine learning model in order to make predictions for new chemical compositions where the electrical conductivity is currently unknown. In Table 3.2, the last line is an example query that could be desired in the industry.

As an example taken from our dataset, Fig. 6.1 illustrates a series of conflicting pairs of data points between two sources. Here the values of EC, with input values extremely close in space and at the same temperature show large variations. Such differences are unacceptable when consulting in process design [22]. When consulting an existing database, faced to such variations in data, as it would be very confusing to decide which information is truthful and which should be discarded.

6.1 Author ranking by sources comparison

In an effort to reduce the noise in databases consisting of experimental points, we introduce a new method of truth discovery using the different sources (research papers in our case) and the amount of conflicting and similar but non-conflicting information between them to create a ranking of reliability.

In Fig. 6.2, we represent a sample of 13 of the 67 sources found in our database. One can see the amount of conflicting information over the amount of similarities.

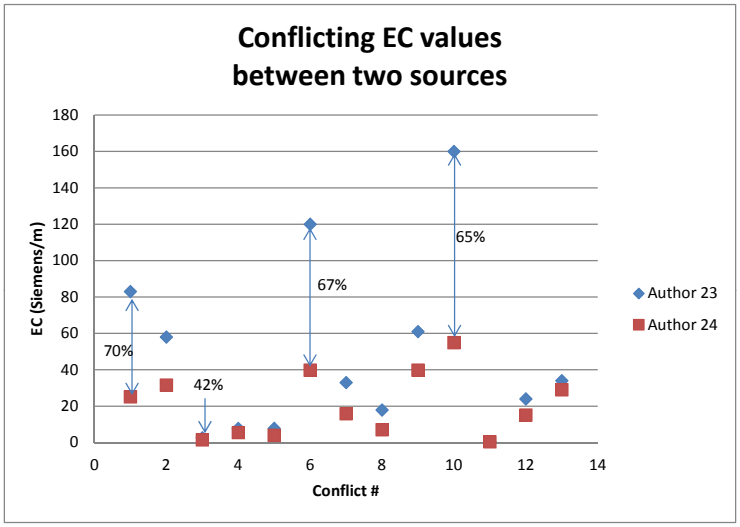


Figure 6.1: Example of conflicting information found in our EC database. Each pair of conflict is shown on the X axis and the Y axis presents the values of EC.

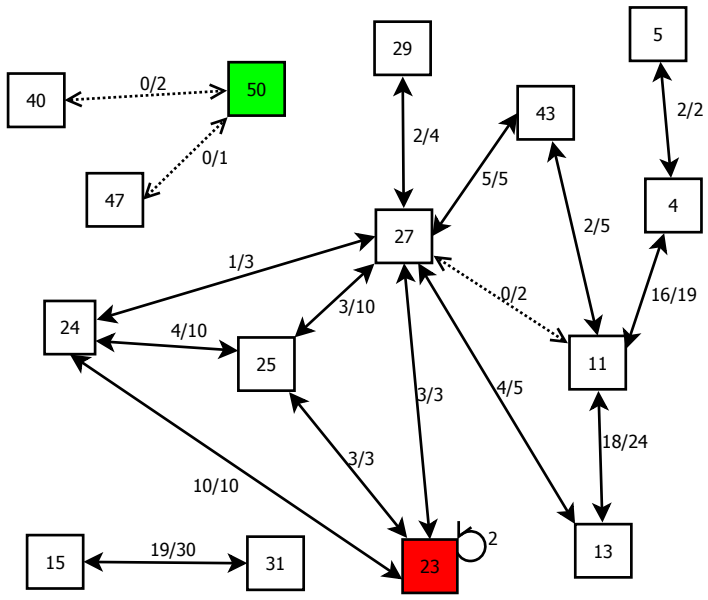


Figure 6.2: Illustration of conflicting information between some sources of the EC database. The squares represent the different sources with their arbitrary numbering, the arrows represent similarities between two sources and the numbers on each arrow represent the amount of conflicts over the amount of similar data points. Dotted arrows indicate that there are no conflict, only agreements between two sources.

In this example, we want to consider source 50 as more reliable than source 23, because 50 has 3 similar data points agreeing with two other sources, whilst 23 has 18 conflicting data points, including 2 within its own data.

Of course, what is regarded as similar information and conflicting is entirely subjective and we had to define our own rules. In this work, we measure the distance between two data points in space using a custom distance equation, following this predicate to compare two points p and q:

$$\forall i \in \mathrm{N} : \left(\frac{|q_i - p_i|}{\sum_{j=1}^{N} q_j} < \epsilon \right) \wedge \left(\frac{|q_i - p_i|}{\sum_{j=1}^{N} p_j} < \epsilon \right) \wedge ((p_i = 0) \leftrightarrow (q_i = 0))$$

Where N is the total number of columns (dimensions), excluding the predicted column (EC) and ϵ is an arbitrary similarity constraint, we used a value of 5%. This equation has been introduced Section 4.2, where it has proven to be a more accurate way of comparing materials properties databases than using a simple euclidean distance. The reason is that we want points to be close in every dimension, as potential chemical interactions between the components can cause a very big difference in the predicted value. In other words, very close points in space can have a very big difference in their value of EC, caused by a small amount of a certain chemical component.

We define a conflict between two authors as two points that are similar, ie. relatively close in space ($\epsilon < 5\%$) but having a difference of more than twice the experimental error in the output used for prediction (EC). To find a reliable value of experimental error, we computed the average discrepancy within each source. That is, for each pair of similar points within a given author, we calculate the average difference in EC. In our case study, we find this average to be of approximately 5%, therefore we considered conflicting information values to be above a 10% difference in EC. By definition, the points that are close in space but where EC is under 10% difference are considered as agreements.

In our work we consider two types of similarities: direct and indirect. For a given source, direct similarities (agreements and conflicts) are the ones that can be found from its own data in relation to other sources. Let us define the list of sources with direct similarites as S_d. An example is illustrated in Fig. 6.3, showing direct similarities for source 23 by black arrows going to sources 24, 25 and 27. Here $S_d = 24, 25, 27$. On the other hand, indirect similarities are the similarities

between our list of similar sources, S_d and other sources. The indirect similarities are illustrated by red arrows on Fig. 6.3. In this example, source 23 has indirect similarities with sources 24, 25, 27, 29, 43, 11 and 13. We consider that indirect similarities are an indication of the reliability of the similar sources. For example, source A could have a lot of conflicting information with source B, but if source B has also a lot of conflicts with a lot of other sources, this means that it may not be very reliable and therefore this information should be of less value than if B was considered very reliable.

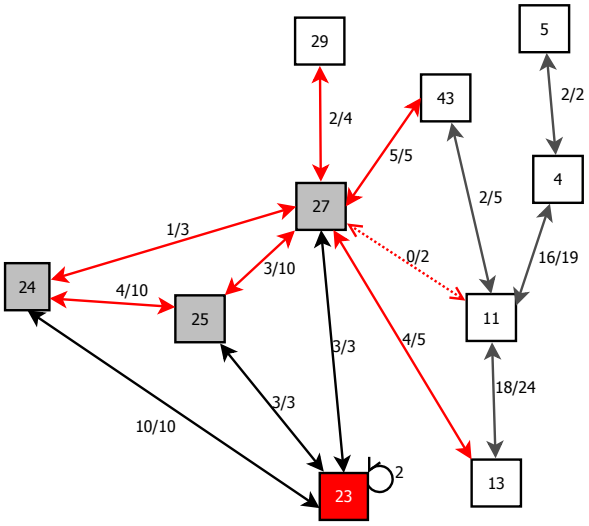


Figure 6.3: Direct similarities of source 23 are shown by black arrows and indirect similarities by red arrows. Here sources 24, 25 and 27 contribute to the indirect similarites for source 23.

Using the amount of similarities, conflicts, agreements and the total amount of data points for each author, we introduce a quality rate Q, giving an estimation of reliability for each source. For a given source, we calculate Q using the following equations:

$$Q = \frac{\alpha Q_d + \beta Q_i + \gamma Q_c}{\alpha + \beta + \gamma} \tag{6.1}$$

$$Q_d = \frac{1 - C_d - A_d}{P} \tag{6.2}$$

$$Q_i = \frac{1 - C_i - A_i}{P} \tag{6.3}$$

$$Q_c = \frac{1 - S_c - S_a}{P} \tag{6.4}$$

Where C_d is the ratio of direct conflicts, A_d is the ratio of direct agreements, C_i is the ratio of indirect conflicts, A_i is the ratio of indirect agreements, S_c is the ratio of authors with conflicting data, S_a is the ratio of sources with agreeing data and P is the number of data points for the given source. Three parameters are introduced in the formula: α, β and γ, allowing weights to be attributed to each type of conflicting information. In this work we have chosen the parameters $\alpha = 2$, $\beta = 1$ and $\gamma = 0.5$. We chose these values because we consider direct conflicts and similarities to be of the most influential on the reliability of a given source. A higher value of Q signifies a higher confidence level for a given author. Here in the case of only one direct conflicting source, for example sources 15 and 31 in Fig. 6.2, the source with the most amount of data will have a higher Q value. In Fig. 6.2, source 23 has two internal conflicts, meaning that two pairs of data points are conflicting within its own dataset. This is considered very unreliable and should have a big effect on Q. We chose to treat these as direct conflicts but it is not added to the total amount of similarities. This can mean that a given source could have a negative value of Q. The percentage of direct conflicts C_d is calculated as follows:

$$C_d = \frac{\sum \text{direct conflicts}}{\sum \text{direct similarities}} \tag{6.5}$$

Similarly, C_i, A_d and A_i are calculated as follows:

$$C_i = \frac{\sum \text{indirect conflicts}}{\sum \text{indirect similarities}} \tag{6.6}$$

$$A_d = \frac{\sum \text{direct agreements}}{\sum \text{direct similarities}} \tag{6.7}$$

$$A_i = \frac{\sum \text{indirect agreements}}{\sum \text{indirect similarities}} \tag{6.8}$$

S_c and S_a are calculated using the following formulae:

$$S_c = \frac{\text{number of conflicted sources}}{\text{number of sources with similarities}} \tag{6.9}$$

$$S_a = \frac{\text{number of agreeing sources}}{\text{number of sources with similarities}} \tag{6.10}$$

Note that the same sources can contribute to both S_c and S_a, as two sources can have conflicting and agreeing data simultaneously.

Once every source has been evaluated, we consider every pair of conflicting information and eliminate the data point where its source has a lower value of Q. This is applied recursively on the entire database until all the conflicted information has been eliminated. Table 6.1 shows an example of two conflicting data points and the values of Q for each source. After the process, the remaining database is pruned and reduced, eliminating noisy information in an attempt to get better predictions.

SiO2	**Al2O3**	**MgO**	**CaO**	**MnO**	**PbO**	**FeO**	**Fe2O3**	**T(K)**	**EC**	**Source**	**Q**
63.40	0	0	36.60	0	0	0	0	1873	16.10	45	0.004
61.38	0	0	38.62	0	0	0	0	1873	20.50	48	0.120

Table 6.1: Example of conflicting datapoints. Here source 48 would be chosen over 45 and the first data point would be elminated from the database.

In order to test the prediction power of SGP, some artificial noise has been generated and introduced in the database. Section 6.2 presents the results of the predictions with various amount of introduced noisy data. In order to keep it realistic, the noisy data had to be close to the existing data points, but have possible conflicting values of EC. Therefore, these points have been produced by taking each existing source and creating a slightly modified version of each data point (randomly +/-5%) but with a possibly conflicting value of EC (randomly +/- 50%). We then choose a random subset of all the generated noisy data and we introduce them in our database prior to testing. It is important to note that from this method, a random portion of the introduced points will not be conflicting information.

In Chapters 4 and 5, we evaluated SGP on prediction of electrical conductivity, using $ln(T * \sigma)$, where T is the temperature and σ is the value of electrical conductivity in Siemens per meter. We showed that this approach provides a significant

improvement on predictions. However, for the truth discovery problem, we are using the non-logarithmic values of electrical conductivity in order to show the full range of errors.

6.2 Results and discussion

First, we test predictions with SGP, using a 10-fold cross-validation technique on the non-filtered database. In a 10-fold cross-validation, the entire database is split in 10 equal subsets. Each subset is then used as a testing set where the model is trained with the remaining 9 subsets. We repeat this procedure 10 times. The average error in percent and root mean square error are then computed over all the tests and this is what we are presenting in this section. Table 6.2 shows the influence of introduced noise on the predictions performed by SGP. Then, we test the same databases when applying our noise reduction technique introduced in Section 6.1. The results are presented as a graph of the RMSE in Fig. 6.4 and in Table 6.2.

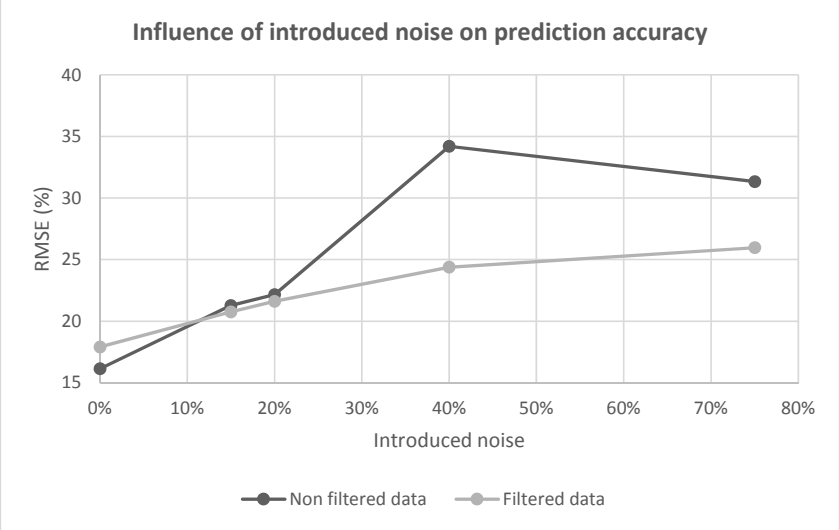


Figure 6.4: Graph showing the influence of the amount of introduced noise on SGP prediction accuracy.

From these results, one can see that SGP is extremely efficient at excluding a low to medium amount of noisy information. When applying our noise reduction technique on the original database, only 7% of data points were removed, and it

	No filtering		**Filtered database**		
Database	**Error (%)**	**RMSE**	**Filtered points (%)**	**Error(%)**	**RMSE**
Original database	14.96	16.13	7	15.11	17.91
15% introduced noise	18.04	21.27	11	18.20	20.76
20% introduced noise	18.65	22.15	12	18.81	21.61
40% introduced noise	28.14	34.19	15	20.86	**24.38**
75% introduced noise	26.51	31.33	15	21.78	**25.96**

Table 6.2: Influence of introduced noise on SGP predictions

explains the fact that there is no improvement on the original predictions. In order to test the robustness of SGP, up to 75% of noisy data points were introduced in the database. When introducing up to 20% of noisy data, the predictions remain acceptable with a RMSE of around 20%. This result is an example of the remarkable robustness of the SGP technique. The biggest effect can be seen around 40% of noise, where the RMSE jumps to 34%. Beyond this amount, as it can be expected, the predictions are actually getting slightly better. This is because the noisy data is overtaking the actual real data and the SGP is actually over fitting. Nevertheless, our noise reduction technique is showing impressive results by keeping the error below the 22% mark. In Table 6.2, the pruned points column shows the amount of conflicting information that has been eliminated using our noise reduction technique. Here we can note that all the introduced noise is not completely removed during the filtering, and this is perfectly normal as some of the random noise can actually be non conflicting information. However, by removing the conflicts, we can improve the predictions by an impressive 10% on the RMSE for the case of 40% introduced noise, which is where we see the most effect on the SGP prediction accuracy.

Even if there is no a major improvement in the prediction accuracy under the bar of 20% introduced semi-noisy data, it is important to note that our filtering technique still managed to remove from 7% to 12% of conflicting information, meaning that the reliability of the database is improved when consulting existing data. Since the introduced data is random, it is not unreasonable to assume that

one quarter to half of this introduced data could be potential agreeing information and thus should not be removed. Therefore, by identifying and removing 12% of conflicting information in the case of 20% introduced noise, one can assume that all or almost all conflicts have been identified and resolved. This is an important point to consider as other modelling techniques would possibly not handle this amount of noise as well as SGP. As a matter of a fact, we tested our approach using a Nearest Neighbour Interpolation model, as implemented in the XonGrid Excel Add-in [12]. The predictions on the original database using this model were mediocre, with an average error of around 45%. However, when using the training database, the average error went down to approximately 20%. We can conclude that for this type of model, conflicting information in the training database has a high influence on the quality of the predictions.

7. Conclusions

7.1 Summary and Conclusions

In Chapter 1, we formally introduce the problem by detailing the motivations behind our work. Because of the cost and time involved in performing experiments, engineers rely on machine learning models to make predictions on materials properties. The databases used for training to make these predictions are collected and assembled from the literature. In conjunction with this data, it may be the case that live data is fed from the plant into the databases, making on-line learning a necessity for the chosen model. Furthermore, optimisation of materials properties requires a large number of predictions to be performed and thus the computational time is critical. From the literature, we know that a Gaussian regression model gives good predictions for the properties we are working with, however, this technique is very costly in terms of computational time. Therefore, we chose to develop a novel method to make on-line learning with Gaussian process regression, called scalable Gaussian process (SGP). Since we are dealing with databases consisting of experimental points, errors can be present and contracting data points included in these databases. In order to improve the prediction accuracy of machine learning methods, as well as to make the databases more reliable when consulting existing data, we also develop a novel truth discovery technique, using the amount of conflicting and agreeing information between the different sources present in each database.

Chapter 2 presents a comprehensive literature review in the domains of prediction of materials properties, Gaussian process regression for machine learning using a large amount of data, clustering of high-dimensional data and truth discovery.

For this research, we have access to five databases for three materials properties: molar volume, electrical conductivity and Martensite start temperature.

Three databases on electrical conductivity are employed in our experiments, one large one and two reduced versions. The databases as well as the detailed composition ranges are described in Section 3.1.

In Section 3.2, we provide a brief description of the machine learning interpolation techniques we chose to compare with our novel SGP as well as evaluate on our datasets. These techniques are: Gaussian process regression, linear interpolation, quadratic interpolation, neural network and dynamic trees.

Our novel method, SGP, is described in details in Chapter 4. We propose a scalable approach to make predictions of materials properties using a Gaussian process regression machine learning model. This approach improves the computational time of a traditional GP, by creating clusters of similar information as input queries, and then using a subset of the entire training database by choosing only the information close to a given query cluster. The calculations are therefore performed by small clusters, or batches, and the reduced training database is also compressed to remove similar information, improving drastically the overall computational time. As it can be expected, our experiments showed that the size of the training matrix influences the calculation time exponentially. While it is clear that a very small training set would lead to poor prediction and that a large set would necessarily produce more accurate predictions, our results with Ms and Electrical Conductivity predictions show that there is no general correlation between the size of the training matrix and the predicted error when using training matrices between $10^{2.7}$ and $10^{4.7}$. We believe that the variation in prediction error is related to the quality of the data in the training matrix. In other words, closely related data in the training set will lead to better prediction. Also, our datasets consist of experimental values, there is a high chance of human error in entire sets of points that could lead to variations in the results. In summary, our SGP has proven to be fast while maintaining a good prediction error. Results on prediction of Martensite Start Temperature as well as Electrical Conductivity demonstrate that the proposed Scalable GP outperforms the other existing methods significantly in terms of efficiency and scalability. Experiments on gas sensor data also prove that our approach can be used successfully not only on material science data, but also for a wider variety of applications, proving its versatility.

In Chapter 5, we present a comprehensive comparison of the machine learning techniques introduced in Chapter 3.2, as well as comparing with our new SGP. This research shows that a material engineer wishing to make predictions on specific sets of data must study the nature of the data in order to make an informed decision. Assisted by computer scientists, one can make the best choice to achieve the most accurate predictions whilst minimizing the computational time. This chapter demonstrate that knowing the behaviour of electrical conductivity data led to more accurate results by using a logarithmic value. Overall, within the tested techniques, the standard Gaussian process regression gives the best prediction accuracy, but is by far the slowest technique. For applications where computational time is an important factor, such as real-time applications, we recommend using a modified version of GPs such as the SGP, proposed in the current manuscript. The constant model of dynaTree could also be a good alternative. This work demonstrates how computer science can be coupled with material engineering, in order to improve material and alloy design [32].

Chapter 6 presents a new truth discovery technique to filter scientific databases consisting of experimental points. When two data points are in conflict, we use the amount of direct and indirect conflicts and agreements in order to make a decision as to which point should be eliminated. We test our approach by making predictions using our introduced SGP, presenting the results in terms of prediction error before and after pruning the database. The results presented in this chapter prove that the SGP interpolation technique is very robust when the ratio of noise, or conflicting data is relatively low. However, predictions start to deteriorate when more and more noisy data is involved. The proposed approach provides an improvement of predictions by 10%. The new produced database can also be considered more reliable when consulting existing information, automatising the conflict resolution process.

In conclusion, this work is not intended to contribute to any new significant findings in the area of material science. However, the methods that were developed support material scientists in their research by providing a low-cost but yet effective and fast alternative to conducting expensive experiments on materials properties.

Our SGP technique allows fast and accurate predictions and can be used in the context of chemical systems optimisations, where on-line learning is essential, while our truth discovery method helps detecting potential errors in published data thus improving the prediction accuracy of machine learning models when using databases consisting of experimental points.

7.2 Future Work

In future work, we will investigate the prediction of other physical properties, and we are planning to integrate the SGP into FactOptimal [22, 23, 24], the optimisation module of FactSage, which is a software system that is created for treating thermodynamic properties and calculations in process metallurgy [3]. We propose to utilize the thermochemistry knowledge and the machine learning approach to achieve more efficient and accurate predictions, which could be used in practical chemical industry. We also plan to improve the web application of SGP [6] by including our truth discovery technique as a preliminary step to to prune the training database. Another improvement to SGP would include the implementation of an automatic analysis of the training data in order to pinpoint areas where data is missing, according to the desired predictions. This test would inform the user that in order to achieve more accurate predictions on the desired points, more experimental values should be collected around a specific region and would be very useful in real life.

Another interesting future development would be to utilise our truth discovery technique on different types of data. We believe that this approach can be very versatile and could be used with any database containing redundant or similar information from different sources. One would need to define a new similarity measure (i.e. new rules for determining what are considered conflicts or agreements), according to the specific problem, but the main idea and the quality rate Q, as explained in Section 6.1 would remain the same.

8. Nomenclature

δ Kronecker delta

μ Mean

κ Electrical Conductivity

σ Variance

D Number of dimensions

GP Gaussian process

Ms Martensite start temperature

MV Molar volume

n Number of training (experimental) points

N_G Gaussian Noise

NS Nash-Sutcliffe model efficiency

Q Quality rate

RMSE Root mean square error

T Temperature

w Width of a Gaussian kernel

Bibliography

[1] C. Bailer-Jones, H. Bhadeshia, and D. MacKay. Gaussian process modelling of austenite formation in steel. *Materials Science and Technology*, 15(3), 1999.

[2] C. Bailer-Jones, H. Bhadeshia, and D. MacKay. Gaussian process modelling of austenite formation in steel. *Materials Science and Technology*, 15(3), 1999.

[3] C. Bale, E. Bélisle, P. Chartrand, S. Decterov, G. Eriksson, K. Hack, I.-H. Jung, Y.-B. Kang, J. Melançon, A. Pelton, C. Robelin, and S. Petersen. Factsage thermochemical software and databases - recent developments. *Calphad*, 33(2):295 – 311, 2009. Tools for Computational Thermodynamics.

[4] C. W. Bale, P. Chartrand, S. A. Degterov, G. Eriksson, K. Hack, R. B. Mahfoud, J. Melançon, A. D. Pelton, and S. Petersen. FactSage thermochemical software and databases. *Calphad-computer Coupling of Phase Diagrams and Thermochemistry*, 26:189–228, 2002.

[5] A. Banerjee, D. B. Dunson, and S. T. Tokdar. Efficient gaussian process regression for large datasets. *Biometrika*, 100(1):75–89, 2013.

[6] E. Belisle. Scalable gaussian process, 2014.

[7] E. Bélisle, Z. Huang, and A. Gheribi. Scalable gaussian process regression for prediction of material properties. In H. Wang and M. Sharaf, editors, *Databases Theory and Applications*, volume 8506 of *Lecture Notes in Computer Science*, pages 38–49. Springer International Publishing, 2014.

[8] E. Bélisle, Z. Huang, and A. Gheribi. Scalable gaussian process regression for prediction of material properties. In *Databases Theory and Applications*, pages 38–49. Springer, 2014.

[9] E. Bélisle, Z. Huang, and A. Gheribi. Truth discovery in material science databases. In H. Wang and M. Sharaf, editors, *Proceedings of the 26th edition of the Australiasian Database Conference (Accepted in March 2105)*. Springer International Publishing, 2015.

[10] E. Bélisle, Z. Huang, and A. Gheribi. Truth discovery in material science databases. In H. Wang and M. Sharaf, editors, *Databases Theory and Applications*, Lecture Notes in Computer Science. Springer International Publishing, 2015.

[11] E. Bélisle, Z. Huang, S. Le Digabel, and A. E. Gheribi. Evaluation of machine learning interpolation techniques for prediction of physical properties. *Computational Materials Science*, 98:170–177, 2015.

[12] B. D. D. BESSES. Xongrid interpolation add-in, 2015.

[13] B. Birol, G. Polat, and M. Saridede. Estimation model for electrical conductivity of molten caf2-al2o3-cao slags based on optical basicity. *JOM*, pages 1–9, 2014.

[14] C. Capdevila, F. Caballero, and C. García de Andrés. Analysis of effect of alloying elements on martensite start temperature of steels. *Materials science and technology*, 19(5):581–586, 2003.

[15] D. Cockcroft, K. Murdock, and J. Mink. Determination of histamine pc20. comparison of linear and logarithmic interpolation. *CHEST Journal*, 84(4):505–506, 1983.

[16] T.-S. Dai, J.-Y. Wang, and H.-S. Wei. An ingenious, piecewise linear interpolation algorithm for pricing arithmetic average options. In M.-Y. Kao and X.-Y. Li, editors, *Algorithmic Aspects in Information and Management*, volume 4508 of *Lecture Notes in Computer Science*, pages 262–272. Springer Berlin Heidelberg, 2007.

[17] O. Dekel and O. Shamir. Vox populi: Collecting high-quality labels from a crowd. 2009.

[18] N. Dodgson. Quadratic interpolation for image resampling. *Image Processing, IEEE Transactions on*, 6(9):1322–1326, 1997.

[19] X. L. Dong, L. Berti-Equille, and D. Srivastava. Truth discovery and copying detection in a dynamic world. *Proceedings of the VLDB Endowment*, 2(1):562–573, 2009.

[20] X. L. Dong, B. Saha, and D. Srivastava. Less is more: Selecting sources wisely for integration. In *Proceedings of the VLDB Endowment*, volume 6, pages 37–48. VLDB Endowment, 2012.

[21] A. Garcia-Junceda, C. Capdevila, F. Caballero, and C. G. de Andres. Dependence of martensite start temperature on fine austenite grain size. *Scripta Materialia*, 58(2):134 – 137, 2008.

[22] A. Gheribi, C. Audet, S. L. Digabel, E. Bélisle, C. Bale, and A. Pelton. Calculating optimal conditions for alloy and process design using thermodynamic and property databases, the factsage software and the mesh adaptive direct search algorithm. *Calphad*, 36(0):135 – 143, 2012.

[23] A. E. Gheribi, S. L. Digabel, C. Audet, and P. Chartrand. Identifying optimal conditions for magnesium based alloy design using the mesh adaptive direct search algorithm. *Thermochimica Acta*, 559(0):107 – 110, 2013.

[24] A. E. Gheribi, C. Robelin, S. L. Digabel, C. Audet, and A. D. Pelton. Calculating all local minima on liquidus surfaces using the factsage software and databases and the mesh adaptive direct search algorithm. *The Journal of Chemical Thermodynamics*, 43(9):1323 – 1330, 2011.

[25] G. Ghosh and G. Olson. Kinetics of f.c.c. b.c.c. heterogeneous martensitic nucleationi. the critical driving force for athermal nucleation. *Acta Metallurgica et Materialia*, 42(10):3361 – 3370, 1994.

[26] G. Ghosh and G. Olson. Kinetics of f.c.c. b.c.c. heterogeneous martensitic nucleationii. thermal activation. *Acta Metallurgica et Materialia*, 42(10):3371 – 3379, 1994.

[27] S. Ghosh and Y. Rudy. Accuracy of quadratic versus linear interpolation in noninvasive electrocardiographic imaging (ecgi). *Annals of Biomedical Engineering*, 33(9):1187–1201, 2005.

[28] M. N. Gibbs and D. J. C. MacKay. Efficient implementation of gaussian processes. Submitted to Statistics and Computing.

[29] R. Gramacy and M. Taddy. dynaTree: An R package implementing dynamic trees for learning and design. Software available at `http://CRAN.R-project.org/package=dynaTree`, 2010.

[30] R. B. Gramacy and H. K. Lee. Gaussian processes and limiting linear models. *Computational Statistics & Data Analysis*, 53(1):123 – 136, 2008.

[31] R. B. Gramacy and H. K. H. Lee. Bayesian treed gaussian process models with an application to computer modeling. *Journal of the American Statistical Association*, 103(483):1119–1130, 2008.

[32] J.-P. Harvey and A. Gheribi. Process simulation and control optimization of a blast furnace using classical thermodynamics combined to a direct search algorithm. *Metallurgical and Materials Transactions B*, 45(1):307–327, 2014.

[33] J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data, 2013. Appears in Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence (UAI2013).

[34] Z. Huang, H. Shen, J. Liu, and X. Zhou. Effective data co-reduction for multimedia similarity search. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, SIGMOD '11, pages 1021–1032, New York, NY, USA, 2011. ACM.

[35] J. C. Hull and A. D. White. Efficient procedures for valuing european and american path-dependent options. *The Journal of Derivatives*, 1(1):21–31, 1993.

[36] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[37] S. Le Digabel. Algorithm 909: NOMAD: Nonlinear Optimization with the MADS Algorithm. *ACM Trans. Math. Softw.*, 37(4), Feb. 2011.

[38] S.-J. Lee and K.-S. Park. Prediction of martensite start temperature in alloy steels with different grain sizes. *Metallurgical and Materials Transactions A*, 44(8):3423–3427, 2013.

[39] H. Malvar, L.-W. He, and R. Cutler. High-quality linear interpolation for demosaicing of bayer-patterned color images. In *Acoustics, Speech, and Signal*

Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on, volume 3, pages iii–485–8 vol.3, 2004.

[40] A. McCallum, K. Nigam, and L. H. Ungar. Efficient clustering of high-dimensional data sets with application to reference matching. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '00, pages 169–178, New York, NY, USA, 2000. ACM.

[41] E. Meijering. A chronology of interpolation: from ancient astronomy to modern signal and image processing. *Proceedings of the IEEE*, 90(3):319–342, 2002.

[42] T. D. Mining. Tiberius data mining predictive modelling software, 2014.

[43] Y. Mualem and S. P. Friedman. Theoretical prediction of electrical conductivity in saturated and unsaturated soil. *Water Resources Research*, 27(10):2771–2777, 1991.

[44] J. Nash and J. Sutcliffe. River flow forecasting through conceptual models part i a discussion of principles. *Journal of Hydrology*, 10(3):282 – 290, 1970.

[45] R. Neal. *Bayesian Learning for Neural Networks*. Lecture Notes in Statistics. Springer New York, 1996.

[46] D. Nguyen-Tuong, M. Seeger, and J. Peters. Model learning with local gaussian process regression. *Advanced Robotics*, 23(15):2015–2034, 2009.

[47] A. O'Hagan and J. F. C. Kingman. Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society. Series B (Methodological)*, 40(1):pp. 1–42, 1978.

[48] P. Payson and C. Savage. Martensite reactions in alloy steels. *Transactions ASM*, 33:261–275, 1944.

[49] C. E. Rasmussen. Gaussian processes for machine learning. MIT Press, 2006.

[50] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[51] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[52] P. Richet. Viscosity and configurational entropy of silicate melts. *Geochimica et Cosmochimica Acta*, 48(3):471 – 483, 1984.

[53] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(1):23–38, 1998.

[54] D. Rummelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323(9):533–535, 1986.

[55] G. Schaller, D. Schaerer, G. Meynet, and A. Maeder. New grids of stellar models from 0.8 to 120 solar masses at z= 0.020 and z= 0.001. *Astronomy and Astrophysics Supplement Series*, 96:269–331, 1992.

[56] B. Schölkopf and A. J. Smola. *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[57] V. S. Sheng, F. Provost, and P. G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622. ACM, 2008.

[58] J. Shi, R. Murray-Smith, D. Titterington, and B. Pearlmutter. Filtered gaussian processes for learning with large data-sets. In R. Murray-Smith and R. Shorten, editors, *Switching and Learning in Feedback Systems*, volume 3355 of *Lecture Notes in Computer Science*, pages 128–139. Springer Berlin Heidelberg, 2005.

[59] A. Skinner and J. Broughton. Neural networks in computational materials science: Training algorithms. *Modelling and Simulation in Materials Science and Engineering*, 3(3):371, 1995.

[60] A. J. Skinner and J. Q. Broughton. Neural networks in computational materials science: training algorithms. *Modelling and Simulation in Materials Science and Engineering*, 3(3):371, 1995.

[61] M. Sloński. Bayesian neural networks and gaussian processes in identification of concrete properties. *Computer Assisted Mechanics and Engineering Sciences*, Vol. 18, nr 4:291–302, 2011.

[62] E. Snelson. Local and global sparse gaussian process approximations. In *Proceedings of Artificial Intelligence and Statistics (AISTATS)*, 2007.

[63] T. Sourmail. Predicting the martensite start temperature (ms) of steels, 2014.

[64] T. Sourmail and C. Garcia-Mateo. Critical assessment of models for predicting the ms temperature of steels. *Computational Materials Science*, 34(4):323 – 334, 2005.

[65] T. Sourmail and C. Garcia-Mateo. A model for predicting the ms temperatures of steels. *Computational Materials Science*, 34(2):213 – 218, 2005.

[66] D. F. Specht. A general regression neural network. *Neural Networks, IEEE Transactions on*, 2(6):568–576, 1991.

[67] A. Stormvinter, A. Borgenstam, and J. Ågren. Thermodynamically based prediction of the martensite start temperature for commercial steels. *Metallurgical and Materials Transactions. A*, 43A(10):3870–3879, 2012. QC 20121029.

[68] Y. Stry, M. Hainke, and T. Jung. Comparison of linear and quadratic shape functions for a hybrid control-volume finite element method. *International Journal of Numerical Methods for Heat and Fluid Flow*, 12:1009 – 1031, 2002.

[69] M. A. Taddy, R. B. Gramacy, and N. G. Polson. Dynamic trees for learning and design. *Journal of the American Statistical Association*, 106(493):109–123, 2011.

[70] H. Tsuboi, A. Chutia, C. Lv, Z. Zhu, H. Onuma, R. Miura, A. Suzuki, R. Sahnoun, M. Koyama, N. Hatakeyama, A. Endou, H. Takaba, C. A. D. Carpio, R. C. Deka, M. Kubo, and A. Miyamoto. An electrical conductivity prediction simulator based on tb-qcmd and kmc. system development and applications. *Journal of Molecular Structure: {THEOCHEM}*, 903(1-3):11 – 22, 2009. Recent advances in the theoretical understanding of catalysis.

[71] J. V. Tu. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology*, 49(11):1225 – 1231, 1996.

[72] R. Urtasun and T. Darrell. T.: Sparse probabilistic regression for activity-independent human pose inference. In *In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[73] R. Wagner. Multi-linear interpolation, 2013.

[74] D. Wang, L. Kaplan, H. Le, and T. Abdelzaher. On truth discovery in social sensing: A maximum likelihood estimation approach. In *Proceedings of the 11th international conference on Information Processing in Sensor Networks*, pages 233–244. ACM, 2012.

[75] R. Weber, H.-J. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In A. Gupta, O. Shmueli, and J. Widom, editors, *VLDB'98, Proceedings of 24rd International Conference on Very Large Data Bases, August 24-27, 1998, New York City, New York, USA*, pages 194–205. Morgan Kaufmann, 1998.

[76] A. d. W. W.G. Vermeulen, P.F. Morris and S. van der Zwagg. Prediction of martensite start temperature using artificial neural network. *Ironmaking and Steelmaking*, 23(5), 1996.

[77] C. K. Williams and C. E. Rasmussen. Gaussian processes for regression. 1996.

[78] D. Wolpert and W. Macready. No free lunch theorems for optimization. *Evolutionary Computation, IEEE Transactions on*, 1(1):67–82, 1997.

[79] X. Yin, J. Han, and P. Yu. Truth discovery with multiple conflicting information providers on the web. *Knowledge and Data Engineering, IEEE Transactions on*, 20(6):796–808, June 2008.

[80] X. Yin and W. Tan. Semi-supervised truth discovery. In *Proceedings of the 20th international conference on World wide web*, pages 217–226. ACM, 2011.

[81] Z. Zhao, J. Cheng, and W. Ng. Truth discovery in data streams: A single-pass probabilistic approach. In *Proceedings of the 23rd ACM International*

Conference on Conference on Information and Knowledge Management, pages 1589–1598. ACM, 2014.