République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique
Université Hadj Lakhdar Batna
Faculté de Technologie

Thèse
Présentée au
Département d'Electronique
Pour l'obtention du diplôme de
Doctorat en Sciences en Electronique

Par
Fouzi DOUAK

Thème:

High-dimensional data analysis using artificial intelligence methods

Devant le jury:

Dr. Ahmed Louchene	Professeur	U. de Batna	Président
Dr. Nabil Benoudjit	Professeur	U. de Batna	Rapporteur
Dr. Farid Melgani	Professeur	U. de Trento (Italie)	Co-rapporteur
Dr. Khaled Melkemi	Professeur	U. de Biskra	Examinateur
Dr. Noureddine Goléa	Professeur	U. de Oum El Bouaghi	Examinateur
Dr. Rédha Benzid	Maître de Conférences (A)	U. de Batna	Examinateur

(2013)

Cette thèse a été réalisée dans le cadre du programme Erasmus Mundus Averroès financé par la Commission Européenne.


بسم الله الرحمن الرحيم

Abstract

In this thesis, new methodologies of machine learning for regression problems are proposed and applied in two main practical fields, which are the chemometrics, and Renewable energies fields. In particular, in the last few years, spectroscopy has represented an important technology for product analysis and quality control in different chemical fields. For example, it has been applied successfully in pharmaceutical, food and textile industries.

Another very interesting field is related to renewable energies. Renewable energies have been of great interest in recent years, as a consequence of increasing population and higher consumption of energy by developing countries, oil resources, natural gas and uranium will be depleted within a few decades. The unavoidable alternative becomes thus the development of renewable energy sources like solar energy, geothermal and wind power. In fact, the best use of renewable energies is an essential factor of development for all countries.

Machine learning methods exhibit the attractive advantage that they can provide very accurate predictors. In this thesis, we concentrate our study on two different methodologies:

In the first one, we propose a two-stage regression approach, which is based on the residual based correction concept (RBC) applied in chemometrics field, in order to improve the accuracy with respect to the single regressor. A comparative study with another approach which exploits differently estimation errors, namely adaptive boosting for regression (AdaBoost.R), is also included. The idea in our proposed method is to correct any adopted regressor, called functional estimator, by analyzing and modeling its residual errors directly in the feature space. RBC is therefore not a regressor but a correction method, whose aim is not to reach the best achievable accuracy for a given data set but to possibly improve the estimation model of a given regressor.

In the second one, the regression process is undertaken by assuming that the training set is composed of a sufficient number of samples in order to obtain reliable and accurate estimations. However, from a practical point of view, the process of collecting training samples is not trivial, because the concentration/wind speed measurements associated with the spectral data have to be performed manually by human experts and thus are subject to errors and costs in terms of time and money. For this reason, the number of available training samples is typically limited and performances can be consequently affected due to data scarcity. A solution to this problem is given by active learning in this thesis. Active learning represents an interesting approach proposed in the literature to address the problem of training sample collection, in which training samples are selected in an iterative way in order to minimize the number of involved samples and the intervention of human users.

The experimental results on different real data sets show the effectiveness of the proposed solutions.

Keywords: Spectrometric data analysis, Renewable energy, Wind speed prediction, Residual-based correction (RBC), Boosting, Active learning, Regression.

ملخص

في هذه الأطروحة، نقترح منهجيات جديدة للتعلم الآلي من أجل المشاكل المتعلقة بالتنبؤ وتطبيقها في مجالين رئيسيين، و هما التحليل الطيفي، و الطاقات المتجددة . على وجه الخصوص، في السنوات القليلة الماضية يمثل التحليل الطيفي تقنية مهمة لتحليل المنتجات و مراقبة الجودة في مجالات كيميائية مختلفة .على سبيل المثال، تم تطبيقه بنجاح في مجال المستحضرات الصيدلانية، المواد الغذائية وصناعة النسيج.

هناك مجال آخر مثير جدا للاهتمام يتعلق بالطاقات المتجددة، التي أصبحت ذات أهمية كبيرة في السنوات الأخيرة، نتيجة للنمو السكاني وزيادة استهلاك الطاقة من قبل البلدان النامية لموارد النفط والغاز الطبيعي واليورانيوم، التي ستستنفذ في غضون بضعة عقود و بالتالي فان تطوير مصادر الطاقات المتجددة سيصبح بديلا لا مفر منه، مثل الطاقة الشمسية، الطاقة الحرارية الجوفية وطاقة الرياح . في الواقع، أفضل استخدام للطاقات المتجددة هو عامل أساسي للتنمية بالنسبة لجميع البلدان.

أساليب التعلم الآلي تحمل ميزة جذابة التي بإمكانها تقديم تنبؤ دقيق جدا .في هذه الأطروحة، نركز دراستنا على منهجين مختلفين :

أولا، نقترح إتباع طريقة التنبؤ في مرحلتين، والتي تقوم على مفهوم التصحيح على أساس الخطأ المتبقي (ت.أ.م) ، المطبقة في مجال التحليل الطيفي ، من أجل تحسين الدقة المتعلقة بالتنبؤ الفردي .دراسة مقارنة مع تقنية أخرى التي تستغل بشكل مختلف تنبأت الأخطاء المتبقية ، المسماة بتدعيم التكيف من أجل التنبؤ، هي أيضا مدرجة في هذه المطروحة .الفكرة المقترحة هي تصحيح أي طريقة معتمدة للتنبؤ، المسماة التنبؤ الوظيفي، من خلال تحليل ووضع نماذج لها أخطاء متبقية مباشرة في الفضاء المميز .ولذلك هذه التقنية ليست أداة تنبؤ ولكن هي طريقة للتصحيح، هدفها ليس للتوصل إلى أفضل دقة ممكنة لمجموعة بيانات معينة ولكن ربما لتحسين نموذج تنبؤ معين.

ثانيا، يتم تنفيذ عملية التنبؤ بافتراض أن مجموعة التعلم الآلي تتكون من عدد كاف من العينات للحصول على تقديرات موثوقة ودقيقة . مع ذلك، من الناحية العملية، فإن عملية جمع عينات التعلم الآلي لها أهمية كبيرة، لأن قياس التركيز / سرعة الرياح يجب أن تأخذ يدويا من قبل الخبراء وبالتالي فهي خاضعة إلى الأخطاء والتكاليف من حيث الوقت والمال .لهذا السبب، فإن عينات التعلم الآلي المتاحة عادة تكون محدودة ويمكن أن تؤثر على أداء التنبؤ نظرا لقلة البيانات المستخدمة. التعلم التلقائي المقترح يمثل مجال مثير للاهتمام في مختلف المنشورات لمعالجة مشكلة التعلم الآلي بإضافة مجموعة من العينات، والتي يتم اختيارها بطرق متكررة من أجل تقليل عدد العينات المعنية والتدخل من الإنسان المستخدم. النتائج التجريبية على مجموعات مختلفة من بيانات حقيقية تظهر فعالية الحلول المقترحة.

الكلمات المفتاحية: تحليل البيانات الطيفي، الطاقة المتجددة، التنبؤ بسرعة الرياح، التصحيح على أساس الخطأ المتبقي ، تدعيم، التعلم التلقائي، التنبؤ.

Résumé

Dans cette thèse, de nouvelles méthodes d'apprentissage pour des problèmes de régression sont proposées et appliquées dans deux grands domaines á savoir la chimiométrie et l'énergie renouvelable. En particulier, au cours des dernières années, la spectroscopie a représenté une technologie importante pour l'analyse des produits et le contrôle de qualité dans les différents domaines de chimie. Par exemple, il a été appliqué avec succès dans les industries pharmaceutique, alimentaire et textile.

L'énergie renouvelable, a fait l'objet d'un grand intérêt au cours des dernières années, par conséquence de la croissance démographique et l'augmentation de la consommation d'énergie par les pays en voie de développement, les ressources en pétrole, gaz naturel et uranium seront épuisées d'ici quelques décennies. La solution inévitable devient ainsi le développement de sources d'énergie renouvelables comme l'énergie solaire, géothermique et éolienne. En fait, la meilleure utilisation des énergies renouvelables est un facteur essentiel du développement pour tous les pays.

Les méthodes d'apprentissage présentent l'avantage attrayant qu'elles puissent fournir des prédicteurs très précis. Dans cette thèse, nous nous concentrons notre étude sur deux différentes méthodologies :

Premièrement, nous proposons une approche de régression en deux étapes, qui est basé sur le concept de la correction de l'erreur résiduelle (RBC) appliquée dans le domaine de la chimiométrie, afin d'améliorer la précision par rapport à un régresseur unique. Une étude comparative avec une autre approche qui exploite différemment les erreurs d'estimation, à savoir le Boosting (AdaBoost.R), est également étudiée. L'idée de la méthode proposée est de corriger tout régresseur, appelé estimateur fonctionnel, par l'analyse et la modélisation de ses erreurs résiduelles directement dans l'espace des caractéristiques. RBC n'est donc pas un régresseur, mais une méthode de correction, dont le but n'est pas d'atteindre la meilleure précision possible pour un ensemble de données, mais pour éventuellement améliorer le modèle d'éstimation pour un régresseur donnée.

Deuxièmement, le procédé de régression est effectué en supposant que l'ensemble d'apprentissage est constitué d'un nombre suffisant d'échantillons afin d'obtenir des estimations fiables et précises. Cependant, d'un point de vue pratique, le processus de collecte d'échantillons d'apprentissage n'est pas trivial, car les mesures de concentration / vitesse de vent associées aux données doivent être effectuées manuellement par des experts humains et sont donc sujets à des erreurs et les coûts en termes de temps et d'argent. Pour cette raison, le nombre d'échantillons d'apprentissage disponibles est généralement limité et les performances peuvent en être affectées en raison de la rareté des données. Une solution à ce problème est donnée par l'apprentissage actif dans cette thèse. L'apprentissage actif représente une approche intéressante proposée dans la littérature pour résoudre le problème de la collecte des échantillons d'apprentissage, dans

lequel les échantillons d'apprentissage sont choisis de façon itérative afin de minimiser le nombre d'échantillons concernés et l'intervention des experts humains.
Les résultats expérimentaux sur différentes jeux de données réelles montrent l'efficacité des solutions proposées.

Mots-clés: Analyse des données de spectrométrie, Énergie renouvelable, Prédiction de la vitesse du vent, Correction basée sur résiduelle (RBC), Boosting, Apprentissage actif, Régression.

Acknowledgements

First of all, I thank Allah, the Most High, for the opportunity He gave me to study, to research and to write this dissertation. Thanks Allah, my outmost thanks, for giving me the ability, the strength, attitude and motivation through this research and to complete this work.
I would like to express my most sincere gratitude to my advisor Prof. Nabil BENOUDJIT and to my co-advisor Prof. Farid MELGANI, for suggesting this area of research and for their solid technical guidance, continuous support and encouragement in my work. A special thanks is addressed to Prof. Farid MELGANI for his permanent availability and responsiveness to my requests and for giving his time and effort without reservation. I would also like to express my appreciation for their guidance during my study. Without which this research would never have reached this point.
I am indebted beyond measure to Prof. Nabil BENOUDJIT for giving me his inestimable advices and valuable guidance during different cycles of my studies M.Sc and Dr.Sc. I have really learned a lot from his comments and suggestions.
I am also grateful to Prof. Khaled MELKEMI, Dr. Redha BENZID, Prof. Noureddine GOLÉA and Dr. Noureddine GHOGGALI who accepted to be my jury members, and devoted their precious time to review my thesis. I also want to thank Prof. Ahmed LOUCHENE for having agreed to preside this jury.
I would like to thank the Averroès Erasmus Mundus program funded by the European Commission about 18 months in university of Trento. I wish to thank the members of the DISI - Dipartimento di Ingegneria e Scienza dell'Informazione, and more particularly to Dr. Edoardo PASOLLI, who have shared their knowledge and experiences with me. I extend additional thanks to Dr. Noureddine GHOGGALI. He has been a continual source of stimulation, help and very useful comments on all my work and most important a good friend. I would particularly like to thank Dr. Redha BENZID and Mr. Toufik BENTRCIA for their encouragement, help and support.
I would like to thank all my friends, colleagues and the staff at the Department of Electronics, University of Batna for their help along the realization of this work.
Finally, I am grateful to my parents, my brothers and my sister for their encouragement and continuous support.

Fouzi Douak

The work compiled in this thesis has been partially supported by the Averroès Erasmus Mundus program funded by the European Commission (unfolding during the period of February 2011 to July 2012).

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction and Dissertation Overview

Contents

1.1 Context and motivations

In order to face the regression problem from a methodological viewpoint, several approaches to parameter estimation have been proposed. In this context, both linear and nonlinear regression methods have been proposed [1–4]. In the literature, one can find three very well-established approaches to envisage a regression task: 1) the supervised approach; 2) the unsupervised approach; and 3) the semi-supervised approach.

The supervised approach algorithm try to learn the input-output relationship $f(x)$ by using a training data set $X = [x_i, y_i], i = 1, ..., n,\ x \in \Re^d$ where d is the feature space dimensionality and the labels y are discrete ($y \in \{1, ..., T\}$, T number of considered classes) for classification problems and real ($y \in \Re$, continuous) value for regression tasks. The supervised learning problem is divided into two types, namely, classification (pattern recognition) and the regression (function approximation). In the regression problem, the task is to find the mapping between input $x \in \Re^d$ and output y. In this context, the learning task in regression is to find the underlying function between some d-dimensional input vectors x_i and scalar outputs y_i. There are two phases when applying supervised learning algorithms for problem-solving as shown in Figure 1.1. The first phase is so-called training phase where the learning algorithms design a mathematical model of a dependency, function or mapping in a regression or classification based on the training data given. While the second one is the test (application) phase, the parameters selecting (models developed) by the supervised approach are used to predict the outputs y_i of the data which are unknown by the learning algorithms in the learning phase [5, 6].

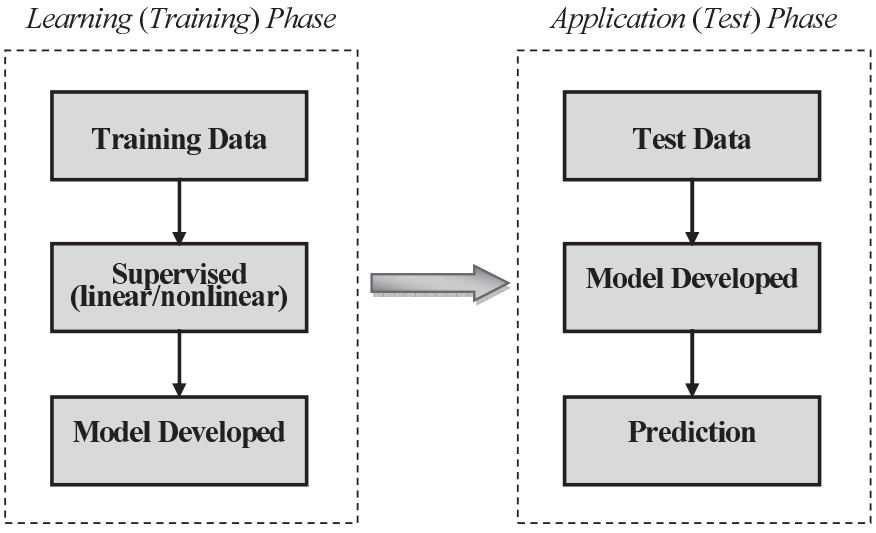


Figure 1.1: Two Phases of Supervised Learning Algorithms.

In the unsupervised approach, there are only raw data $x_i \in \Re^d$, no labels data y_i are available. Several algorithms to face this issue have been proposed, such as clustering techniques, principal

component analysis (PCA), and independent component analysis (ICA) [7–9].
Another method so-called semi-supervised learning can be considered as an attractive solution. Their underlying idea is to exploit unlabeled samples that are readily available at zero cost from the data sets under analysis, during the design of the regression model to compensate the deficit in labeled samples [10–13]. The cause of an appearance of the unlabeled data points is usually an expensive, difficult and slow process of obtaining labeled data. Thus, labeling brings additional costs and often it is not feasible. As a result, the goal of a semi-supervised learning algorithm is to predict the labels of the unlabeled data by taking the entire data set into account [5].
This thesis focuses firstly on the application of linear and nonlinear regression methods in the field of food industry (near infrared spectroscopy) and secondly on the wind prediction (renewable energy) in different regions of Algeria.

In the last few years, the field of spectroscopy is continuously working on means to improve, optimize and gain a better understanding of the way food productions are run. The increased focus on food quality creates a big challenge for the industry into development and control of food productions. Ensuring the quality of food products needs monitoring and evaluation of every step from the raw material, to the production, to the final product, and in the distribution [14]. A general system of information-extraction from near infrared spectroscopy can be described by the block diagram shown in Figure 1.2.

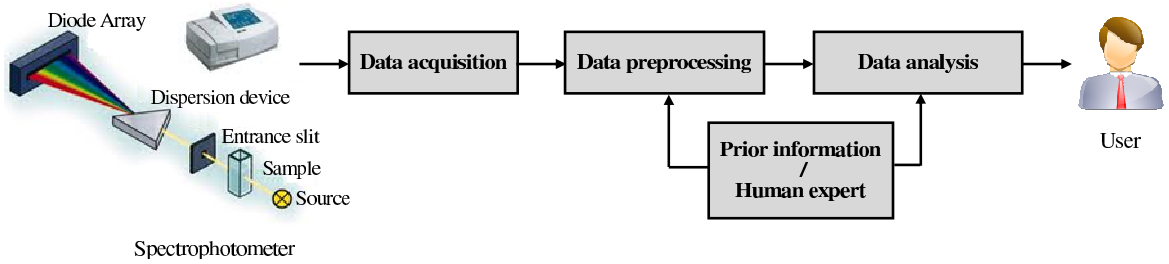


Figure 1.2: Flow chart of a general system of information extraction from Near infrared spectroscopy.

In the data-acquisition phase, modelling in laboratory where all measurements of variables must be carried out and where parameters of the model (linear or non-linear) must also be estimated. According on the reason of the application, the user may or may not be concerned in the spectral contributions originating from the physical characteristics of the sample. Where these characteristics are significant, the user may choose to work with the raw spectral data, otherwise some preprocessing is usually carried out [15]. Several spectral preprocessing methods exist, two categories of the most widely used methods in data preprocessing are scatter correction methods and spectral derivatives (smoothing methods) [16].

In particular, automatic or semi-interactive calibration is a critical process monitoring and control in real time as they provide the data from which relevant process and product information and conclusions should be extracted. Near Infrared spectroscopy is rich in information which is representative of both the physical and chemical characteristics of the sample, these characteristics have to be applied to take into account, for instance, pH, pressure, temperature, treatment that can removes or isolates the measurement sample in real time, etc. Recently, automatic approaches to the analysis of spectroscopy data have attracted the attention of several researchers. This is a direct result of both the growing interest of end-users in the capabilities of spectroscopy and the scientific community's realization that the automation of traditional "manual" analysis techniques (i.e., based on the intervention of human experts) offers great advantages in terms of time and economic cost. In this context, one of the main objectives of this dissertation is to provide a modest contribution to the automatic or semi-interactive techniques. Finally, the resulting preprocessed data can be fed to the analysis stage, which aims at extracting from the data set of spectroscopy information (product) being of interest to a given end user. In analytical chemistry, several linear calibration methods are applied to solve quantitative problems with the argument that the relation between the chemical composition and the measured signal is linear [17]. However, there are many others where nonlinearity is present. In [18] discusses important sources of nonlinearity in near-infrared spectroscopy, namely :1) deviations from the Beer-Lambert law, which are typical of highly absorbing samples; 2) nonlinear detector responses; 3) drifts in the light source; 4) interactions between analytes; 5) nonlinearity between diffuse reflectance/transmittance data and chemical data. When the nonlinearity is significant, one can use truly nonlinear calibration techniques, for example, support vector machine (SVM).

The exploitation of renewable energy and especially of wind power is receiving increased attention the last years under the influence of novel guidelines adopted for energy management, and the concerns for global warming and climate change. In this framework the accurate estimation of the wind speed for short or long forecasting horizons is of primarily significance but not always easy to be attained due to the variable nature and the complexity of the environmental conditions that are implicated. In the literature, several wind speed prediction methods can be found. They can be divided into three categories: statistical methods [19–22], physical methods [19, 23], and machine learning methods including neural networks and support vector machines since the estimation of the wind speed can be designed as a nonlinear regression problem [24–36].

1.2 Problems and solutions

The focus of this thesis is the development of models and algorithms for prediction. In this work, we concentrate our study on two well known issues of the machine learning community:

how to improve the prediction accuracy of a single regressor and how to deal with the issue of a limited labeled training samples.
Concerning the first issue, quality control of production systems and authenticity testing of products are increasing in importance in food industry, since they represent the new required issues to compete in the present-day market. Both production systems and food products can be described as complex systems, where several factors can interact and play a fundamental rule: consequently all these factors should be monitored and their synergic effects controlled.
There is also a need in the food industry to rationalise and improve quality and process controls. The modern production systems require fast and automatic on-line monitoring, which should be able to extract the maximum amount of available information, in order to assure the optimal system functioning. On the other hand, food products acquire a higher value when their authenticity is protected, controlled and assured: in fact, consumers are more oriented towards purchasing food products of a certified origin. Consequently, during recent years there has been increasing interest in the origin authentication of food products, since authenticity can be often associated with food quality [37].
The use of traditional analytical techniques does not always match with these constraints, because they can be time-consuming and expensive, while fast and cheap methods are essential, in order to assure a continuous monitoring. As a results novel analytical techniques have been used for these issues, since they enable more rapid and non-invasive characterisation of foods: nuclear magnetic resonance (NMR), near infrared spectroscopy (NIR), electronic sensors and image analysis are only a few of the involved analytical methods. A common property of these techniques is also the production of a large amount of spectra, so that several data sets are usually obtained and must be interpreted. Summarising, quality and authenticity control faces with complex systems, described by a large amount of data: hence, specific tools should be used in order to assure an effective prediction. Regression and classification can provide these specific tools: in the last years regression proved to be able to handle a huge amount of data, to process them, and to give useful results that can be explained by the users [38].

- In this context, in general all the three approaches (supervised, unsupervised and semi-supervised) use only a single regressor in order to estimate the prediction. However, using a single regressor usually is not capable of providing high accuracies over the entire input space. This is due to the fact that any estimator provides usually an accuracy depending on the region of the input space to which the analyzed pattern belongs to. In this dissertation a new method based on a residual-based correction (RBC) concept applied in chemometrics field. Its underlying idea is to correct any adopted regressor, called functional estimator, by analyzing and modeling its residual errors directly in the feature space. RBC is therefore not a regressor but a correction method, whose aim is not to reach the best achievable accuracy for a given data set but to possibly improve the estimation model of a given (poor or accurate) regressor. For this reason, we have proposed a regression approach that

consists in correcting a given estimator by exploiting its systematic errors in the feature space [39].

The second interesting field in this dissertation is renewable energy, as a consequence of increasing population and higher consumption of energy by developing countries, oil resources, natural gas and uranium will be depleted within a few decades. The unavoidable alternative becomes thus the development of renewable energy sources like solar energy, geothermal and wind power. In fact, the best use of renewable energies is an essential factor of development for all countries. Algeria is a country rich in renewable energy resources, thanks to its geographical location and its large area which offer it great opportunities to find renewable energy sources (e.g., solar, wind, and geothermal) [40–42]. Focusing to the particular case of wind power, the estimation of wind speed in the short or the long-term represents an important target to evaluate the possibility to create new wind turbines or to predict the wind power production of existing ones [43]. Wind energy is seen as a green power technology for having less impacts on the environment. Wind energy plants generate no air pollutants or greenhouse gases. At the end of 2009, worldwide wind powered generators capacity was 159.2GW. All wind turbines installed worldwide are generating 340 TWh/year, which is about 2 of worldwide electricity usage. Understanding the site-specific nature of wind is a critical phase in planning a wind energy project and detailed knowledge of wind on-site is needed to estimate the utility of a wind energy project [44].
The main reasons for invested and developed of renewable energies in Algeria are: 1) they constitute a solution economically viable to provide energy services to the rural isolated populations in particular in the Great South areas, where the demand consists essentially in satisfying basic energy requirements (light, refrigeration, television and radio), 2) they allow a sustainable development because of their inexhaustible character and of their limited impact on the environment (protection of the environment) and contribute to the safeguarding of our fossil resources, 3) the monetisation of these energy resources can have only positive repercussions as regards of regional balance and creation of jobs [45].

- In the aforementioned works, the regression process is undertaken by assuming that the training set is composed of a sufficient number of samples in order to obtain reliable and accurate estimations. However, from a practical point of view, the process of collecting training samples is not trivial, because the concentration/wind speed measurements have to be performed manually by human experts and thus are subject to errors and costs in terms of time and money. For this reason, the number of available training samples is typically limited and performances can be consequently affected due to data scarcity. A solution to this problem is given by semi-supervised approaches, in which the unlabeled samples are exploited during the design of the regression model in order to compensate for the deficit in labeled samples. By unlabeled samples, we mean samples whose spectral values are known,

but for which the corresponding concentration/wind speed values are unknown. In the data classification context, another solution to the problem of training sample collection is given by the active learning approach [46]. Starting from a small training set, additional samples are selected from a large amount of unlabeled data. These samples are labeled by the expert and added to the training set. The process is iterated until a stop criterion is reached. Active learning strategies have been applied successfully in different fields in classification [46–50]. Similarly, the active learning approach has been studied for regression problems by the machine learning and statistics communities, in which it is also known as optimal experimental design. After the seminal paper by Cohn et al. [51], in which active learning has been applied to two statistically-based learning architectures, such as mixtures of Gaussian and locally weighted regression, several works have appeared in the last few years. For instance, in [52], the authors focus on the problem of local minima in active learning for neural networks, and two probabilistic solutions are proposed. In [53], after introducing the fundamental limits in a minimax sense of active and passive learning for various function classes, some strategies based on a tree-structured partition of the data are presented. In [54], considering linear regression scenarios, a method using the weighted least squares learning based on the conditional expectation of the generalization error is proposed. In [55], the authors apply the query by committee approach in the regression context. The main idea is to train a committee of learners and query the labels of the samples where the committee's predictions differ, thus minimizing the variance of the learner by training on samples where the variance is largest. In [56], solving the problems of active learning and model selection at the same time is suggested in order to improve further the generalization performance. In [57], a solution to the problem of pool-based active learning in linear regression is proposed. In [58], the authors develop a strategy for kernel-based linear regression, in which the proposed greedy algorithm employs a minimum-entropy criterion, derived using a Bayesian interpretation of RR. Note that while semi-supervised methods integrate unlabeled samples automatically (without the intervention of human experts) in the learning process, active learning works differently. Indeed, its aim is to minimize the number of unlabeled samples to be labeled by human experts. For such a purpose, it resorts to smart strategies for selecting the most significant unlabeled samples, that is, those which, if labeled, would most improve the classification/regression model. For this reason, the second objective of this dissertation is to propose new methodologies of active learning in different application fields. In particular, two main fields have been considered, namely chemometrics [59,60], and wind speed prediction [61]. Our motivation and contribution in this dissertation is that the active learning has not yet been explored for these two fields of interest.

1.3 Organization of the thesis

This thesis is organized into six chapters. In Chapter 2, we present the background information about the utilized data sets, in order to better understand the work present in this dissertation. Specifically, we use two different field data sets: 1) near infrared spectroscopy data sets, and 2) wind speed data sets. In Chapter 3, we present briefly the different methods of linear and non-linear regression that we use in this dissertation. In Chapter 4, we propose a two-stage regression approach, which is based on the residual correction concept. Its underlying idea is to correct any given regressor by analyzing and modeling its residual errors in the input space. We report and discuss results of experiments conducted on two different data sets in infrared spectroscopy and designed in such a way to test the proposed approach by: 1) varying the kind of adopted regression method used to approximate the chemical parameter of interest. Partial least squares regression (PLSR), support vector machine (SVM) and radial basis function neural network (RBF) methods are considered; 2) adopting or not a feature selection strategy to reduce the dimension of the space where to perform the regression task. A comparative study with another approach which exploits differently estimation errors, namely adaptive boosting for regression (AdaBoost.R), is also included. In Chapter 5, we introduce an active learning approach for the estimation of chemical concentrations from spectroscopic data. Its main objective is to opportunely collect training samples in such a way as to minimize the error of the regression process while minimizing the number of training samples used, and thus to reduce the costs related to training sample collection. In particular, we propose two different active learning strategies, developed for regression approaches, based on partial least squares regression (PLSR), ridge regression (RR), kernel ridge regression (KRR) and support vector regression (SVR). The first strategy is based on adding samples that are distant from the current training samples in the feature space, while the second one uses a pool of regressors in order to select the samples with the greatest disagreements among the different regressors of the pool. For SVR, a specific strategy based on the selection of the samples distant from the support vectors is proposed. Similarly, in the active learning approach is used for regression problems in the renewable energy field. In particular, we consider the problem of the estimation of wind speed in Algeria. In this case, the proposed strategies are specifically developed for kernel ridge regression (KRR). In particular, we propose three different active learning strategies. The first strategy uses a pool of regressors, while the second one relies on the idea to add samples that are distant from the available training samples, and the last strategy is based on the selection of samples which exhibit a high expected prediction error. Finally, general conclusions on the methodological and experimental developments conveyed by the present dissertation are drawn in Chapter 6.

The main contributions of the thesis are:

1. with respect to design of the regressors [39]:

    - A new correction method for spectroscopy data set using linear and nonlinear regression is proposed.

2. with respect to the problem of training sample collection [59–62]:

    - In order to use and to improve forecasting/concentration systems, the quality of the predictions has to be evaluated, where "quality" and "quantity" refer to a judgement of how good or bad the prediction. In the case of data sets such as wind speed or spectroscopy, the easiest way to get an idea of the quality and quantity sample collection of this data sets is by using the active learning.
    - We investigate and develop different techniques and methods for the active learning of spectroscopy and wind speed data sets. In particular, we address several issues associated to different regression tasks of data sets (supervised and unsupervised strategies).

Chapter 2

Data Sets Description

Contents

2.1 Introduction

In this chapter, we give an overview of some data sets used in the following chapters. The goal here is to give all the necessary information to understand the work presented in this thesis. In particular, two main fields have been considered, namely chemometrics, and renewable energy. The first field is the chemometrics data, and more particularly the spectral data, which is a high-dimensional data. Spectra are obtained from the analysis of the absorbance or reflectance of the light at different wavelengths on physical or chemical products. The objective of spectral analysis it to estimated a chemical parameters from product. The second field data set is considered as a low-dimensional data. The renewable energy (wind speed data) depends on meteorological variables such as relative humidity, temperature, etc. We use the data collected from ten stations located in Algeria. In each station, different physical input parameters, such as temperature measurements and average relative humidity, which in turn provides in output an estimate of the mean wind speed (m/s). We are interested in long term wind prediction in Algeria.
Depending on the applied fields, some other information can be needed in order to develop a model.
The organization of this chapter is as follow. Section 2.2 describe the near infrared spectroscopy data sets, while in Section 2.3, we present the wind speed data sets. Finally, we finish by a conclusion.

2.2 Near infrared spectroscopy

Spectroscopy is an important technology for product analysis and quality control in different chemical fields. For example, it has been applied successfully in the pharmaceutical [63, 64], food [65] and textile industries [66]. Chemical analysis by spectroscopy relies on the fast acquisition of a large number of spectral data, which can be analyzed in order to yield accurate estimations of the concentration of the chemical component of interest in a given product.
Chemometrics can be defined as the chemical discipline that uses the methods and theory developed in mathematical, statistical, and computer sciences to design or select optimal measurement procedures and experiments, and to provide maximum relevant chemical information by analyzing chemical data [16, 67, 68].
In analytical chemistry infrared spectroscopy (IR) is mainly used for the analysis of organic components. The qualitative assessment of organic components is performed for the identification of unknown compounds, or for the determination of the chemical structure of the components. In addition, IR analysis may be used for quantification of the components. IR spectroscopy is also known as vibration spectroscopy, since the spectra arise from transitions between the vibrational energy levels of a covalent bond of a molecule. The infrared spectrum, which ranges from 1 μm to 1000 μm, is part of the electromagnetic spectrum and is surrounded by the visible and

microwave regions (Figure 2.1). The IR region may be further subdivided in the near infrared, the Mid infrared and the Far infrared regions [69]. The NIR spectroscopy region is between 700-2500 nm (14300-4000 cm^{-1}). Mid infrared and Far infrared light can be situated in the 2500-10,000 nm and 10-1000 μm range, respectively.

The instruments that measure electromagnetic radiation have several concepts and components in common. Shared instrumental components are discussed in some detail in a later section. Photometric instruments measure light intensity without consideration of wavelength. Most instruments today use filters (photometers), prisms, or gratings (spectrometers) to select (isolate) a narrow range of the incident wavelength. Radiant energy that passes through an object will be partially reflected, absorbed, and transmitted. Electromagnetic radiation is described as photons of energy traveling in waves. The relationship between wavelength and energy E is described by Planck's formula [70]:

$$E = hv, \tag{2.1}$$

where h is a constant (6.62×10^{-27} erg sec), known as Planck's constant, and v is frequency. Because the frequency of a wave is inversely proportional to the wavelength, it follows that the energy of electromagnetic radiation is inversely proportional to wavelength. Figure 2.2 shows this relationship. Electromagnetic radiation includes a spectrum of energy from short-wavelength, highly energetic gamma rays and X-rays on the left in Figure 2.1 to long-wavelength radio frequencies on the right. Visible light falls in between, with the color violet at 400-nm and red at 700-nm wavelengths being the approximate limits of the visible spectrum [70].

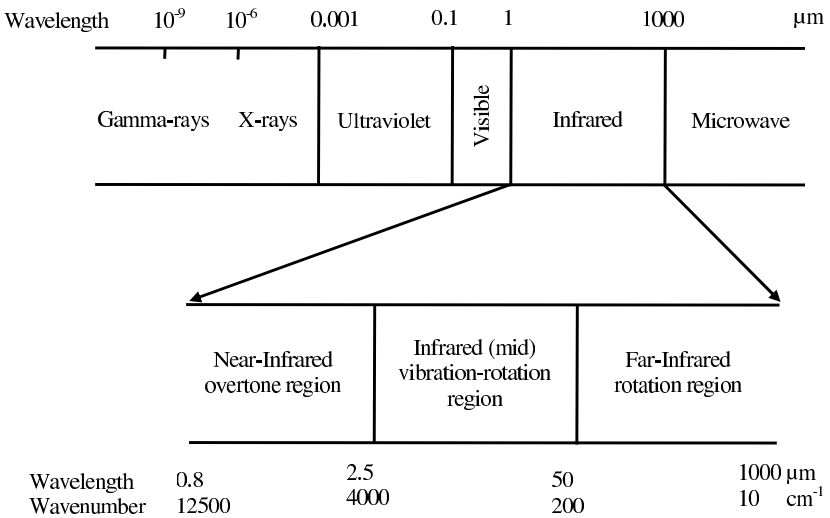


Figure 2.1: Regions of the electromagnetic spectrum.

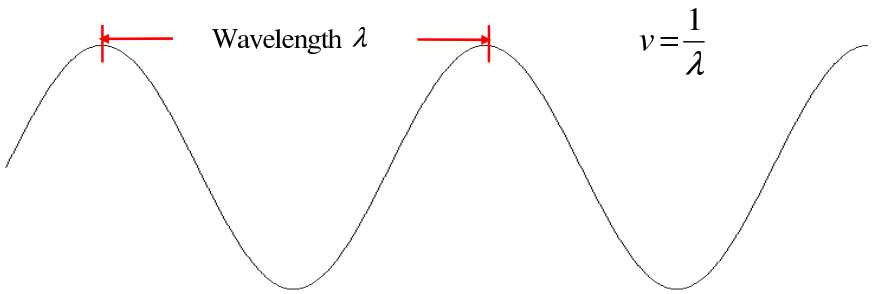


Figure 2.2: Electromagnetic Spectrum.

2.2.1 Relating absorbance to concentration

Spectroscopy deals with the interaction of electromagnetic radiation with matter. Depending on the energy of the electromagnetic radiation, different oscillations are excited. This excitation involves absorption of the corresponding energy of the oscillation involved. For example, microwave radiation excites rotational motions of the molecules, infrared radiation excites vibrational modes, near-infrared radiation excites overtones and combination frequencies, ultraviolet and visible radiation excite electronic transitions [71]. Spectroscopic methods used within the food industry include ultraviolet and visual spectroscopy, fluorescence spectroscopy, nuclear magnetic resonance, microwave absorption, ultrasound transmission, and infrared techniques such as IR and NIR, and Raman spectroscopy covering most regions of the electromagnetic spectrum (see Figure 2.1). In the present thesis the spectroscopic technique have been applied is Near infraRed spectroscopy (NIR) (see Chapter 4 and Chapter 5). The relationship between absorbance and concentration is given by the Lambert-Beer's Law:

$$A = Mcd, \tag{2.2}$$

where A is the absorbance, M is the molar absorptivity, c is the concentration, and d is the sample path length.
In a more practical sense, the absorbance is defined as the negative logarithm of the transmittance. This is given mathematically as:

$$A = -\log T = -\log \frac{I}{I_0}, \tag{2.3}$$

where I is the intensity of the light beam on the sample, I_0 is the intensity of the light beam after having passed the sample and T is transmittance. In infrared transmittance the relationship between the absorbance and the concentration $\log \frac{1}{T}$ is described by the Beer-Lambert's law, i.e. this relation is linear [72]. However, that law should not be applied to near-infrared diffuse reflectance $\log \frac{1}{R}$, because of the light scattering [73] and the light path length shortening [74], as shown in Figure 2.3.

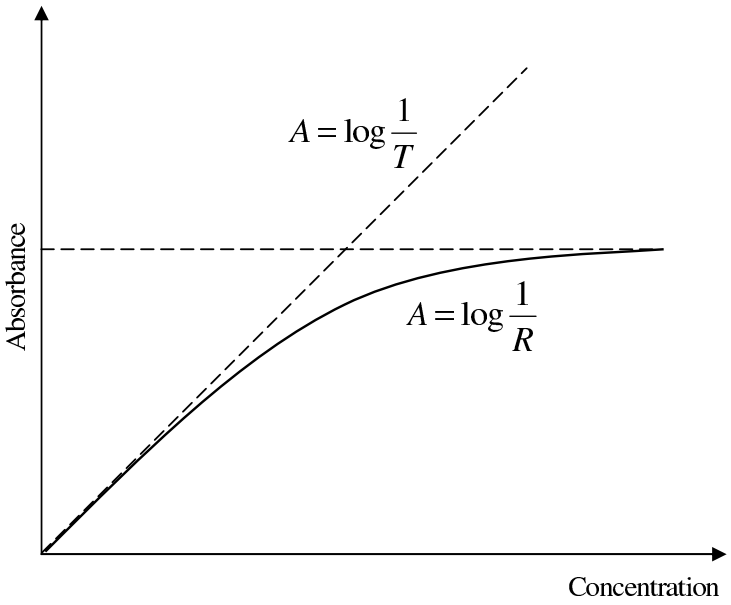


Figure 2.3: Absorbance according concentration

2.2.2 High-dimensional data

In the past, high dimensionality used to mean four or five attributes. Nowadays, we have to deal with, comparatively, super high-dimensional data, described by thousands of attributes. Data are nowadays complex and high-dimensional. In scene analysis, face recognition, document categorization, speech recognition, optical character recognition, spectral and hyperspectral analysis, and many others, one has to deal with data that are described by many attributes, and consequently described with high-dimensional. Each data element is thus viewed as a point in a vector space whose dimension is the number of numerical features necessary to describe each data element [75, 76]. Feature selection is a step of major importance for the high-dimensional data sets. Indeed, learning with high-dimensional data is generally a complicated task due to many undesirable facts denoted by the term *curse of dimensionality*. Among the various approaches to feature selection, principal component analysis (PCA) is very popular method that is able to adapt to this problem of dimensionality were developed.
In case the number of variables p is larger than the number of observations n, if X contains more variables than observations ($p >> n$) its covariance structure can be estimated by means of a PCA. In general, PCA constructs a new set of $k << p$ variables, called loadings, which are linear combinations of the original variables and which contain most of the information. These loading vectors span a k-dimensional subspace. Projecting the observations onto this subspace yields the scores t_i which for all $i = 1, ..., n$ satisfy [77]. In following are three data sets illustrative of the context and needs of high-dimensional data analysis (This section gives additional information about the near-infrared spectroscopy data sets used in the Chapter 4 and Chapter 5).

2.2.2.1 Orange juice

The first data set deals with the problem of determining sugar (saccharose) concentration in orange juice samples by near-infrared reflectance spectroscopy [78]. In this case, training (for model learning and selection) and test (for model assessment) sets contain respectively 149 and 67 samples, with 700 spectral variables that are the absorbance (log 1/R) at 700 wavelengths between 1100 and 2500 nm (where R is the light reflectance on the sample surface). The saccharose concentration ranges from 0 to 95.2% by weight. Figures 2.4-(a) and 2.4-(b) shows the spectra of orange juice used in the training and test sets, respectively. Figure 2.5 shows all the spectra of the orange juice data set. Both spectra 130 and 194 are considered as outliers in figure 2.5, where in Figure 2.6 gives a typical example of a score plot for two first principal components (Pc1-Pc2), after the application of the PCA on the 218 orange juice spectra. In Figure 2.6, two dense regions and few outliers can be seen, and we can consider that samples 130 and 194 are outliers, and can consequently be eliminated from the orange juice data set [79].

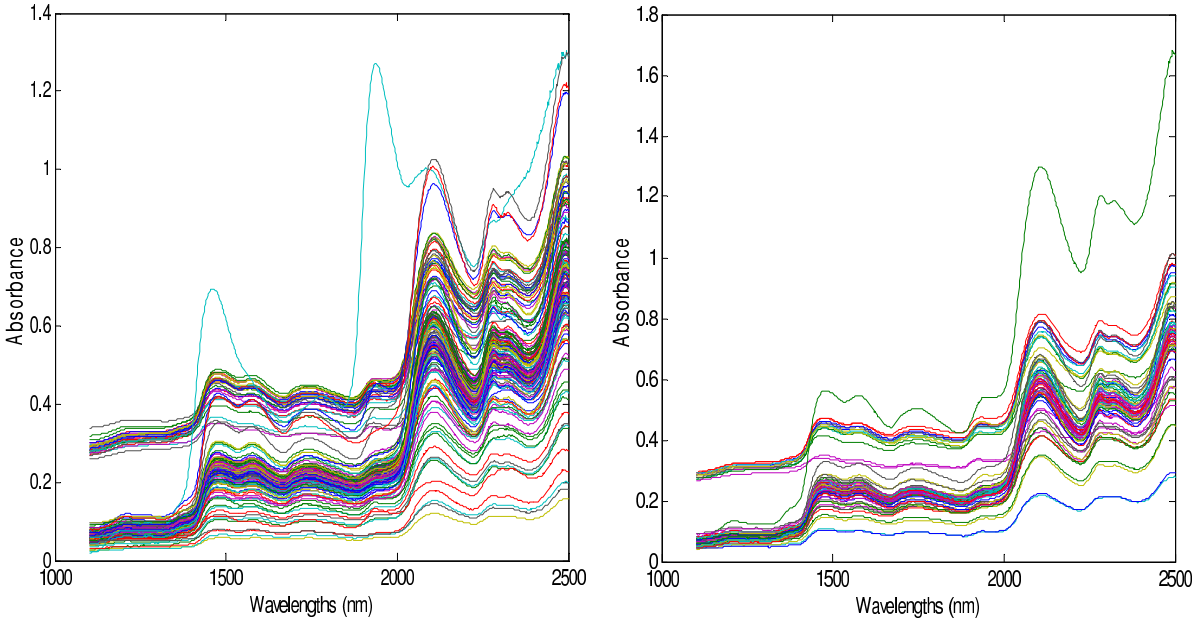


Figure 2.4: Near-infrared reflectance spectra of orange juice data set.

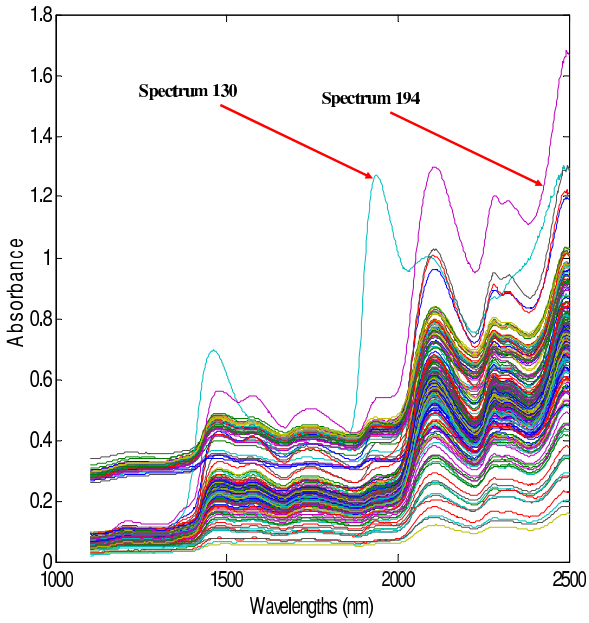


Figure 2.5: All the spectra of the orange juice data set.

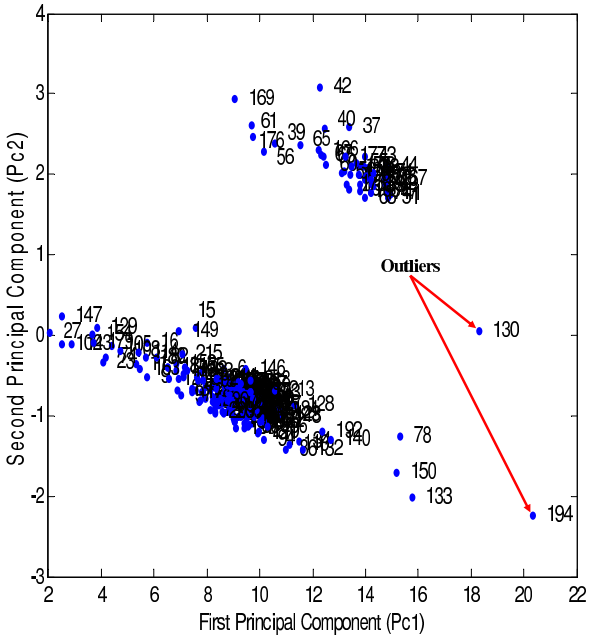
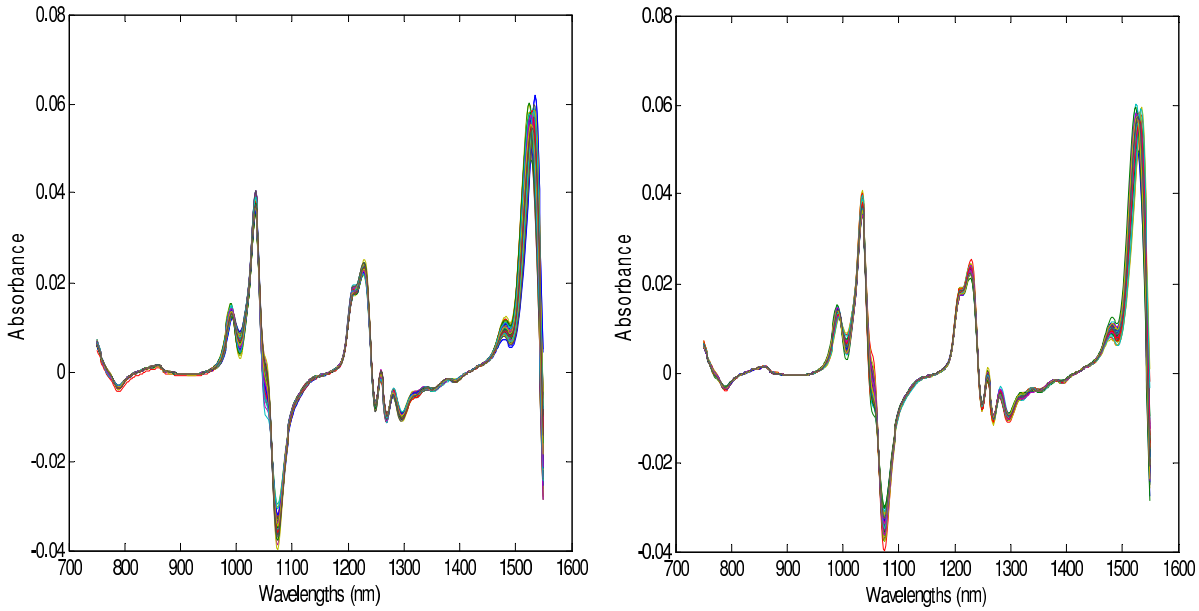


Figure 2.6: Principal components analysis (Pc1-Pc2) of the orange juice data set.

2.2.2.2 Diesel

The second data set refers to multispectral acquisitions of diesel fuels [80]. It was built by the Southwest Research Institute in order to develop instrumentation to evaluate fuel on battle fields. Along with the spectral acquisitions, different properties are available, such as boiling point at 50% recovery, cetane number, density, freezing temperature, total aromatics and viscosity. The data set contains only summer fuels, and outliers were removed. In our experiments, we consider one of the most difficult prediction tasks in this data set, that is, the prediction of the cetane number of the fuel. All spectra range from 750 to 1550 nm, discretized into 401 wavelength values. The data set contains 20 high leverage spectra, shown in Figure 2.7-(a), and 225 low leverage spectra, the latter being separated into two subsets labeled a and b. As suggested by the providers of the data, we have built a learning set with the high leverage spectra and a subset of the low leverage spectra (thus yielding 133 spectra). Figure 2.7-(b) show the test set is made up of the low leverage spectra of subset b (gathering 112 spectra). In Figures 2.8 and 2.9 the all original spectra and their principal components analysis are shown. Note that the spectra look already pre-processed and no outlier was detected.



(a) Training set (b) Test set

Figure 2.7: High leverage spectra (after centering and reduction) from the diesel data set.

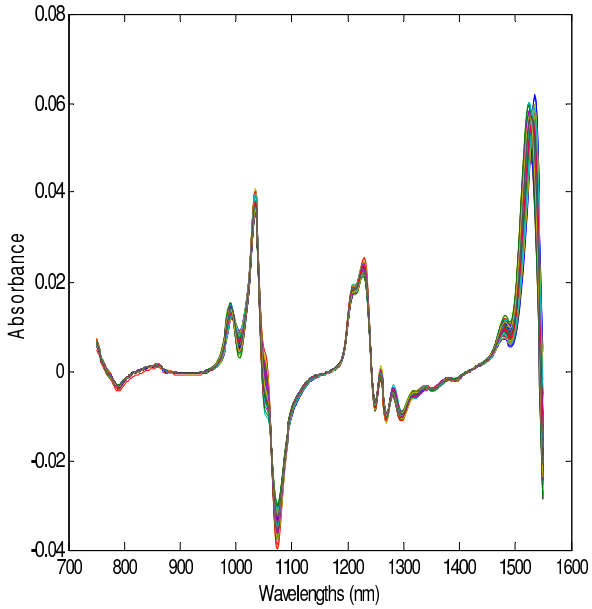


Figure 2.8: All the spectra of the diesel data set.

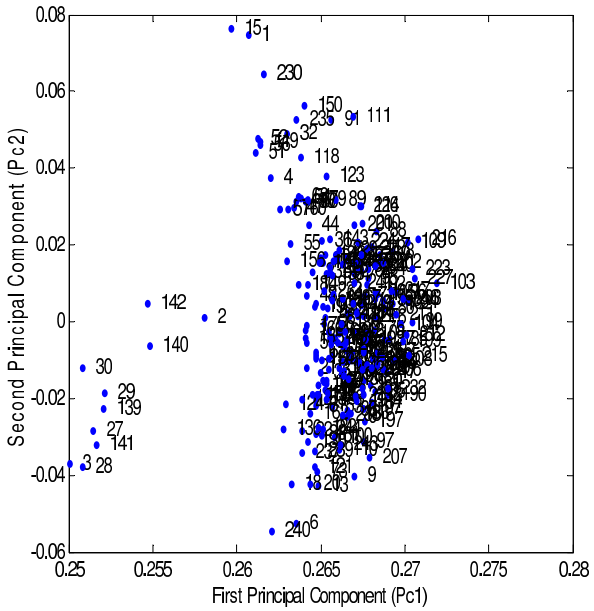


Figure 2.9: Principal components analysis (Pc1-Pc2) of the diesel data set.

2.2.2.3 Tecator

The third data set deals with the determination of the fat content of meat samples analyzed by near infrared transmittance spectroscopy [81]. The spectra have been recorded on a Tecator Infratec Food and Feed Analyzer working in the wavelength range 850-1050 nm. The spectrometer records light transmittance through the meat samples at 100 wavelengths in the specified range. The corresponding 100 spectral variables are absorbance defined by the measured transmittance. Each sample contains finely chopped pure meat with different moisture, fat and protein contents. Those contents, measured in percent by weight, are determined by analytic chemistry and range from 0.9 to 49.1%. The data set contains 172 training samples and 43 test samples (see Figure 2.10). The spectra are normalized according to the SNV method (standard normal variance, mean equal to zero and variance equal to 1). From the 215 original spectra, 2 outliers where detected. Figures 2.11 and 2.12 presents the all original spectra and the outlier detection procedure. It should be noted that the outliers detected in the tecator data set will not be eliminated [82].

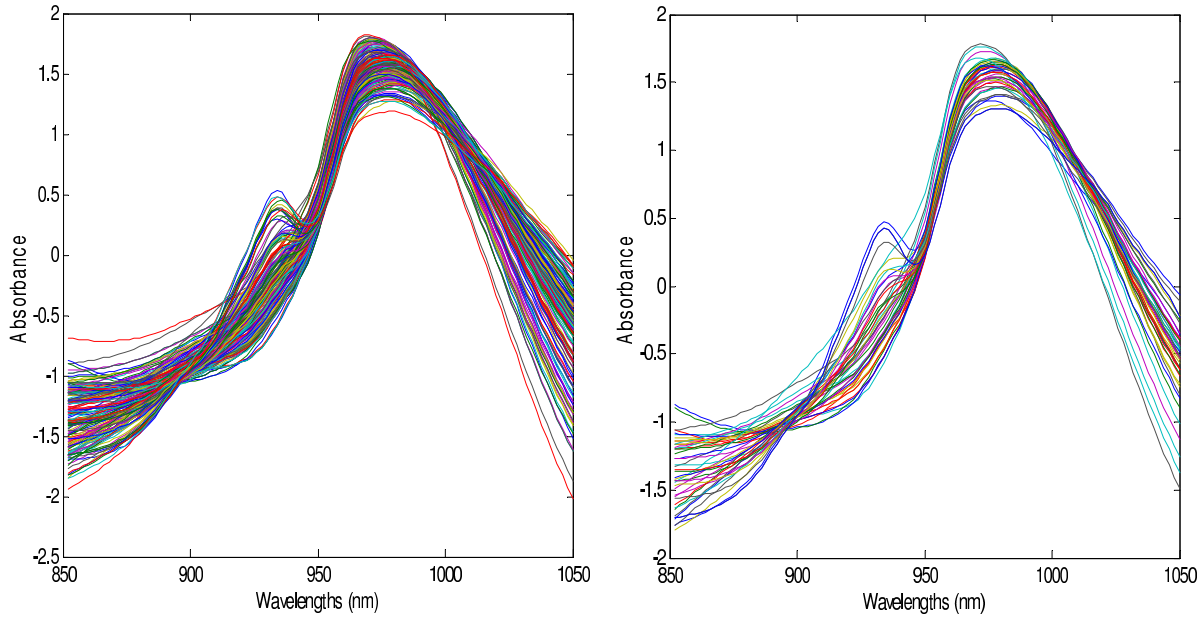


Figure 2.10: Near-infrared spectra of tecator data set.

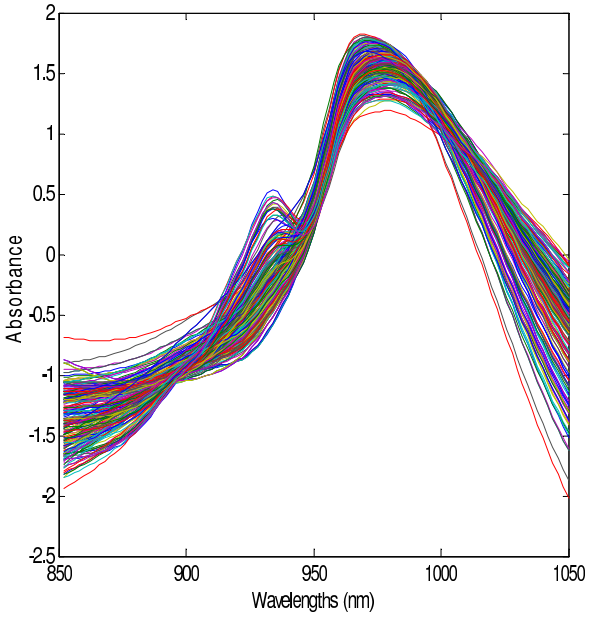


Figure 2.11: All the spectra of the tecator data set.

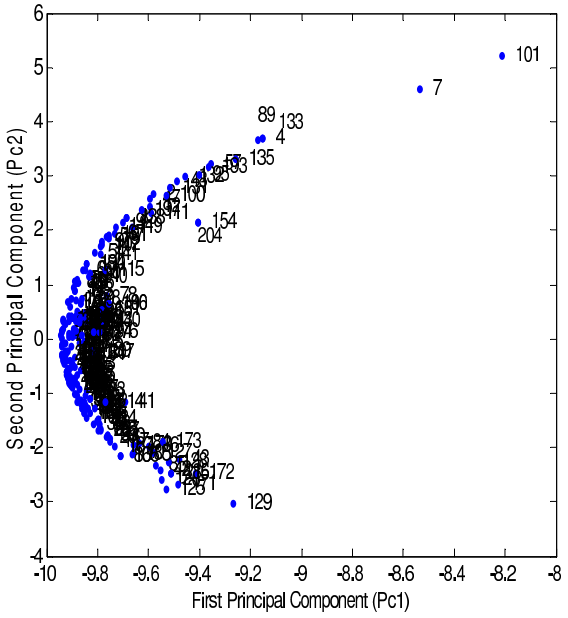


Figure 2.12: Principal components analysis (Pc1-Pc2) of the tecator data set.

2.3 Wind speed

2.3.1 Nature of the wind

The energy available in the wind varies as the cube of the wind speed, so an understanding of the characteristics of the wind resource is critical to all aspects of wind energy exploitation, from the identification of suitable sites and predictions of the economic viability of wind farm projects through to the design of wind turbines themselves, and understanding their effect on electricity distribution networks and consumers.

From the point of view of wind energy, the most striking characteristic of the wind resource is its variability. The wind is highly variable, both geographically and temporally. Furthermore this variability persists over a very wide range of scales, both in space and time. The importance of this is amplified by the cubic relationship to available energy. On a large scale, spatial variability describes the fact that there are many different climatic regions in the world, some much windier than others. These regions are largely dictated by the latitude, which affects the amount of insolation. Within any one climatic region, there is a great deal of variation on a smaller scale, largely dictated by physical geography the proportion of land and sea, the size of land masses, and the presence of mountains or plains for example. The type of vegetation may also have a significant influence through its effects on the absorption or reflection of solar radiation, affecting surface temperatures, and on humidity [43].

2.3.2 Geographical variation in the wind resource

The winds are driven almost entirely by the sun's energy, causing differential surface heating. The heating is most intense on land masses closer to the equator, and obviously the greatest heating occurs in the daytime, which means that the region of greatest heating moves around the earth's surface as it spins on its axis. Warm air rises and circulates in the atmosphere to sink back to the surface in cooler areas [43]. The tendency of climate remain which lead to clear climatic differences between regions. These differences are tempered by more local topographical and thermal effects [43,83]. The study of geographical distribution of wind speeds, characteristic parameters of the wind, topography and local wind flow and measurement of the wind speed are very essential in wind resource assessment for successful application of wind turbines [84]. The mountains and hills result in local regions of increased wind speed. This is partly a result of altitude - the earth's boundary layer means that wind speed generally increases with height above ground, and hill tops and mountain peaks may 'project' into the higher wind-speed layers. It is also partly a result of the acceleration of the wind flow over and around hills and mountains, and funneling through passes or along valleys aligned with the flow. Equally, topography may produce areas of reduced wind speed, such as sheltered valleys, areas in the lee of a mountain ridge or where the flow patterns result in stagnation points [43]. For example, Figure 2.13 shows a geographical map of the distribution of wind speed in January and July.

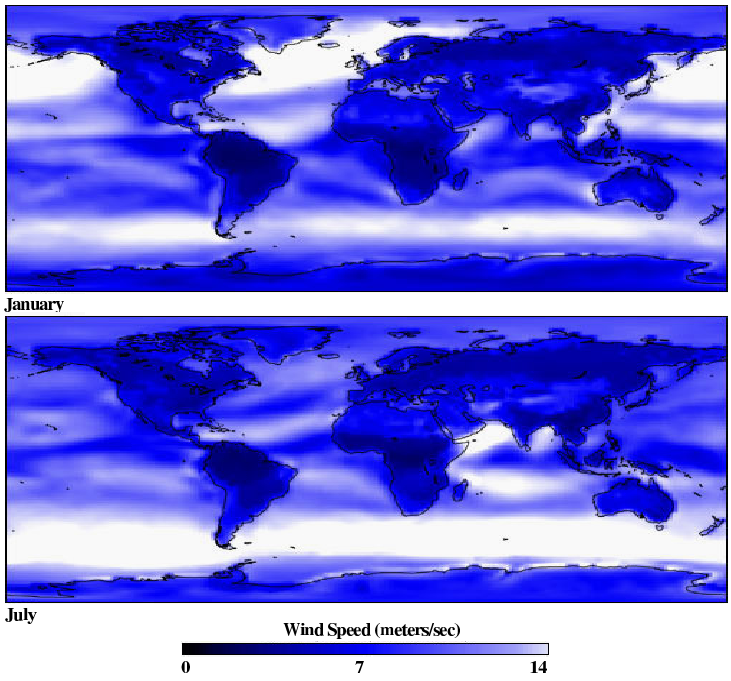


Figure 2.13: Seasonal world wind resource map in January and July.

The thermal effect is another important factor may also result in considerable local variations. Coastal regions are often windy because of differential heating between land and sea. This effect may also be produced by differences in altitude. Thus cold air from high mountains can sink down to the plains below, causing quite strong and highly stratified 'downslope' winds [43].

2.3.3 Long-term wind speed variations

There is evidence that the wind speed at any particular location may be subject to very slow long-term variations. Although the availability of accurate historical records is a limitation [85]. Clearly these may be linked to long term temperature variations for which there is ample historical evidence. There is also much debate at present about the likely effects of global warming, caused by human activity, on climate, and this will undoubtedly affect wind climates in the coming decades. Apart from these long-term trends there may be considerable changes in windiness at a given location from one year to the next. These changes have many causes. They may be coupled to global climate phenomena, changes in atmospheric particulate resulting from volcanic eruptions, and sunspot activity. These changes add significantly to the uncertainty in predicting the energy output of a wind farm at a particular location during its projected lifetime [43].

2.3.4 Classification according to time horizons

The various forecasting models in the literature can be different based on the forecasting horizons, which can vary according to the required application and the technique used. The forecasting system is divided into four categories according to time horizons: very short term, short term, medium term, or long term. The time span is different in various literature descriptions. The specific classification is listed in Table 2.1, this table also presents the application for each prediction horizon [86, 87].

Table 2.1: Classification of different time horizons.

Time Horizon	Range	Application Purpose
Very short term (in minutes)	Few seconds to 30 minutes ahead	• Electricity Market Clearing • Wind Turbine Control
Short term (in hours)	30 minutes to 48(or 72) hours ahead	• Economic Load Dispatch Planning • Load Increment/Decrement Decisions
Medium term (in days)	48(or 72) hours to 1 week ahead	• Generator Online/Offline Decisions (Arrangements for Maintenance) • Unit Commitment Decisions
Long term (in years)	1 week to 1 year or more ahead	• Maintenance Scheduling to Obtain Optimal Operating Cost • The Feasibility Study for Design of the Wind Farm

2.3.5 Wind prediction

In general, different methods are used for wind prediction. The easiest ones are based on climatology or averages of past production values. They may be considered as reference forecasting methods since they are easy to implement. The famous of these reference methods is certainly persistence. This is one of the simplest models. It is very effective for very short term forecasting. It's based on the idea that under similar conditions the next forecast data point will be approximately the same or constant to present data point value, due the simplicity, reduced complexity and low cost implementation. This model is very popular. However, this model suffer from a drawback, which is the larger the time horizon is the bigger the prediction error. Advanced approaches for wind power forecasting necessitate predictions of meteorological variables as input. Then, they differ in the way predictions of meteorological variables are converted to predictions of wind power production, through the so-called power curve [43].
Wind power generation is directly linked to weather conditions and thus the first aspect of wind power forecasting is the prediction of future values of the necessary weather variables at the level of the wind farm. This is done by using numerical weather prediction (NWP) models. Such models are based on equations governing the motions and forces affecting motion of fluids. From the knowledge of the actual state of the atmosphere, the system of equations allows to estimate what the evolution of state variables, e.g. temperature, velocity, humidity and pressure, will be at a series of grid points. The meteorological variables that are needed as input for wind power prediction obviously include wind speed and direction, but also possibly temperature, pressure and humidity. The distance between grid points is called the spatial resolution of the NWPs. The main disadvantage of using this model is that the cost of implementation, complexity involved in it takes a long time in processing to train the model [88].

In the literature, several wind speed prediction methods can be found. They can be divided into three categories: physical methods, statistical methods, and machine learning methods.

2.3.5.1 Physical approach

It is based on a detailed physical considerations to predict the future wind speed like terrain, obstacle, pressure, and temperature. Sometimes they are only the first step to forecast the wind, which is supplied as auxiliary input of other statistical models. Numeric weather prediction (NWP) is developed by meteorologists for large-scale area weather prediction. NWP is a physical approach to wind forecasting, it is operate by solving complex mathematical models that use weather data [86, 89].

2.3.5.2 Statistical approach

The statistical approach is based on training with measurement data and uses difference between the predicted and the actual wind speeds in immediate past to tune model parameters, it is not based on any predefined mathematical model and rather it is based on statistical linear and nonlinear models [86, 90, 91].

2.3.5.3 Machine learning approach

Models based on machine learning techniques such as Artificial Neural Network (ANN), bayesian networks, fuzzy logic, support vector machine (SVM) and hybrid models, are used for the wind speed data in recent years, because of their excellent ability to learn non-linear relationships from experience, many researchers found these techniques to be effective for wind speed and power output prediction [24–36].
In this context, different architecture and types of artificial neural network is presented. For instance, Fadare [24] compared three different artificial neural networks (ANNs) applied to wind speed in Nigeria and used different configurations of the ANN. Li [92] investigated a method to do one-step-ahead prediction of wind power generation using recurrent multilayer perceptron neural networks (RMLP). Results showed that the RMLP model performed better in 1-hour prediction than that for 10-min prediction. In [25], three types of neural networks, namely, adaptive linear element, back propagation, and radial basis function, are compared. Mohandes [29] used neural networks (ANN) to forecast the mean of monthly and daily wind speed. The forecasting accuracy was compared with the autoregressive (AR) model. The results indicated that the ANN model outperformed the AR model for all examined forecasting horizons. In [28], introduced the support vector machines (SVM) for wind speed prediction and compared it with the multilayer perceptron neural networks (MLP). The results proved that the SVM model is better than MLP model. In [33], hybridization of linear regression (MLR, SVM-linear) and nonlinear regression (ANN, SVM-Gaussian, SVM-polynomial). Sfetsos [93] examined and

compared various artificial intelligence based forecasting models based on time series analysis. The models examined in this study include ARMA models, feed-forward and recurrent neural networks, Adaptive Neuro-Fuzzy Inference Systems (ANFIS) and Neural Logic Networks (NLN). In [94], used an artificial neural network (ANN) to predict the average hourly wind speed and the related power production.
Among several nonlinear regression approaches, KRR, which is a kernel version of the ridge regression (RR), has a very good generalization performance by selecting the suitable regularization parameter [95]. The nonlinear maps can be approximated by means of kernel ridge regression, an extension of linear ridge regression based on kernel functions. KRR offers the advantage of being fast to evaluate, requiring only a single matrix inversion which depends on the number of points but is independent of the number of attributes in the input space [96]. The focus of this thesis lies on machine learning approach for wind speed prediction.

2.3.6 Low-dimensional data

Many data analysis borrowed from statistics or machine learning were designed for low-dimensional, large sample data, based on a scheme that the human mind can apprehend, in this case we do not need to features selection.
Let us consider a set of n training samples $X = (x_1, ..., x_n)$ with $x_i = (x_{i1}, ..., x_{id})$ represented in the d-dimensional feature space $\Re^d$. In this section of wind speed data set, we assume in addition that the data are low-dimensional. Here, this means that d should at least be smaller than n (This section gives additional information about the wind speed data sets used in the Chapter 5).

2.3.6.1 Geographic location of Algeria

Algeria's geographic location has several advantages for extensive use of most of the renewable energy (solar and wind). Algeria situated in the centre of North Africa between the 38-35° of latitude north and 8-12° longitude east, has an area of 2,381,741 km^2 [97].
Administratively speaking, Algeria is divided into 48 provinces and lies, in the north, on the coast of the Mediterranean Sea. The length of the coastline is 2400 km. In the west Algeria borders with Morocco, Mauritania and occidental Sahara, in the southwest with Mali, in the east with Tunisia and Libya, and in the southeast with Niger (see Figure 2.14) [41].

Figure 2.14: Algeria's geographic location.

2.3.6.2 Wind speed data sets in Algeria

We used different wind speed data measurement stations distributed over the vast Algerian territory, which cover a period of ten years (between January 1st, 2001 to December 31st, 2010) based on ten different stations in Algeria, namely, Tlemcen, Chlef, Alger, Annaba, Djelfa, Batna, El Oued, Ghardaia, Adrar and Tamanrasset. These stations are distributed in different regions of the country as depicted in Figure 2.15.

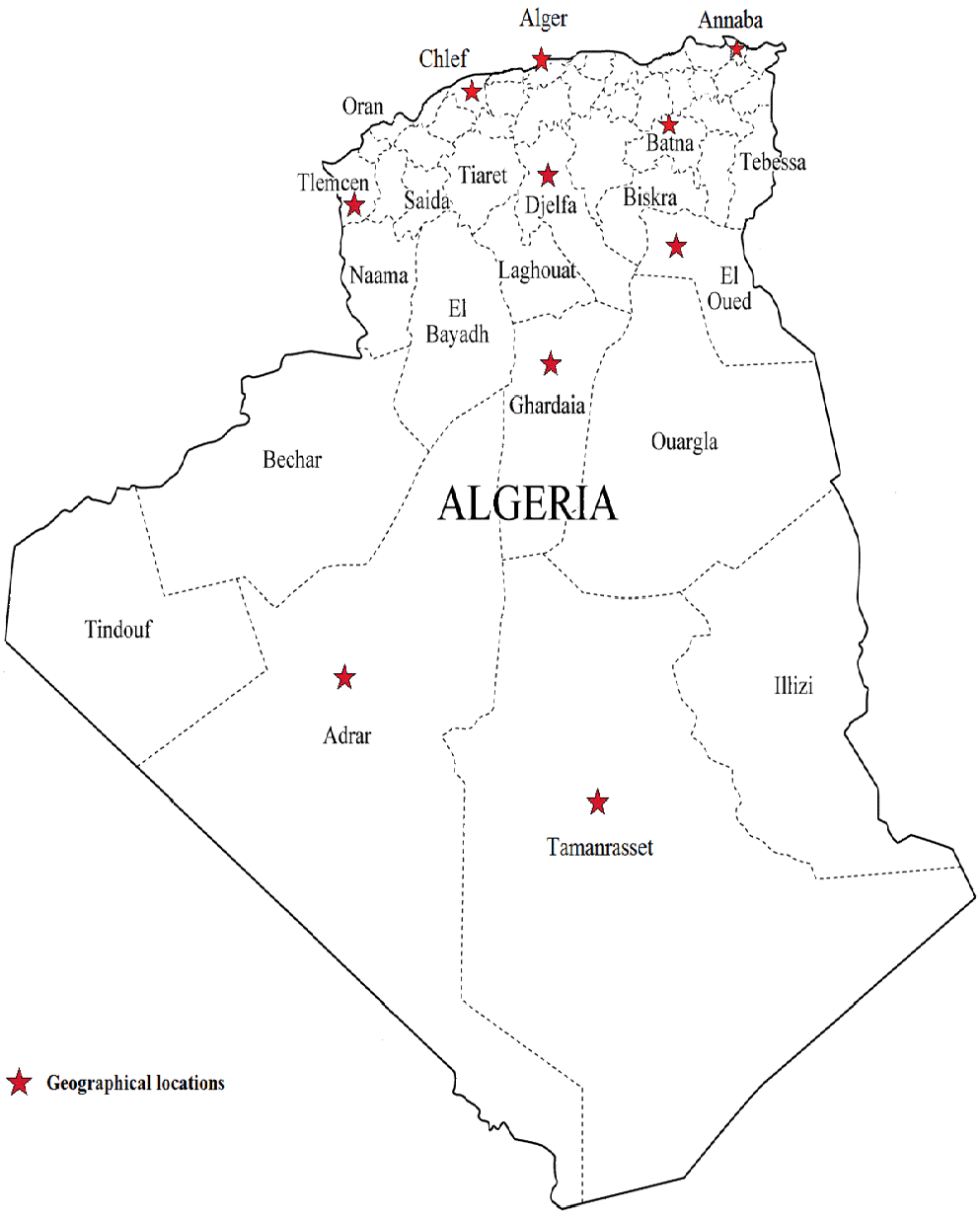


Figure 2.15: Geographical location of the meteorological stations considered in the experiments.

Table 2.2 provides the exact location of these stations, their altitude as well as the related number of daily measurements used for training and testing the investigated methods. The month number, the day number (within the month), three temperature measurements (average, maximum and minimum temperatures) and the average relative humidity are used as input features for the prediction, which in turn provides in output an estimate of the mean wind speed.

Table 2.2: Information about the meteorological stations considered in the experiments.

	Location			Data set information	
Name	**Latitude**	**Longitude**	**Altitude (m)**	**# training samples (days)**	**# test samples (days)**
Tlemcen	35.01	-1.46	247	2519	1095
Chlef	36.21	1.33	143	1679	961
Alger	36.76	3.1	12	965	1013
Annaba	36.83	7.81	4	2514	1090
Djelfa	34.33	3.25	1144	2401	1042
Batna	35.75	6.18	1052	2415	1086
El Oued	33.5	6.11	63	2473	954
Ghardaia	32.4	3.81	450	2486	1092
Adrar	27.88	-0.28	263	2262	1001
Tamanrasset	22.8	5.51	1364	2319	1087

For each station, the wind speed data set was divided into two sets: 1) a training set, from 1st January 2001 to 31st December 2007; and 2) a test set, from 1st January 2008 to 31st December 2010. Figures 2.16-2.19 illustrates ten different wind speed data sets used in the experimental analysis.

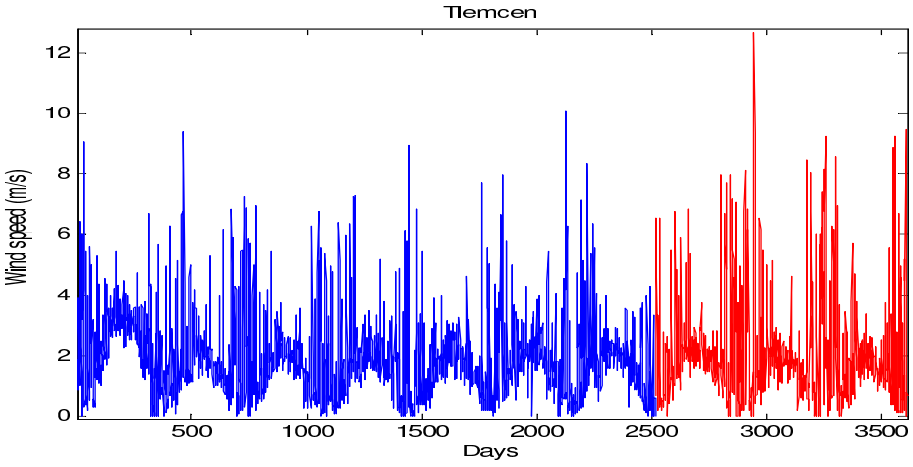


Figure 2.16: Daily wind speed behavior for Tlemcen station. The blue curve show the training samples, while the red curve show the test samples.

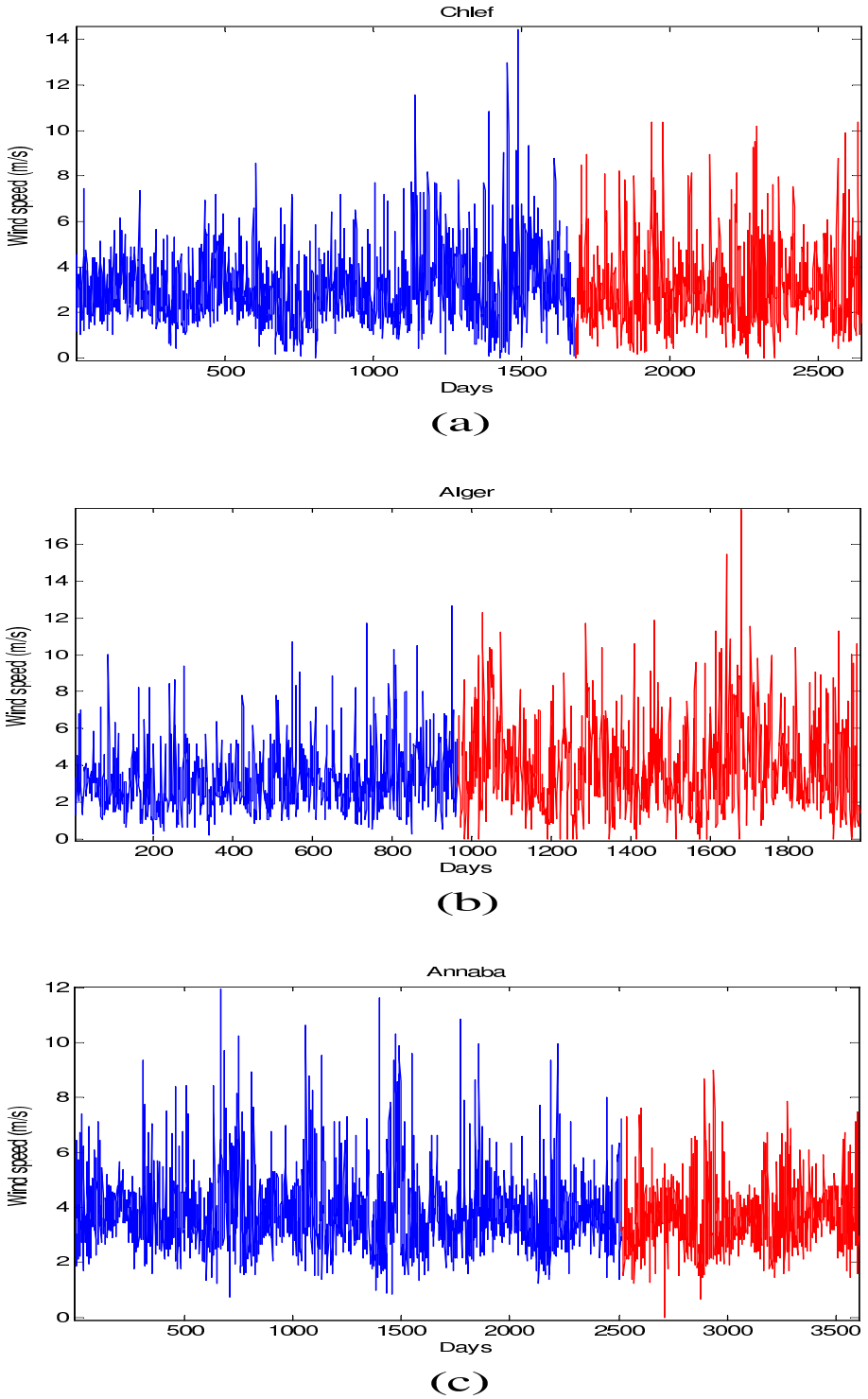


Figure 2.17: Daily wind speed behavior for (a) Chlef, (b) Alger and (c) Annaba stations. The blue curve show the training samples, while the red curve show the test samples.

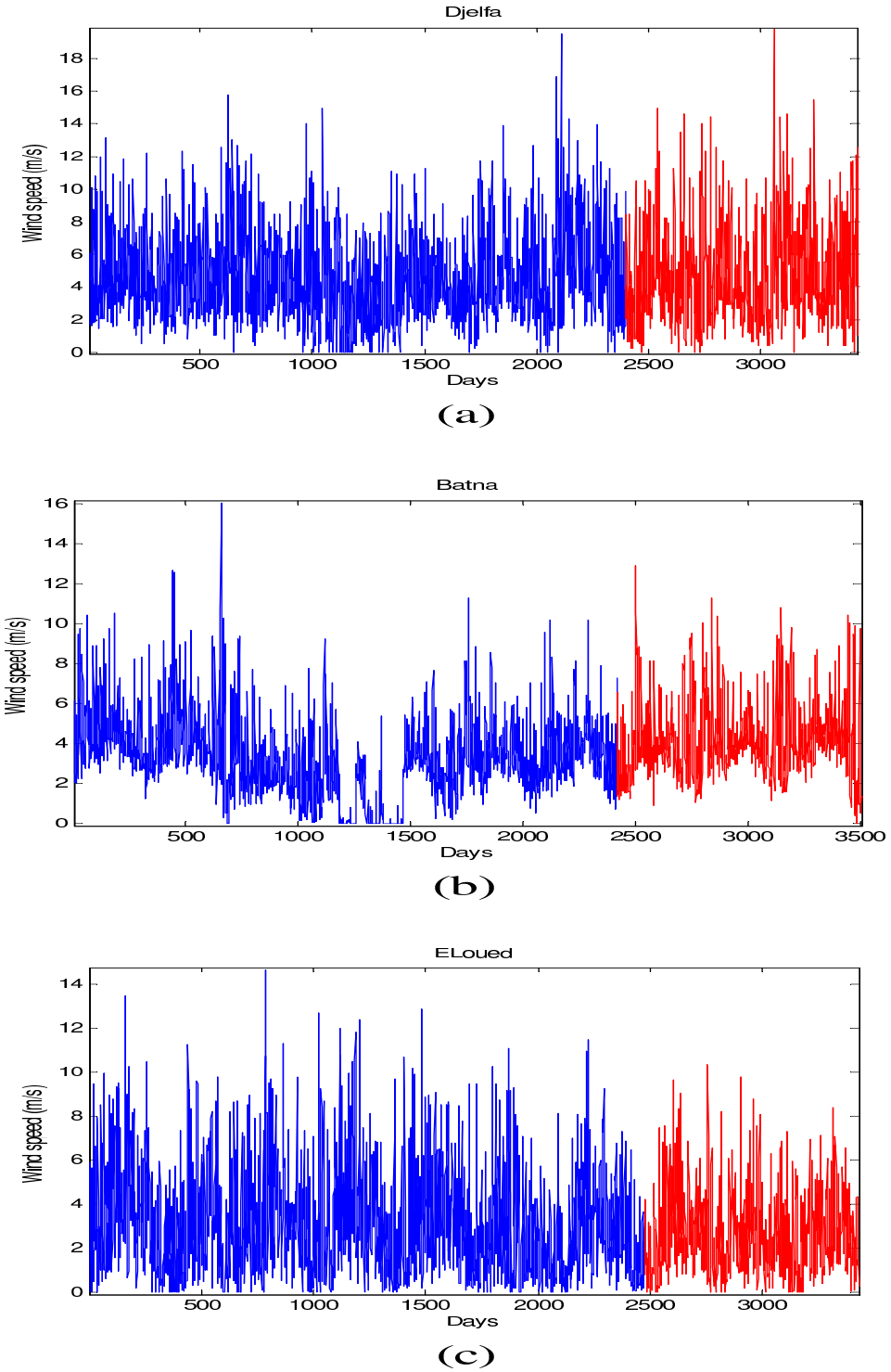


Figure 2.18: Daily wind speed behavior for (a) Djelfa, (b) Batna and (c) El Oued stations. The blue curve show the training samples, while the red curve show the test samples.

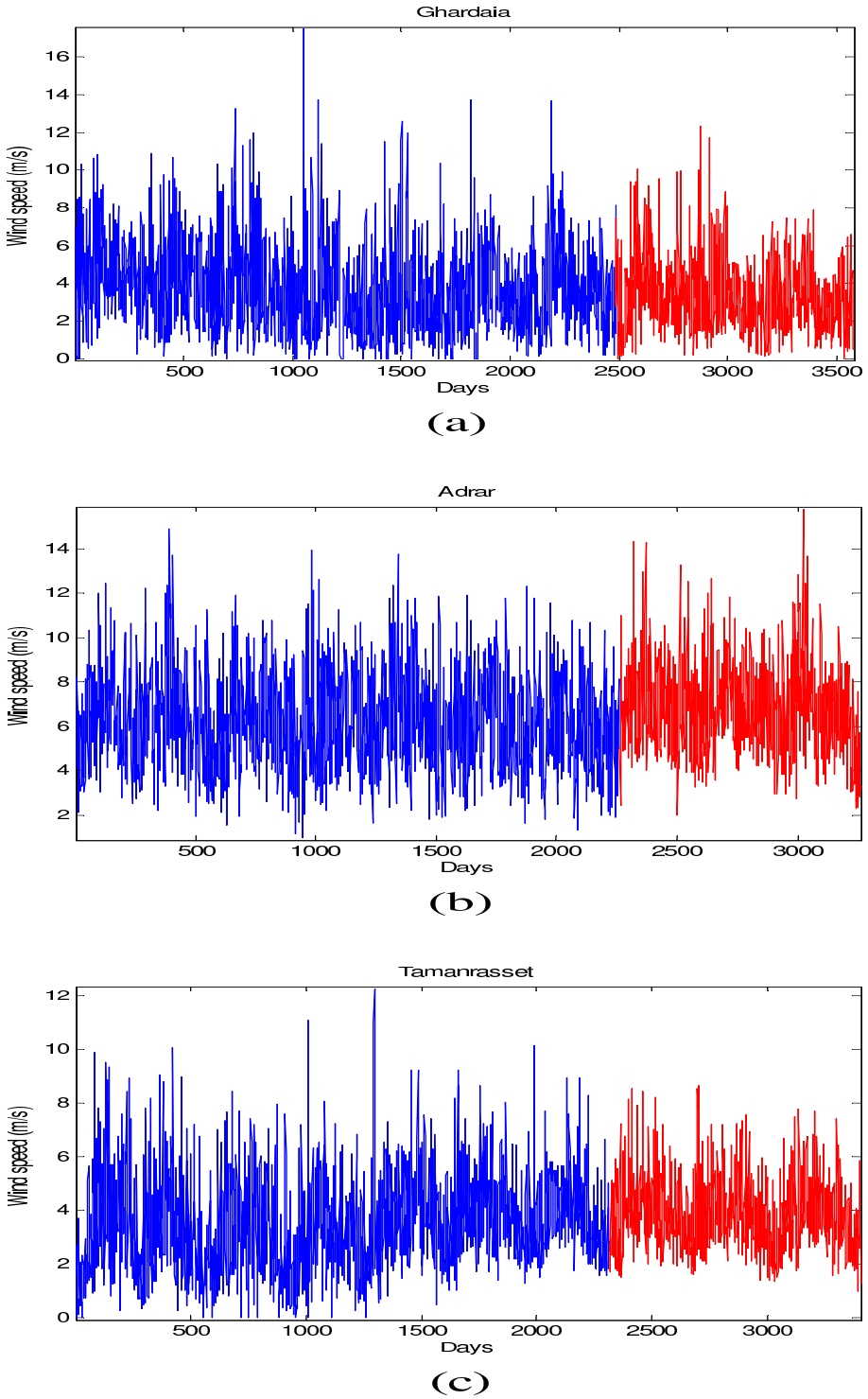


Figure 2.19: Daily wind speed behavior for (a) Ghardaia, (b) Adrar and (c) Tamanrasset stations. The blue curve show the training samples, while the red curve show the test samples.

2.4 Conclusion

This chapter has presented a literature review on spectroscopy and forecasting of wind speed data sets. After the representation of the data sets, it can be noted that in the case of spectroscopy data, the variation of the concentration can be considered as a linear or nonlinear problem. However, in the case of wind speed we have a nonlinear estimation problem. Visualization is the paramount aim of data analysis. Several methods to visualize the data with a low-dimensional representation are of great help. As seen in this chapter among the most used methods is the projection of data (e.g. PCA). A two-dimensional representation is often preferred as representations of higher dimensional require training to be used properly, this representation provides interesting information about the data distribution and other such as outliers.
In the second section of this chapter, we have introduced the most important principles related to the wind energy, its nature, geographical variation in the wind resource and the long-term wind speed variations. Finally, the representation of the ten data sets used in the following chapters is provided. This main constituents is related to a good understanding of its characteristics.
Since in the last years, these data sets have so increased in uses and applications that now modern learning techniques are usually combined with them. The objective of the next chapters is to describe and used regression methods to predict the concentration of food product and wind speed prediction by means of a classical and new methods.

Chapter 3

Linear and Nonlinear Regression

Contents

3.1 Introduction

In the literature, two main approaches of regression have been proposed. The first is based on linear models, such as the multiple linear regression (MLR), the principal component regression (PCR) and the partial least square regression (PLSR) methods. The MLR is a simpler approach for calibration model creation than PCR and PLS, because it performs regression directly on the original variables while PCR carries out regression on principal components that do not necessarily have a physical meaning and PLSR finds a projection subspace by exploiting the target variable. However, due to the colinearity between original spectral variables, overfitting problems can be encountered in MLR [98]. PCR first consists of applying a principal components analysis (PCA) to the matrix of the spectral data. Then, PCA replaces the original spectral variables (typically redundant) by principal components (linear combinations of the original variables), which contain most of the conveyed information and have the advantage of being uncorrelated [99]. The most important principal components are then used as inputs for a multiple linear regression (MLR) [100]. PLSR aims at finding linear projections which exhibit the maximum correlation with the target (output) variable; a linear regression model is then estimated in the subspace defined by the projected coordinates [101]. Linear regression has the advantage of being simple and cheap in terms of computation load, but is not reliable if the true relationship between the inputs and the output is nonlinear, unless opportune preprocessing methods are adopted. The second approach makes use of nonlinear models such as artificial neural networks and support vector machines (SVM) [2, 3, 102, 103]. The most popular neural networks are the multilayer perceptrons (MLP) which are composed of layers of neurons (computing units). MLP have some drawbacks associated with the training process: 1) no guarantee of convergence toward the global optimum, 2) long training time, 3) empirical architecture definition and free parameter setting. Moreover, they are affected by a serious risk of overfitting the training data in particular when increasing the size of the network [104, 105]. However, radial basis function neural network (RBFN) and SVM have proved to perform nonlinear multivariate function estimation and nonlinear regression tasks in an effective way. RBFN are based on a single hidden layer of neurons using Gaussian transfer functions, and a linearly activated output layer. In comparison with MLP, RBFN appear to offer some advantages such as robustness to noise and much faster training procedures. SVM rely on the principle of structural risk minimization, which promises good generalization capability and very good performance when just few training samples are available [106, 107].

In this chapter we present briefly the different methods of linear and nonlinear regression that we used in this thesis. In Section 3.2, the basic concepts of two linear regression and two linear projection techniques, being, least squares (LS), ridge regression (RR), principal component regression (PCR) and partial least squares regression (PLS), respectively, are recalled. Section 3.3, three nonlinear regression methods, kernel ridge regression (KRR), support vector regression (SVR), radial basis function neural network (RBFN), respectively, are described.

Finally, conclusions in Section 3.4.

3.2 Linear regression methods

3.2.1 Linear regression

The problem of linear regression consists in finding a linear function:

$$f(x) =< w \cdot x > +b, \tag{3.1}$$

where $w \in \Re^d$ is a vector of weights and $b \in \Re$ is the bias. The weights and bias are the parameters of the model. That best interpolates a given set of training points labeled from $Y \subseteq \Re$. Geometrically this corresponds to a hyperplane fitting the given points. Figure 3.1 shows a one dimensional linear regression function. The distance shown as ξ in the figure is the error for the particular training example.
The estimation of the best-known solution is related of choosing the line that minimizes the sum of the squares of the distances from the training points. This technique is known as least squares, and is known to be optimal in the case of linear targets corrupted by Gaussian noise [107].

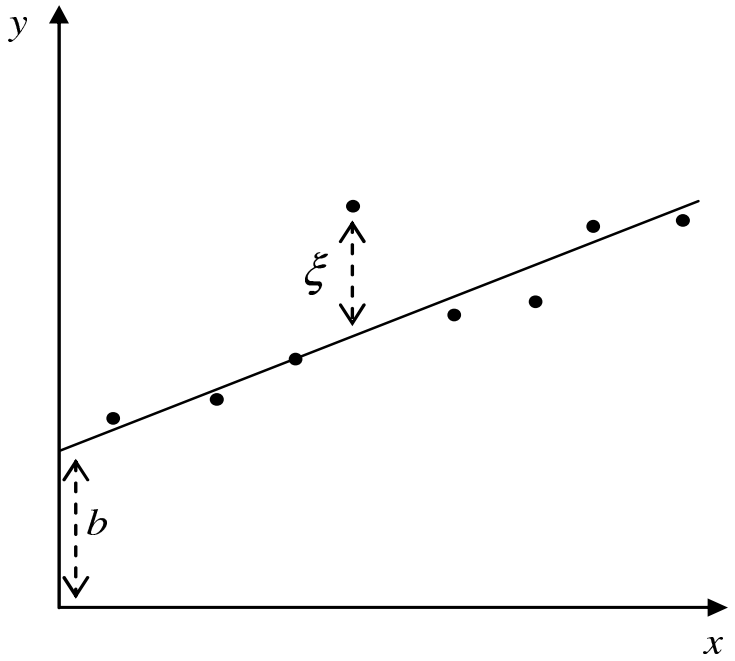


Figure 3.1: A one dimensional linear regression function.

3.2.1.1 Least squares (LS)

The prediction of a dependent variable y from independent variable $x_1, x_2, ..., x_n$ is defined by the equation:

$$y = f(x) =< w \cdot x > +b. \tag{3.2}$$

The least squares approach prescribes choosing the parameters (w, b) to minimize the sum of the squared deviations of the data,

$$L(w, b) = \sum_{i=1}^{l} (y - < w \cdot x_i > -b)^2. \tag{3.3}$$

The function L is known as the square loss function as it measures the amount of loss associated with the particular choice of parameters by a sum of squares. The loss function L is minimized by differentiating with respect to the parameters (w, b), and setting the resulting $n + 1$ linear expressions to zero [107]. This is best expressed in matrix notation by setting $\hat{w} = (w', b)'$, and $\hat{X}$ is represented by a vector:

$$\hat{X} = \begin{pmatrix} \hat{x}_1' \\ \hat{x}_2' \\ . \\ . \\ . \\ \hat{x}_l' \end{pmatrix}, where\ \hat{x}_i' = (x_i', 1)'. \tag{3.4}$$

With this notation the vector of output discrepancies becomes

$$y - \hat{X}\hat{w}, \tag{3.5}$$

with y a column vector. Hence, the loss function can be written as

$$L(\hat{w}) = (y - \hat{X}\hat{w})'(y - \hat{X}\hat{w}). \tag{3.6}$$

Taking derivatives of the loss and setting them equal to zero,

$$\frac{\partial L}{\partial \hat{w}} = -2\hat{X}'y + 2\hat{X}'\hat{X}\hat{w} = 0, \tag{3.7}$$

yields the well-known normal equations

$$\hat{X}'\hat{X}\hat{w} = \hat{X}y, \tag{3.8}$$

and, if the inverse of $\hat{X}'\hat{X}$ exists, the solution of the least squares problem is

$$\hat{w} = (\hat{X}'\hat{X})^{-1}\hat{X}'y. \tag{3.9}$$

If $\hat{X}'\hat{X}$ is singular, the pseudo-inverse can be used, or else the technique of ridge regression described below can be applied [107].

3.2.1.2 Ridge regression (RR)

Because of the high degree of collinearity as d increases, the matrix $\hat{X}'\hat{X}$ tends to be singular, and the estimates become unstable, thus leading to an inappropriate regression model. In order to solve this problem, one can resort to RR for which:

$$\hat{w} = (\hat{X}'\hat{X} + \lambda I_n)^{-1}\hat{X}'y, \tag{3.10}$$

where λ is the ridge parameter ($\lambda \in \Re, \lambda \geq 0$) and I_n is the identity matrix with the $(n+1, n+1)$ entry set to zero [107, 108].
The ridge regression algorithm minimizes the penalized loss function [107]:

$$L(w, b) = \lambda < w \cdot w > + \sum_{i=1}^{l} (< w \cdot x_i > + b - y_i)^2, \tag{3.11}$$

so that the parameter λ controls a trade-off between low square loss and low norm of the solution.
Note that ridge regression also admits a dual representation. The solution needs to satisfy $\frac{\partial L}{\partial w} = 0$, which gives the following expression for the hypothesis:
$\lambda w = -\sum_i (< w \cdot x_i > b - y_i)x_i$, which implies that there exist scalars $\alpha_i = -\frac{1}{\lambda}(< w \cdot x_i > b - y_i)$, such that the solution can be written as $w = \sum_i \alpha_i x_i$.
Once we know that the solution can be expressed in dual form we can derive conditions that α must satisfy. We can express the duality condition in vector form by expressing the weight vector in terms of the vector α:

$$w = X'\alpha, \tag{3.12}$$

where X is the matrix with the last column (of 1s) removed. We can rewrite Equation (3.10) as follows, where we have set b to zero for simplicity:

$$\begin{aligned} L(w) &= \lambda\alpha^{'}XX^{'}\alpha + \sum_{i=1}^{l}(\alpha^{'}Xx_i - y_i)^2 \\ &= \lambda\alpha^{'}G\alpha + \sum_{i=1}^{l}((G\alpha)_i - y_i)^2 \\ &= \lambda\alpha^{'}G\alpha + (G\alpha - y)^{'}(G\alpha - y) \\ &= \lambda\alpha^{'}G\alpha + \alpha^{'}GG\alpha - 2y^{'}G\alpha + y^{'}y, \end{aligned} \tag{3.13}$$

where $G = XX^{'} = G^{'}$. Taking derivatives with respect to α and setting to zero we obtain the equation

$$2G(\lambda\alpha + G\alpha - y) = 0. \tag{3.14}$$

This equation will be satisfied if

$$(\lambda I + G)\alpha = y, \tag{3.15}$$

giving a predictive function of

$$f(x) = y^{'}(\lambda I + G)^{-1}z, \tag{3.16}$$

where $z_i =< x \cdot x_i >$. Note how this dual equation depends on the Gram matrix of inner products of the training examples, $G = XX^{'}$ [107].

3.2.2 Linear projection techniques

Two of the most popular multivariate projections techniques are Principal Component Regression (PCR) and Partial Least Squares Regression (PLSR). They both use a linear inner relation.

3.2.2.1 Principal component regression (PCR)

One problem with multivariate data is that the sheer volume may make it difficult to see patterns and relationships. The aim of many methods of multivariate analysis is data reduction. Quite frequently there is some correlation between the variables, so some of the information is redundant. In PCR, Principal component analysis (PCA) [109] is a technique for reducing the amount of data when there is correlation present. It is worth stressing that it is not a useful technique if the variables are uncorrelated [110]. In the first step, the PCA are calculated. PCA are intimately related to covariance matrix. For instance, the loading vectors of Principal Component Analysis (PCA) can be extracted either from a mean centered data set X or from the corresponding covariance matrix $C = (X^T X)/(N-1)$, with N is the number of samples in X.

To identify the loadings from X, this can be done using e.g. the NIPALS algorithm [111] or the Singular Value Decomposition (SVD) [112], among others, can be employed.
In the final step in PCR, the output variable, y is regressed on the principal components using ordinary multiple linear regression (MLR) [113].
PCR can be characterized as an unsupervised method, since the principal components are not found using information about the output variable y. Instead, PCR is a variance maximizing method, because only those principal components which contribute the most to the variability in X are considered [114].

3.2.2.2 Partial least squares regression (PLS)

Another strategy to adjust the flexibility of a linear model is to reduce the dimensionality of the input vectors by projecting them on a small dimensional linear subspace. Let us consider a set of labeled samples $L = \{x_l, y_l\}_{l=1}^{n}$, where $x_l = [x_{l1}, ..., x_{ld}] \in \Re^d$ represents a vector of d spectral acquisitions and/or processed features and $y_l \in \Re$ is the associated target value, that is, the measurement of the concentration value of interest. Let us aggregate all x_l's $l = 1, ..., n$ into a $n \times d$ feature matrix X and all y_l's $(l = 1, ..., n)$ into a target vector y so that $L = \{X, y\}$. The goal is to deduce from the set of labeled samples L the function $f(.)$ so that $y = f(x)$.
Contrary to PCR, PLSR is a supervised method, where the influence of y is incorporated when the latent variables are found. PLSR aims to find a linear regression model by projecting data to a new space [115–118]. In particular, it tries to find the multidimensional direction in the space X that explains the maximum multidimensional variance direction in the space y. The user has to supply the number ℓ_v of latent factors in the regression. If ℓ_v equals the rank of the matrix X, the method yields simply the least squares regression estimates. After centering the input X and y, the following steps are performed for each latent factor k $(k = 1, ..., \ell_v)$ [113]:

1. find the weight vector w_k by maximizing the covariance between the linear combination $X_{k-1}w_k$ and y under the constraint that $w_k^{'}w_k = 1$. This corresponds to find the unit vector w_k that maximizes $w_k^{'}X_{k-1}^{'}y_{k-1}$, i.e., the scaled covariance between X_{k-1} and y_{k-1}

$$w_k = \frac{X_{k-1}^{'}y_{k-1}}{\left\|X_{k-1}^{'}y_{k-1}\right\|}. \tag{3.17}$$

2. find the factor score t_k as the projection of X_{k-1} on w_k, so that the X-residuals E

$$X_{k-1} = t_k w_k + E. \tag{3.18}$$

Since $w_k^{'}w_k = 1$, the solution is

$$t_k = X_{k-1}w_k. \tag{3.19}$$

3. regress X_{k-1} on t_k to find the loadings $p_k^{'}$

$$X_{k-1} = t_k p_k^{'} + E. \tag{3.20}$$

The least square solution is given by

$$p_k = X_{k-1}^{'} t_k / t_k^{'} t_k. \tag{3.21}$$

4. regress y_{k-1} on t_k to find q_k , so that the y-residuals F

$$y_{k-1} = t_k q_k + F. \tag{3.22}$$

The solution is given by

$$q_k = y_{k-1}^{'} t_k / t_k^{'} t_k. \tag{3.23}$$

5. subtract $t_k p_k^{'}$ from X_{k-1} in order to obtain X_k. Similarly, y_k is obtained by subtracting $t_k q_k^{'}$ from y_{k-1}. After the computation of the latent factors, the matrix X is deflated by subtracting $t_k q_k^{'}$ from X. In this way, the model refers to the residuals after previous dimension E instead of relating to the variables X themselves

$$E = X_{k-1} - t_k p_k^{'}, \tag{3.24}$$

$$F = y_{k-1} - t_k q_k^{'}. \tag{3.25}$$

Replacing X_{k-1} and y_{k-1} by the residuals E and F and increasing k of one, we obtain

$$X_k = E, \tag{3.26}$$

$$y_k = F, \tag{3.27}$$

$$k = k + 1. \tag{3.28}$$

The regression coefficients b are given by

$$b = W\left(P^{'}W\right)^{-1}q, \tag{3.29}$$

where $W = (w_1|w_2|...|w_{\ell_v})$, $P = (p_1|p_2|...|p_{\ell_v})$, $q^{'} = (q_1|q_2|...|q_{\ell_v})$. Finally, the prediction of a generic sample x^* is given by

$$y^* = x^*b. \tag{3.30}$$

3.3 Nonlinear regression methods

3.3.1 Kernel ridge regression (KRR)

Kernel ridge regression (KRR) is a nonlinear regression method, which exploits the so-called kernel trick, namely a nonlinear kernel function applied in the original space for defining an inner product in a transformed space of higher dimensionality $k(x,z) =< \phi(x)\cdot\phi(z) >$ (x and z are two generic points of the original space, and $\phi(x)$ and $\phi(z)$ their respective transform) [107]. The (primal) problem can be written as follows:

$$\begin{aligned} &\text{minimise } \lambda\left\|w\right\|^2 + \sum_{i=1}^{l}\xi_i^2, \\ &\text{subject to } y_i - < w\cdot\phi(x_i) >= \xi_i,\ i = 1,...,l. \end{aligned} \tag{3.31}$$

from which the following Lagrangian can be derived:

$$L(w,\xi,\alpha) = \lambda\left\|w\right\|^2 + \sum_{i=1}^{l}\xi_i^2 + \sum_{i=1}^{l}\alpha_i(y_i - < w\cdot\phi(x_i) > -\xi_i). \tag{3.32}$$

Differentiating and imposing stationarity, we obtain:

$$\begin{aligned} w &= \tfrac{1}{2\lambda}\sum_{i=1}^{l}\alpha_i\phi(x_i), \\ \xi_i &= \tfrac{\alpha_i}{2}. \end{aligned} \tag{3.33}$$

The reformulation of Equation (3.32) leads to $\alpha = 2\lambda(K + \lambda I)^{-1}y$ and to the following final KRR prediction function:

$$f(x) = k^{'}(K^{'} + \lambda I)^{-1}y. \tag{3.34}$$

where K is the so-called Gram matrix with entries $K_{ij} =< \phi(x_i)\cdot\phi(x_j) >$, and k is a vector with

entries $K_i =< \phi(x_i) \cdot \phi(x_i) >$ with $i = 1, ..., n$. n stands for the number of training samples. I is the identity matrix and λ the ridge parameter, $\lambda \in \Re, \lambda \geq 0$. This last controls the degree of regularization of the regression function. In the implementation of KRR, a typical choice of kernel function is the Gaussian kernel (or radial basis function), which expresses the components of the kernel matrix as follows [119, 120]:

$$K_{ij} = K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right), \tag{3.35}$$

where $\sigma > 0$ represents the kernel width. The best value of both σ and λ can be empirically estimated in different ways. In this work, we will adopt the well-known k-fold cross-validation procedure.

3.3.2 Support vector regression (SVR)

In the SVM regression approach [106, 107], the goal is to find a function $f(x)$ that has at most ε deviation from the desired targets y_i and, at the same time, is as smooth as possible. This is obtained by mapping the data from the original d-dimensional domain to a higher dimensional feature space, i.e., $\Phi(x) \in \Re^{d'}(d' > d)$ both in order to increase the flatness of the function and to approximate it in a linear way as follows:

$$f(x) = w.\Phi(x) + b. \tag{3.36}$$

The optimal linear function in the higher dimensional feature space is the one that minimizes the cost function defined as:

$$\Psi(w, \xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*). \tag{3.37}$$

This cost function minimization is subject to the following constraints:

$$\begin{cases} y_i - (w.\Phi(x_i) + b) \leq \varepsilon + \xi_i \\ (w.\Phi(x_i) + b) - y_i \leq \varepsilon + \xi_i^* \end{cases} \quad i = 1, 2, ..., N \tag{3.38}$$

and

$$\xi_i, \xi_i^* \geq 0, \ i = 1, 2, ..., N, \tag{3.39}$$

where the ξ_i and ξ_i^* are the slack variables introduced to account for samples that do not lie

in the ε-deviation tube. Constant C represents a regularization parameter. The formulation of the error function is equivalent to dealing with a so-called ε-insensitive loss function typically defined as [121] :

$$|\xi|_\varepsilon = \begin{cases} 0, & if\ |\delta| \leq \varepsilon \\ |\delta| - \varepsilon, & otherwise \end{cases} \tag{3.40}$$

where δ represents the deviation with respect to the desired target. This means that the differences between the targets and the estimated values are tolerated inside the ε-tube (error smallest than ε), while a linear penalty is assigned to estimates lying outside the ε-insensitive tube (see Figure 3.2).

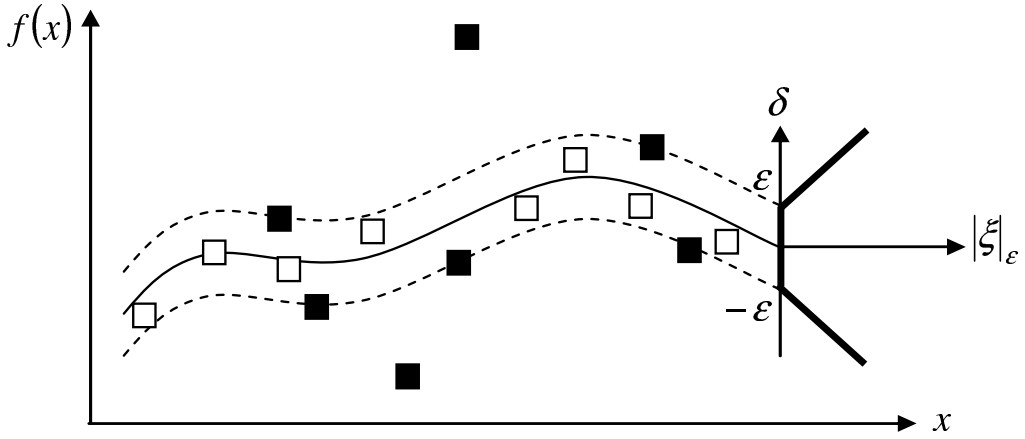


Figure 3.2: Example of ε-insensitive tube and error function used in the SVM-based regression technique. Filled squares data are support vectors. Hence, SVs can appear only on the tube boundary or outside the tube.

The above optimization problem can be reformulated through a Lagrange functional. The Lagrange multipliers can be found by a dual optimization leading to a QP solution [106, 107]. We may state the dual problem for nonlinear regression using support vector machine as follows: given the training set $\{(x_i, y_i)\}_{i=1}^N$, find the Lagrange multipliers $\{\alpha_i\}_{i=1}^N$ and $\{\alpha_i^*\}_{i=1}^N$ that maximize the objective function:

$$Q(\alpha_i, \alpha_i^*) = \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) - \varepsilon \sum_{i=1}^{N} y_i(\alpha_i + \alpha_i^*) - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) K(x_i, x_j), \tag{3.41}$$

subject to the following constraints:

$$\begin{aligned} &\sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0 \\ &0 \leq \alpha_i \leq C, i = 1, 2, ..., N \\ &0 \leq \alpha_i^* \leq C, \end{aligned} \tag{3.42}$$

where C is a user-specified constant. The final result is a function of the data conveniently expressed in the original (lower) dimensional feature space as:

$$f(x) = \sum_{i \in S} (\alpha_i - \alpha_i^*) K(x_i, x) + b^*, \tag{3.43}$$

where K is a kernel function. S is the subset of indices ($i = 1, 2, ..., N$) corresponding to the nonzero Lagrange multipliers. The Lagrange multipliers weight each training sample according to its importance in determining a solution. The training samples associated to nonzero weights are called support vectors. In S, margin support vectors that lie within the ε-insensitive tube and non-margin support vectors that correspond to errors coexist. The kernel must satisfy the condition imposed by the Mercer's theorem so that it can correspond to an inner product in the transformed (higher) dimensional feature space. Examples of common kernels that fulfill Mercer's condition are the linear kernel, the polynomial kernel functions, the Gaussian radial basis functions, and the hyperbolic tangent kernel [106, 107].

3.3.3 Radial basis function neural network (RBFN)

Radial-Basis Function Networks (RBFN) can be used for a wide range of applications, primarily because they can approximate any function and their training is faster compared to Multilayer Perceptrons (MLP). This fast learning speed comes from the fact that RBFN have just two layers (see Figure 3.3) of parameters (centers, widths and weights). The RBFN is based on the idea of approximating a function $F(x)$ through a linear combination of radial basis functions Ψ [99, 122–125]:

$$\hat{F}(x) = \sum_{j=1}^{P} \lambda_j \Psi_j(\|x - c_j\|), \tag{3.44}$$

where P, λ_j , and c_j are the number, the weight and the center (prototype) of the radial functions, respectively. A typical choice for the radial basis function is the Gaussian kernel:

$$\Psi_j \left(\|x - c_j\| \right) = \exp\left(-\frac{1}{2} \left(\frac{\|x - c_j\|}{\sigma_j} \right)^2 \right), \tag{3.45}$$

where σ_j is the width parameter of the j^{th} hidden unit (basis function) of the hidden layer.

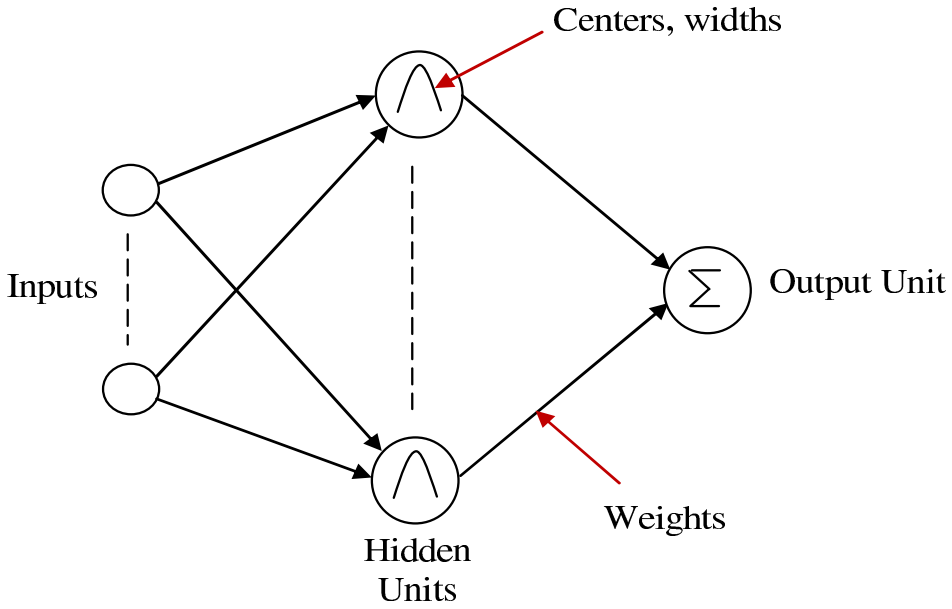


Figure 3.3: Architecture of a Radial Basis Function Neural Network (RBFN).

For a given RBFN architecture based on the Gaussian kernel (i.e., for a fixed value of P), the training algorithm consists in finding the parameters λ_j, c_j and σ_j such that $\hat{F}(x)$ fits the desired function $F(x)$ as best as possible. Since $F(x)$ is unknown, the goodness of fit is measured empirically by means of the available training samples. Briefly, the training of the hidden layer, which is equivalent to the computation of the kernel parameters (c_j and σ_j), is performed by applying the k-means clustering algorithm (with $k = P$) to the available training samples. In this work, for the sake of simplicity, we will assume that all kernel functions have the same width ($\sigma_j = \sigma$) [126]. The training of the output layer (i.e., the estimation of the λ_j parameter) is accomplished by formulating the estimation problem as a linear system of equations solved according to the pseudo-inverse technique.

3.4 Conclusion

In this chapter, we have highlighted the basic concept regarding linear and nonlinear methods. Some approaches are based on mathematical models that relate the concentration of the studied parameters to the measures acquired by the spectroscopy or wind speed. Other approaches are based on the use of regression methodologies that estimate parameter concentration/speed on the basis of interpolation techniques applied to a set of available training samples.
Various alternative regression procedures have been described for the analysis of data in which the predictor variables are highly correlated such as principal component regression, partial least squares regression and ridge regression.

Among the nonlinear methods, neural networks represent a promising tool for an accurate evaluation of parameter values. In this research we have chosen to use Radial Basis Functions Networks (RBFN), because they can be used for a wide range of applications and their training is faster compared to the popular multilayer perceptrons (MLP). This fast speed comes from the fact that RBFN have just two layers of parameters which can be determined sequentially. RBFN allows the modelling of nonlinear data using a linear approach, which is therefore fast, with the additional benefit of avoiding the problem of local minima usually encountered when using MLP.

Another more recent and promising regression approach based on support vector machines (SVMs). It is noteworthy that RBFN and SVM are two regression models sharing common properties. In particular, the regression model is expressed as a linear combination of kernel distances. Their main difference relies on the way the combination weights are inferred from the training samples. While RBFN are derived from a minimum empirical risk principle, SVM take origin from a structural risk minimization which in theory makes them less subject to overfitting problems.

Among several kernel-based approaches, kernel ridge regression (KRR), a kernel version of the ridge regression (RR), makes it possible to perform a sparse nonlinear regression by constructing a linear regression function in feature space.

The different regression methods presented in this chapter used in all the following chapters. There are several questions about these methods of single regression. The performance of some single regressors, the behavior of the systematic residuals influence by the accuracy improvement and the use of the correction of regressors are all major subject, that will be addressed in the next chapter.

Chapter 4

Residual Correction Concept for Spectroscopic Data Sets Regression

Contents

4.1 Introduction

Near infrared spectroscopy (NIR) is widely used in food and pharmaceutical industries for analysis and quality control. In reflection-based NIR spectroscopy, NIR radiation is guided into the product, and some of the backscattered radiation is captured and related to variables of interest via chemometric techniques. As the backscattered radiation spectrum is affected by both the scattering and absorption properties of the product, it provides information about its physical structure as well as its chemical composition. From the NIR spectrum, quantitative and qualitative information can thus be obtained with regression and classification models, respectively [127–129].

From a methodological point of view, the problem of concentration estimation can be viewed as an inverse modeling issue in which it is necessary to define a model that relates the acquired observations to the concentration of interest. Typically, the choice of the regression algorithm depends on the statistical distribution of the data under study and the related noise, which have a direct impact on its prediction performance [120, 127, 130]. In the literature, two main approaches of regression have been proposed. The first is based on linear models, while the second approach makes use of nonlinear models.

In this chapter, we propose a two-stage regression approach that is based on a residual-based correction (RBC) concept. Its underlying idea is to correct any adopted regressor, called functional estimator, by analyzing and modeling its residual errors directly in the feature space. RBC is therefore not a regressor but a correction method, whose aim is not to reach the best achievable accuracy for a given data set but to possibly improve the estimation model of a given (poor or accurate) regressor. Experimental results based on PLSR, SVM and RBFN regression methods and conducted on two different data sets in infrared spectroscopy point out that the proposed approach can improve the estimation accuracy achieved by traditional regression or by adaptive boosting. The work was published in [39].

The remaining part of the chapter is organized as follows. In section 4.2, we present briefly the adaptive boosting technique for regression. Section 4.3 describes the proposed residual-based correction approach. Section 4.4 reports the experimental results obtained on two infrared spectroscopy data sets. Finally, Section 4.5 draws the advantages and drawbacks of the present method.

4.2 Adaptive boosting for regression (AdaBoost.R)

The method proposed in this work relies thus on an opportune exploitation of the errors generated by a given regressor. Another way to exploit estimation errors can be found implemented in the popular AdaBoost.R (adaptive boosting for regression) which is based on the boosting principle [131]. Other variants derived from the boosting principle are available in [132–139]. In the following, we describe briefly this conceptually interesting alternative [140].

First, all samples in the original training set are allocated with equal weights $w_i^{(1)} = 1, i = 1, ..., N$. Then, for each calculation cycle $t = 1, ..., T$ (T being the total number of iterations), the following steps are performed:

Algorithm 1 AdaBoost.R

1: For each sample, a probability value is computed as follows:

$$f_i^{(t)} = \frac{w_i^{(t)}}{\sum_{i=1}^{N} w_i^{(t)}}, i = 1, ..., N. \tag{4.1}$$

According to the resulting sampling probability distribution $f^{(t)}$, N samples are picked from the original training set to yield the so-called boosting set. Note that samples with high probability values may appear several times in the boosting set.

2: Train a regressor $R^{(t)}$ on the current boosting set and generate the estimates for the original training samples $y_i^{(t)}, i = 1, ..., N$.

3: Compute a loss value for each sample of the original training set with one of the following three loss functions:

$$L_i^{(t)} = \frac{|\hat{y}_i^{(t)} - y_i|}{\underset{i}{max}\{|\hat{y}_i^{(t)} - y_i|\}}, \text{(Linear)} \tag{4.2}$$

$$L_i^{(t)} = \frac{|\hat{y}_i^{(t)} - y_i|^2}{\underset{i}{max}\{|\hat{y}_i^{(t)} - y_i|^2\}}, \text{(Square)} \tag{4.3}$$

$$L_i^{(t)} = 1 - exp\left[-\frac{|\hat{y}_i^{(t)} - y_i|}{\underset{i}{max}\{|\hat{y}_i^{(t)} - y_i|\}}\right], \text{(Exponential)} \tag{4.4}$$

4: Calculate the mean loss:

$$\bar{L}^{(t)} = \sum_{i=1}^{N} L_i^{(t)} f_i^{(t)}. \tag{4.5}$$

5: Let $\beta_t = \frac{\bar{L}^{(t)}}{1-\bar{L}^{(t)}}$ and update the weight of each sample in the original training set by:

$$w_i^{(t+1)} = w_i^{(t)} \beta_t^{(1-L_i^{(t)})}. \tag{4.6}$$

6: If $t \leq T$, go to ***Step* 1**. At the end of the training process, each regression model $R^{(t)}(t = 1, ..., T)$ given by an unknown sample x will provide a prediction $y^{(t)}$. The resulting T predictions are combined as follows to obtain a final prediction.

(***Step* A**) Sort the T predictions $y^{(t)}(t = 1, ..., T)$ in increasing order to get $y^{(n_1)} \leq y^{(n_2)} \leq ... \leq y^{(n_J)}$, where $n_j(j = 1, ..., J; J = T)$ is a permutation of the original cycle number t.

(***Step* B**) Take the sum of $log(1/\beta_{n_j})$ over j from the smallest term to the r-th term n_r, when the following inequality starts to be satisfied:

$$\sum_{j=1}^{r} log(1/\beta_{n_j}) \geq \frac{1}{2} \sum_{j=1}^{J} log(1/\beta_{n_j}). \tag{4.7}$$

(***Step* C**) Take the final prediction as the one yielded by the n_r-th regressor.

While, in our approach, a regressor is devoted to the modelling of the estimation errors to play the role of error corrector, in Adaboost.R the errors are exploited to build a cascade of regressors, where the last one is very likely to be trained only on difficult training samples. In other words, by moving in the cascade, the regressors become more and more specialized in handling difficult samples. The concepts and implementations of the two approaches are thus different though they take origin from the same idea, namely *exploit errors to improve the estimation process.*

4.3 Proposed residual regression

4.3.1 Description

Let us consider a set of N training samples $x_i (i = 1, 2, ..., N)$ represented in the d-dimensional measurement space $\Re^d$. Let us assume that a target $f(x_i) \in \Re (i = 1, 2, ..., N)$ is associated to each variable x_i.
Let us consider an estimator (called functional estimator) whose task is to provide an estimation model $\hat{f}(x)$ of the concentration of the chemical parameter of interest in the input feature space. No constraint is imposed on the choice of the functional estimator; it can be based on any kind of regression approach. The idea behind the proposed approach consists in exploiting the systematic residuals (i.e., errors) that are generated by the functional estimator for correcting it. For such purpose, a second estimator (called residual corrector) is trained to accomplish the task of estimating the residuals of the functional estimator. In other words, the residual estimator analyzes and models in the feature space the error function associated with the functional estimator and defined as:

$$e(x) = f(x) - \hat{f}(x). \tag{4.8}$$

The training phase related to the residual correction is depicted in Figure 4.1. It is worth recalling that, during the training phase, $x_i (i = 1, ..., N)$ represents a training sample for which $f(x_i)$ stands for its corresponding (known) target value while $\hat{f}(x_i)$ refers to the corresponding estimate generated by the functional estimator. The residual corrector is a regressor of the same kind of the functional estimator. This means that if, for instance, PLSR is adopted as regression method for the functional estimator, it is too for the residual corrector with the difference that the functional estimator is trained on sample pairs $(x_i, f(x_i))$ with $i = 1, 2, ..., N$, while the training of the residual corrector is performed on sample pairs $(x_i, e(x_i))$ with $i = 1, 2, ..., N$.
Once the training phase is finished, the correction (global estimation) phase takes place. Any unknown sample x is given in input to both the functional estimator and the residual corrector, the former providing an estimate of the chemical parameter concentration $\hat{f}(x)$ while the latter yielding an estimate of the corresponding error $\hat{e}(x)$. A simple addition of the contributions of the two estimators (see Figure 4.2) makes it possible to obtain a global estimation model

$\hat{F}(x)$ (of the parameter of interest) with a higher expected accuracy. This aspect is discussed in greater detail in the following sub-section.

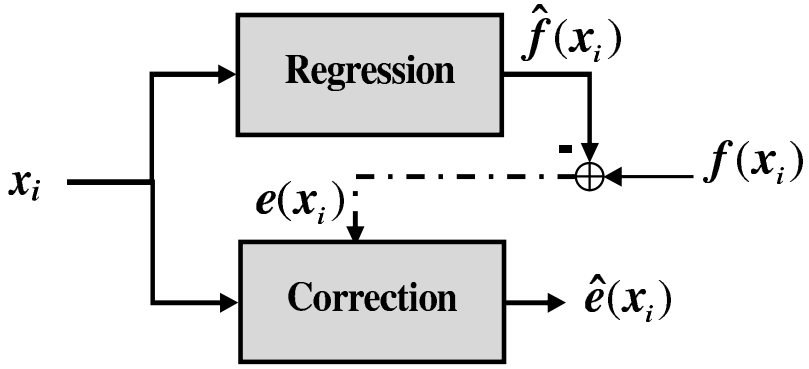


Figure 4.1: Block diagram illustrating the training phase of the residual correction.

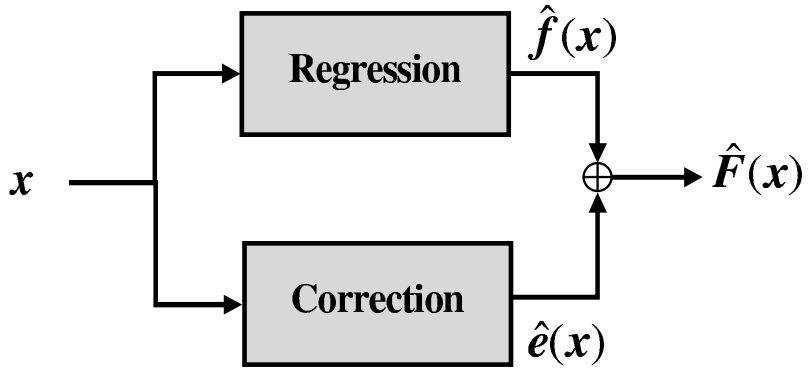


Figure 4.2: Block diagram of the proposed approach in the global estimation phase.

4.3.2 Theoretical considerations

Let us try to understand in which conditions the residual correction can help in improving the accuracy of the estimation. To do so, let us adopt as an accuracy measure the commonly used mean squared error (MSE). The MSE of the functional estimator (MSE_F) can be calculated in the following way:

$$MSE_F = E\left\{(f(x) - \hat{f}(x))^2\right\}. \tag{4.9}$$

Similarly, the MSE of the residual corrector (MSE_R) is defined as:

$$MSE_R = E\left\{(e(x) - \hat{e}(x))^2\right\}. \tag{4.10}$$

The MSE of the global estimation system (MSE_S) composed of the functional and residual corrector takes the following expression:

$$\begin{aligned} MSE_S &= E\left\{[f(x) - \hat{F}(x)]^2\right\} = E\left\{(e(x) - \hat{e}(x))^2\right\} \\ &\Rightarrow MSE_S = MSE_R. \end{aligned} \tag{4.11}$$

From Equation (4.11), it is interesting to point out that the accuracy of the global estimation system is independent on the one of the functional estimator. It only depends on the accuracy of the residual corrector. In other words, in order to obtain a system that is capable to improve the accuracy with respect to the functional estimator, it is necessary that:

$$MSE_R \leq MSE_F. \tag{4.12}$$

For the sake of simplicity, we will drop in the following the feature vector term x. The condition expressed in Equation (4.12) can be written as:

$$E\left\{(e - \hat{e})^2\right\} \leq E\left\{e^2\right\}. \tag{4.13}$$

After development, we get:

$$E\left\{\hat{e}^2\right\} \leq 2E\left\{e \cdot \hat{e}\right\}. \tag{4.14}$$

The residual estimate $\hat{e}$ can be written as:

$$\hat{e} = e + \Delta e, \tag{4.15}$$

where Δe represents the error incurred by the residual estimator. Such an error can be considered as a second-order residual. Thus, the condition in Equation (4.14) can be rewritten as:

$$E\left\{\hat{e}^2\right\} \leq 2E\left\{e^2\right\} + 2E\left\{e \cdot \Delta e\right\} \tag{4.16}$$

Let us assume that the second-order residual Δe is of zero-mean and let us adopt the worst case reasoning that is, we suppose that the first and second-order residuals, e and Δe respectively, are independent of each other. Such reasoning aims at understanding the limits of the proposed approach in the most unfavorable conditions since the correlation between the first and second-order residuals is neglected. Accordingly, the condition in Equation (4.16) can be simplified to:

$$E\left\{\hat{e}^2\right\} \leq 2E\left\{e^2\right\}. \tag{4.17}$$

This relationship means that, in order to improve the accuracy with respect to the functional estimator, it is enough to find a residual estimator characterized by a residual estimate power less than two times the actual residual power. Since such a condition is weakly constraining, it can be concluded that it is very likely to find and design a residual estimator that will permit to exploit the information contained in the systematic residuals of the functional estimator and, accordingly, to improve its accuracy. At the limit, if one adopts only a zero-order residual estimator such that:

$$\hat{e} = \bar{e} = E\{e\}, \tag{4.18}$$

the expected accuracy of the global estimator must necessarily be improved (unless the residuals are already with zero-mean) since the following condition is always verified:

$$\bar{e}^2 \leq E\left\{e^2\right\}. \tag{4.19}$$

4.4 Experimental results on spectroscopic data set

4.4.1 Experimental design

All results are given in terms of the normalized mean square error (NMSE) criterion achieved on the test set, which is expressed as:
without correction:

$$NMSE = \frac{\frac{1}{N_T}\sum_{i=1}^{N_T}\left(y_i - \hat{f}(x_i)\right)^2}{var(y)}, \tag{4.20}$$

with correction:

$$NMSE = \frac{\frac{1}{N_T}\sum_{i=1}^{N_T}\left(y_i - \hat{F}(x_i)\right)^2}{var(y)} = \frac{\frac{1}{N_T}\sum_{i=1}^{N_T}\left(e_i - \hat{e}_i\right)^2}{var(y)}, \tag{4.21}$$

where N_T is the number of test samples, $var(y)$ is the variance of the output values. This last, which plays the role of normalizing constant, is estimated on all available samples (i.e., on both training and test samples). Accuracy comparison will be expressed in terms of gain:

$$Gain\,[\%] = 100 \times \frac{NMSE(reference\ regressor) - NMSE(considered\ regressor)}{NMSE(reference\ regressor)}. \tag{4.22}$$

Another interesting criterion used in statistical tests is $t - test$. Van der Voet [141] proposed a randomization $t - test$ to compare the predictive accuracy of two models $\hat{f}(x_i)$ (without correction) and $\hat{F}(x_i)$ (with correction RBC) using the distribution of prediction errors. The

application of randomization $t-test$ to the prediction method comparison problem is defined as follows:

$$MSE_F = \frac{\sum_{i=1}^{N_T} e(x_i)^2}{N_T}, \tag{4.23}$$

$$MSE_S = \frac{\sum_{i=1}^{N_T} e_{RBC}(x_i)^2}{N_T}, \tag{4.24}$$

where $e(x_i)$ and $e_{RBC}(x_i)$ are the N_T-dimensional vectors of the predictive errors and the mean squared error of prediction given by:

$$e(x_i) = y_i - \hat{f}(x_i), \tag{4.25}$$

$$e_{RBC}(x_i) = y_i - \hat{F}(x_i). \tag{4.26}$$

A comparison between models without correction and with correction (RBC) is made using the test statistic (T):

$$d_i = e(x_i)^2 - e_{RBC}(x_i)^2, \tag{4.27}$$

$$T = MSE_F - MSE_S = \frac{\sum_{i=1}^{N_T} d_i}{N_T} = \bar{d}, \tag{4.28}$$

where d_i is the difference of predictive errors of dimension $N_T \times 1$ and $\bar{d}$ the mean of difference. The $t-test$ for one-sided alternative hypothesis $MSE_F > MSE_S$, the test proceeds as follows:

Algorithm 2 The $t-test$ for one-sided alternative hypothesis.

1: Calculate d_i.
2: Compute T, for the actual evaluation data ($T^{'}$).
3: Repeat Steps A and B m times:

   A: Fix random signs to d_i.

   B: Calculate $T = \bar{d}$.

4: Calculate the significance level $p = \frac{k}{m+1}$, where m is the number of randomization trials, and k is the rank of $T^{'}$ among the randomization values of T when ranked from decreasing order.

The main objective of the experiments was to assess thoroughly the proposed correction technique. For such purpose we conducted experiments: 1) by adopting a feature selection strategy

(namely, the Sequential Forward-Backward selection [99, 142] using k-fold cross-validation (k = 4) on the training set) so that to perform both the regression and correction tasks in the resulting feature subspace (performance is measured using k-fold cross-validation ($k = 4$) on the training set); and 2) by working directly in the original (hyperdimensional) input space. In all what follows, we assume that the choice of the best parameter is based on the minimum of the normalized mean square error k-fold cross-validation (NMSE CV).

Due to the nature of NIR spectra, the large baseline regions without chemical information and many non significant variables, variable selection can become necessary. In forward selection, the variables are added to the model one at a time. At each step, each variable that is not already in the model is tested for inclusion in the model. Forward selection has drawbacks, including the fact that addition of new variables may render one or more of the already included variables non significant. An alternate approach is backward selection where all the variables are included from the start. Then the bad features are then removed. With an initial subset in each iteration to which features are added/removed (Forward/Backward). On the both data sets, orange juice and tecator, the best results for forward selection is 19 and 12, respectively, as shown in Figure 4.3-(a) and Figure 4.3-(b). Forward/Backward selects 25 variables for the juice data set and 10 variables for the tecator (see Figures 4.3-(a) and 4.3-(b)).

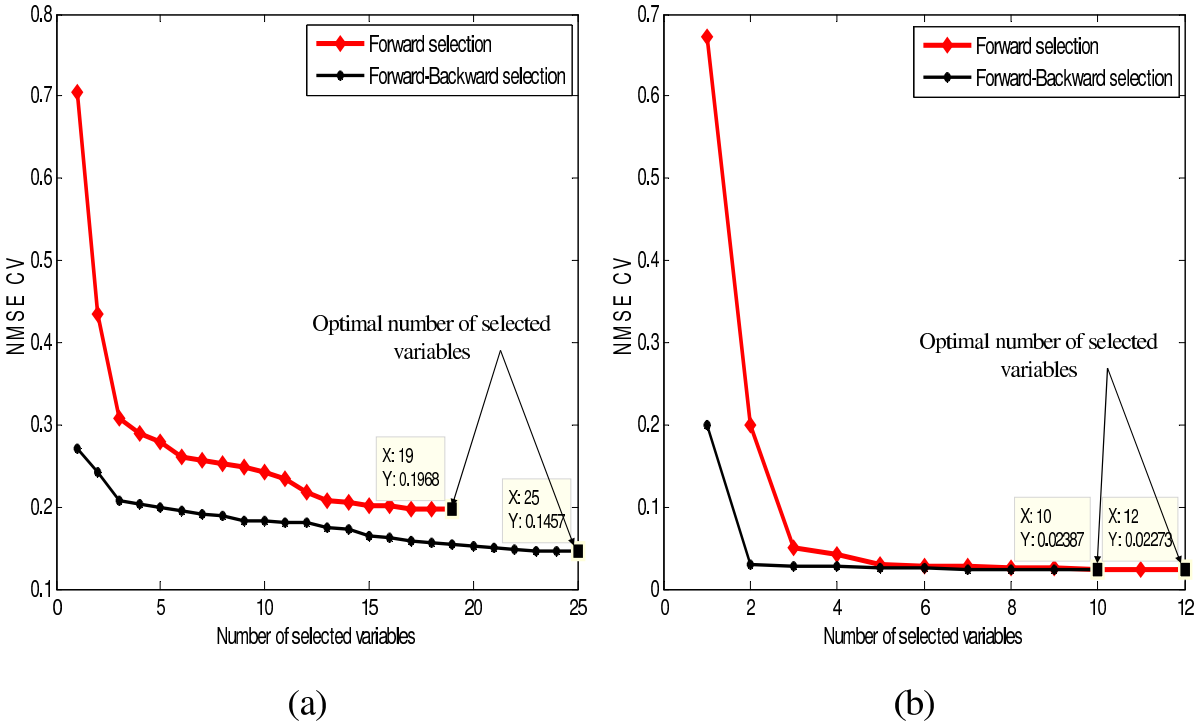


Figure 4.3: NMSE k-fold cross-validation with respect to the number of selected variables, (a) orange juice and (b) tecator data set.

Concerning the parameter setting of PLSR, the optimal number of latent variables was estimated in the range [1, 20]. Figures 4.4-(a) and 4.4-(b) show the evaluation of the normalized mean square error k-fold cross-validation (NMSE CV) according to the number of latent variable, on the orange juice and tecator data set, respectively. For the orange juice, the optimal number of latent variables is equal to 12. The same optimal number of latent variables in the case of the tecator data set.

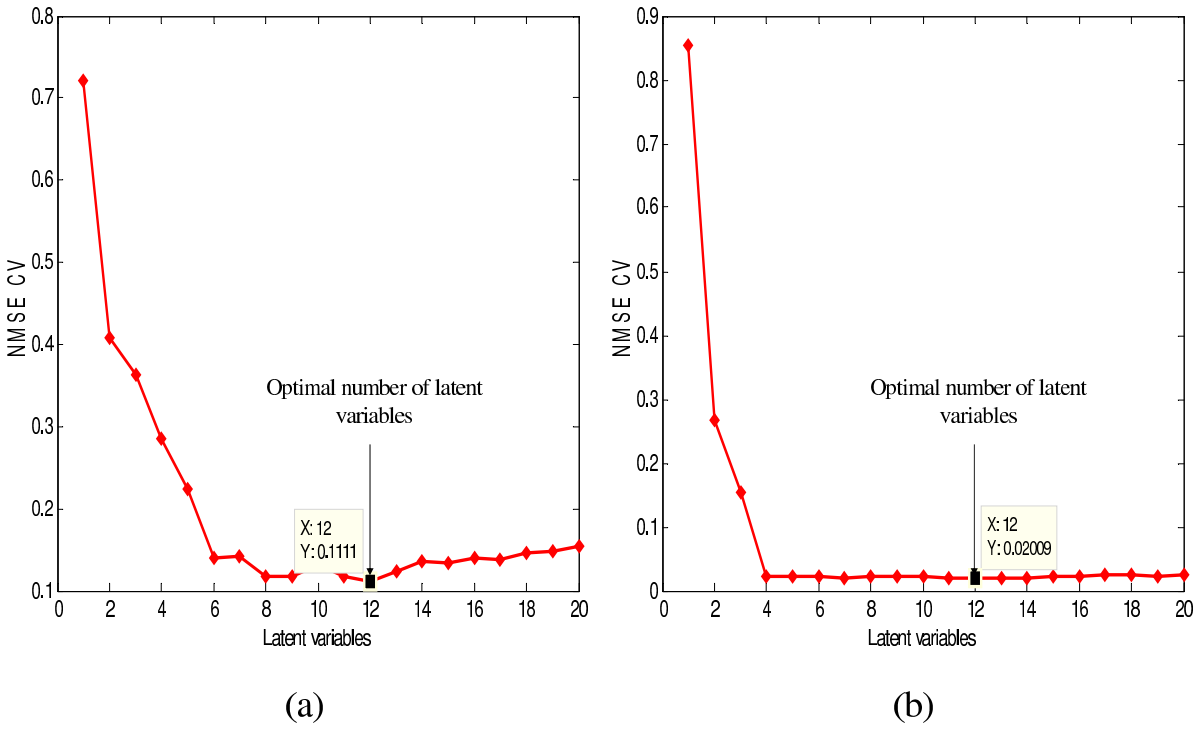


Figure 4.4: NMSE k-fold cross-validation with respect to the number of ℓ_v latent variables, (a) orange juice and (b) tecator data set.

For both experimental scenarios, we considered two different kinds of regressors: i) a nonlinear SVM based on the Gaussian kernel function (SVM-RBF), and ii) a nonlinear RBFN. For the SVM-RBF, it was necessary to set three parameters, i.e., the width parameter of the Gaussian kernel (γ), the regularization parameter (C), and the size of the insensitive loss tube (ε). These parameters were adjusted empirically so as to minimize the prediction error using k-fold cross-validation (k=4).

In particular, C, γ and ε were varied from 10^{-4} to 10^{3} , from 10^{-4} to 10^{3} and from 10^{-4} to 10^{-1}, respectively. For the RBFN, as mentioned above, the widths of the radial basis functions were chosen the same for all hidden units ($\sigma_j = \sigma$). Another parameter to set is the number of hidden units (P). We carried out experiments varying the value of this parameter from 2 to 40, while the width (σ) of the radial functions was varied in the range 10^{-3} to 50. The best

values of these two parameters were obtained through k-fold cross-validation (k=4). Finally, in order to complete our experimental study, the results of the RBC are compared with those of AdaBoost.R.

4.4.2 Results with RBC

In this section, we highlight the results obtained by usual regression and with RBC on the orange juice data set by considering the two scenarios (with and without features selection, respectively) denoted by the words "Subset" and "All", respectively. The first experiment using RBFN-subset gives a value of NMSE equal to 0.1611 with regression and to 0.1574 with RBC. In this experiment the best number of selected spectral variables (by using the Sequential Forward-Backward selection) was 25 spectral variables for the two cases, with regression and with RBC, respectively. The best architecture of the RBFN network was 19 and 15 hidden units, respectively, and σ was equal to 11.88 and 0.008 with regression and with RBC, respectively, the optimization of the parameters of RBFN with subset features selection is shown in Figures 4.5-(a) and 4.5-(b).

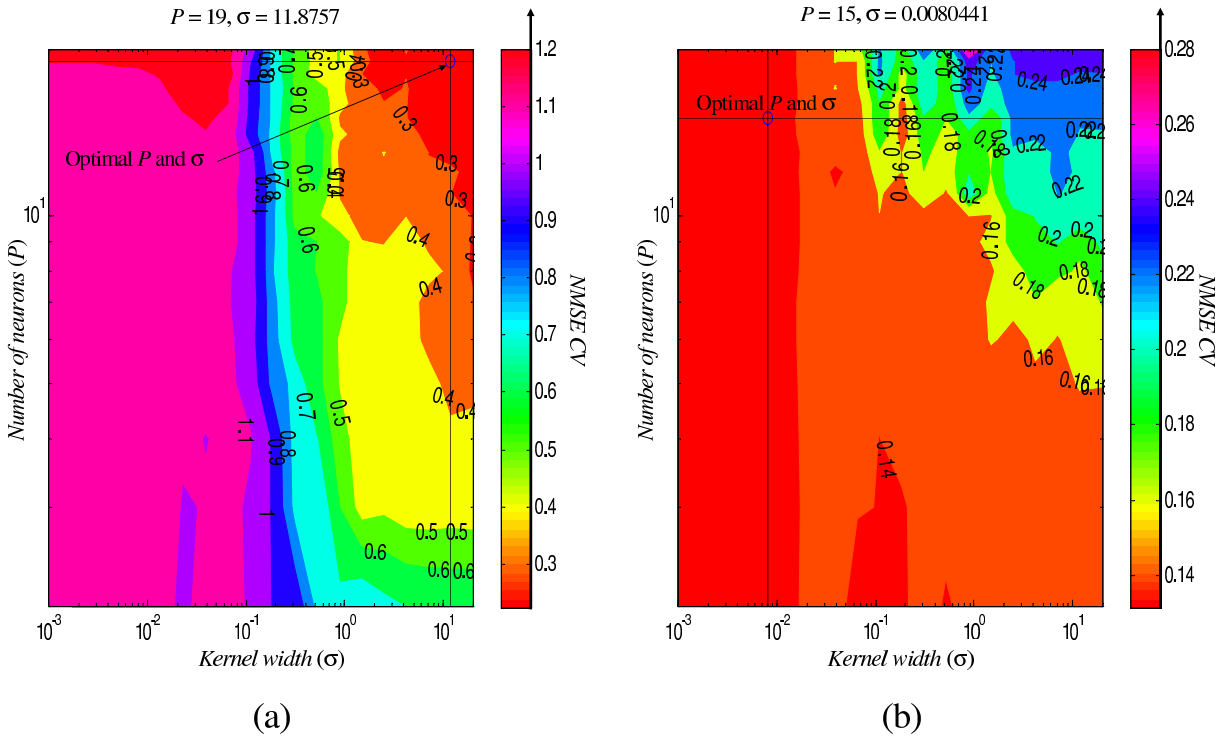


Figure 4.5: RBFN-Subset optimization number of neurons in hidden layer (P) and the width parameter of the Gaussian kernel (σ) for the orange juice data set, (a) Regression and (b) Correction.

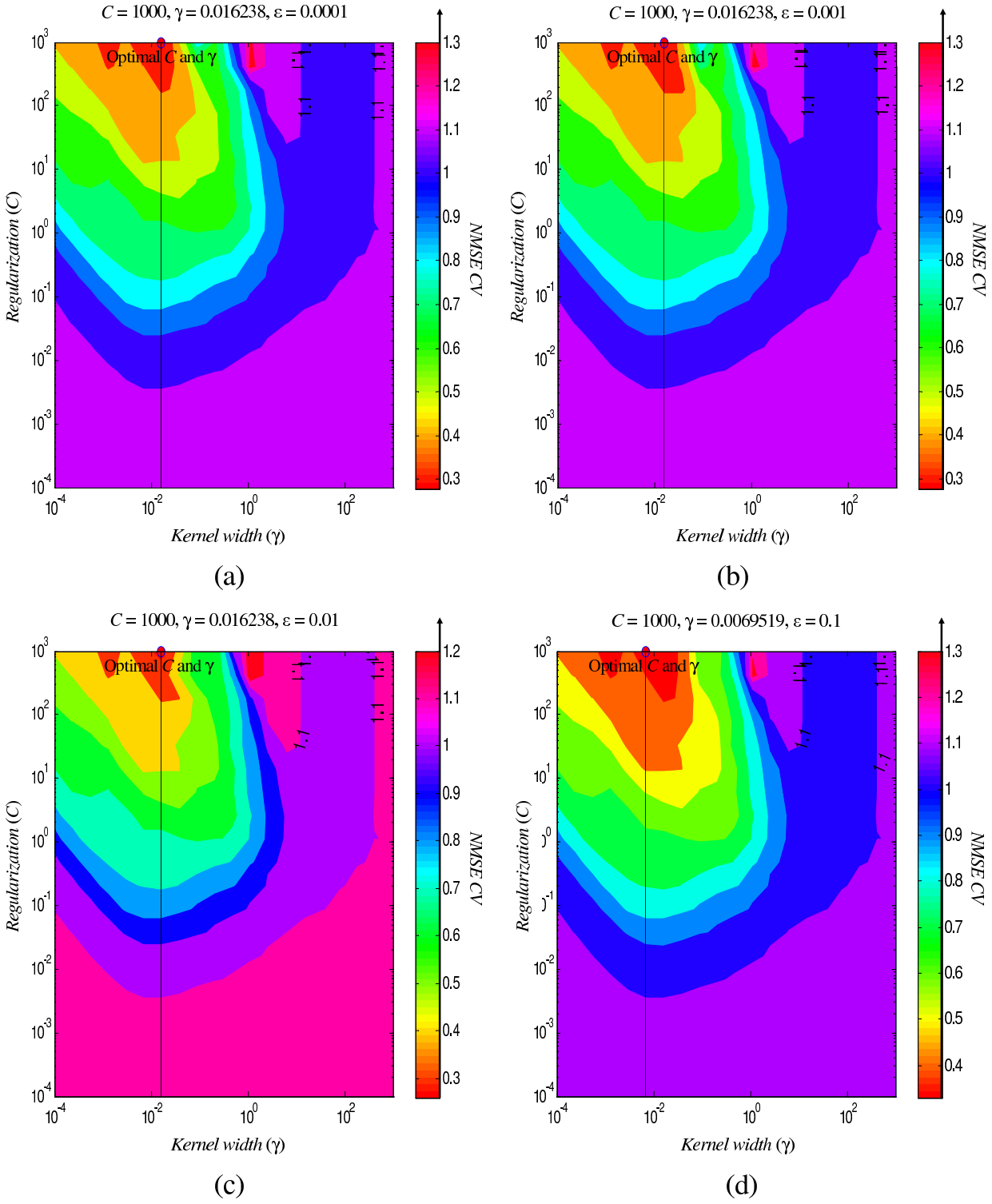


Figure 4.6: SVM-Subset optimization of parameters C, γ and ε for the orange juice data set in the case of Regression. (a) $\varepsilon = 0.0001$, (b) $\varepsilon = 0.001$, (c) $\varepsilon = 0.01$, and (d) $\varepsilon = 0.1$.

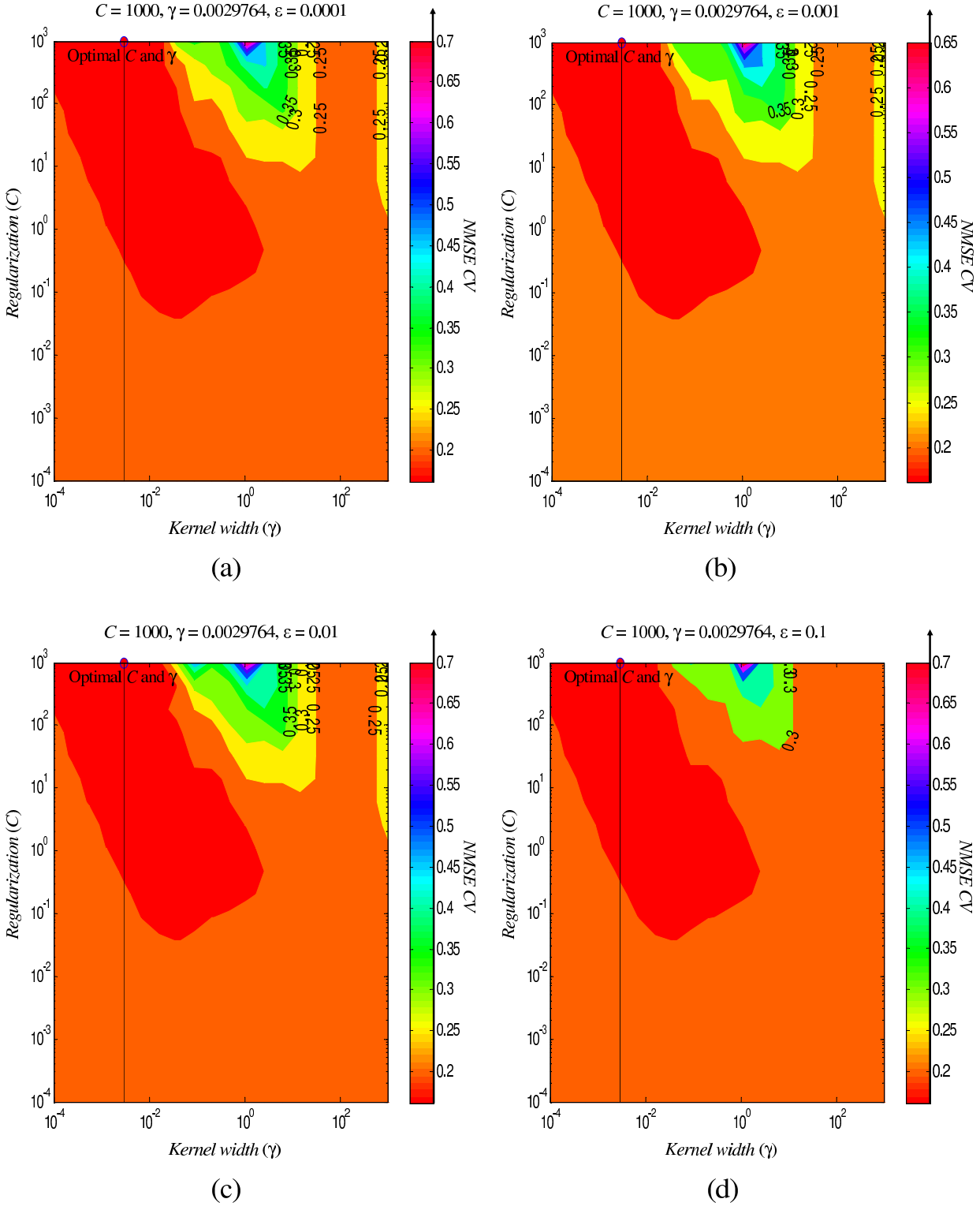


Figure 4.7: SVM-Subset optimization of parameters C, γ and ε for the orange juice data set in the case of Correction. (a) $\varepsilon = 0.0001$, (b) $\varepsilon = 0.001$, (c) $\varepsilon = 0.01$, and (d) $\varepsilon = 0.1$.

The second experiment shows the results obtained with the second nonlinear regression method used in this work, namely SVM-RBF. By considering the SVM-RBF parameters C, γ and ε, we variate C and γ for fixed values of ε ($10^{-4}, 10^{-3}, 10^{-2}$ and 10^{-1}), we calculate NMSE CV for each case and we retain the best values for which NMSE CV is minimal. The set of results for single regression (without correction) and with RBC (with correction) approach are reported in Figures 4.6 and 4.7, respectively. The NMSE is equal 0.3191 with regression and to 0.2497 with RBC. Also, in this experiment the number of selected spectral variables by using the forward selection is 25 for both cases. The optimal parameters C, γ and ε were equal to 10^3, 0.0162, 0.01 and 10^3, 0.003, 0.1 for regression and RBC, respectively.

In the case of using all spectral variables, the best model (RBFN-All) obtained on the test set corresponds to an error value of 0.1255 and 0.1212 for regression and RBC, respectively. The correction model used in this experiment was based on a RBFN. It was characterized by 13 and 3 neuron in the hidden layer and their corresponding width σ was equal to 50 and 0.0941, respectively, as indicated in Figure 4.8.

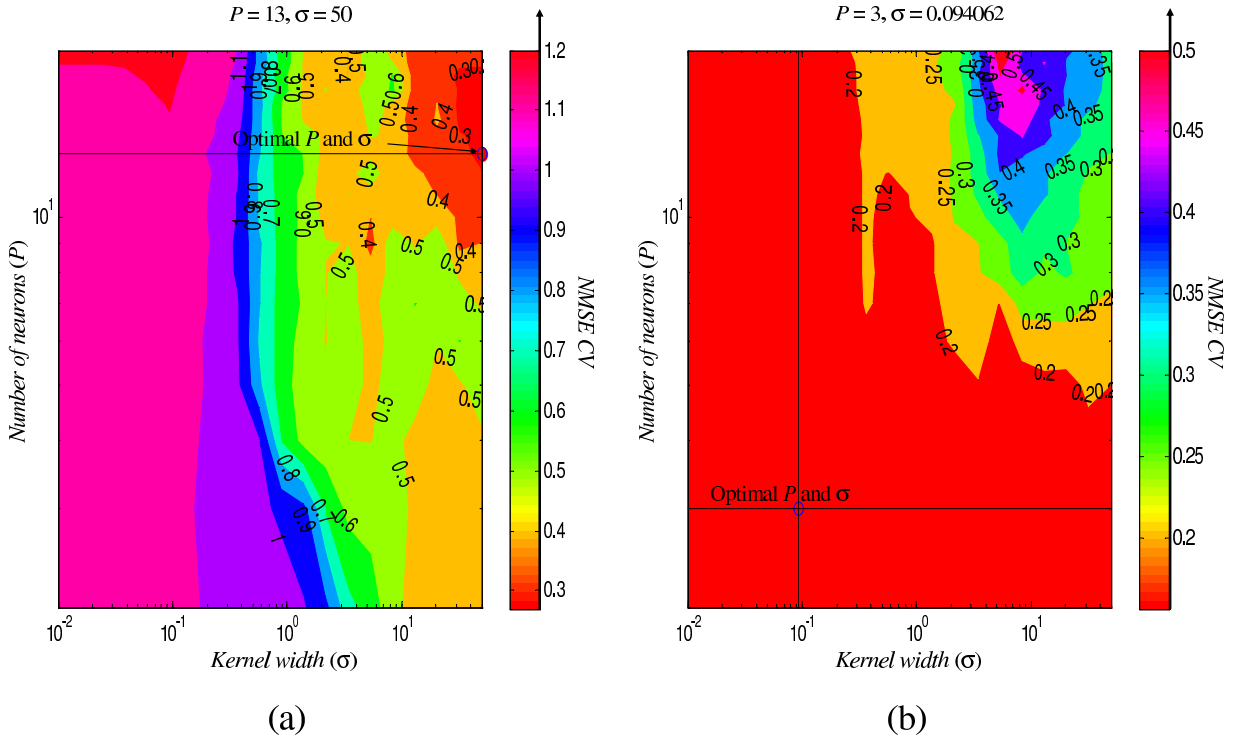


Figure 4.8: RBFN-All optimization number of neurons in hidden layer (P) and the width parameter of the Gaussian kernel (σ) for the orange juice data set, (a) Regression and (b) Correction.

For the last experiment, concerning the orange juice data set using all spectral variables, the

smallest NMSE obtained by the SVM-RBF with regression and with RBC were 0.2488 and 0.2081, respectively. The optimization parameter is similar to the previous mentioned one for the SVM-RBF structure, where the results with single regression and with RBC are shown in Figures 4.9 and 4.10, respectively. The best values of the parameter C were 183.2981 and 10^3, the optimal kernel width parameter γ was found equal to 0.0013 and 0.0001, and ε to 0.01 and 0.1 for regression and RBC, respectively.

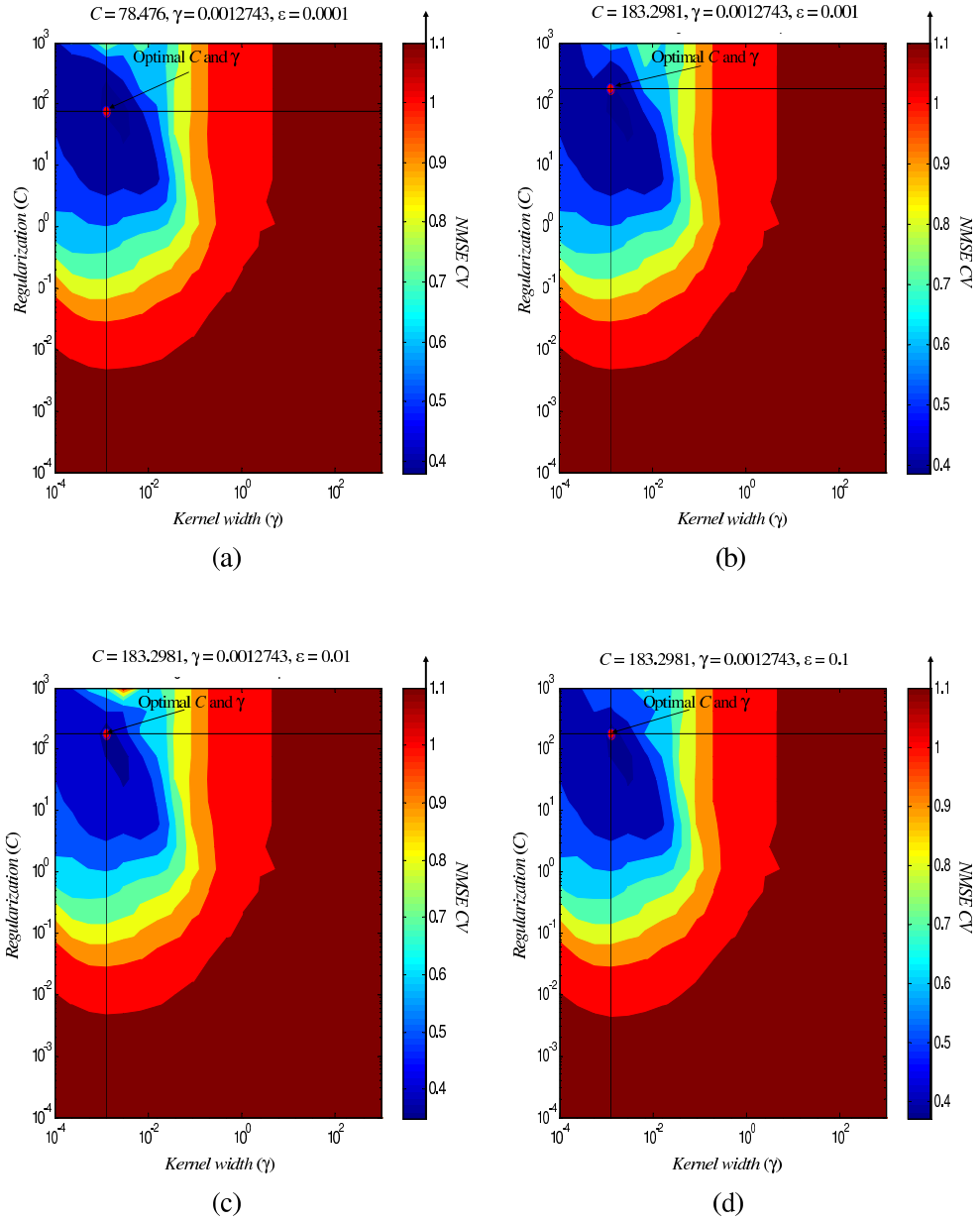


Figure 4.9: SVM-All optimization of parameters C, γ and ε for the orange juice data set in the case of Regression. (a) $\varepsilon = 0.0001$, (b) $\varepsilon = 0.001$, (c) $\varepsilon = 0.01$, and (d) $\varepsilon = 0.1$.

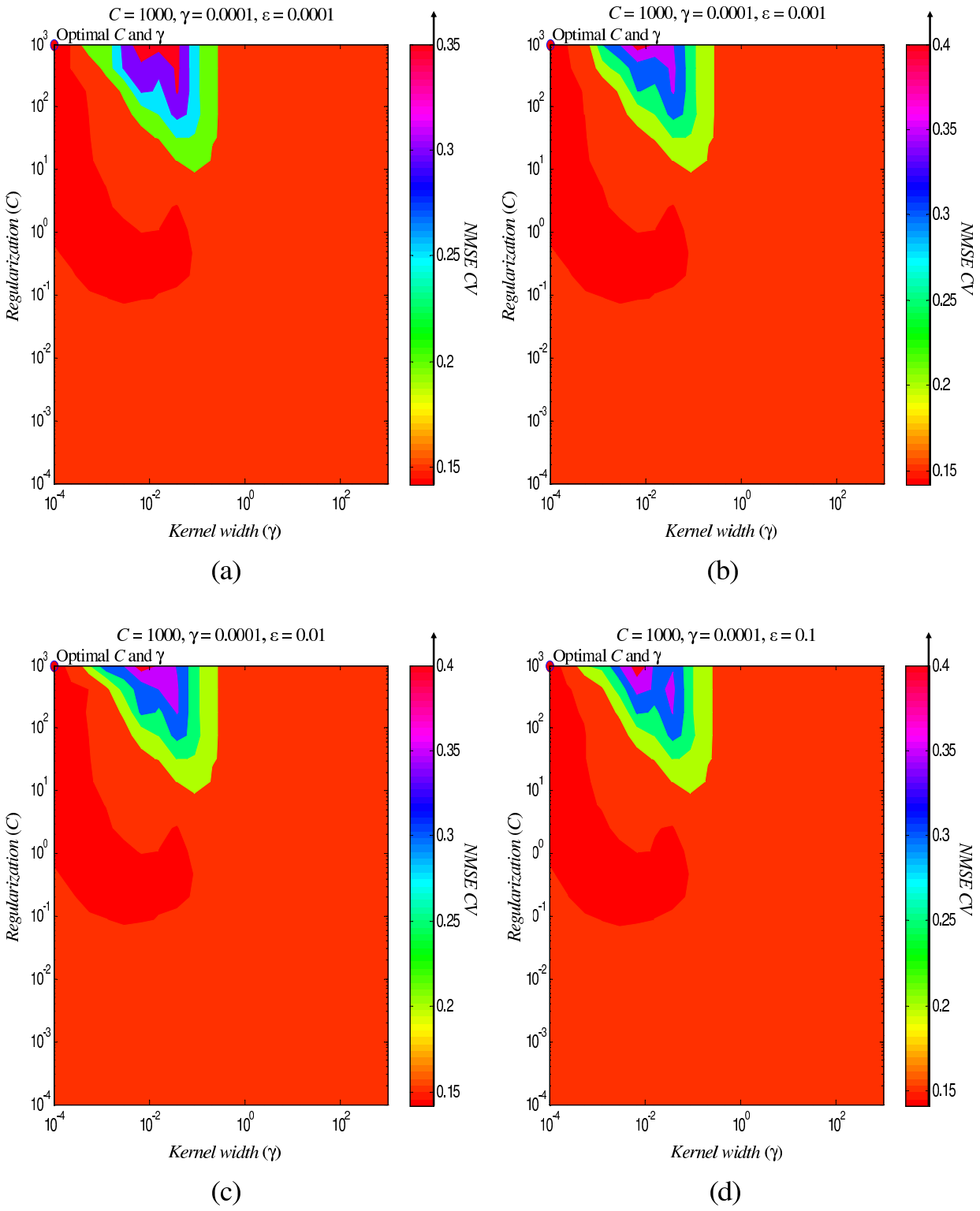


Figure 4.10: SVM-All optimization of parameters C, γ and ε for the orange juice data set in the case of Correction. (a) $\varepsilon = 0.0001$, (b) $\varepsilon = 0.001$, (c) $\varepsilon = 0.01$, and (d) $\varepsilon = 0.1$.

As shown in Table 4.1, for all regression strategies (PLSR, RBFN-Subset, SVM-Subset, RBFN-All and SVM-All), the correction-based (RBC) improves the accuracy, though not all the improvements are statistically significant as discussed in the next subsection. On an average, the proposed RBC approach provides an accuracy gain of the order of 8.96 %, with gains varying from 0.98 % to 21.75 %. The smallest improvement was achieved with PLSR (from 0.1626 to 0.1610).

Table 4.1: NMSE achieved on the orange juice data set by the five regression methods implemented without and with residual based correction (RBC)

Regression Method	**Features**	**Regression**		**RBC**		**Gain [%]**
		NMSET	**Time [s]**	**NMSET**	**Time [s]**	
PLSR	13	0.1626	0.03	0.1610	0.03	0.98
RBFN-Subset	25	0.1611	207	0.1574	200	2.30
SVM-Subset	25	0.3191	485	0.2497	479	21.75
RBFN-All	700	0.1255	351	0.1212	352	3.43
SVM-All	700	0.2488	1803	0.2081	1783	16.36

Similar experiment was performed on the other data set, namely the tecator data set. The obtained results are reported in Table 4.2. In brief, the achieved accuracy gains were equal to 14.79 %, 10.35 %, 29.27 %, 8.33 % and 32 % for the PLSR (12 features), RBFN-Subset (10 features), SVM-Subset (10 features), RBFN-All and SVM-All, respectively. On an average, for this data set, the gain is equal to 18.95 %.

Table 4.2: NMSE achieved on the tecator data set by the five regression methods implemented without and with residual based correction (RBC)

Regression Method	**Features**	**Regression**		**RBC**		**Gain [%]**
		NMSET	**Time [s]**	**NMSET**	**Time [s]**	
PLSR	12	0.0284	0.01	0.0242	0.01	14.79
RBFN-Subset	10	0.0029	302	0.0026	308	10.35
SVM-Subset	10	0.0041	473	0.0029	442	29.27
RBFN-All	100	0.0024	459	0.0022	468	8.33
SVM-All	100	0.0025	664	0.0017	649	32

4.4.3 Results with AdaBoost.R

As mentioned in the previous methodological section, AdaBoost.R represents a conceptually interesting alternative to our approach. For the sake of coherence, we thus assessed it on the two data sets and the five regression methods we considered in our experimental framework. Moreover, we tested the three different loss functions defined in Equations (4.1)-(4.3). Since all the boosting sets are constructed by drawing samples from the original training set according to the sampling probability distributions, the parameters optimized based on the original training set can be a good approximation to those for regressor modeling in every cycle [136]. Another parameter to tune in AdaBoost.R is the iteration number, which is critical since it may lead to overfitting problems. Indeed, a large number of iterations may create a very complex cascade of regressors, built in such a way to fit perfectly the training samples, i.e., to minimize the empirical risk. To contain the overfitting issue, it is therefore desirable to keep the number of iterations as small as possible [143]. In our case, the maximum number of iterations was fixed to 100 and the best iteration number was estimated by minimizing the NMSE on the training set. Figure 4.11 shows the behaviors of the Boosting (linear, square and exponential) for the PLSR with iterations versus the NMSE for orange juice data set. The optimal value of iteration number on this data set is 19, 3 and 15 for linear, square and exponential, respectively. Therefore, we repeated the experiments also on a Boosting of the same data set, obtained from (RBFN, SVM)-Subset and (RBFN, SVM)-All are shown in Figure 4.12. Similar experiments were performed on the second data set tecator. The results achieved on the test set are given in Table 4.3.

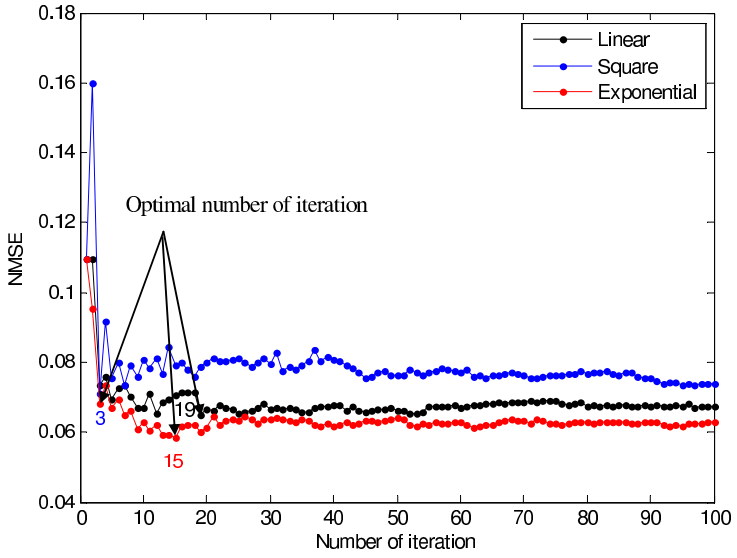


Figure 4.11: Behaviors of the Boosting-PLSR (linear, square and exponential) obtained by iterations for orange juice data set.

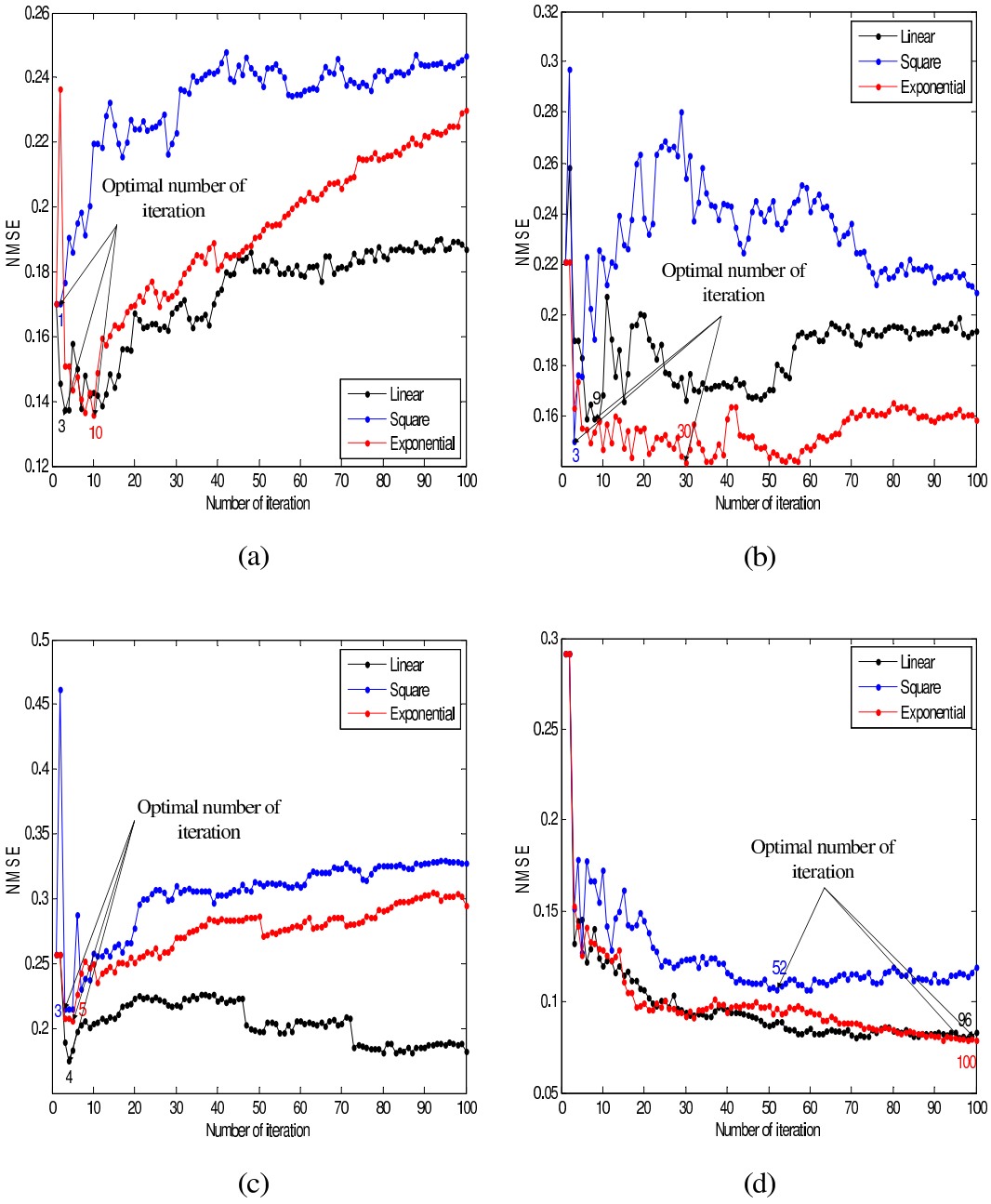


Figure 4.12: Behaviors of the Boosting (linear, square and exponential) obtained by iterations for subset orange juice data set. (a) Boosting-RBFN-Subset, (b) Boosting-SVM-Subset, (c) Boosting-RBFN-All, (d) Boosting-SVM-All.

Table 4.3: Results in terms of NMSE and processing time achieved by adaptive boosting regression (AdaBoost.R).

Regression Method	**Features**	**Loss Function**			**Time [s]**
		Linear	**Square**	**Exponential**	
Orange juice data set					
PLSR	13	0.2083	**0.1744**	0.2130	19.6
RBFN-Subset	25	0.1519	**0.1506**	0.2273	47.6
SVM-Subset	25	**0.2896**	0.2968	0.3226	96.9
RBFN-All	700	**0.1684**	0.1975	0.2256	186.9
SVM-All	700	**0.2593**	0.2736	0.2605	5494.8
Tecator data set					
PLSR	12	**0.0197**	0.0204	0.0212	5.7
RBFN-Subset	10	**0.0034**	0.0042	0.0045	50.4
SVM-Subset	10	**0.0027**	0.0038	0.0030	77.9
RBFN-All	100	0.0039	0.0048	**0.0035**	125.4
SVM-All	100	0.0031	0.0030	**0.0028**	386.8

These results point out how the choice of the loss function can be important. It actually depends on the density of noise characterizing the data (e.g., Gaussian, Laplacian), which a priori is not trivial to know. The linear loss function appears in general to be a good option.

For a direct comparison, we have gathered the results obtained by RBC and AdaBoost.R in Table 4.4. Moreover, for the sake of assessing the statistical significance of the prediction error decreases, we performed the statistical $t-test$. In particular, the $p-value$ associated with the simple regression and the best model among RBC and AdaBoost.R are reported (see Table 4.4). The error decrease is said statistically significant if the $p-value$ is less than the 5 % level. 6 cases (5 for RBC and 1 for Adaboost.R) among 10 cases (2 data sets and 5 regressors) exhibit a statistically significant decrease in the prediction error. Other 2 cases are close to the 5 % threshold. This confirms that our method can improve the results of the original functional estimator. When this is note situation, it does not worsen them in any case, making it more attractive than AdaBoost.R.

Table 4.4: Results in terms of NMSE and statistical test (*p-value*) achieved by the regression method, the proposed RBC approach and the best adaptive boosting regression method.

Regression Method	**Regression**	**RBC**	**Boosting**	***p-value***
Orange juice data set				
PLSR	0.1626	**0.1610**	0.1744	0.86
RBFN-Subset	0.1611	0.1574	**0.1506**	0.56
SVM-Subset	0.3191	**0.2497**	0.2896	**0.025**
RBFN-All	0.1255	**0.1212**	0.1684	0.115
SVM-All	0.2488	**0.2081**	0.2593	**0.015**
Tecator data set				
PLSR	0.0284	0.0242	**0.0197**	**0.03**
RBFN-Subset	0.0029	**0.0026**	0.0034	**0.02**
SVM-Subset	0.0041	0.0029	**0.0027**	0.08
RBFN-All	0.0024	**0.0022**	0.0035	**0.035**
SVM-All	0.0025	**0.0017**	0.0028	**0.005**

4.5 Conclusion

In this chapter, we have presented a regression approach that consists in correcting a given estimator by exploiting its systematic errors in the feature space. The rationale is that it is rather difficult to obtain a single estimator capable of providing high accuracies over the entire input space. This is due to the fact that any estimator provides usually an accuracy depending on the region of the input space to which the analyzed pattern belongs to. It is noteworthy that the proposed RBC is not a regressor but a correction method, whose aim is not to reach the best achievable accuracy for a given data set but to possibly improve the estimation model of a given (poor or accurate) regressor. The performance of RBC was assessed on the basis of two different chemometric data sets.
The following conclusions could be drawn from the obtained experimental results:

- It is shown that using a residual correction strategy can improve the estimation accuracy. Indeed, the achieved gains vary from 0.98 % to 32 % depending on the data set and the regression tool adopted. Not all these gains are statistically significant but, at the same time, no case of accuracy decrease has been observed.

- The proposed approach is independent from the kind of functional estimator used to approximate the chemical parameter of interest. Its general nature makes it applicable with any regression method.

- It is lowly sensitive to the optimization problem of the functional estimator, thus allowing a significant gain of time in the design of this latter. By contrast, a greater attention should be given in the choice and the design of the residual corrector.
- Compared with AdaBoost.R, RBC performs generally better despite it just involves an additional regressor in the system while AdaBoost.R makes use of a relatively large number of regressors. Its original way to exploit the estimation errors makes it less exposed to the overfitting risk with respect to AdaBoost.R.

Machine learning methods exhibit the attractive advantage that they can provide very accurate predictors. However, their accuracy depends on the quality and the quantity of samples used to train the considered predictor. A solution to the training sample scarcity is given by active learning approaches. The next chapter originates from a study on the use of active learning to spectroscopy and wind speed.

Chapter 5

Active Learning Methods

Contents

5.1 Introduction

In this chapter, we investigate a new strategy of the active learning approach for regression applied to spectroscopy/wind-speed prediction. The idea in our proposed method is to minimize the error of the prediction for concentration/speed in such a way as to minimize the quantity of training samples used, and thus to reduce the costs related to the training sample collection. For this reason, we propose to select the most significant sample among a large number of training samples by using active learning for regression problem. To solve this problem, we propose three general active learning strategies (i.e., applicable with any regression method) and one method specifically for support vector regression. The work was published in [59–61].
The first strategy uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors. The second one relies on the idea to add samples that are distant from the available training samples, while the third strategy is based on the selection of samples which exhibit a high expected prediction error. For support vector regression, a specific strategy based on the selection of the samples distant from the current support vectors is proposed. These strategies are tested on different linear and nonlinear methods, namely PLSR, RR, KRR and SVR.
To illustrate the capabilities of the proposed strategies, we conduct an experimental study in different fields. In particular, two main fields have been considered, namely chemometrics and wind speed.
The remaining part of the chapter is organize as follows. In Section 5.2, we introduce the general criterion for active learning. In Section 5.3, the four active learning strategies proposed in this chapter are described. Section 5.4 presents the experiments results on Spectroscopic and wind speed prediction data sets. Finally, conclusions are drawn in Section 5.5.

5.2 Active learning

Active learning is a subfield of machine learning and, more generally, artificial intelligence. The key hypothesis is that if the learning algorithm is allowed to choose the data from which it learns to be "curious", if you will it will perform better with less training. Why is this a desirable property for learning algorithms to have? Consider that, for any supervised learning system to perform well, it must often be trained on hundreds (even thousands) of labeled instances. Sometimes these labels come at little or no cost, but for many other more sophisticated supervised learning tasks, labeled instances are very difficult, time consuming, or expensive to obtain [46]. For examples, in speech recognition, information extraction, classification and filtering, near infrared spectroscopy, wind speed.
Active learning systems attempt to overcome the labeling (i.e. manually labeling of samples for each concept) by asking queries in the form of unlabeled instances to be labeled by an human expert. In this way, the active learner aims to achieve high accuracy using as few labeled in-

stances as possible, thereby minimizing the cost of obtaining labeled data. Active learning is well-motivated in many modern machine learning problems where data may be abundant but labels are scarce or expensive to obtain.

Figure 5.1 illustrates the active learning cycle [46]. A learner may begin with a small number of instances in the labeled training set L, request labels for one or more carefully selected instances, learn from the query results, and then leverage its new knowledge to choose which instances to query next. Once a query has been made, there are usually no additional assumptions on the part of the learning algorithm. The new labeled instance is simply added to the labeled set L, and the learner proceeds from there in a standard supervised way [46].

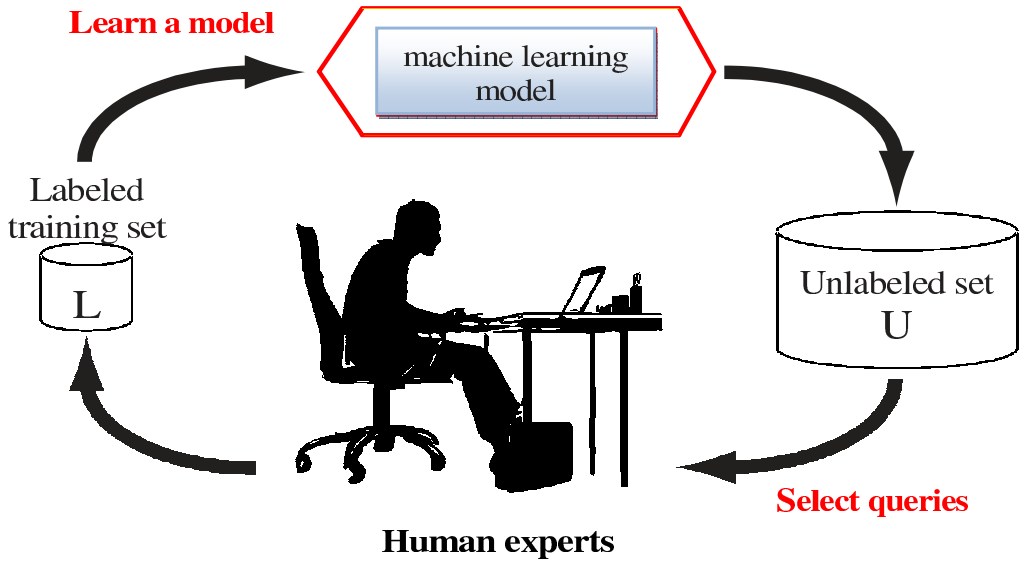


Figure 5.1: Active learning cycle.

A performance example of regression based active learning is shown in Figures 5.2 and 5.3. These figures show regression curves derived from the regression (PLSR or SVR). In Figure 5.2, the curve is obtained from training from nine data points. However, the Figure 5.3 is drawn closer to the target function. The main difference between both figures is that we have added more data points to the significant figure on the 5.3 so that data points are now well distributed. Hence, one of the advantages of active learning in the area with data points of the paucity of data add samples to draw an accurate regression function.

There are several strategies in which active learners may pose queries, and there are also several different query strategies that have been used to decide which instances are most informative. Section 5.3 describes, all the proposed active learning strategies.

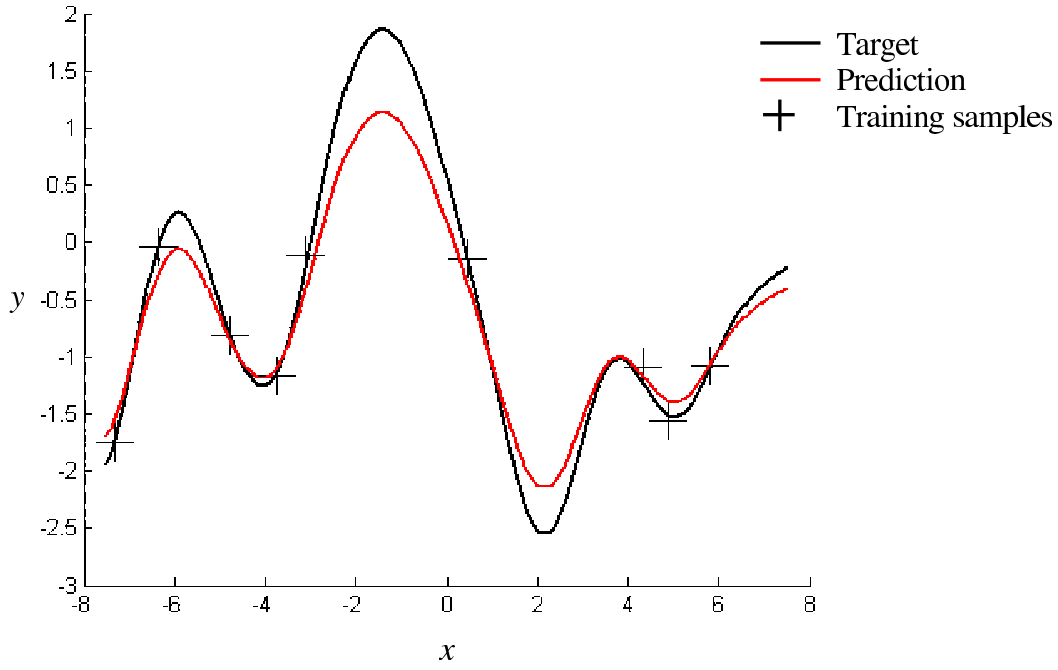


Figure 5.2: Performance example of regression.

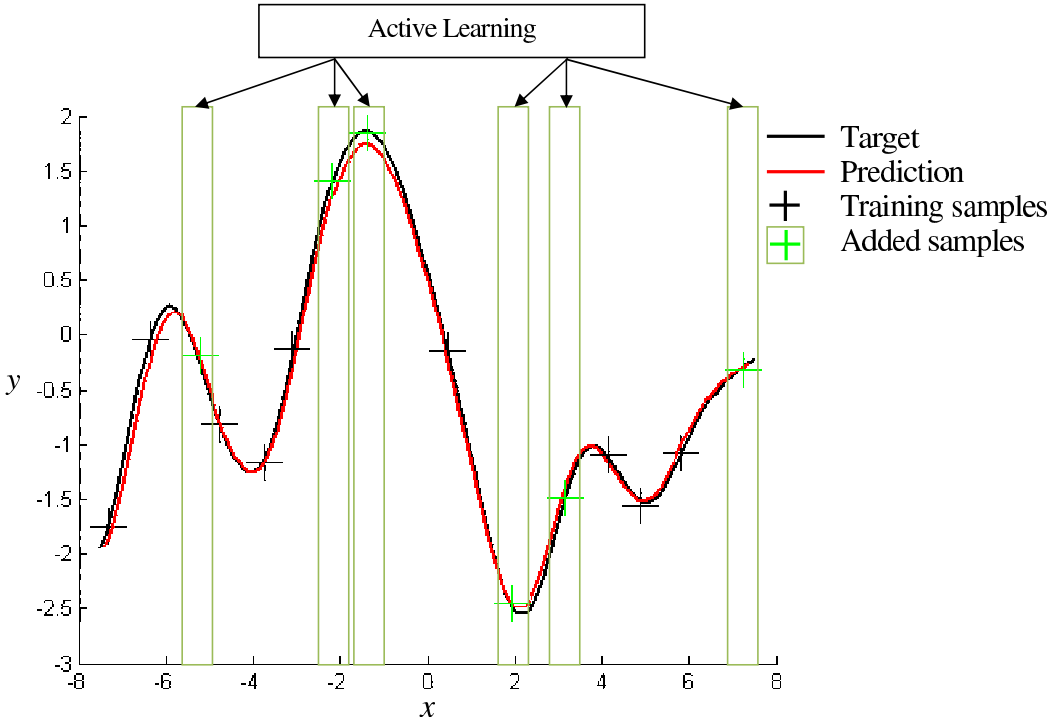


Figure 5.3: Performance example of regression based active learning.

5.3 Proposed active learning methods

Let us consider a training set composed initially of n labeled samples $L = \{x_l, y_l\}_{l=1}^{n}$ and an additional learning set composed of m-unlabeled samples $U = \{x_u\}_{u=n+1}^{n+m}$, with $m >> n$. The initial training samples (L) required by the active learning process were selected randomly from the whole training set U. In order to increase the training set L with a series of samples chosen from the learning set U and labeled by the expert, an active learning algorithm has the task of choosing them properly so that to minimize the error of the regression process while minimizing the number of learning samples to label, and thus to reduce the costs related to the training sample collection.

In Figure 5.4, we show the generic flow chart of the active learning approach for regression problems proposed in this work. Starting from the initial and small labeled training set L, the unlabeled samples of the learning set U are evaluated and sorted using an opportune criterion f_{AL}. In particular, we suppose for convention that the criterion f_{AL} has to be minimized. At this point, from the sorted samples U_s, the first N_s samples are selected, where N_s is the number of samples to be added in the training set L. Finally, the selected samples $U_s^{'}$ are labeled by the human expert and added to the training set L. The entire process is iterated until the predefined convergence condition is satisfied (e.g., the total number of samples to add to the training set is reached, or the accuracy improvement on an independent calibration/validation set over the last iterations becomes insignificant). In the following algorithm, we describe the different steps of the active learning approach:

Algorithm 3 Resumes the general steps of the active learning approach.

Input:

L: initial training set, composed of n labeled samples.

U: learning set, composed of m ($m >> n$) unlabeled samples.

N_s: number of samples to add at every iteration of the active learning process.

Start

1: Sort the learning set U using the criterion f_{AL} in order to obtain the set U_s.
2: Select the first N_s samples from U_s.
3: Label the selected samples $U_s^{'}$.
4: Add the labeled samples $L_s^{'}$ to the training set L and remove them from U.
5: Go to Step (1) until the predefined convergence condition is satisfied.

End

Output:

L: final training set

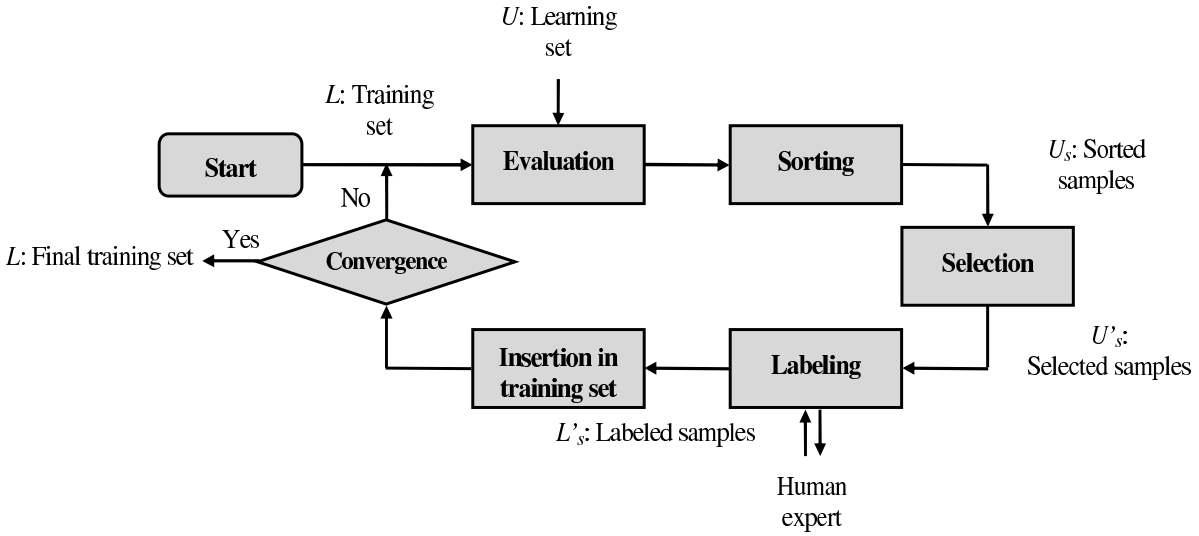


Figure 5.4: Flow chart of the proposed active learning approach.

In the next subsections, we present the different active learning strategies proposed in this thesis.

5.3.1 Proposed general active learning strategies

5.3.1.1 Pool of regressors (PAL)

The first strategy, named PAL in the rest of the chapter, is based on a pool of regressors, being, PLSR, RR, KRR, or SVR depending on the adopted regression method. Considering the original training set L, q training subsets are constructed by a down sampling of L. PAL can be achieved by two possible sampling: 1) sampling in the spectral domain, in the case of High-Dimensional data (spectroscopy data); 2) sampling in the training samples, in the case of Low-Dimensional data (wind speed data). Each training subset is considered independently from each other and used to train a different regressor. In this way, q parallel regressors are used. As depicted in Figure 5.5 .Therefore, q different estimations are obtained for each sample. Let $\mu_{u,ks}$ be the prediction value yielded for sample x_u $(u = n+1, n+2, ..., n+m)$ by regressor r_{ks} $(ks = 1, 2, ..., q)$. For each sample, the predictions are exploited to calculate the variance value on them:

$$f_{PAL,u} = \frac{1}{q} \sum_{ks=1}^{q} (\mu_{u,ks} - \bar{\mu}_u)^2, \tag{5.1}$$

where

$$\bar{\mu}_u = \frac{1}{q} \sum_{ks=1}^{q} \mu_{u,ks}. \tag{5.2}$$

The samples characterized by the greater disagreements between the different regressors, i.e.

the greater values of variance, are selected. Indeed, a high disagreement means that the corresponding sample has been estimated with high uncertainty, and thus adding it to the training set could be useful to improve the regression process. In the following, the different steps of the PAL technique are summarized:

Algorithm 4 Resumes the proposed methodology based on the pool of regressors.

Input:

L: initial training set.

U: learning set.

q, number of training subsets.

N_s: number of samples to add at every iteration of the active learning process.

1: Considering the current training set L, construct q different training subsets L_g $(g = 1, 2, ..., q)$ by sampling the training samples or spectral domain.
2: Predict the target value of each sample x_u of the learning set U for each regressor r_{ks} $(ks = 1, 2, ..., q)$ beforehand trained on its corresponding training subset.
3: Compute the variance on the predictions $f_{PAL,u}$ given by the different regressors using Equation (5.1).
4: Set $f_{PAL}(u) = -f_{PAL,u}$.
5: Select and label the N_s samples with the greater disagreements between the different regressors.

Output:

$L_s^{'}$: new subset of labeled samples to add in the training set.

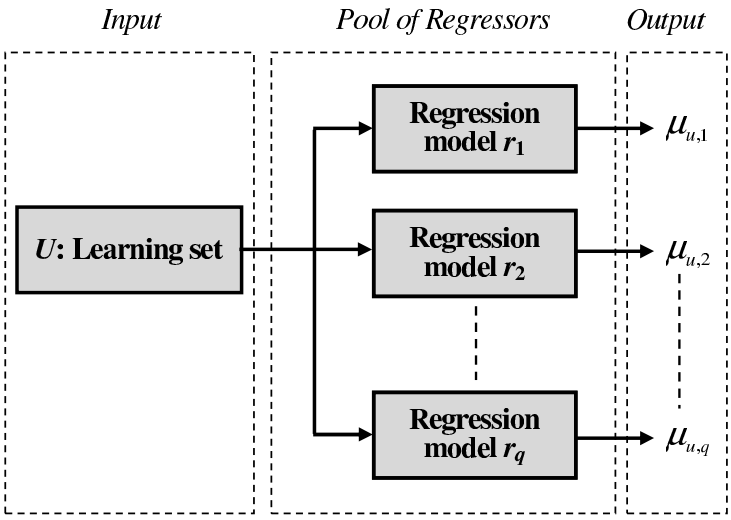


Figure 5.5: Block diagram of the method based a pool active learning (PAL).

5.3.1.2 Distance from the closest training sample (DAL)

The second strategy (DAL) consists to calculate for each sample x_u $(u = n+1, n+2, ..., n+m)$ the Euclidean distances $D_u \in \Re^n = [D_{u,1}, D_{u,2}, ..., D_{u,n}]$ in the feature domain from the samples x_l $(l = 1, 2, ..., n)$ already composing the current training set:

$$D_{u,l} = \|\mathrm{x}_u - \mathrm{x}_l\| . \tag{5.3}$$

After that, for each sample x_u $(u = n+1, n+2, ..., n+m)$, the closest training sample $tr_{min,u}$ is identified and the corresponding distance value $D_{min,u}$ is considered to define the selection criterion:

$$f_{DAL}(u) = -D_{min,u} = -\min_{l=1,...,n} \{D_{u,l}\}. \tag{5.4}$$

In this way, we favor the selection of sample placed in areas of the original space not covered by training samples and avoid to choose samples similar to those already present in the current training set. In the following, the different phases of the DAL strategy are summarized:

Algorithm 5 Synthesizes the proposed strategy based on the distance from the closest training sample.

Input:

L: initial training set.

U: learning set.

N_s: number of samples to add at every iteration of the active learning process.

1: Compute the Euclidean distances $D_u \in \Re^n = [D_{u,1}, D_{u,2}, ..., D_{u,n}]$ from the n different training samples for each sample x_u $(u = n+1, n+2, ..., n+m)$ of the learning set U.
2: Identify the training sample $tr_{min,u}$ closest to each sample x_u.
3: Consider the distance value $D_{min,u}$ associated with the training sample $tr_{min,u}$.
4: Set $f_{DAL}(u) = -D_{min,u}$.
5: Select and label the N_s most distant samples.

Output:

L_s': new subset of labeled samples to add in the training set.

5.3.1.3 Residual regression (RSAL)

The third strategy, denoted RSAL, is based on the residual regression approach described in [39] (see Chapter 4) and is depicted in Figure 5.6. Its underlying idea consists in exploiting the systematic residuals (i.e., errors) which are generated by the predictor, i.e., the regression

model (see Figure 5.6). For such purpose, a second model (called residual model) is trained to accomplish the task of estimating the residuals of the predictor. In other words, the residual estimator analyzes and models the error function associated with the predictor and defined as:

$$e(x_l) = y_l - \hat{f}(x_l). \tag{5.5}$$

It is worth recalling that, during the training phase x_l $(l = 1, 2, ..., n)$, represents a training sample for which y_l stands for its corresponding target value while $\hat{f}(x_l)$ refers to the corresponding estimate generated by the predictor. The residual model is a regressor of the same kind of the predictor. This means that if, for instance, KRR is adopted as regression method for the predictor, it is too for the residual regression with the difference that the predictor is trained on sample pairs (x_l, y_l) with $(l = 1, 2, ..., n)$, while the training of the residual regressor is performed on sample pairs $(x_l, e(x_l))$ with $(l = 1, 2, ..., n)$.
Once the training phase is finished, the residual phase takes place. All unlabeled samples x_u $(u = n + 1, n + 2, ..., n + m)$ are given in input to the residual regressor for the purpose of estimating their corresponding residual $\hat{f}_r(x_l)$, on its turn, exploited to define the selection criterion of RSAL:

$$f_{RSAL,u} = \left| \hat{f}_r(x_u) \right|. \tag{5.6}$$

Accordingly, RSAL favors the selection of samples for which the predictor exhibits a higher likelihood of error. The RSAL algorithm can be summarized as follows:

Algorithm 6 Synthesizes the proposed method based on the residual regression.

Input:
L: initial training set.
U: learning set.
N_s: number of samples to add at every iteration of the active learning process.

1: Train an estimator $\hat{f}(\cdot)$ with the training set L.
2: Compute the residuals on L using Equation (5.5).
3: Train a residual regression model $\hat{f}_r(\cdot)$ with the resulting training set $(x_l, e(x_l))$.
4: Estimate the residuals on the unlabeled samples U by means of $\hat{f}_r(\cdot)$.
5: Using Equation (5.6), compute $f_{RSAL,u}$.
6: Set $f_{RSAL,u}(u) = -f_{RSAL,u}$.
7: Select and label the N_s samples characterized by the highest residual values.

Output:
L'_s: new subset of labeled samples to add in the training set.

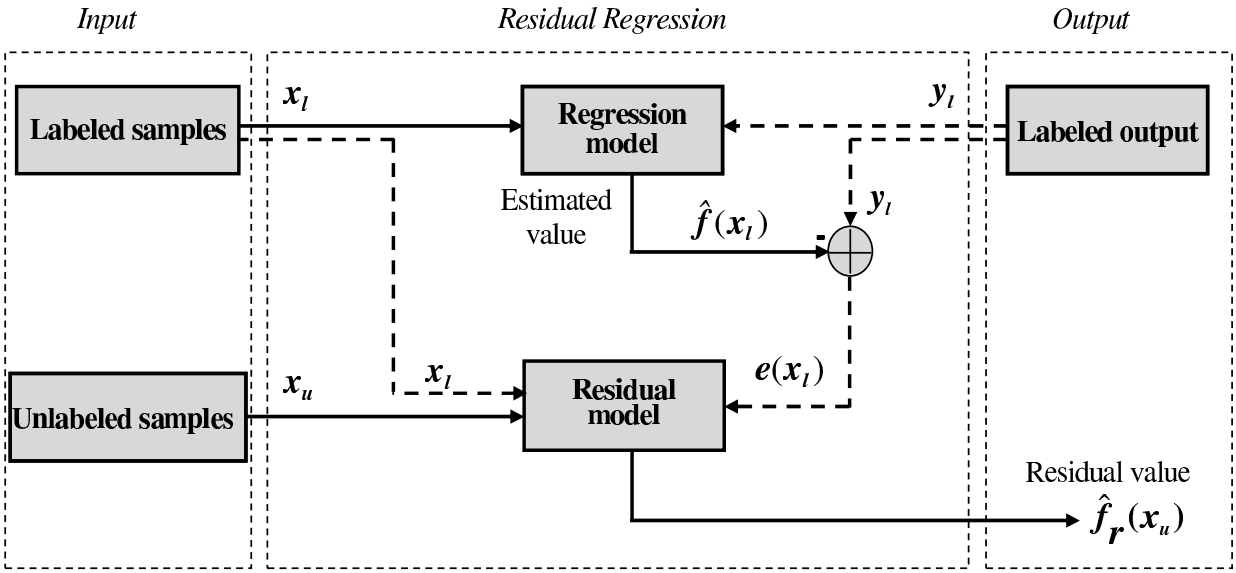


Figure 5.6: Block diagram of the method based a residual active learning (RSAL).

5.3.2 Active learning strategy for SVR

5.3.2.1 Distance from the support vectors (SVR-DAL)

The proposed method (SVR-DAL) for SVR is very similar to DAL presented previously for PLSR, RR, KRR, or SVR. However, while for DAL we calculate the distances with respect to all training samples, in this case we consider only the training samples identified as support vectors (SVs) after the regressor training on L. This is motivated by the fact that while for DAL all training samples contribute to describing the regression model, for SVR only the SVs are necessary to define the regression function. Moreover, more complex sorting and selection strategies are performed in order to take into account the distribution in the feature space of the samples. First, for each sample x_u $(u = n+1, n+2, ..., n+m)$, the closest support vector $s_{min,u}$ is identified and the corresponding distance value $D_{min,u}$ is calculated. Then, we define as $\tilde{\alpha}_u$ the absolute value of the Lagrange multiplier associated with $s_{min,u}$, that is, $\tilde{\alpha}_u = |\alpha_u - \alpha_u^*|$. We recall that the Lagrange multipliers weight each training sample according to its importance in determining the final solution. The most important training samples are those for which the corresponding Lagrange multipliers are, in absolute terms, equal to the regularization parameter C. At this point, the samples of the learning set U are ordered first by function of the value $\tilde{\alpha}_u$ and then by function of the distance value $D_{min,u}$. Finally, an additional constraint is included in the sample selection step since, at each iteration, it is required that the selected samples cannot share the same closest support vector. In this way, we tend to select most distributed samples possible in the feature space.

Algorithm 7 SVR Active learning based on distance from the support vectors.

Input:

L: initial training set.

U: unlabeled samples.

N_s: number of samples to add at every iteration of the active learning process.

1: Identify the S_n support vectors of the regressor on the training set L.
2: Compute the Euclidean distances $D_u \in \Re^{Sn} = [D_{u,1}, D_{u,2}, ..., D_{u,Sn}]$ from the S_n different support vectors for each sample x_u $(u = n+1, n+2, ..., n+m)$ of the learning set U.
3: Identify the support vector $s_{min,u}$ closest to each sample x_u.
4: Consider the distance value $D_{min,u}$ associated with the support vector $s_{min,u}$.
5: Consider the absolute value $\tilde{\alpha}_u$ of the Lagrange multiplier associated with the support vector $s_{min,u}$.
6: Set $f_{DAL-SVR}(u) = -\tilde{\alpha}_u$, $f_{DSVAL-SVR}(u) = -D_{\min,u}$.
7: Select first by function of $f_{DAL-SVR}$ and then by function of $f_{DSVAL-SVR}$ if the same value of $f_{DAL-SVR}$ is obtained for successively sorted samples. During the selection, if a sample shares the same closest SV with respect to a previously selected sample, skip it.

Output:

$L_s^{'}$: new subset of labeled samples to add in the training set.

5.4 Experimental design

Two experiments are described in order to assess the accuracy and performance of the active learning. In Subsection 5.4.1, an active learning in chemometrics data set is considered. In Subsection 5.4.2, the second experiment refers to a renewable energy data set. In order to assess the performances of different strategies, three different performance metrics are used: the root mean square error (RMSE), the mean absolute error (MAE), the normalized mean square error (NMSE), and the standard deviation STD of the error E (i.e., RMSE, MAE, and NMSE):

$$RMSE = \sqrt{\frac{1}{t}\sum_{i=1}^{t}(y_i - \hat{y}_i)^2}, \tag{5.7}$$

$$MAE = \frac{1}{t}\sum_{i=1}^{t}|y_i - \hat{y}_i|, \tag{5.8}$$

$$NMSE = \frac{\frac{1}{t}\sum_{i=1}^{t}(y_i - \hat{y}_i)^2}{\mathrm{var}(y)}, \tag{5.9}$$

$$STD = \sqrt{\frac{1}{N_R - 1}\sum_{j=1}^{N_R}(E_j - \frac{1}{N_R}\sum_{j=1}^{N_R}E_j)^2}, \tag{5.10}$$

where y_i and $\hat{y}_i$ indicate the actual and forecast wind or concentration measurements values of

the sample i, t is the number of test samples. The parameter $var(y)$ represents the variance on the output values and plays the role of normalizing constant. It is calculated on all available samples (i.e., both learning and test samples), N_R is the number of run ($N_R = 10$).

5.4.1 Experiments on spectroscopic data set

In the following subsections, two general active learning strategies (PAL, DAL) and one method specifically for SVR (SVR-DAL) applied to chemometrics field, developed for linear and nonlinear regression approaches, namely PLSR, RR, KRR and SVR. In the present study, the above three strategies are tested and compared with them.

For all spectroscopic data sets, the initial training samples required by the active learning process were selected randomly from the learning set U. For the diesel data set, starting from 33 samples, the active learning algorithms were run until all the learning samples were added to the training set, adding 20 samples at each iteration. Similarly, 20 samples were added at each iteration by starting from 49 and 32 samples for the orange juice and tecator data sets, respectively.

The details of the experimental setup on the different data sets are summarized in Table 5.1. To yield statistically reliable results, the entire active learning process was run ten times, each time with a different initial training set. At each run, the initial training samples were chosen in a completely random way. Linear and nonlinear regressors were also trained on the entire learning set in order to have a reference-training scenario, called "full" training (named PLSR-Full, RR-Full, KRR-Full and SVR-Full, respectively). On the one hand, the regression results obtained in this way represent a lower boundary for the errors. On the other hand, we expect that the upper error boundary will be given by the completely random selection strategy (named PLSR-Random, RR-Random, KRR-Random and SVR-Random, respectively). We recall that the purpose of any active learning strategy is to converge to the performance of the "full" training scenario faster than the random selection method. Regression performances were evaluated on the test sets in terms of the normalized mean square error (NMSE).

Table 5.1: Data set information and experimental setup for the different data sets.

	Data set information			**Experimental setup**	
Name	**# features**	**# learning samples**	**# test samples**	**# initial training samples**	**# samples added at each iteration**
Diesel	401	133	112	33	20
Orange juice	700	149	67	49	20
Tecator	100	172	43	32	20

Concerning the parameter setting, in the case of PLSR, the optimal number of latent variables was estimated in the range [1, 20]. For RR, the value of the ridge parameter λ is in the range $[2^{-23}, 2^{-6}]$. For the KRR and SVR, we adopted a Gaussian kernel. This choice is motivated by the generally good prediction accuracy associated with this kernel. For KRR, the value of the kernel width (σ) was chosen in the interval $[2^3, 2^7]$, while λ is comprised in $[2^{-23}, 2^{-6}]$. For SVR, the regularization (C) and kernel width (γ) parameters were tuned empirically in the ranges $[2^{-9}, 2^9]$ and $[2^{-11}, 2^3]$, respectively. For all algorithms, parameter values were estimated by k-fold cross-validation ($k = 5$). Regarding the number q of regressors for the active learning strategy based on the pool, it was fixed to four in all experiments.

5.4.1.1 Experimental results

Figures 5.7-5.9 report the results obtained for the diesel, orange juice, and tecator data sets, respectively, by evolving the active learning process. In particular, the graphs refer to (a) PLSR, (b) RR, (c) KRR and (d) SVR in terms of NMSE and standard deviation. First, we note that, before starting the active learning process, poor performances were obtained, both in terms of prediction errors (NMSE) and related standard deviations (STD). This result can be expected because of the small number of training samples used to train the regressors, which has also a direct impact on the regression model quality as shown by the strong variability (STD) of the prediction errors. Another expected result is given by the improvement of performances when additional samples are inserted in the training set. This results in graphs with an approximately monotonous decreasing behavior of NMSE and STD, which tend to converge to the results yielded by the "full" regressors, for which the entire learning set is exploited to train the model. Although such decreasing is verified for both active and random selection, we note that in general the active methods allow a faster convergence to the "full" result with respect to the random strategy, both in terms of NMSE and STD. In particular, the improvements in terms of STD indicate greater levels of stability in defining the regression model. While for the random selection the entire set of learning samples is necessary to converge for all experiments, in some cases the active learning process allows complete convergence using just a subset of the learning set. Moreover, before convergence, the proposed active learning strategies give, in general, an improvement with respect to the random one. This means that similar values of prediction errors can be obtained using a minor quantity of training samples, which implies a reduction in the expert work and a decrease in the computational time necessary to train the regressor. Among the proposed strategies, the PAL method, based on the pool of regressors, yields, in general, better results with respect to those of DAL (based on the distances in the feature space between labeled and unlabeled samples). This is verified for PLSR, RR, KRR and SVR.

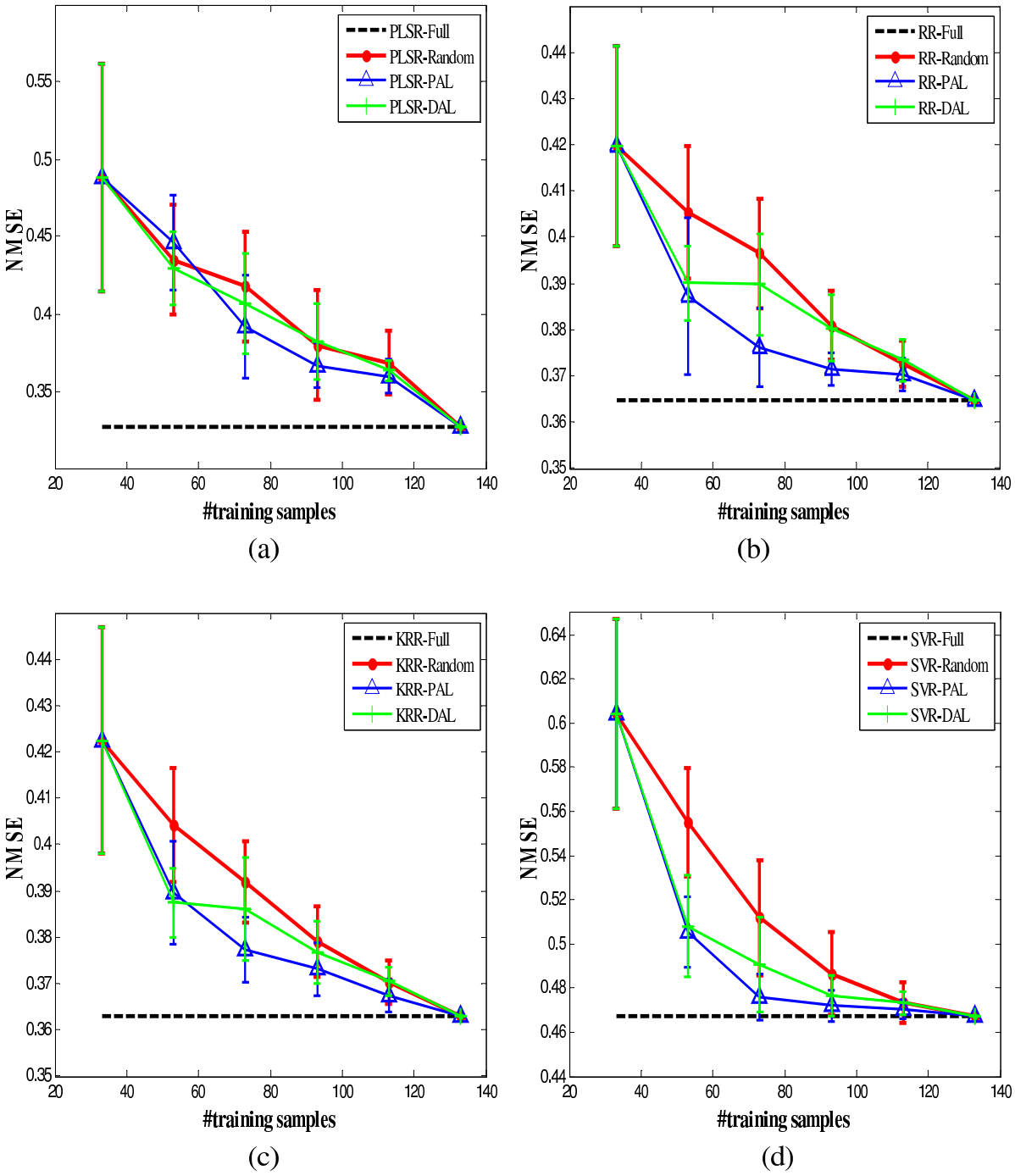


Figure 5.7: Performances achieved on the diesel data set for (a) PLSR, (b) RR, (c) KRR and (d) SVR in terms of NMSE and standard deviation. Each graph shows the results in function of the number of interactions. All results are averaged over ten runs of the approaches. (PLSR, RR, KRR, SVR)-Full = full, (PLSR, RR, KRR, SVR)-Random = random, (PLSR, RR, KRR, SVR)-PAL = pool of regressors, (PLSR, RR, KRR)-DAL = distance from the closest training sample in features space, SVR-DAL = distance from the support vectors.

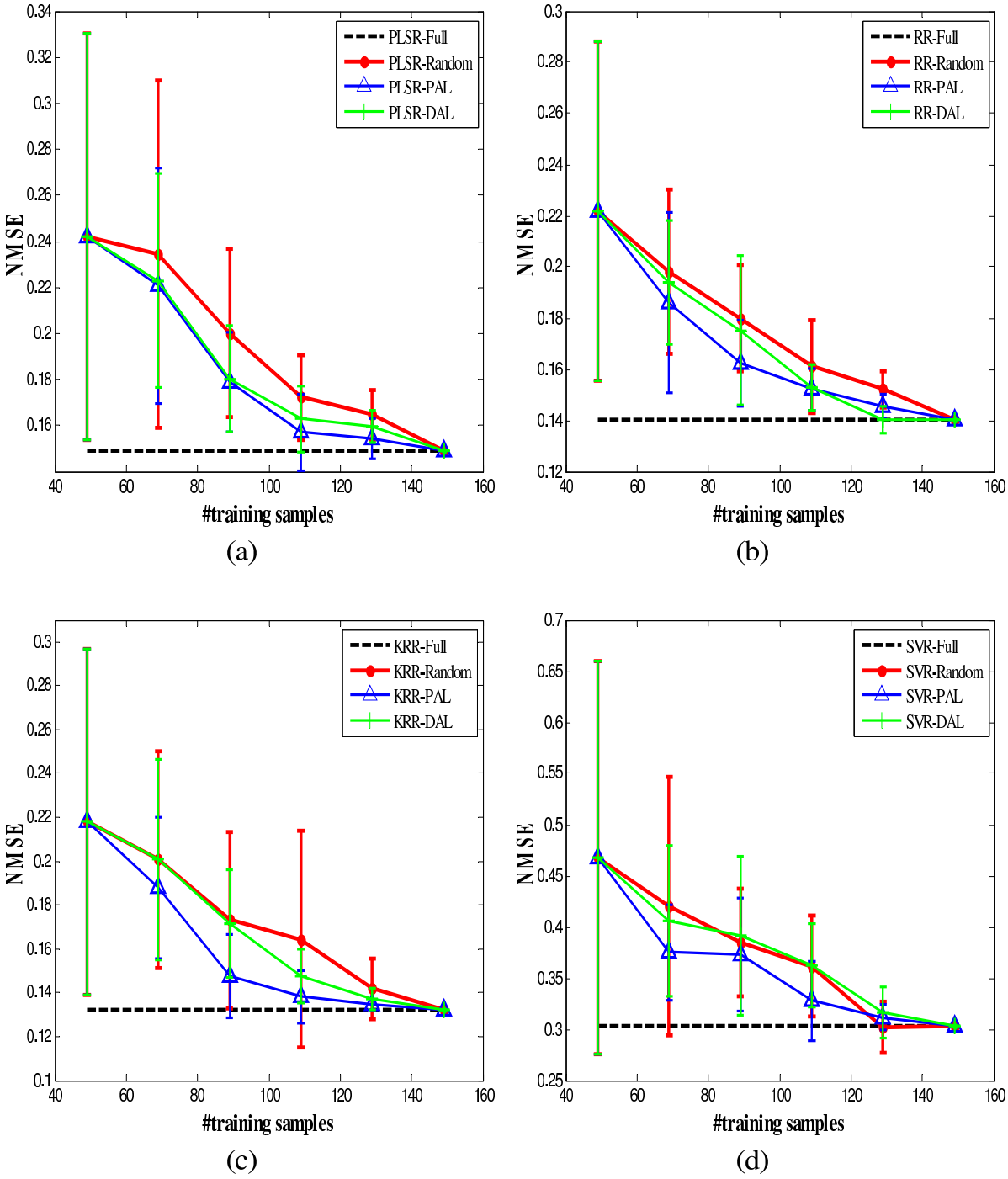


Figure 5.8: Performances achieved on the orange juice data set for (a) PLSR, (b) RR, (c) KRR and (d) SVR in terms of NMSE and standard deviation.

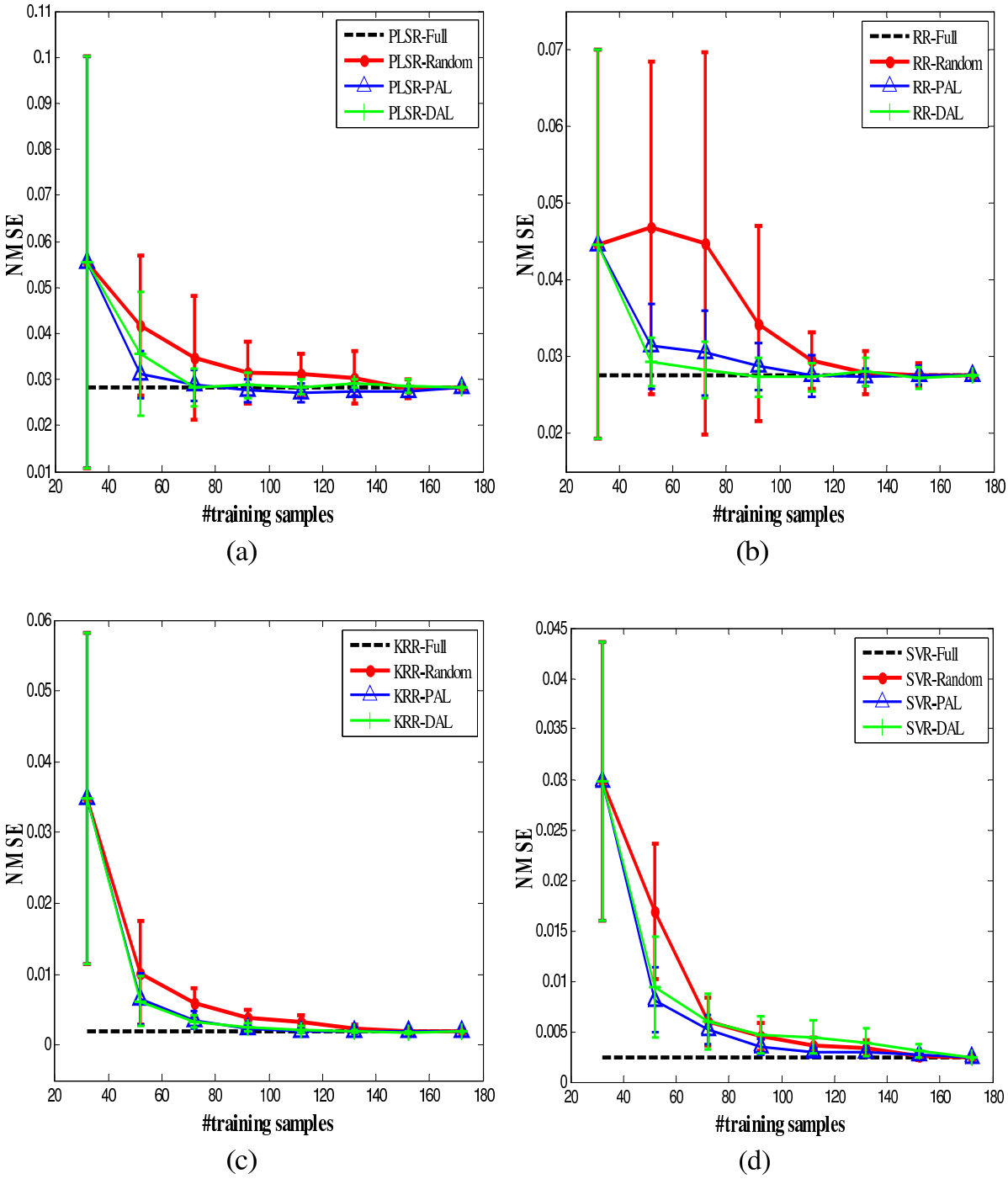


Figure 5.9: Performances achieved on the tecator data set for (a) PLSR, (b) RR, (c) KRR and (d) SVR in terms of NMSE and standard deviation.

The obtained results are shown in greater detail in Table 5.2 (a)-(c), for the diesel, orange juice, and tecator data sets, respectively. In particular, we considered the performances obtained when 40 additional samples are inserted in the training set. Therefore, the number of training samples equal to 73, 89 and 72 are considered for the different data sets, respectively. We report the values of NMSE and the corresponding STD. The best results are highlighted in bold font. Moreover, for PLSR we indicate the number of optimal latent variables estimated automatically by k-fold cross-validation, while for SVR we show the number of support vectors identified in the training process. As can be seen, the proposed strategies are characterized by better performances with respect to the random method from different points of view. First, better values of accuracies are obtained using the same number of training samples. Then, better values of standard deviations associated with the prediction errors are verified.

Table 5.2: NMSE, Standard Deviation (STD), # latent variables, # support vectors obtained for the PLSR, RR, KRR and the SVR on (a) the diesel, (b) the orange juice, and (c) the tecator data sets.

		Linear regression					**Nonlinear regression**				
		PLSR			**RR**		**KRR**		**SVR**		
Method	**# training samples**	**NMSE**	**STD**	**# latent variables**	**NMSE**	**STD**	**NMSE**	**STD**	**NMSE**	**STD**	**# support vectors**
(a)											
Full	133	0.3270	–	7	0.3648	–	0.3630	–	0.4672	–	110
Initial	33	0.4881	0.0733	4.4	0.4197	0.0217	0.4225	0.0243	0.6044	0.0428	31.6
Random	73	0.4176	0.0356	4.8	0.3965	0.0118	0.3921	0.0088	0.5120	0.0261	65.5
PAL		**0.3918**	0.0335	5.5	**0.3762**	**0.0086**	**0.3774**	**0.0070**	**0.4759**	**0.0103**	68
DAL		0.4064	**0.0325**	4.9	0.3898	0.0109	0.3861	0.0111	0.4907	0.0216	66.2
(b)											
Full	149	0.1493	–	13	0.1405	–	0.1320	–	0.3038	–	142
Initial	49	0.2422	0.0883	9.8	0.2218	0.0662	0.2180	0.0790	0.4679	0.1921	47.7
Random	89	0.2001	0.0365	10.6	0.1799	0.0209	0.1731	0.0400	0.3848	**0.0526**	89
PAL		**0.1790**	**0.0215**	13.3	**0.1624**	**0.0169**	**0.1477**	**0.0188**	**0.3730**	0.0551	88.7
DAL		0.1803	0.0231	11.6	0.1752	0.0294	0.1714	0.0246	0.3918	0.0778	89
(c)											
Full	172	0.0282	–	10	0.0275	–	0.0019	–	0.0024	–	151
Initial	32	0.0556	0.0448	4.5	0.0446	0.0253	0.0349	0.0234	0.0298	0.0138	30
Random	72	0.0348	0.0134	9.6	0.0447	0.0249	0.0060	0.0020	0.0060	0.0024	65.6
PAL		0.0288	**0.0033**	8.2	0.0304	0.0055	0.0035	0.0012	**0.0052**	**0.0015**	65.8
DAL		**0.0282**	0.0042	9.6	**0.0283**	**0.0037**	**0.0033**	**0.0009**	0.0060	0.0027	66.6

5.4.2 Experiments on wind speed data set

Similarly to the previous Subsection 5.4.1, the active learning approach is applied in the renewable energy field. In particular, we consider the problem of the estimation of wind speed in Algeria. In this case, the proposed strategies are specifically developed for kernel ridge regression (KRR). In particular, three different active learning strategies (PAL, DAL, RSAL) are applied and tested.

In all the following experiments and for all wind speed data sets, the initial training samples required by the active learning process were selected randomly from the whole training set U. Starting from 100 samples, the active learning algorithms were run until all the learning samples were added to the training set, adding 50 samples at each iteration. The entire active learning process was run ten times, each time with a different initial training set to yield statistically reliable results. At each run, the initial training samples were chosen in a completely random way.

Nonlinear regressors were trained on the entire learning set in order to have a reference-training scenario, called "full" training (named KRR-Full). On the one hand, the regression results obtained in this way represent a lower bound for the errors. On the other hand, we expect that the upper error bound will be given by the completely random selection strategy (KRR-Random). We recall that the purpose of any active learning strategy is to converge to the performance of the "full" training scenario faster than the random selection method.

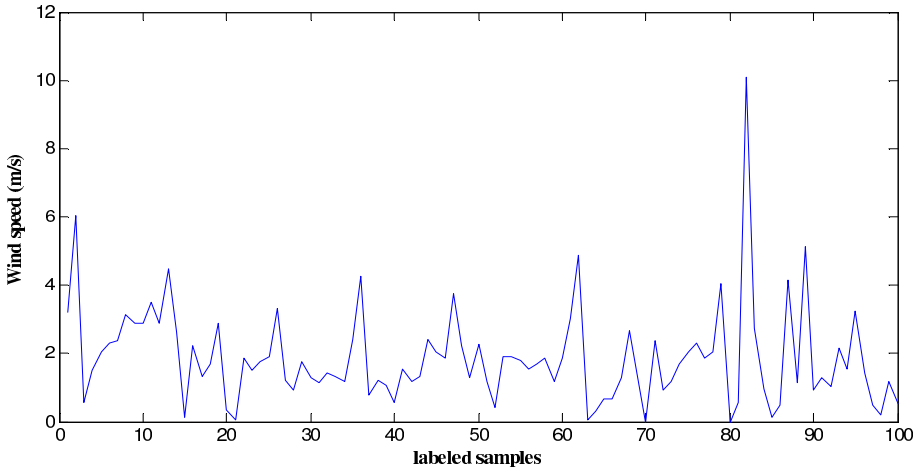
Regression performances were evaluated on the test sets by means of three different performance metrics which are the root mean square error (RMSE), the mean absolute error (MAE), and the normalized mean square error (NMSE).

Regarding the KRR parameter setting, the value of the kernel width was chosen in the interval $[10, 100]$, while λ was tuned in the range $[10^{-7}, 10^{-4}]$. All parameter values were estimated by k-fold cross-validation (k=5). Regarding the number q of regressors for the active learning strategy based on the pool, it was fixed to four in all experiments.

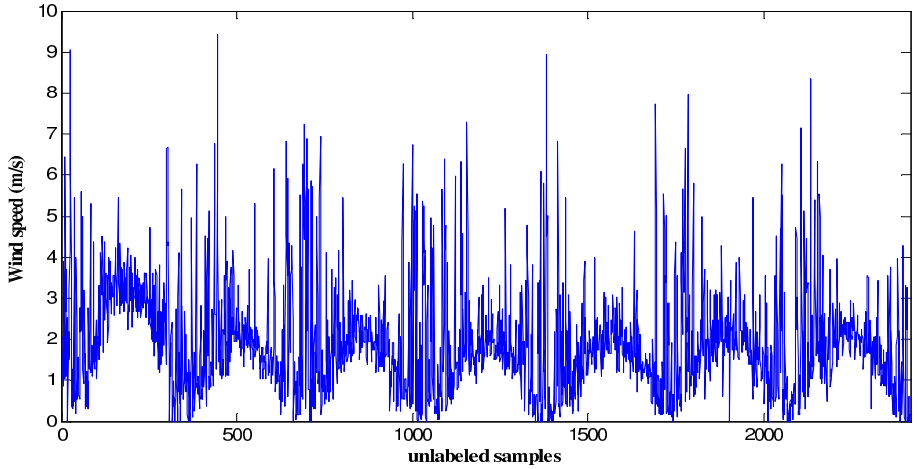
5.4.2.1 Experimental results

Let us first focus on a specific station, for instance the one located in Tlemcen. With the available 100 training samples, we trained a KRR predictor whose parameters were tuned by k-fold cross-validation as mentioned above. Repeating this operation 10 times, each with a different initial training set and averaging the error incurred on the test samples, KRR yields a RMSE equal to 1.6647 as reported in Table 5.3-(a). As expected such an error is higher than what could be obtained by KRR-Full, namely by involving all the available 2519 training samples (instead of just 100 training samples, see Figures 5.10-(a) and 5.10-(b)) in which case the RMSE is equal to 1.4034. Now the question which arises is: ***would it be possible to get closer to the error of KRR-Full by however consuming (much) less training samples?*** To answer to this

question, we ran the three proposed active learning strategies (PAL, DAL, RSAL) as described in the previous section. Figures 5.11-(a), 5.11-(b) and 5.11-(c) show an example of sample selection on the Tlemcen data set by the three methods (i.e., PAL, DAL and RSAL). In this example, the best 50 unlabeled samples are identified by each method. Note that the distribution of the samples is not uniform along the time index as the sample selection is guided by a criterion which follows a concept of variance maximization for PAL, distance maximization for DAL, and residual error maximization for RSAL. Subsequently, for each method, each selected sample was labeled by the expert. By "labeled", we mean that the true target is assigned to the sample, making it a new training sample ready to integrate the current training set.

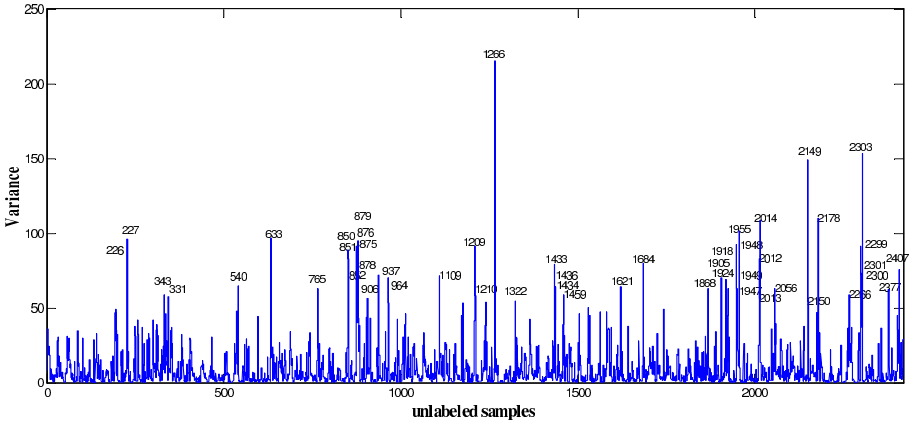


(a)

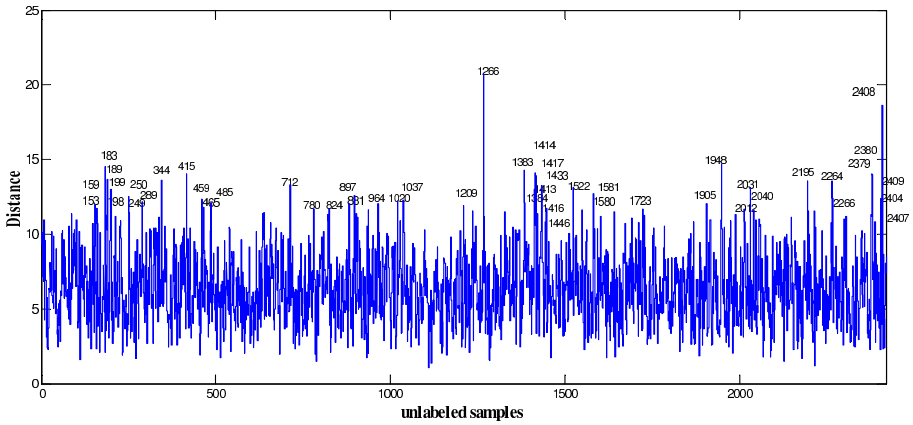


(b)

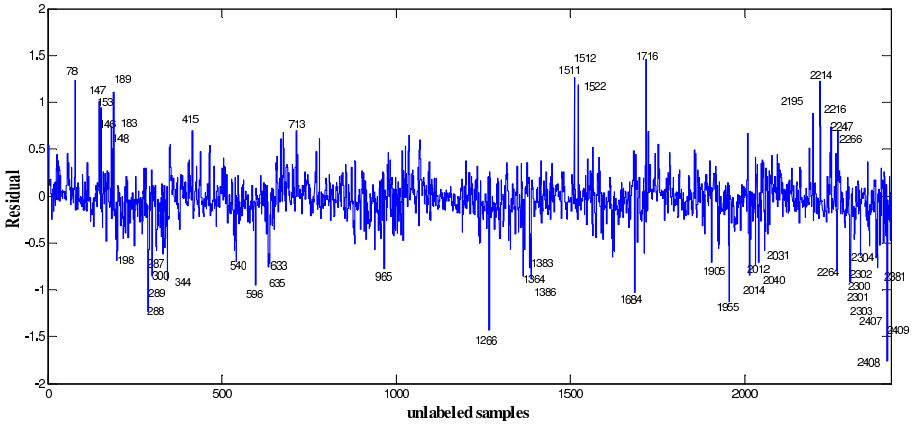
Figure 5.10: Illustration of learning samples, (a) 100 initial training samples, (b) 2419 unlabeled samples, respectively.



(a)



(b)



(c)

Figure 5.11: Illustration of selected samples. (a), (b) and (c) samples selected by PAL, DAL and RSAL, respectively.

To better understand the behavior of the proposed methods, in Figure 5.12 (a)-(c) we show the evolution at each iteration of the variance, distance and the residual on the Tlemcen data set. According to the previous figure (Figure 5.11), we can represent the evaluation of the proposed methods as function of the number of iterations. For the first to fourth iterations, the number of training samples are 100, 150, 200 and 250 (at each iteration we are add 50 sample), respectively. It is interesting that these values decrease with each iteration, so we have a decrease in the values of the variance, distance and residual only when a sufficient number of samples were added to the training set. The decrease of the values of the different criteria's proposed means that at each iteration the difficult samples for prediction are added to the training set. However, these new samples are very informative and thus allow improving the generalization performance.

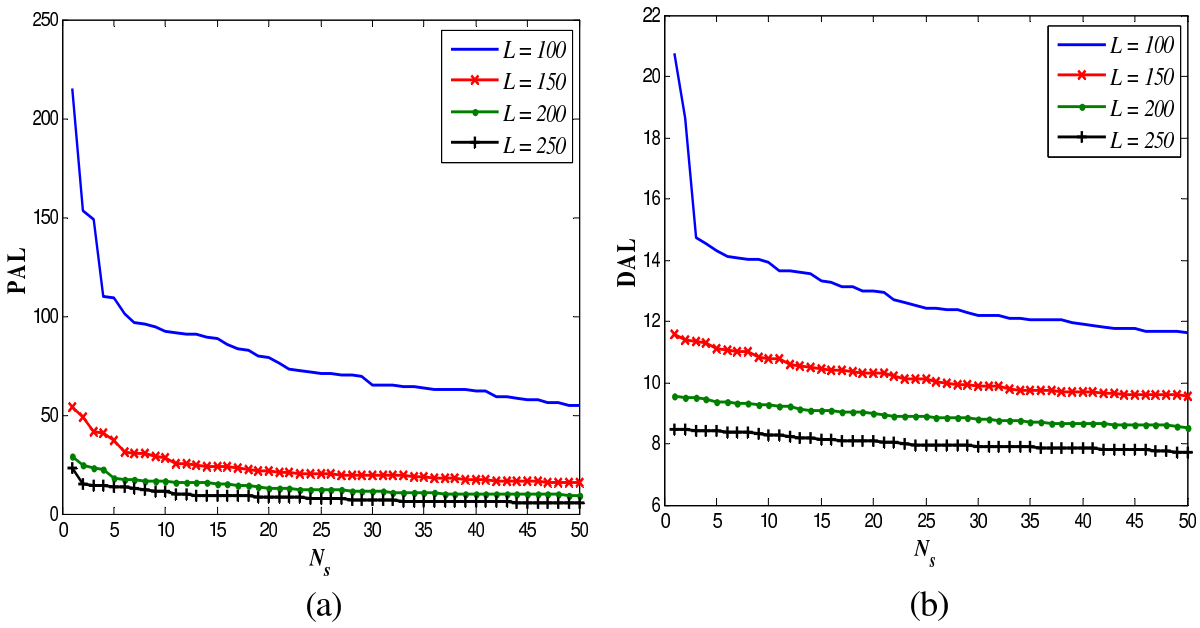
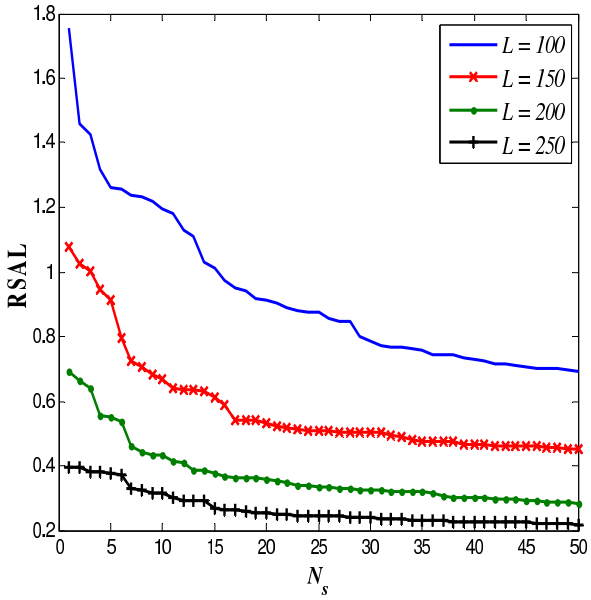


Figure 5.12: Example of active learning of (a) and (b) the evolution of sample selection at the first to fourth iteration (100, 150, 200 and 250 training samples) by the PAL and DAL, respectively.



(c)

Figure 5.12 (Continued): Example of active learning of (c) the evolution of sample selection at the first to fourth iteration (100, 150, 200 and 250 training samples) by the RSAL. Each graph shows the results in function of the number of samples to add at each iteration (N_s).

The above process was repeated several times, each by detecting and labeling the best 50 unlabeled samples. Figure 5.13-(a) shows the behavior of the RMSE by pushing the active learning process up to 1000 training samples. What can be expected and is confirmed in this figure is that the larger the number of training samples the smaller the error. An interesting observation is that the error can go also below the one achieved by KRR-Full. This means that the active learning is a way to filter the training set by potentially discarding those noisy samples which impact negatively on the prediction model. Note that all three active learning methods behave better than the simple random sample selection strategy (KRR-Random). This desirable behavior is also expressed in terms of standard deviation of the RMSE (STD), which means that the active learning is more stable than random sampling. Among the active learning methods, RSAL is the one which converges faster and thus is less sample-demanding with respect to PAL and DAL. In greater detail, as shown in Table 5.3-(a), with already 250 training samples, RSAL achieves a result very close to KRR-Full (RMSE=1.4439 against 1.4034). For achieving a similar result, the random sampling requires twice the number of training samples (i.e., 500 samples).

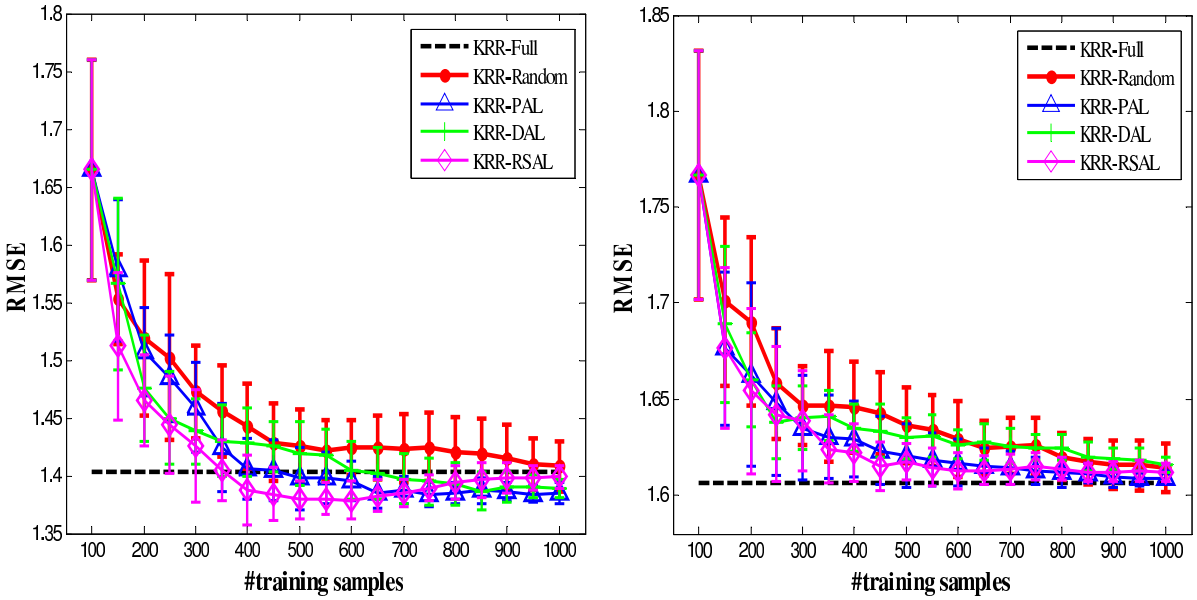


Figure 5.13: Performances achieved by the investigated methods on the (a) Tlemcen and (b) Chlef data sets in terms of RMSE and standard deviation of RMSE versus the number of selected samples. All results are averaged over ten runs.

Moving to the other nine stations (see Figures 5.13-(b), 5.14, 5.15, and Tables 5.3, 5.4, 5.5, 5.6), whatever the error measure (RMSE, MAE and NMSE), active learning confirms that a smart selection of the samples permits to optimize the construction of the training set, namely reaching a good accuracy while using a contained number of training samples.

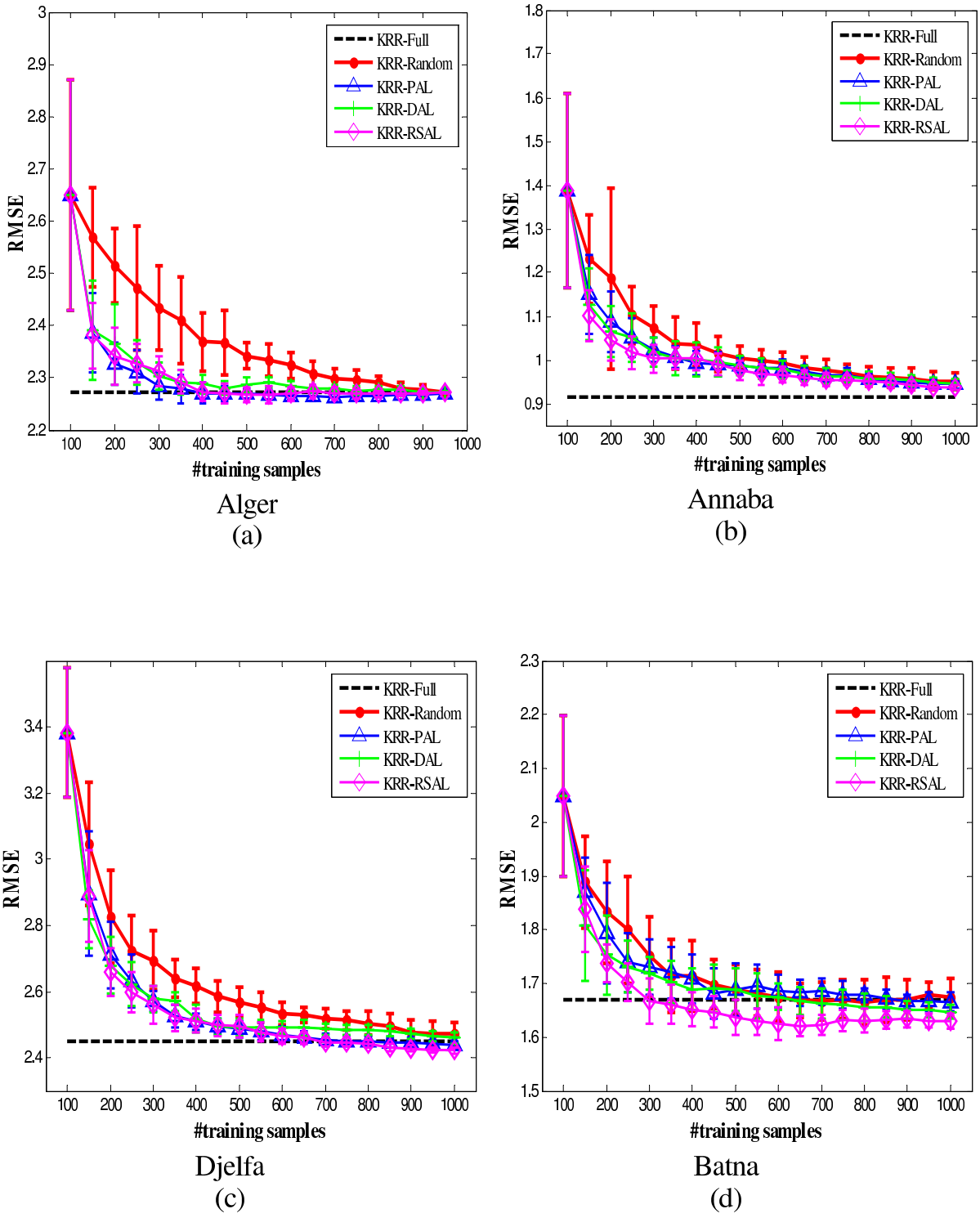


Figure 5.14: Performances achieved by the investigated methods on the (a) Alger, (b) Annaba, (c) Djelfa and (d) Batna data sets in terms of RMSE and standard deviation of RMSE versus the number of selected samples.

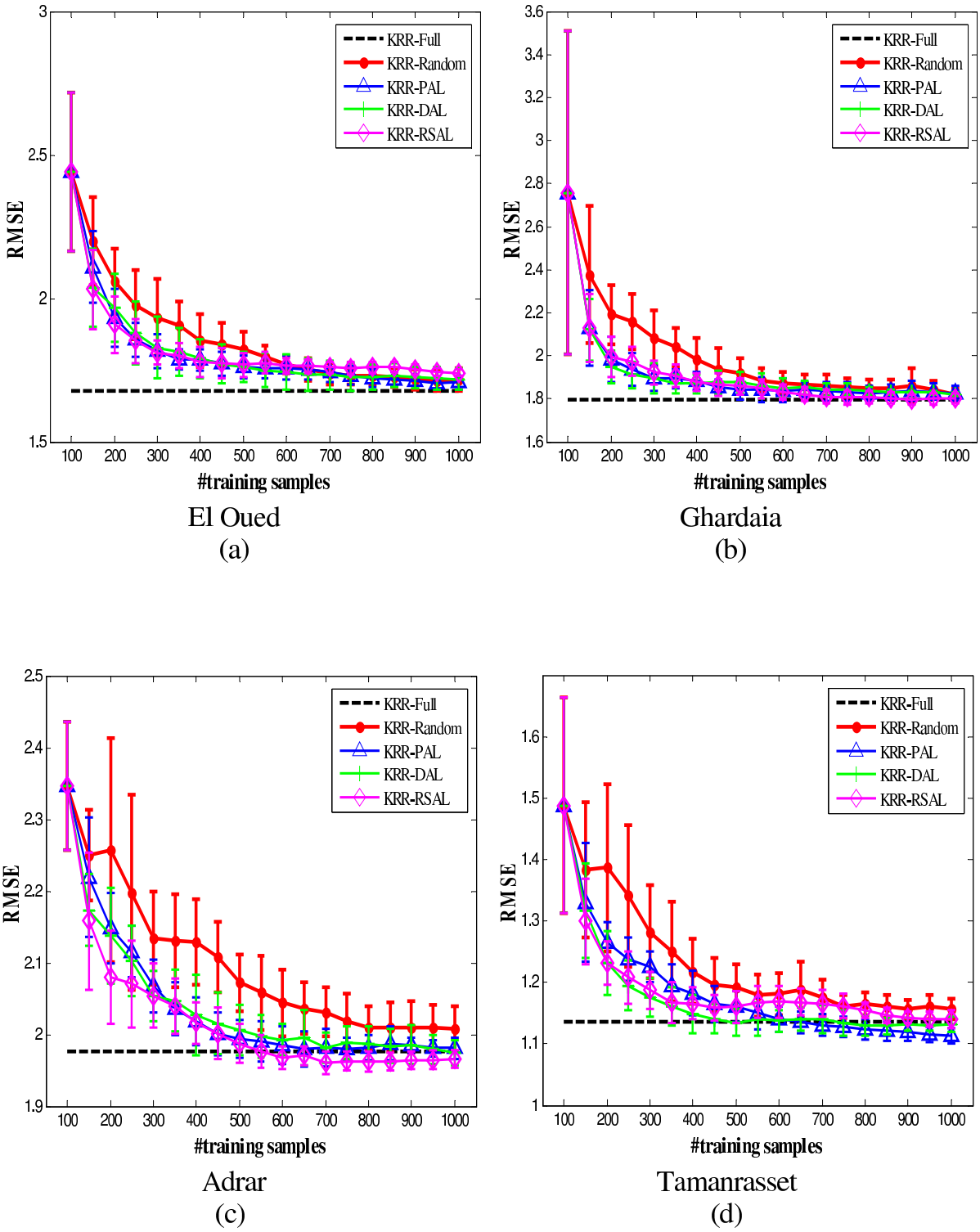


Figure 5.15: Performances achieved by the investigated methods on the (a) El Oued,(b) Ghardaia, (c) Adrar and (d) Tamanrasset data sets in terms of RMSE and standard deviation of RMSE versus the number of selected samples.

Table 5.3: RMSE Standard Deviation (STD), MAE Standard Deviation (STD), NMSE Standard Deviation (STD), computation times obtained by the KRR predictor on Tlemcen, Chlef and Alger data sets.

Location	**Method**	**Full**	**Initial**	**Random**	**PAL**	**DAL**	**RSAL**	**Random**	**PAL**	**DAL**	**RSAL**
Tlemcen	**# training samples**	2519	100	250				500			
	RMSE	1.4034	1.6647	1.5023	1.4847	1.4496	**1.4439**	1.4256	1.3976	1.4192	**1.3793**
	± STD	–	0.0951	0.0719	**0.0375**	0.0401	0.0425	0.0311	0.0274	0.0273	**0.0162**
	MAE	0.8925	1.1647	0.9869	0.9869	0.9552	**0.9539**	0.9252	0.9127	0.9123	**0.9024**
	± STD	–	0.0718	0.0421	**0.0228**	0.0296	0.0284	0.0207	0.0184	0.0130	**0.0117**
	NMSE	0.9483	1.3382	1.0889	1.0620	1.0124	**1.0045**	0.9789	0.9408	0.9701	**0.9161**
	± STD	–	0.1566	0.1041	**0.0538**	0.0558	0.0589	0.0424	0.0365	0.0373	**0.0215**
	Time [s]	368.00	0.36	1.55	2.40	1.57	3.07	8.96	11.02	9.22	18.24
Chlef	**# training samples**	1679	100	250				500			
	RMSE	1.6068	1.7667	1.6584	1.6485	**1.6379**	1.6422	1.6367	1.6206	1.6298	**1.6173**
	± STD	–	0.0645	0.0287	0.0383	**0.0186**	0.0353	0.0198	0.0166	0.0109	**0.0095**
	MAE	1.1494	1.2824	1.1971	**1.1901**	1.1924	1.1957	1.1694	1.1712	**1.1690**	1.1755
	± STD	–	0.0507	0.0371	**0.0228**	0.0256	0.0285	0.0188	0.0138	0.0110	**0.0089**
	NMSE	0.9851	1.1923	1.0497	1.0374	**1.0238**	1.0294	1.0222	1.0021	1.0135	**0.9980**
	± STD	–	0.0875	0.0364	0.0489	**0.0233**	0.0445	0.0248	0.0205	0.0134	**0.0117**
	Time [s]	145.58	0.35	1.56	2.34	1.55	2.92	9.14	11.04	9.11	18.31
Alger	**# training samples**	965	100	250				500			
	RMSE	2.2717	2.6496	2.4716	**2.3111**	2.3296	2.3280	2.3419	2.2695	2.2862	**2.2678**
	± STD	–	0.2199	0.1181	0.0416	0.0425	**0.0371**	0.0255	**0.0095**	0.0146	0.0150
	MAE	1.7556	1.9871	1.8848	**1.7880**	1.7954	1.7981	1.7969	**1.7556**	1.7717	1.7570
	± STD	–	0.1217	0.0639	0.0266	0.0265	**0.0196**	0.0167	0.0059	0.0141	**0.0044**
	NMSE	1.1374	1.5570	1.3491	**1.1776**	1.1965	1.1947	1.2090	1.1352	1.1520	**1.1335**
	± STD	–	0.2710	0.1304	0.0427	0.0439	**0.0383**	0.0263	**0.0095**	0.0148	0.0151
	Time [s]	37.44	0.34	1.49	2.34	1.56	3.18	9.10	10.94	9.09	18.12

Table 5.4: RMSE Standard Deviation (STD), MAE Standard Deviation (STD), NMSE Standard Deviation (STD), computation times obtained by the KRR predictor on Annaba, Djelfa and Batna data sets.

Location	Method	Full	Initial	Random	PAL	DAL	RSAL	Random	PAL	DAL	RSAL
Annaba	**# training samples**	2514	100	250				500			
	RMSE	0.9149	1.3873	1.1037	1.0521	1.0509	**1.0172**	1.0046	0.9884	0.9874	**0.9772**
	± STD	–	0.2228	0.0655	0.0449	0.0559	**0.0381**	0.0267	**0.0189**	0.0215	0.0227
	MAE	0.6674	0.9887	0.8007	0.7835	0.7795	**0.7524**	0.7315	0.7321	0.7288	**0.7203**
	± STD	–	0.1230	0.0443	0.0334	0.0413	**0.0266**	0.0191	**0.0150**	0.0165	0.0197
	NMSE	0.5241	1.2329	0.7651	0.6941	0.6931	**0.6487**	0.6322	0.6118	0.6107	**0.5981**
	± STD	–	0.4416	0.0945	0.0591	0.0748	**0.0500**	0.0335	**0.0233**	0.0268	0.0277
	Time [s]	339.34	0.34	1.55	2.35	1.61	3.07	8.86	11.00	9.04	18.32
Djelfa	**# training samples**	2401	100	250				500			
	RMSE	2.4486	3.3832	2.7241	2.6319	2.6226	**2.5989**	2.5691	**2.4877**	2.4917	2.4949
	± STD	–	0.1942	0.1047	0.0789	0.0655	**0.0611**	0.0437	**0.0250**	0.0262	0.0332
	MAE	1.8747	2.5119	2.0775	2.0197	2.0110	**1.9957**	1.9719	1.9157	**1.9127**	1.9336
	± STD	–	0.1379	0.0757	0.0582	**0.0476**	0.0523	0.0329	**0.0229**	0.0239	0.0291
	NMSE	0.8589	1.6444	1.0644	0.9930	0.9858	**0.9680**	0.9457	**0.8865**	0.8894	0.8918
	± STD	–	0.1874	0.0809	0.0588	0.0493	**0.0455**	0.0324	**0.0178**	0.0187	0.0238
	Time [s]	277.88	0.35	1.56	2.40	1.61	3.15	9.24	11.32	9.25	18.82
Batna	**# training samples**	2415	100	250				500			
	RMSE	1.6692	2.0479	1.7999	1.7395	1.7314	**1.7028**	1.6883	1.6878	1.6882	**1.6365**
	± STD	–	0.1496	0.0996	0.0552	0.0480	**0.0354**	0.0471	0.0495	0.0388	**0.0322**
	MAE	1.2653	1.5623	1.3649	1.3344	1.3142	**1.2965**	1.2747	1.2904	1.2766	**1.2374**
	± STD	–	0.1223	0.0703	0.0447	0.0579	**0.0319**	0.0313	0.0431	0.0393	**0.0233**
	NMSE	0.7294	1.1033	0.8505	0.7929	0.7854	**0.7594**	0.7467	0.7464	0.7465	**0.7014**
	± STD	–	0.1691	0.0947	0.0495	0.0435	**0.0316**	0.0418	0.0438	0.0346	**0.0276**
	Time [s]	280.01	0.35	1.56	2.43	1.58	3.12	9.17	11.15	9.22	18.60

Table 5.5: RMSE Standard Deviation (STD), MAE Standard Deviation (STD), NMSE Standard Deviation (STD), computation times obtained by the KRR predictor on El Oued and Ghardaia data sets.

Location	Method	Full	Initial	Random	PAL	DAL	RSAL	Random	PAL	DAL	RSAL
El Oued	**# training samples**	2473	100	250				500			
	RMSE	1.6793	2.4412	1.9765	1.8565	1.8786	**1.8517**	1.8229	1.7627	**1.7606**	1.7694
	± STD	–	0.2746	0.1210	**0.0588**	0.1105	0.0770	0.0608	**0.0396**	0.0536	0.0416
	MAE	1.3071	1.8397	1.5390	1.4796	1.4858	**1.4754**	1.4233	1.4059	**1.3912**	1.4154
	± STD	–	0.1677	0.0933	**0.0528**	0.0895	0.0638	0.0412	**0.0376**	0.0456	0.0424
	NMSE	0.5763	1.2317	0.8010	0.7049	0.7234	**0.7018**	0.6797	0.6352	**0.6340**	0.6401
	± STD	–	0.2753	0.0941	**0.0450**	0.0866	0.0584	0.0458	**0.0284**	0.0381	0.0300
	Time [s]	294.08	0.35	1.54	2.41	1.56	3.11	9.17	11.06	9.10	18.41
Ghardaia	**# training samples**	2486	100	250				500			
	RMSE	1.7940	2.7561	2.1581	1.9365	**1.9104**	1.9718	1.9164	**1.8438**	1.8749	1.8582
	± STD	–	0.7512	0.1278	0.0771	**0.0618**	0.0669	0.0722	0.0484	0.0486	**0.0394**
	MAE	1.4017	1.9001	1.6341	1.5252	**1.4946**	1.5567	1.4930	**1.4541**	1.4744	1.4736
	± STD	–	0.2288	0.0752	0.0585	**0.0515**	0.0596	0.0556	0.0363	0.0505	**0.0300**
	NMSE	0.7239	1.8227	1.0508	0.8446	**0.8216**	0.8754	0.8271	**0.7651**	0.7911	0.7769
	± STD	–	1.2250	0.1216	0.0679	**0.0531**	0.0592	0.0624	0.0403	0.0412	**0.0328**
	Time [s]	294.42	0.34	1.50	2.38	1.52	3.03	9.08	11.01	9.04	18.20

Table 5.6: RMSE Standard Deviation (STD), MAE Standard Deviation (STD), NMSE Standard Deviation (STD), computation times obtained by the KRR predictor on Adrar and Tamanrasset data sets.

Location	Method	Full	Initial	Random	PAL	DAL	RSAL	Random	PAL	DAL	RSAL
Adrar	**# training samples**	2262	100	250				500			
	RMSE	1.9764	2.3466	2.1982	2.1155	2.1031	**2.0702**	2.0721	1.9940	2.0057	**1.9876**
	± STD	–	0.0889	0.1367	**0.0358**	0.0491	0.0600	0.0391	**0.0269**	0.0352	0.0270
	MAE	1.5274	1.8145	1.6770	1.6533	1.6347	**1.6129**	1.6042	1.5500	1.5581	**1.5428**
	± STD	–	0.0642	0.0577	**0.0291**	0.0413	0.0472	0.0313	0.0213	0.0283	**0.0175**
	NMSE	0.9400	1.3268	1.1669	1.0773	1.0649	**1.0322**	1.0336	0.9570	0.9684	**0.9509**
	± STD	–	0.1006	0.1521	**0.0367**	0.0494	0.0597	0.0387	**0.0259**	0.0341	0.0260
	Time [s]	238.07	0.34	1.45	2.32	1.53	2.99	8.98	10.95	9.02	18.15
Tamanrasset	**# training samples**	2319	100	250				500			
	RMSE	1.1370	1.4880	1.3412	1.2383	**1.1948**	1.2083	1.1912	1.1616	**1.1349**	1.1605
	± STD	–	0.1759	0.1148	**0.0352**	0.0400	0.0426	0.0374	0.0250	**0.0224**	0.0258
	MAE	0.8714	1.0951	0.9905	0.9487	**0.9169**	0.9220	0.9090	0.8837	**0.8679**	0.8804
	± STD	–	0.0955	0.0658	0.0324	**0.0318**	0.0385	0.0248	0.0216	**0.0201**	0.0217
	NMSE	0.4960	0.8602	0.6947	0.5888	**0.5483**	0.5608	0.5449	0.5180	**0.4944**	0.5170
	± STD	–	0.2160	0.1222	**0.0336**	0.0368	0.0396	0.0342	0.0223	**0.0196**	0.0232
	Time [s]	314.01	0.36	1.55	2.33	1.48	2.87	9.09	11.03	9.23	18.16

For an easiest readability of these results, we averaged them over the ten stations and reported the outcome in Table 5.7. This last shows that it is possible to save about 75 % of the samples (from 2200 to 500 training samples), while keeping almost unchanged the prediction accuracy. If one desires to save more training samples, active learning demonstrate to be the right approach to construct a small training set and in the same time to get a good prediction performance. Slight differences are observed between the three active learning methods. The best one is however RSAL in particular if one takes into consideration the stability as performance criterion.
Regarding the computation times, the Random, PAL, DAL and RSAL consumed 1.53, 2.37, 1.56 and 3.05 seconds for learning the prediction model on 250 training samples, and 9.08, 11.05, 9.13 and 18.33 seconds for the case of 500 training samples, respectively. As expected, the RSAL strategy results the most time demanding as it involves the training of two models without bagging the training sets. It is however noteworthy that computation times are much shorter for the proposed active learning strategies with respect to the KRR-Full (258.88 [s]).

Table 5.7: Average RMSE Standard Deviation (STD), Average MAE Standard Deviation (STD), Average NMSE Standard Deviation (STD), and average computation time obtained on the ten data sets.

Method			**RMSE±STD**		**MAE±STD**		**NMSE±STD**		**Time [s]**
# training samples	**∼2200**	**Full**	1.6901	–	1.2712	–	0.7919	–	258.88
	100	**Initial**	2.1931	0.2237	1.6147	0.1184	1.3310	0.3130	0.35
	250	**Random**	1.8934	0.0989	1.4153	0.0625	0.9881	0.1031	1.53
		PAL	1.8015	0.0503	1.3709	**0.0381**	0.8973	0.0496	2.37
		DAL	1.7909	0.0532	1.3580	0.0443	0.8855	0.0516	1.56
		RSAL	**1.7835**	**0.0496**	**1.3559**	0.0396	**0.8775**	**0.0486**	3.05
	500	**Random**	1.7669	0.0403	1.3299	0.0292	0.8620	0.0382	9.08
		PAL	1.7214	0.0287	1.3071	0.0236	0.8198	0.0268	11.05
		DAL	1.7279	0.0299	1.3063	0.0262	0.8270	0.0279	9.13
		RSAL	**1.7149**	**0.0263**	**1.3038**	**0.0209**	**0.8124**	**0.0239**	18.33

5.5 Conclusion

In this chapter, we presented a study focused on active learning applied in two different application fields, namely spectroscopy and wind speed forecasting. In particular, for spectroscopy we propose three active learning methods to construct the training set for a prediction based on the PLSR, RR, KRR and SVR. Two different active learning strategies (PAL, DAL) applicable with any regression method and a specific strategy for the support vectors regression (SVR-DAL). By

contrast, we propose three active learning strategy (PAL, DAL, RSAL) applied to wind speed prediction in Algeria, developed for regression approach based on the kernel ridge regression (KRR).

Starting from a small and suboptimal training set, an iterative process selects from a set of unlabeled data the samples which are considered very significant for the prediction process, i.e., those able to give smaller prediction errors while minimizing the number of required training samples and thus the costs for collecting the final training set.

In global, four methods are proposed, the first strategy uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors, while the second one relies on the idea to add samples that are distant from the available training samples. The third strategy is based on the selection of samples which exhibit a high expected prediction error, and the last, a specific strategy for SVR based on the selection of the samples distant from the current support vectors.

The experimental results obtained on different data set fields show good capabilities of the proposed strategies for selecting significant samples. In general, the proposed methods are characterized by higher performances in terms of errors and stability, with respect to a completely random selection strategy.

For similar error values, active learning reduces substantially the number of required training samples when compared to random sampling. The best active learning strategy appears the one based on pool of regression (PAL) for spectroscopy and residual estimation (RSAL) for wind speed prediction. It is, however, the most computationally demanding since it needs the training of different regressors to build the pool for spectroscopy and residual for wind speed.

Chapter 6

Final Conclusions and Future Works

Contents

6.1 Contributions and conclusion

In this thesis, several methodological aspects and theoretical solutions have been presented and investigated to circumvent some of the most important issues that the chemometrics and wind speed community has been facing recently. Such issues have been identified with the objective to increase the weight of the new technology in numerous applications. In particular, the methodological issues that have been considered in the present thesis are the: i) Residual Correction Concept; ii) Active Learning Regression. In the following, we briefly summarize the conclusions drawn for each of the addressed topics.

For the first issue, in Chapter 4, we proposed a new method based on a residual-based correction (RBC) concept. Its underlying idea is to correct any adopted regressor, by analyzing and modeling its residual errors directly in the feature space. RBC is therefore not a regressor but a correction method, whose aim is not to reach the best achievable accuracy for a given data set but to possibly improve the estimation model of a given regressor. Experimental results based on linear and nonlinear regression methods and conducted on different data sets in infrared spectroscopy point out that the proposed approach can improve the estimation accuracy achieved by traditional regression and compared with another well-known method based on exploiting the errors to improve the estimation process so-called AdaBoost.R.
The use of residual correction strategy can improve the estimation accuracy with respect to single predictor. The proposed approach is independent from the kind of functional estimator used to predict the chemical concentrations of interest. Comparison with AdaBoost.R, RBC performs better than AdaBoost.R and less exposed to the overfitting risk with respect to AdaBoost.R. However, when compared with traditional regressor in terms of time computation, the RBC method needs a huge time computation to tune the best parameters of the model during the training phase.

For the second issue, in Chapter 5, we introduced the active learning approach for regression problems to estimate the chemical concentrations from spectroscopic data. Some strategies are tested on different linear and nonlinear methods. The proposed general active learning is divided into two strategies; the first method is based on adding samples that are distant from the current training samples in the feature space, while the second is based on the pool of regressors. Other methods proposed specifically for SVM, the method based on the pool of regressors and an additional strategy based on the selection of the samples distant from the current support vectors are presented. The experimental results on three different real data sets show higher performances of the proposed strategies in terms of both accuracy and stability with respect to a completely random selection strategy. Comparing them, the best active strategy appears the one based on the pool of regressors for both linear and nonlinear methods.
Similarly to the previous section, in Chapter 5 the active learning has been applied for regression

problems for wind speed prediction, but in this case we proposed an active learning strategy developed for regression approach based only on the kernel ridge regression (KRR). In particular, we proposed three different active learning strategies. The first strategy uses a pool of regressors in order to select the samples with the greater disagreements between the different regressors. The second one relies on the idea to add samples that are distant from the available training samples, while the third strategy is based on the selection of samples which exhibit a high expected prediction error. The experimental results obtained on ten different Algerian stations show good capabilities of the proposed strategies for selecting significant samples. In general, the proposed methods are characterized by higher performances in terms of errors (RMSE, MAE and NMSE) and stability, with respect to a completely random selection strategy. By averaging over the three error measures, the error reduction is of the order of 5 % and the improvement in terms of stability (STD) is of the order of 40 %. For similar error values, active learning reduces substantially the number of required training samples when compared to a random sampling. The best active learning strategy appears to be the one based on residual estimation (RSAL). Computational cost is the main drawback of the active learning approach. This drawback is however widely compensated by the reduction of required training samples, which involves a substantial benefit in terms of time and economic cost incurred by the demanding training sample collection as well as the prediction of the learning model when compared to a full-learning. Moreover, in some cases, active learning has shown to be able to outperform PLSR-Full, RR-Full, KRR-Full and SVR-Full regarding of accuracy thanks to its intrinsic ability to filter out the training samples.

6.2 Perspectives and future work

The contributions provided in this thesis were mainly focused on the development of new strategies for exploiting errors to improve the estimation process and new active learning methodologies to address the problem of training sample collection for regression problem. Such contributions have been critically analyzed considering the state-of-the-art of the related research topics, and have been compared with reference approaches by means of in-depth testing experiments. The results turned out to be satisfactory, and confirmed that the research reported in this dissertation have made interesting contributions to the faced methodological issues. We cite here some possible futures research directions.

1. **Residual correction concept:**

    - The correction mechanism could be implemented as well beyond the first level of correction. For instance, a second level of correction would involve the use of a third regressor (second corrector) to correct the errors incurred by the first corrector. A third level of correction would require in total four regressors, and so on. Experiments

would be needed to analyze up to which level the correction process could be envisioned without being trapped in generalization problems like AdaBoost.R. In order to limit the overfitting risk, weights could be assigned to the different correctors to control their possible misleading effects.

2. **Active learning:**

   - In this thesis, we focused on PLSR, RR, KRR and SVR, the active selection of the training samples could be used in combination with other supervised regression approaches, for example gaussian process.
   - The initial training set was chosen in a random way, more sophisticated initialization strategies could be envisioned in order to further improve the performances of the active learning process.
   - Other active learning strategies could be considered for SVR in particular, for instance by combining sample distance, pool disagreement and Lagrange multiplier information sources.
   - Another interesting idea is to adopt a hybrid strategy which will take advantage of both semi-supervised and active learning. In the beginning, the most significant samples are selected by the active learning technique. After that, the selected samples are automatically labeled (zero cost) with the semi-supervised approach which has the advantage in this case to eliminate the human expert and thus resulting in fully-automatic hybrid approach.

List of Publications

Published Journal Papers

[**J.1**] **F. Douak**, N. Benoudjit, and F. Melgani, "A two-stage regression approach for spectroscopic quantitative analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 109, no. 1, pp. 34-41, 2011.

[**J.2**] L. Douha, N. Benoudjit, **F. Douak**, and F. Melgani, "Support vector regression in spectrophotometry: An experimental study," *Critical Reviews in Analytical Chemistry*, vol. 42, no. 3, pp. 214-219, 2012.

[**J.3**] **F. Douak**, F.Melgani, N. Alajlan, E. Pasolli, Y. Bazi, and N. Benoudjit, "Active learning for spectroscopic data regression," *Journal of Chemometrics*, vol. 26, no. 7, pp. 374-383, 2012.

[**J.4**] **F. Douak**, F.Melgani, and N. Benoudjit, "Kernel ridge regression with active learning for wind speed prediction," *Applied Energy*, vol. 103, pp. 328-340, 2013.

Conference Proceedings

[**C.1**] **F. Douak**, F. Melgani, E. Pasolli, and N. Benoudjit, "SVR active learning for product quality control," *in Information Science, IEEE 2012 11th International Conference on Signal Processing and their Applications (ISSPA)*, Montreal, Quebec, Canada, 3-5 July 2012, pp. 1113-1117.

[**C.2**] **F. Douak**, N. Benoudjit, and F. Melgani, "Design of a multiblock general regression neural network for wind speed prediction in algeria," *in IEEE 8th International Workshop on Systems, Signal Processing and their Applications (WOSSPA2013)*, Algiers, Algeria, 12-15 May 2013, pp. 390-395.

Bibliography

[1] M. Kutner, C. Nachtsheim, and J. Neter, *Applied linear regression models*. McGraw-Hill Irwin, Boston, MA, 2004. (page 2)

[2] V. Vapnik, *The nature of statistical learning theory*. Springer-Verlag, New York, USA, 1995. (pages 2, 33)

[3] D. Patterson, *Artificial neural networks, theory and applications*. Prentice-Hall, Upper Saddle River, 1996. (pages 2, 33)

[4] D. M. Bates and D. G. Watts, *Nonlinear regression: iterative estimation and linear approximations*. Wiley Online Library, 1988. (page 2)

[5] T.-M. Huang, V. Kecman, and I. Kopriva, *Kernel based algorithms for mining huge data sets: Supervised, Semi-supervised, and Unsupervised learning*. Springer-Verlag Berlin Heidelberg, 2006. (pages 2, 3)

[6] D. L. Olson and D. Delen, *Advanced data mining techniques*. Berlin Heidelberg: Springer-Verlag, 2008. (page 2)

[7] A. Paoli, F. Melgani, and E. Pasolli, "Clustering of hyperspectral images based on multiobjective particle swarm optimization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 12, pp. 4175–4188, 2009. (page 3)

[8] I. Jolliffe, *Principal component analysis, 2nd Edn.* New York: Springer-Verlag, 2002. (page 3)

[9] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural networks*, vol. 13, no. 4, pp. 411–430, 2000. (page 3)

[10] X. Zhu, "Semi-supervised learning literature survey," Comput. Sci. Tech. Rep. 1530, Univ. Wisconsin-Madison, Tech. Rep., 2005. (page 3)

[11] Y. Bazi and F. Melgani, "Semisupervised pso-svm regression for biophysical parameter estimation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 6, pp. 1887–1895, 2007. (page 3)

[12] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," *Synthesis lectures on artificial intelligence and machine learning*, vol. 3, no. 1, pp. 1–130, 2009. (page 3)

[13] Y. Bazi and F. Melgani, "Semisupervised gaussian process regression for biophysical parameter estimation," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2010, pp. 4248–4251. (page 3)

[14] V. T. Svensson, "Handling complex data in food analysis: a chemometric approach," Ph.D. dissertation, University of Copenhagen, Faculty of Life Sciences, Department of Food Science, Quality & Technology, 2008. (page 3)

[15] S. S. Sekulic, J. Wakeman, P. Doherty, and P. A. Hailey, "Automated system for the on-line monitoring of powder blending processes using near-infrared spectroscopy: Part ii. qualitative approaches to blend evaluation," *Journal of pharmaceutical and biomedical analysis*, vol. 17, no. 8, pp. 1285–1309, 1998. (page 3)

[16] A. Rinnan, F. v. d. Berg, and S. B. Engelsen, "Review of the most common pre-processing techniques for near-infrared spectra," *TrAC Trends in Analytical Chemistry*, vol. 28, no. 10, pp. 1201–1222, 2009. (pages 3, 11)

[17] V. Centner, O. De Noord, and D. Massart, "Detection of nonlinearity in multivariate calibration," *Analytica chimica acta*, vol. 376, no. 2, pp. 153–168, 1998. (page 4)

[18] C. Miller, "Sources of non-linearity in near-infrared methods," *NIR news*, vol. 4, no. 6, pp. 3–5, 1993. (page 4)

[19] M. Lei, L. Shiyan, J. Chuanwen, L. Hongling, and Z. Yan, "A review on the forecasting of wind speed and generated power," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 4, pp. 915–920, 2009. (page 4)

[20] J. Morales, R. Minguez, and A. Conejo, "A methodology to generate statistically dependent wind speed scenarios," *Applied Energy*, vol. 87, no. 3, pp. 843–855, 2010. (page 4)

[21] H. Liu, E. Erdem, and J. Shi, "Comprehensive evaluation of arma–garch (-m) approaches for modeling the mean and volatility of wind speed," *Applied Energy*, vol. 88, no. 3, pp. 724–732, 2011. (page 4)

[22] E. Erdem and J. Shi, "Arma based approaches for forecasting the tuple of wind speed and direction," *Applied Energy*, vol. 88, no. 4, pp. 1405–1414, 2011. (page 4)

[23] M. Lange and U. Focken, *Physical approach to short-term wind power prediction*. Springer-Verlag, New York, LLC, 2009. (page 4)

[24] D. Fadare, "The application of artificial neural networks to mapping of wind speed profile for energy application in nigeria," *Applied Energy*, vol. 87, no. 3, pp. 934–942, 2010. (pages 4, 23)

[25] G. Li and J. Shi, "On comparing three artificial neural networks for wind speed forecasting," *Applied Energy*, vol. 87, no. 7, pp. 2313–2320, 2010. (pages 4, 23)

[26] ——, "Application of bayesian model averaging in modeling long-term wind speed distributions," *Renewable Energy*, vol. 35, no. 6, pp. 1192–1202, 2010. (pages 4, 23)

[27] M. Mohandes, S. Rehman, and S. Rahman, "Estimation of wind speed profile using adaptive neuro-fuzzy inference system (anfis)," *Applied Energy*, vol. 88, no. 11, pp. 4024–4032, 2011. (pages 4, 23)

[28] M. Mohandes, T. Halawani, S. Rehman, and A. Hussain, "Support vector machines for wind speed prediction," *Renewable Energy*, vol. 29, no. 6, pp. 939–947, 2004. (pages 4, 23)

[29] M. Mohandes, S. Rehman, and T. Halawani, "A neural networks approach for wind speed prediction," *Renewable Energy*, vol. 13, no. 3, pp. 345–354, 1998. (pages 4, 23)

[30] G. Kariniotakis, G. Stavrakakis, and E. Nogaret, "Wind power forecasting using advanced neural networks models," *IEEE transactions on Energy Conversion*, vol. 11, no. 4, pp. 762–767, 1996. (pages 4, 23)

[31] A. More and M. Deo, "Forecasting wind with neural networks," *Marine structures*, vol. 16, no. 1, pp. 35–49, 2003. (pages 4, 23)

[32] P. Erto, A. Lanzotti, and A. Lepore, "Wind speed parameter estimation from one-month sample via bayesian approach," *Quality and Reliability Engineering International*, vol. 26, no. 8, pp. 853–862, 2010. (pages 4, 23)

[33] H. Bouzgou and N. Benoudjit, "Multiple architecture system for wind speed prediction," *Applied Energy*, vol. 88, no. 7, pp. 2463–2471, 2011. (pages 4, 23)

[34] T. Barbounis, J. Theocharis, M. Alexiadis, and P. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Transactions on Energy Conversion*, vol. 21, no. 1, pp. 273–284, 2006. (pages 4, 23)

[35] T. Barbounis and J. Theocharis, "A locally recurrent fuzzy neural network with application to the wind speed prediction using spatial correlation," *Neurocomputing*, vol. 70, no. 7, pp. 1525–1542, 2007. (pages 4, 23)

[36] ——, "Locally recurrent neural networks for wind speed prediction using spatial correlation," *Information Sciences*, vol. 177, no. 24, pp. 5775–5797, 2007. (pages 4, 23)

[37] D.-W. Sun, *Infrared spectroscopy for food quality analysis and control*. Academic Press, 2009. (page 5)

[38] D. Ballabio, "Chemometric characterisation of physical-chemical fingerprints of food products," Ph.D. dissertation, Dipartimento di Scienze e Tecnologie Alimentari e Microbiologiche, vol. Dottorato di Ricerca in Biotecnologie degli Alimenti. Milano: Università degli Studi di Milano, 2006. (page 5)

[39] F. Douak, N. Benoudjit, and F. Melgani, "A two-stage regression approach for spectroscopic quantitative analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 109, no. 1, pp. 34–41, 2011. (pages 6, 9, 47, 75)

[40] Y. Himri, A. Boudghene Stambouli, B. Draoui, and S. Himri, "Review of wind energy use in algeria," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 4, pp. 910–914, 2009. (page 6)

[41] A. Stambouli, "Promotion of renewable energies in algeria: Strategies and perspectives," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 2, pp. 1169–1181, 2011. (pages 6, 24)

[42] F. Chellali, A. Khellaf, A. Belouchrani, and A. Recioui, "A contribution in the actualization of wind map of algeria," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 2, pp. 993–1002, 2011. (page 6)

[43] T. Burton, D. Sharpe, N. Jenkins, and E. Bossanyi, *Wind energy handbook*. Wiley, 2001. (pages 6, 19, 20, 21, 22)

[44] A. Stambouli, Z. Khiat, S. Flazi, and Y. Kitamura, "A review on the renewable energy development in algeria: Current perspective, energy scenario and sustainability issues," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 7, pp. 4445–4460, 2012. (page 6)

[45] A. Ainouche and H. Ainouche, "Promotion of renewable energies in algeria for a sustainable development and better future for next generations," *Renewable Energy journal, Algiers*, 2006. [Online]. Available: http://www.worldenergy.org/documents/p000983.doc (page 6)

[46] B. Settles, "Active learning literature survey," *Comput Sci Tech Rep 1648, University of Wisconsin, Madison*, 2009. (pages 7, 69, 70)

[47] P. Mitra, C. Murthy, and S. K. Pal, "A probabilistic active support vector learning algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 3, pp. 413–418, 2004. (page 7)

[48] S. Zomer, M. Del Nogal Sánchez, R. G. Brereton, and J. L. Perez Pavon, "Active learning support vector machines for optimal sample selection in classification," *Journal of chemometrics*, vol. 18, no. 6, pp. 294–305, 2004. (page 7)

[49] E. Pasolli and F. Melgani, "Active learning methods for electrocardiographic signal classification," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 6, pp. 1405–1416, 2010. (page 7)

[50] E. Pasolli, F. Melgani, and Y. Bazi, "Support vector machine active learning through significance space construction," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 3, pp. 431–435, 2011. (page 7)

[51] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *J. Artif. Intell. Res*, vol. 4, pp. 129–145, 1996. (page 7)

[52] K. Fukumizu, "Statistical active learning in multilayer perceptrons," *IEEE Transactions on Neural Networks*, vol. 11, no. 1, pp. 17–26, 2000. (page 7)

[53] R. Castro, R. Willett, and R. Nowak, "Faster rates in regression via active learning," *Advances in Neural Information Processing Systems*, vol. 18, pp. 179–186, 2006. (page 7)

[54] M. Sugiyama, "Active learning in approximately linear regression based on conditional expectation of generalization error," *The Journal of Machine Learning Research*, vol. 7, pp. 141–166, 2006. (page 7)

[55] R. Burbidge, J. J. Rowland, and R. D. King, "Active learning for regression based on query by committee," in *Intelligent Data Engineering and Automated Learning-IDEAL*. Springer, 2007, pp. 209–218. (page 7)

[56] M. Sugiyama and N. Rubens, "A batch ensemble approach to active learning with model selection," *Neural Networks*, vol. 21, no. 9, pp. 1278–1286, 2008. (page 7)

[57] M. Sugiyama and S. Nakajima, "Pool-based active learning in approximate linear regression," *Machine learning*, vol. 75, no. 3, pp. 249–274, 2009. (page 7)

[58] J. Paisley, X. Liao, and L. Carin, "Active learning and basis selection for kernel-based linear models: A bayesian perspective," *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2686–2700, 2010. (page 7)

[59] F. Douak, F. Melgani, N. Alajlan, E. Pasolli, Y. Bazi, and N. Benoudjit, "Active learning for spectroscopic data regression," *Journal of Chemometrics*, vol. 26, no. 7, pp. 374–383, 2012. (pages 7, 9, 69)

[60] F. Douak, F. Melgani, E. Pasolli, and N. Benoudjit, "Svr active learning for product quality control," in *Information Science, IEEE 2012 11th International Conference on Signal Processing and their Applications (ISSPA)*, Montreal, Quebec, Canada, 3-5 July 2012, pp. 1113–1117. (pages 7, 9, 69)

[61] F. Douak, F. Melgani, and N. Benoudjit, "Kernel ridge regression with active learning for wind speed prediction," *Applied Energy*, vol. 103, pp. 328–340, 2013. (pages 7, 9, 69)

[62] F. Douak, N. Benoudjit, and F. Melgani, "Design of a multiblock general regression neural network for wind speed prediction in algeria," in *IEEE 8th International Workshop on Systems, Signal Processing and their Applications (WOSSPA2013)*, Algiers, Algeria, 12-15 May 2013, pp. 390–395. (page 9)

[63] S. S. Sekulic, H. W. Ward, D. R. Brannegan, E. D. Stanley, C. L. Evans, S. T. Sciavolino, P. A. Hailey, and P. K. Aldridge, "On-line monitoring of powder blend homogeneity by near-infrared spectroscopy," *Analytical Chemistry*, vol. 68, no. 3, pp. 509–513, 1996. (page 11)

[64] M. Blanco, J. Coello, A. Eustaquio, H. Iturriaga, and S. Maspoch, "Analytical control of pharmaceutical production steps by near infrared reflectance spectroscopy," *Analytica chimica acta*, vol. 392, no. 2, pp. 237–246, 1999. (page 11)

[65] Y. Ozaki, R. Cho, K. Ikegaya, S. Muraishi, and K. Kawauchi, "Potential of near-infrared fourier transform raman spectroscopy in food analysis," *Applied spectroscopy*, vol. 46, no. 10, pp. 1503–1507, 1992. (page 11)

[66] M. Blanco, J. Coello, J. M. Garcia Fraga, H. Iturriaga, S. Maspoch, and J. Pagès, "Determination of finishing oils in acrylic fibers by near-infrared reflectance spectroscopy," *Analyst*, vol. 122, no. 8, pp. 777–781, 1997. (page 11)

[67] D. L. Massart, B. Vandeginste, S. Deming, Y. Michotte, and L. Kaufman, *Chemometrics: a textbook*. Elsevier Amsterdam, 1988. (page 11)

[68] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003. (page 11)

[69] M. Volmer, *Infrared spectroscopy in clinical chemistry, using chemometric calibration techniques*. University Library Groningen, 2001. (page 12)

[70] M. Bishop, J. Duben-Engelkirk, and E. Fody, *Clinical chemistry: principles, procedures, correlations*. Lippincott Williams & Wilkins Philadelphia, PA, USA, 2000. (page 12)

[71] Y. Ozaki, W. F. McClure, and A. A. Christy, *Near-infrared spectroscopy in food science and technology*. Wiley-Interscience, 2006. (page 13)

[72] V. Acha, H. Naveau, and M. Meurens, "Extractive sampling methods to improve the sensitivity of ftir spectroscopy in analysis of aqueous liquids." *Analusis*, vol. 26, no. 4, pp. 157–163, 1998. (page 13)

[73] S. C. Ahalt and F. J. E., "Vector quantization using artificial neural networks models," in *Proceedings of the International Workshop on Adaptive Method and Emergent Techniques for Signal Processing and Communications*, 1993, pp. 42–62. (page 13)

[74] M. Meurens, "Acquisition et traitement du signal spectrophotométrique," in *La spectroscopie infrarouge et ses applications analytiques, D. Bertrand et E. Dufour. Collection sciences et techniques agroalimentaires*, 2000, pp. 199–211. (page 13)

[75] D. Donoho, "High-dimensional data analysis: The curses and blessings of dimensionality. lecture delivered at the mathematical challenges of the 21st century conference of the american math. society, los angeles, august 6-11," 2000. (page 14)

[76] M. Verleysen, "Learning high-dimensional data," *Limitations and Future Trends in Neural Computation*, pp. 141–162, 2003. (page 14)

[77] S. Verboven and M. Hubert, "Libra: a matlab library for robust analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 75, no. 2, pp. 127–136, 2005. (page 14)

[78] *Dataset provided by Prof. Marc Meurens, Université catholique de Louvain, BNUT, meurens@bnut.ucl.ac.be. orange juice datasets*. [Online]. Available: http://www.ucl.ac.be/mlg/ (page 15)

[79] F. Rossi, A. Lendasse, D. François, V. Wertz, and M. Verleysen, "Mutual information for the selection of relevant variables in spectrometric nonlinear modelling," *Chemometrics and intelligent laboratory systems*, vol. 80, no. 2, pp. 215–226, 2006. (page 15)

[80] *Diesel data set*. [Online]. Available: http://www.eigenvector.com/data/SWRI/ (page 16)

[81] *Tecator meat sample dataset*. [Online]. Available: http://lib.stat.cmu.edu/datasets/tecator (page 18)

[82] C. Krier, F. Rossi, D. François, and M. Verleysen, "A data-driven functional projection approach for the selection of feature ranges in spectra with ica or cluster analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 91, no. 1, pp. 43–53, 2008. (page 18)

[83] J. Adams, M. Maslin, and E. Thomas, "Sudden climate transitions during the quaternary," *Progress in Physical Geography*, vol. 23, no. 1, pp. 1–36, 1999. (page 20)

[84] G. Joselin Herbert, S. Iniyan, E. Sreevalsan, and S. Rajapandian, "A review of wind energy technologies," *Renewable and Sustainable Energy Reviews*, vol. 11, no. 6, pp. 1117–1145, 2007. (page 20)

[85] J. Palutikoff, X. Guo, and J. Halliday, "The reconstruction of long wind speed records in the uk," in *Proceedings of the Thirteenth British Wind Energy Association Conference*, 1991, pp. 275–280. (page 21)

[86] S. Soman, H. Zareipour, O. Malik, and P. Mandal, "A review of wind power and wind speed forecasting methods with different time horizons," in *North American Power Symposium (NAPS)*, vol. 4, Sep. 2010, pp. 1–8. (pages 21, 23)

[87] S. Han, "Study of short-term wind power prediction method, north china electric power university," pp. 3–4, Jun. 2008. (page 21)

[88] P. Pinson, "Estimation of the uncertainty in wind power forecasting," Ph.D. dissertation, Paris, France: Ecole des Mines de Paris, 2006. (page 22)

[89] L. Landberg, "Short-term prediction of the power production from wind farms," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 80, no. 1, pp. 207–220, 1999. (page 23)

[90] C. W. Potter and M. Negnevitsky, "Very short-term wind forecasting for tasmanian power generation," *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 965–972, 2006. (page 23)

[91] M. Lange and U. Focken, "New developments in wind energy forecasting," in *Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE*, 2008, pp. 1–8. (page 23)

[92] S. Li, "Wind power prediction using recurrent multilayer perceptron neural networks," in *IEEE Power Engineering Society General Meeting*, vol. 4, 2003. (page 23)

[93] A. Sfetsos, "A comparison of various forecasting techniques applied to mean hourly wind speed time series," *Renewable Energy*, vol. 21, no. 1, pp. 23–35, 2000. (page 23)

[94] P. Fonte and J. Quadrado, "Ann approach to wecs power forecast," in *10th IEEE Conference on Emerging Technologies and Factory Automation*, vol. 1, 2005, pp. 1069–1072. (page 24)

[95] C. Saunders, A. Gammerman, and V. Vovk, "Ridge regression learning algorithm in dual variables," in *Proceedings of the 15th International Conference on Machine Learning*. Morgan Kaufmann, 1998, pp. 515–521. (page 24)

[96] C. Orsenigo and C. Vercellis, "Kernel ridge regression for out-of-sample mapping in supervised manifold learning," *Expert Systems with Applications*, vol. 39, no. 9, pp. 7757–7762, 2012. (page 24)

[97] *African Economic Outlook*, 2007–2008. [Online]. Available: www.oecd.org/dev/publications/africanoutlook (page 24)

[98] B. Hemmateenejad, R. Miri, M. Akhond, and M. Shamsipur, "Qsar study of the calcium channel antagonist activity of some recently synthesized dihydropyridine derivatives. an application of genetic algorithm for variable selection in mlr and pls methods," *Chemometrics and intelligent laboratory systems*, vol. 64, no. 1, pp. 91–99, 2002. (page 33)

[99] N. Benoudjit, E. Cools, M. Meurens, and M. Verleysen, "Chemometric calibration of infrared spectrometers: selection and validation of variables by non-linear models," *Chemometrics and Intelligent Laboratory Systems*, vol. 70, no. 1, pp. 47–53, 2004. (pages 33, 43, 54)

[100] T. Eklöv, P. Mårtensson, and I. Lundström, "Selection of variables for interpreting multivariate gas sensor data," *Analytica Chimica Acta*, vol. 381, no. 2, pp. 221–232, 1999. (page 33)

[101] P. Geladi, "Some recent trends in the calibration literature," *Chemometrics and Intelligent Laboratory Systems*, vol. 60, no. 1, pp. 211–224, 2002. (page 33)

[102] J. Zupan and J. Gasteiger, *Neural networks for chemists: an introduction*. Verlag Chemie, Weinheim, 1993. (page 33)

[103] B. Schölkopf and A. Smola, *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press,Cambridge, MA, USA, 2001. (page 33)

[104] A. García-Reiriz, P. Damiani, M. Culzoni, H. Goicoechea, and A. Olivieri, "A versatile strategy for achieving the second-order advantage when applying different artificial neural networks to non-linear second-order data: Unfolded principal component analysis/residual bilinearization," *Chemometrics and Intelligent Laboratory Systems*, vol. 92, no. 1, pp. 61–70, 2008. (page 33)

[105] P. Fidencio, R. Poppi, and J. de Andrade, "Determination of organic matter in soils using radial basis function networks and near infrared spectroscopy," *Analytica Chimica Acta*, vol. 453, no. 1, pp. 125–134, 2002. (page 33)

[106] B. Smola, A. Schölkopf, "A tutorial on support vector regression: Tech. rep," Royal Holloway College, Univ. London, London, U.K., NeuroCOLT Tech. Rep. NC-TR-98-030, Tech. Rep., 1998. (pages 33, 41, 42, 43)

[107] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. 1st ed. Cambridge, U.K.: Cambridge University Press, 2000. (pages 33, 34, 35, 36, 37, 40, 41, 42, 43)

[108] A. Gusnanto, Y. Pawitan, J. Huang, and B. Lane, "Variable selection in random calibration of near-infrared instruments: ridge regression and partial least squares regression settings," *Journal of Chemometrics*, vol. 17, no. 3, pp. 174–185, 2003. (page 36)

[109] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1, pp. 37–52, 1987. (page 37)

[110] J. Miller and J. Miller, *Statistics and chemometrics for analytical chemistry*. sixth ed., England, 2010. (page 37)

[111] H. Wold, "Nonlinear estimation by iterative least squares procedures," *Research papers in statistics*, pp. 411–444, 1966. (page 38)

[112] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936. (page 38)

[113] N. Benoudjit, "Variable selection and neural networks for high-dimensional data analysis: application in infrared spectroscopy and chemometrics," Ph.D. dissertation, Universite Catholique de Louvain, Belgium, 2003. (page 38)

[114] L. T. Aarhus, "Nonlinear empirical modeling using local pls models," Ph.D. dissertation, University of Oslo, 1994. (page 38)

[115] S. Wold, H. Martens, and H. Wold, "The multivariate calibration problem in chemistry solved by the pls method," *Matrix Pencils*, vol. 973, pp. 286–293, 1983. (page 38)

[116] S. Wold, A. Ruhe, H. Wold, and W. Dunn III, "The collinearity problem in linear regression. the partial least squares (pls) approach to generalized inverses," *SIAM Journal on Scientific and Statistical Computing*, vol. 5, no. 3, pp. 735–743, 1984. (page 38)

[117] P. Geladi and B. Kowalski, "Partial least-squares regression: a tutorial," *Analytica chimica acta*, vol. 185, pp. 1–17, 1986. (page 38)

[118] A. Lorber, L. Wangen, and B. Kowalski, "A theoretical foundation for the pls algorithm," *Journal of Chemometrics*, vol. 1, no. 1, pp. 19–31, 1987. (page 38)

[119] N. Benoudjit, C. Archambeau, A. Lendasse, J. A. Lee, and M. Verleysen, "Width optimization of the gaussian kernels in radial basis function networks." in *ESANN*, vol. 2, 2002, pp. 425–432. (page 41)

[120] L. Douha, N. Benoudjit, F. Douak, and F. Melgani, "Support vector regression in spectrophotometry: An experimental study," *Critical Reviews in Analytical Chemistry*, vol. 42, no. 3, pp. 214–219, 2012. (pages 41, 47)

[121] L. Bruzzone and F. Melgani, "Robust multiple estimator systems for the analysis of biophysical parameters from remotely sensed data," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 43, no. 1, pp. 159–174, 2005. (page 42)

[122] J. Park and I. Sandberg, "Approximation and radial-basis-function networks," *Neural computation*, vol. 5, no. 2, pp. 305–316, 1993. (page 43)

[123] T. Poggio and F. Girosi, "Networks for approximation and learning," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1481–1497, 1990. (page 43)

[124] Y.-S. Hwang and S.-Y. Bang, "An efficient method to construct a radial basis function neural network classifier," *Neural Networks*, vol. 10, no. 8, pp. 1495–1503, 1997. (page 43)

[125] R. J. Howlett and L. C. Jain, *Radial basis function networks 2: new advances in design*. Physica-Verlag Heidelberg Printed in Germany, 1st edition, 2001. (page 43)

[126] M. Orr, "Introduction to radial basis function networks," Technical Reports, www.anc.ed.ac.uk/~mjo/papers/intro.ps, Tech. Rep., 1996. (page 44)

[127] O. Devos, C. Ruckebusch, A. Durand, L. Duponchel, and J. Huvenne, "Support vector machines (svm) in near infrared (nir) spectroscopy: Focus on parameters optimization and model interpretation," *Chemometrics and Intelligent Laboratory Systems*, vol. 96, no. 1, pp. 27–33, 2009. (page 47)

[128] B. Nicolai, K. Theron, and J. Lammertyn, "Kernel pls regression on wavelet transformed nir spectra for prediction of sugar content of apple," *Chemometrics and intelligent laboratory systems*, vol. 85, no. 2, pp. 243–252, 2007. (page 47)

[129] D. Massart, B. Vandeginste, and L. Buydens, *Handbook of chemometrics and qualimetrics*. Elsevier Science, 1997, vol. 20. (page 47)

[130] D. François, "High-dimensional data analysis: optimal metrics and feature selection," Ph.D. dissertation, Universite Catholique de Louvain, Belgium, 2007. (page 47)

[131] Y. Freund and R. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," in *Computational learning theory*. Springer, 1997, pp. 23–37. (page 47)

[132] J. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, pp. 367–378, 2002. (page 47)

[133] ——, "Multivariate adaptive regression splines," *The annals of statistics*, pp. 1–67, 1991. (page 47)

[134] L. Breiman, J. Friedman, R. Olshen, and C. Stone, "Classification and regression trees," *Wadsworth International Group*, 1984. (page 47)

[135] X. Shao, X. Bian, and W. Cai, "An improved boosting partial least squares method for near-infrared spectroscopic quantitative analysis," *Analytica chimica acta*, vol. 666, no. 1, pp. 32–37, 2010. (page 47)

[136] Y. Zhou, J. Jiang, W. Lin, H. Zou, H. Wu, G. Shen, and R. Yu, "Boosting support vector regression in qsar studies of bioactivities of chemical compounds," *European journal of pharmaceutical sciences*, vol. 28, no. 4, pp. 344–353, 2006. (pages 47, 63)

[137] M. Zhang, Q. Xu, and D. Massart, "Boosting partial least squares," *Analytical chemistry*, vol. 77, no. 5, pp. 1423–1431, 2005. (page 47)

[138] H. Shinzawa, J. Jiang, and Y. Ritthiruangdej, P.and Ozaki, "Investigations of bagged kernel partial least squares (kpls) and boosting kpls with applications to near-infrared (nir) spectra," *Journal of chemometrics*, vol. 20, no. 8-10, pp. 436–444, 2006. (page 47)

[139] Y. Zhou, J. Jiang, H. Wu, G. Shen, R. Yu, and Y. Ozaki, "Dry film method with ytterbium as the internal standard for near infrared spectroscopic plasma glucose assay coupled with boosting support vector regression," *Journal of chemometrics*, vol. 20, no. 1-2, pp. 13–21, 2006. (page 47)

[140] H. Drucker, "Improving regressors using boosting techniques," in *Proceedings of the 14th International Conference on Machine Learning, Nashville, TN*, July 8-12 1997, pp. 107–115. (page 47)

[141] H. van der Voet, "Comparing the predictive accuracy of models using a simple randomization test," *Chemometrics and Intelligent Laboratory Systems*, vol. 25, no. 2, pp. 313–323, 1994. (page 52)

[142] N. Benoudjit, D. Francois, M. Meurens, and M. Verleysen, "Spectrophotometric variable selection by mutual information," *Chemometrics and Intelligent Laboratory Systems*, vol. 74, no. 2, pp. 243–251, 2004. (page 54)

[143] O. Maimon and L. Rokach, *Data mining and knowledge discovery handbook*. Springer-Verlag New York Inc, 2005. (page 63)