TOOLS FOR PREDICTION OF ENVIRONMENTAL PROPERTIES OF CHEMICALS BY QSAR/QSPR WITHIN REACH

AN APPLICABILITY DOMAIN PERSPECTIVE

PhD dissertation by:

SAHIGARA FAIZAN ABDULRAZAK

DEPARTMENT OF ENVIRONMENTAL SCIENCES
UNIVERSITY OF MILANO-BICOCCA

ACADEMIC YEAR: 2012/2013
PHD ENROLMENT CYCLE: XXV

TOOLS FOR PREDICTION OF ENVIRONMENTAL PROPERTIES OF CHEMICALS BY QSAR/QSPR WITHIN REACH

AN APPLICABILITY DOMAIN PERSPECTIVE

SAHIGARA FAIZAN ABDULRAZAK

TUTOR: PROF. ROBERTO TODESCHINI

CO-TUTORS: DR. VIVIANA CONSONNI
DR. DAVIDE BALLABIO

Preface

This research was funded within the Marie Curie Initial Training Network - Environmental ChemOinformatics (ECO-ITN). Aimed at developing the careers for Environmental Cheminformatians, this Initial Training Network (ITN) has been mainly implemented to provide advanced training in both environmental and computational approaches. This ITN is functioning within several research groups located in 5 EU countries over a period of four years until September 2013. Additionally, external collaborations with other research networks and industrial partners open doors to new future opportunities for the ECO participants. Internal trainings at other ECO partner groups facilitate a better way of knowledge exchange within the training network while the flexibility to opt for external collaborators allow participants to take their research a step ahead on a global level.

One of the important considerations within the new European legislation on chemicals and their safe use REACH (Registration, Evaluation, Authorization and restriction of CHemicals) is to minimize the number of animal testing by replacing them with suitable alternatives such as in-silico methods, wherever possible. The primary goals of ECO-ITN can be fits well with these considerations since the trainees within this project are exposed to several state-of-the-art computational approaches which can then be applied to towards the development of novel automated strategies for risk assessment of chemicals.

Thesis outline

As the title suggests, this work is mainly focussed at providing an Applicability Domain (AD) perspective towards the QSAR/QSPR models predicting environmental properties relevant to REACH regulations. A well-defined AD is one of the prerequisites for a predictive model before it is considered as validated for regulatory purposes. The main idea behind compiling this thesis is to provide the reader with all the major insights towards defining a model's AD where it can reliably predict the modelled endpoint for new test samples.

The thesis contents are divided into three major parts summarized as follows:

The first section is an introductory part which guides through the scope of validated QSARs within REACH. A regulatory insight is presented towards the consideration of QSAR methodologies as one of the alternatives to animal testing and the possibility to use its reliable predictions directly or include them as supplementary information within a Weight of Evidence approach. The major principles towards QSAR validation are briefly discussed with a particular attention towards the prerequisite to have a well-defined AD for reliable predictions.

The second section initially discusses several classical approaches proposed in the existing literature towards defining the AD of QSAR models in its descriptor space. In theory, all these approaches attempted to characterize the interpolation space where a model is capable of making reliable predictions. The major highlights for each approach include a) the basic strategy followed to characterize the interpolation space and b) the major advantages and/or limitations in addressing the model's AD. Later, a novel AD approach based on the classical *k*-Nearest Neighbours principle is introduced which also features the major highlight of the thesis. This discussion includes the motivation behind proposing the new approach followed by the description of the underlying algorithm. Finally, an AD perspective is provided towards the application of a novel pseudo-distance called *Locally-centred Mahalanobis distance* for outlier detection. The results derived from this newly proposed outlier detection approach provides an excellent platform to better understand the impact of extreme training outliers on the defined AD using different AD approaches as well as to verify if the test samples detected as outliers in the training space could hint for them being unreliably predicted and thus, likely to get excluded from the model's AD.

The final section of this thesis work discusses the results derived implementing previously introduced classical and two novel AD approaches on several QSAR models from the existing literature. Some of these models predicting significant environmental properties were intended to contribute towards REACH implementation and thus, served as ideal case studies to better evaluate for their AD. The performance of both the novel AD approaches was evaluated with respect to the classical methodologies. Moreover, presence of consensus test samples excluded from the model's AD with different approaches, further allowed to reflect upon the similar trends within the underlying algorithms and also added to the confidence in rendering those test samples being unreliably predicted.

Last but not the least, general conclusions and future prospects for this thesis work were briefly discussed. All the relevant research articles accepted by the scientific journals were listed and reported in the appendix.

Acknowledgements

Big thanks to the entire Milano Chemometrics and QSAR Research group. Undoubtedly, first of all I would like to thank Prof. Roberto Todeschini for believing in my abilities and making me a part of his research group. There was so much I learnt from you in the past three years and there is much more beyond this to explore. At this point, I would like extend my thanks to Dr. Viviana Consonni and Dr. Davide Ballabio, my co-tutors who always made sure that my research was being carried out in a right direction. Viviana, thanks for looking after all the bureaucratic aspects and other issues at the university allowing me to focus on my research tasks. Davide, your valuable suggestions and hints made my life easier at work. Andrea and Alberto, thanks for always being there for me whenever I needed your assistance and suggestions. Kamel, it had been a pleasure working with you, sharing the same lab as well as the project. Matteo, your friendly nature and passion to help was very much appreciated. Francesca, the newest and youngest member in the group, you had been more like a friend than a colleague to me. Thanks to the administrative staff at the University of Milano-Bicocca for looking after my reimbursements and other financial matters.

I would like to take this opportunity to thank the Marie Curie Actions for funding my research within the Environmental Chemoinformatics Initial Training Network. ECO project manager Eva Schlosser, thanks for keeping us always updated and for your support and guidance whenever needed. A big thanks to the research groups of Dr. Igor Tetko and Prof. Ian Nicholls for their wonderful hospitality during my internships. My special thanks to the entire CEHTRA family, most importantly Dr. Paul Thomas who gave wings to my career life. Carine, Carole, Rija and Julien, it had been a great pleasure working with you as a part of my industrial training in Lyon and keeping in touch since then. Special thanks to Andrew Worth for his valuable suggestions and guidance. Your work on applicability domain helped me throughout my doctoral research. Prof. Marco Vighi, it was a pleasure to know your ideas and opinions about QSARs. I appreciated your guidance whenever needed. Stefan Brandmaier, you had been an amazing colleague,

collaborator and a fantastic friend! Thanks for making me a part of your research article.

My special thanks to Primina Monga and the Brignolis for their wonderful hospitality and affection. In the past three years, I never felt I was away from home. Let me take this opportunity to thank the Lombardos for their unconditional love and help in improving on my Italian skills. Gianfranco, learning Golf from you was so much fun. You and your entire family, especially Anna Maria always made me feel so proud.

Eva or shall I say Evanthia, you had been the biggest support as a colleague and as a friend, through most of my doctoral degree. Despite of the several disagreements and meaningless fights we had, our friendship never changed. My best friends Shruti and Vinaya, your presence in my life makes me feel complete and precious. Domenico, officially my first Italian friend, thanks to you and your family for making me feel at home since the day I arrived in Milan. Frantonio, Pierangela and Emmanuelle, thanks for your support and guidance. I thoroughly enjoyed your company and the good times we spent together. Alberto, thanks for dragging me along to the most amazing restaurants and gelato shops I have ever visited in town. Giorgio, discovering the city of Milan could not be as interesting without your company. I really enjoyed visiting several interesting places and always appreciated your valuable suggestions. Aggeliki, you had been a big support in all good and bad times.

Ivan and Manuela, since the day I met both of you, lunch hours had been amongst the most amazing moments at the university campus. Anu, it had been a pleasure to meet you. Thanks for always motivating me to accomplish my thesis goals. Paola, Simone and Ivan, thanks for your company in making the weekends so much fun. Jessica and Jessica, Cesare and Matteo, thanks for your company, it means a lot to me.

Contents

PART I. BACKGROUND **Page**

1. Introduction

PART II. THEORY

2. Classical ways of characterizing the interpolation space

3. A novel k-Nearest Neighbours based Applicability Domain evaluation

4. Outlier detection from an Applicability Domain perspective

PART III. APPLICATIONS

5. Case studies

Contents

PART I

BACKGROUND

Chapter 1

Introduction

As an alternative to animal testing, provisions for Quantitative Structure Activity Relationship (QSAR) predictions towards regulatory purposes are well-discussed and documented within the framework of a new European Community regulation for the safe use of chemicals – REACH. This chapter discusses the regulatory perspective towards the acceptance of QSARs and introduces the major principles for their validation, paying particular attention towards defining their Applicability Domain in order to differentiate the reliable predictions from extrapolations.

1.1 Scope of QSARs within REACH framework

REACH is a European legislation on chemicals that came into force in 2007. It is mainly focussed on the risk assessment of chemicals for their safe use [1]. As a part of this regulation, a major responsibility lies on the industry towards risk management by providing all the necessary information about the chemicals and their properties. The outcome of REACH is mainly aimed at enhancing the human health protection and minimizing the environmental hazards by the safer handling of chemicals as well as replacement of hazardous chemicals with suitable alternatives [2,3].

One of the major objectives of implementing REACH is to minimize the animal testing. To achieve this, usefulness of non-testing approaches has been highlighted and as a result, REACH encourages the use of cost-effective methods like QSARs, Read-Across approaches and expert systems.

The possibility to train QSARs based on high quality and reliable data can allow evaluation of several physicochemical and biological properties for various chemicals relevant to REACH. Moreover, the results derived from QSARs can also be used as a part of Weight of Evidence (WoE) approach. Thus, QSARs and other relevant approaches can be significant for REACH in filling the data gaps prevailing towards the evaluation of several chemical properties. Depending on the reliability in their predictions, the QSAR models can directly replace the test data otherwise can be used as supplementary information to improve the transparency in evaluations [2,3].

QSARs are based on the principle that similar chemical structures can lead to similar biological activities. In general, QSARs can be thought as a combination of data analysis and statistical methods that are aimed towards finding a trend within the descriptor values of chemicals, which in turn can explain the corresponding trend in their biological activities [4]. A basic workflow of a QSAR includes data collection and pretreatment, followed by implementation of a model development technique (for instance, Linear Regression, Artificial Neural Network and so on) and finally evaluating the model performance through internal and external validation.

Enormous experimentally derived data for several significant endpoints is readily available from the existing literature. This data collection can be an excellent input to train models towards predicting several physicochemical and biological activities for new test compounds. This idea was realised in the past decades and consequently, several QSAR models emerged since then predicting different endpoints. From time to time, more efficient algorithms were proposed towards model development, thus a range of different methodologies were in place from a Simple Linear Regression to Artificial Neural Networks.

1.2 QSAR predictions may be reliable yet restricted

In theory, applicability of QSAR models irrespective of their predictive reliability is limited. These limitations of a model can be referred to its structural domain and the response space which defines the scope of that model [5]. Usually, the predictive models are trained using a limited set of chemical structures. The level of structural diversity reflected within a training set strongly relies on the information contained for instance, functional groups present, chemical categories covered and so on. For instance, a QSAR model trained using only aromatic structures may not be useful in predicting a test set of aliphatic structures. The resulting predictions will be unreliable as they will be beyond the scope of that model. Thus, it is reasonable to expect that the scope of local models is limited, though it shouldn't be confused with their predictive ability.

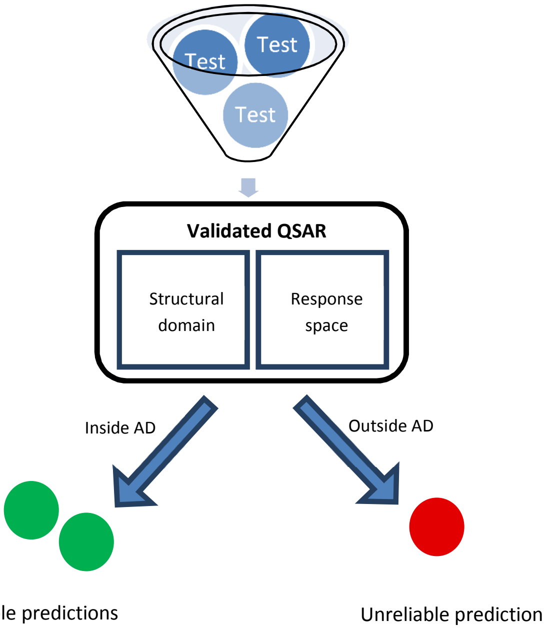

Figure 1.1 *Evaluating the reliability in prediction for new test samples*

It is crucial that a model is used for predicting only those test samples that are structurally similar to the samples used for training purpose [5-9]. It makes sense because structural similarity implies similarity in the descriptor values, which in turn can fit the trend in deriving a modelled endpoint. In other words, test samples must fall within the structural domain described by its descriptor space. Since, a model is usually aimed to identify a reliable trend between the descriptor values and the modelled endpoint, the prediction of a structurally similar test sample is likely to fall within the response domain of the training samples. Figure 1.1 informs that test samples satisfying the limitations of a model within its structural domain and response space fall within the Applicability Domain (AD) and are thus, associated with a reliable prediction. On the contrary, those excluded from the AD where unreliably predicted.

1.3 Validated QSARs for their regulatory acceptance

As discussed in the earlier section, QSARs can be thought amongst one of the promising non-testing approaches towards regulatory use. However, to ensure that the QSAR predictions are reliable, several conditions are necessary to be met by such predictive models. The regulatory authorities need to make sure that a QSAR model was strictly validated before being applied for regulatory assessment of chemicals. Before a QSAR model can be accepted for regulatory use, its validity has to be demonstrated, the test sample being predicted has to fall within the AD of that model and reliability in the modelling approach has to be well-documented in order to provide the transparency in the underlying algorithm.

No formal adoption procedure is suggested for QSARs within REACH. Thus, information provided to the regulatory authorities towards demonstrating the model's validity and reliability in its predictive ability will be evaluated in deciding upon the adequacy of a model and its predictions for regulatory acceptance [3]. To address validation procedure, REACH referred to the principles for QSAR validation adopted by OECD in 2004. These principles are internationally agreed and each of them highlights several key aspects relevant to the regulatory acceptance of QSAR models [3,5-10].

For its regulatory consideration, a validated QSAR must be associated with these principles listed in the following order [3,10]:

a) A defined endpoint

As several experimental methods and conditions are feasible towards prediction of a given physicochemical property or a biological effect, the first OECD principle provides information about the endpoint being modelled.

b) An unambiguous algorithm

As several modelling approaches have been proposed from time to time, the second principle tries to bring transparency in the algorithm used towards model development.

c) A defined domain of applicability

In theory, the applicability of a QSAR model is limited to the chemical that are structurally similar to those used to train that model. The third principle tries to highlight this feature and informs about the limitations of a proposed model in its structural domain and response space.

d) Appropriate measures of goodness-of-fit, robustness and predictivity

To better evaluate for the model's performance, it is essential to understand if it's robust, is not overfitted and is able to reliably predict the modelled endpoint for external test samples. To achieve this, the fourth principle for model validation provides with all the necessary information derived performing an internal and external validation using the training and an external test set, respectively.

e) A mechanistic interpretation, if possible

The mechanistic relevance between the set of descriptors used towards model development and the endpoint being modelled, can further add to the confidence in a model, however, it is also understandable that deriving such mechanistic interpretation is not always possible and thus, the fifth principle recommends a model developer to provide mechanistic basis for the descriptors and its relevance to the modelled endpoint, whenever possible.

1.4 Applicability domain for reliable predictions

As discussed earlier, the third principle of QSAR validation deals with defining model's AD. It is one of the prerequisites to have a well-defined AD before a model can be considered as validated. Several approaches have been discussed from time to time in the existing literature towards defining a model's AD and an entire section of this thesis is dedicated discussing these methodologies [5].

In theory, all these approaches attempted to characterize the interpolation space for reliable predictions using different algorithms [6,11-13]. The efficiency of a strategy can be estimated based on its ability to maximize the retention of reliable test predictions. Depending on the nature of endpoint being modelled, QSAR models can be divided into two major categories, regression and classification models. Regression models are implemented for quantitative endpoints, such as LC50 in aquatic toxicity, Bioconcentration factor and so on. On the other hand, classification models deal with endpoints of qualitative nature, for instance if a test molecule is ready biodegradable or not ready biodegradable, is a carcinogen or non-carcinogen. In a case of regression model, the reliability measure is quantitative where a lowest prediction error is desirable, while in the case of classification models, the underlying algorithm tries to achieve reliability by maximizing the allocation of test molecules to their correct classes.

If a test molecule is associated with a very high prediction error or is allocated to a wrong class, the reliability in its prediction decreases. There can be several reasons behind deriving an unreliable prediction for instance, the new test molecule contains some specific functional groups that are unknown to the training space, the test molecule reacts with a specific mode of action which cannot be described well with the set of descriptors used for training that model or there are no structurally similar training molecules identified for a given test molecule. There may be several other explanations behind deriving an unreliable prediction; however, most of them converge to a single conclusion that is the test molecule could be beyond the scope of that model.

One of the major concerns about QSARs from a regulatory perspective is the reliability in their predictions. A QSAR model with a defined a domain of applicability makes predictions with a defined level of reliability. When this model is applied to a new set of test molecules, the resulting predictions that fall within its AD can be associated with that given level of reliability. In other words, there exists a trade-off between the applicability of a model and the reliability in its predictions. Thus, from a regulatory perspective, a prediction falling outside the model's AD is associated with a lower level of reliability. A well-defined AD can allow the regulatory authorities to better evaluate the structural domain in which a model can predict reliably and prevents from extrapolating beyond the scope of that model [2-3].

There are several ways in which a model's AD could be addressed. For instance, in a model's descriptor space, the defined AD can be thought to be restricted to the test molecules with relevant descriptor values; in a mechanistic domain the defined AD can be limited to the test molecules acting based on the same mode of action represented by the training set molecules; in a metabolic domain, the AD can be defined based on the possibilities of the molecules to undergo transformation or get metabolized [2-3]. With growing awareness about the QSAR validation for its regulatory acceptance, the development and implementation of different AD strategies has become one of the promising areas of research in the field of QSAR in the current years.

From time to time, more efficient approaches have been proposed overcoming several of the prevailing issues, however until now, no strategy towards defining a model's AD has been officially accepted or recognized [14]. Nevertheless, emerging awareness towards non-testing approaches is likely to keep the QSARs in focus. A joint effort between regulators, industry and researchers can shape a better future of such alternative methods

PART II

THEORY

Chapter 2

Classical ways of characterizing the interpolation space

This chapter discusses several classical approaches towards defining the Applicability Domain of a QSAR model in its descriptor space. The major focus is given on the methodology used to characterize the interpolation space where the model is expected to make reliable predictions. Most of the discussed approaches were associated with their own advantages and limitations. Their implementation on a two-dimensional simulated dataset and the resulting contour plots allowed a better understanding of their defined domain of applicability.

2.1 An introduction to the AD methodologies

Characterization of the interpolation space is very significant to define the AD for a given QSAR model. This characterised space can be associated with reliable predictions derived from the model and helps the user to evaluate the reliability in prediction for a given query molecule Depending upon how efficiently the interpolation space is defined, the clarity and transparency in distinguishing quality predictions from extrapolations also improves. Several AD approaches have been already proposed and primarily they all differ in the way how they characterize the interpolation space defined by the descriptors used. They can be classified into following four major categories based on the methodology used for interpolation space characterization in the model's descriptor space: range-based methods, geometric methods, distance-based methods and Probability Density Distribution based methods [5-6,11-13].

This chapter discusses all the above-mentioned classical approaches which were then implemented on the two-dimensional simulated datasets shown in Figures 2.1 and 2.2.

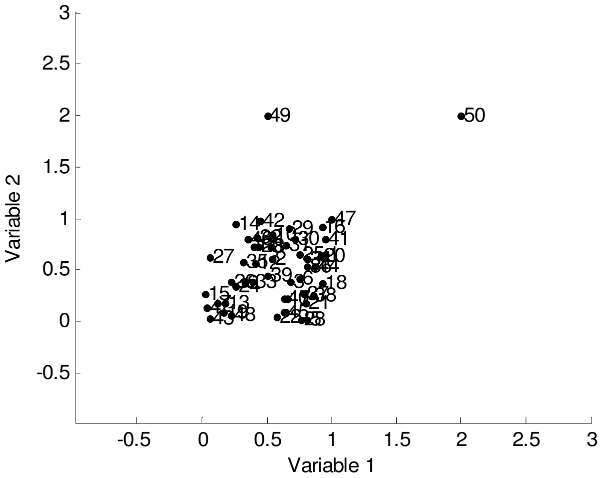

Figure 2.1 *Scatter plot for the first simulated dataset*

As shown in Figure 2.1, the first simulated dataset consists of a cluster with 48 training samples and 2 isolated samples (49 and 50) which were localized distant from each other as well as the cluster.

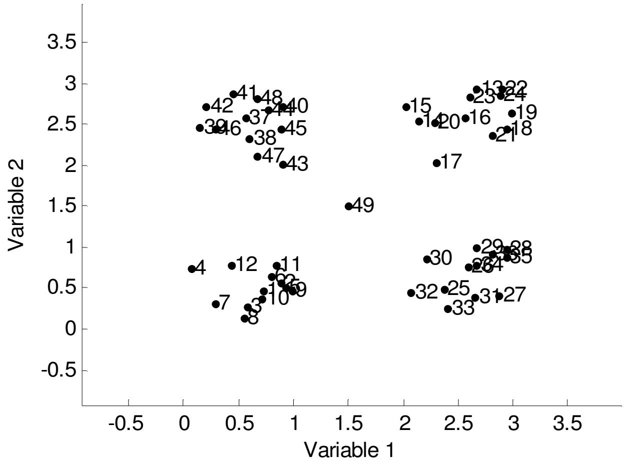

Figure 2.2 *Scatter plot for the second simulated dataset*

As shown in Figure 2.2, the second simulated dataset comprised of four clusters of samples and an isolated sample (49) between them.

The AD defined implementing each of these approaches was visualized using contour plots for the simulated datasets derived projecting several data points enough to fill its training space. These plots allowed a better understanding of the features relevant to the interpolation space characterized with these existing approaches and wherever possible, also reflected the prevailing drawbacks in their methodologies.

2.2 Range-based and Geometric Methods

These are considered as the simplest methods to characterize a model's interpolation space.

2.2.1 Bounding Box

This approach considers the range of individual descriptors used to build the model. Assuming a uniform distribution, resulting domain of applicability can be imagined as a Bounding Box which is a p-dimensional hyper-rectangle defined on the basis of maximum and minimum values of each descriptor used to build the model. The sides of this hyper-rectangle are parallel with respect to the coordinate axes. However, there are several drawbacks associated with this approach: since only descriptor ranges are taken into consideration, empty regions in the interpolation space cannot be identified and also the correlation between descriptors cannot be taken into account [11,12].

Figure 2.3 provides with the contour plot implementing Bounding Box on the simulated datasets introduced earlier. As shown in the Figure 2.3a, the characterized interpolation space accounts for a considerable empty space between the cluster and two isolated samples.

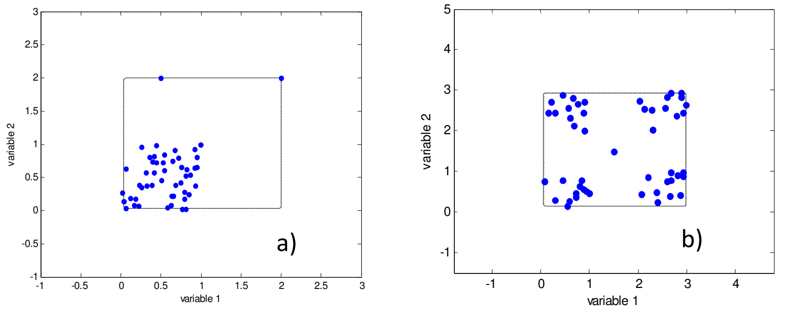

Figure 2.3 *Contour plots for the simulated datasets derived implementing Bounding Box. First simulated dataset (2.3a) and second simulated dataset (2.3b)*

This implies that the presence of one or more outliers in the training extremities can have a huge impact on the defined AD, which is not desirable. Figure 2.3b provides with the contour plot for the second simulated dataset using Bounding Box. As expected, empty regions between the clusters were considered within the AD as a result of which the isolated sample (49) was rendered as reliable.

2.2.2 PCA Bounding Box

Principal Component Analysis (PCA) transforms the original data into a new coordinate system by the rotation of axes, such that the new axes are orthogonal to each other and aligned in the direction having maximum variance within the data. These new axes are called Principal Components (PCs) representing the maximum variance within the dataset [15]. A M-dimensional hyper-rectangle (where M is the number of significant components) is obtained similar to the previous approach by considering the projection of the molecules in the principal component space, however taking into account the maximum and minimum values for the PCs. The implementation of Bounding Box with PCA can overcome the problem of correlation between descriptors but empty regions within the interpolation space still remains an issue [11-13]. Moreover, selection of appropriate number of components is significant to implement this approach. For all the case studies discussed in this thesis, only those PCs having eigenvalues greater than the average eigenvalue (which corresponds to 1 when data are autoscaled) were considered. This criterion was chosen in order not to include the influence of noise that is taken into account by the remaining PCs with lower eigenvalues. However, in the case of two dimensional datasets (like the simulated dataset being discussed here), by default both the resulting PCs were considered.

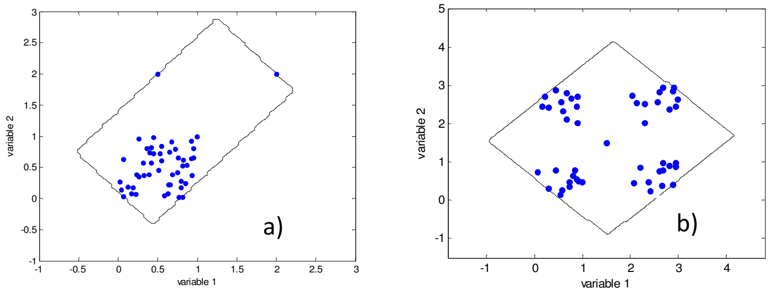

Figure 2.4 *Contour plot for the simulated datasets derived implementing PCA Bounding Box. First simulated dataset (2.4a), second simulated dataset (2.4b).*

The contour plot in Figure 2.4a was derived implementing the PCA Bounding Box approach on the first simulated dataset. As clear from the figure, the issue of accounting for undesirable empty regions in the defined interpolation space still prevails. As shown in the Figure 2.4b, like the earlier approach, PCA bounding box included unnecessary empty regions between the clusters within the defined AD for the second simulated dataset.

2.2.3 Convex Hull

With this geometric approach, interpolation space is defined by the smallest convex area containing the entire training set. Implementing a Convex Hull could be challenging with increasing data complexity [16]. For two or three dimensional data, several algorithms are proposed; however, increase in dimensions contributes to the order of complexity. This could be a major drawback for this approach since in practice, not all the QSARs are limited to a small number of molecular descriptors. Several descriptors at times are needed to efficiently identify the trends in the modelled endpoint. Thus in theory, the implementation of this approach is limited to QSAR models with very limited number of descriptors. Apart from this issue, set boundaries are analysed without considering the actual data distribution. Similar to the range-based approaches, Convex Hull cannot identify the potential internal empty regions within the interpolation space [11-12].

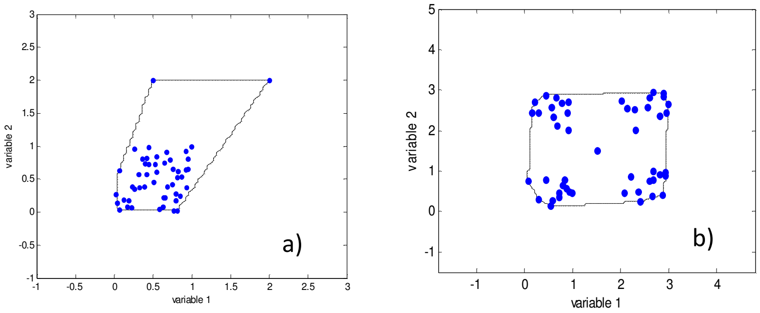

Figure 2.5 *Contour plots derived for the simulated datasets implementing Convex Hull. First simulated dataset (2.5a), second simulated dataset (2.5b).*

Figure 2.5a shows the convex hull defined for the first simulated dataset. The defined hull reflects the interpolation space for reliable predictions. Like the range-based approaches, this strategy cannot overcome the existing limitation towards accounting for the empty regions. The AD defined for the second simulated dataset is shown in the contour plot of Figure 2.5b. The derived convex hull enclosed all the four clusters within a common interpolation space thus including the empty regions between the clusters within the defined AD.

The implementation of this approach in this case was quite simple as the simulated datasets were two-dimensional. In practice, QSAR models can have much higher level of complexity with multiple descriptors which could render this approach quite time consuming.

2.3 Distance-Based Methods

These approaches calculate the distance of test molecules from a defined point, (usually the data centroid) within the descriptor space of the training data. The general idea is to compare the distances measured between this defined point and the test molecules with a pre-defined threshold. The threshold is a user-defined parameter and is set to maximize the separation of dense regions within the original data. However, the cut-off value does not entirely reflect the actual data density [5-6,11-13]. No strict rules were evident from the literature about defining thresholds for distance-based approaches and thus it is up to the user how to define them.

2.3.1 Centroid-based distance approach

In this approach, the distances of the training molecules from their centroid are calculated and based on a user-defined criterion, a cut-off distance value is considered as the threshold. For all the case studies dealt in this thesis, the distance value of the training molecules from their centroid corresponding to the 95th percentile was considered as the threshold. Later, the distances of the test samples from the centroid of the training set were derived and compared with the threshold value. If they were lesser or equal to the threshold, those test molecules were included within the model's AD, else discarded.

In theory, this approach can be implemented using a wide range of distance measures available in the literature however, for all the case studies dealt in this thesis work, following three distance measures will be considered: Euclidean, Manhattan and Mahalanobis distances.

Table 2.1 *Formulas for different distance measures*

Distance measure	**Formula**
Euclidean	$d_{st} = \sqrt{\sum_{j=1}^{p} \left(x_{sj} - x_{tj} \right)^2}$
Manhattan	$d_{st} = \sum_{j=1}^{p} \left\| x_{sj} - x_{tj} \right\|$
Mahalanobis	$d_{st} = \sqrt{(\mathbf{x}_s - \mathbf{x}_t)^{\mathrm{T}} \mathbf{S}^{-1} (\mathbf{x}_s - \mathbf{x}_t)}$ where **S** is the covariance matrix

Given a multidimensional matrix **X** whose rows represent molecules and columns their corresponding descriptor values, Table 2.1 provides with the formulas to derive three different distances between two objects *s* and *t* described by *p* variables. x_{sj} and x_{tj} represent the *j*th variable describing the objects *s* and *t*, respectively. $\mathbf{x}_s$ and $\mathbf{x}_t$ represent the *p*-dimensional vectors for the objects s and t, respectively [17].

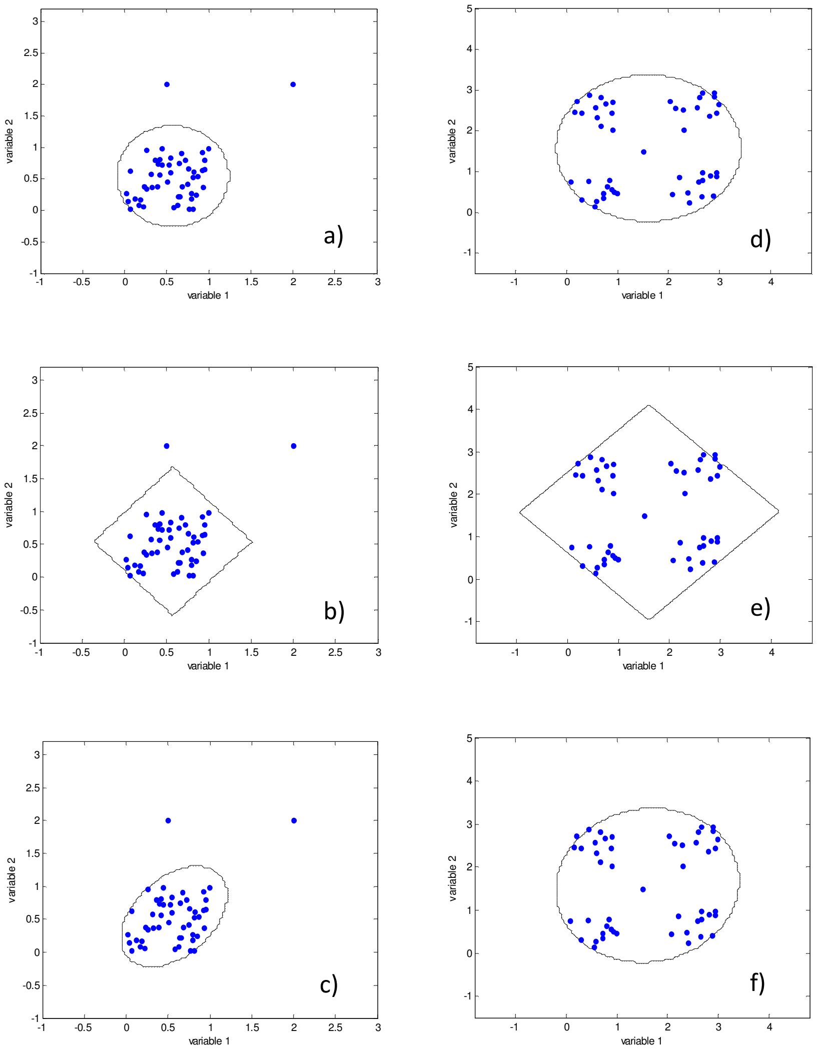

Figure 2.6 *Contour plots derived for the simulated datasets implementing centroid-based distance approach. First simulated dataset: Euclidean (2.6a), Manhattan (2.6b), Mahalanobis (2.6c). Second simulated dataset: Euclidean (2.6d), Manhattan (2.6e), Mahalanobis (2.6f).*

Iso-distance contours constitute the regions having constant distance measures and generally their shapes differ with approaches according to the distance measure considered, for example, ellipsoids for Mahalanobis or spherical for Euclidean distances [12].

Figure 2.6 shows the contour plots derived on the both simulated datasets using three different distance measures. As the threshold was set to 95th percentile, the two isolated training samples were not included in the defined AD with all the three distance measures. The interpolation space mainly represented the regions around the cluster; the only difference was in the shape of the iso-distance contours depending on the distance measure used.

Approaches based on calculating leverages are also quite recommended for defining the AD of a QSAR model [18]. Leverage of a query chemical is proportional to its Mahalanobis distance measure from the centroid of the training set. For a given descriptor matrix **X** with rows as molecules and columns representing the descriptor values, its leverage matrix (**H**) is obtained with the following equation :

$$\mathbf{H} = \mathbf{X}\left(\mathbf{X}^{\mathbf{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathbf{T}} \tag{2.1}$$

where **X** is the model matrix while $\mathbf{X}^{\mathrm{T}}$ is its transpose matrix.

Diagonal values in the **H** matrix represent the leverage values for different molecules in a given dataset. The molecules that are far from the centroid will be associated with higher leverages and are considered to be influential in model building. Leverage is proportional to Hotellings T^2 statistic and Mahalanobis distance measure but can be applied only on the regression models. The approach can be associated with a threshold, generally 2.5 times the average of the leverage that corresponds to p+1/n where p is the number of model descriptors while n is the number of training molecules. A query chemical with leverage higher than the warning leverage can be associated with unreliable predictions. Such chemicals are outside the descriptor space and thus be considered outside the AD [11-13].

Figure 2.7 shows the contour plots derived on both the simulated datasets using leverage approach. Based on the above-discussed threshold, the defined AD for the first dataset (Figure 2.7a) was in the form of an ellipsoid oriented in the direction showing maximum variance in the data.

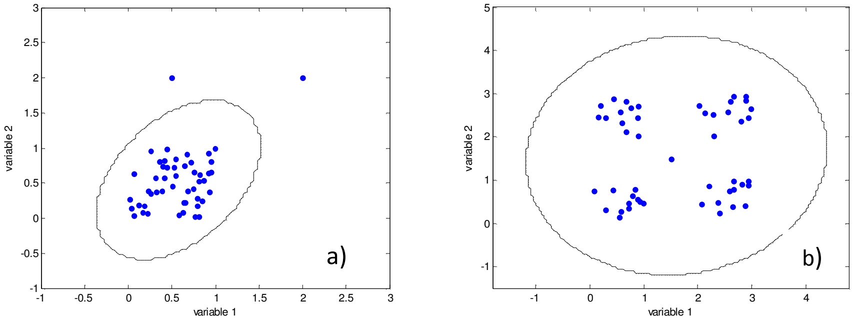

Figure 2.7 *Contour plots derived for the simulated datasets implementing Leverage approach. First simulated dataset (2.7a), second simulated dataset (2.7b)*

The defined AD didn't include the two isolated training samples and the prevailing issue of accounting for empty regions within the training space seems partially resolved here. At a first glance, both the isolated samples are clearly potential outliers in the training space. As a result, it would be reasonable to expect a minimum possible influence of such isolated samples on the resulting AD. The use of above-discussed statistically significant threshold excludes these two outliers and their surrounding descriptor space from the resulting AD, indicating that these isolated samples have no role to play in defining the interpolation space. Thus, the resulting AD was mainly surrounded around the extremities of the huge cluster. On the hand, the defined AD for the second simulated dataset (Figure 2.7b) resembled the AD defined with Euclidean and Mahalanobis distances using the centroid based approach.

2.3.2 K-Nearest Neighbours based approaches

This set of approaches is based on providing similarity measure for a new test molecule with respect to the molecules within the training space. The similarity is accessed by finding the distance of a test molecule from nearest training molecule or its average distances from *k* nearest neighbours in the training set. If these distance values are within the user defined threshold, the test molecule with higher similarity is indicated to have higher number of training neighbours and therefore, is considered to be reliably predicted. Thus, similarity to the training set molecules is significant for this approach in order to associate a test molecule with a reliable prediction [9]. Two variants of the kNN-based approach were implemented.

The first variant of the kNN-based AD approach [9, 19] was implemented by calculating average distances of all the training samples from their *k* nearest neighbours since the choice of thresholds didn't follow any strict rules in the existing literature, the value corresponding to 95th percentile in this vector of average distances was considered as general threshold. If the average distance of a test sample from its *k* nearest training neighbours was lesser than or equal to the threshold value, the test sample was retained within the AD.

Usually for classification purposes where kNN-based approaches are quite commonly applied, a smaller number of nearest neighbours is preferred to avoid any sort of bias. In theory, this makes sense because a higher number of *k* neighbours could take into account training neighbours which may not be significant towards structural similarity. In the literature, a small number of neighbours like $k = 3$ or 5 are quite commonly used to implement different kNN-based approaches.

Figure 2.8 provides with the contour plots derived for both the simulated datasets implementing three different distance measures. To derive the plots, the approach was implemented taking 5 nearest neighbours ($k = 5$) into account. The differences between the defined AD using different distance measures were clearer for the second dataset. The AD was more adapted to the shape of the clusters for Mahalanobis distances (Figure 2.8f) while some empty regions were included in the defined AD with the Manhattan distance (Figure 2.8e).

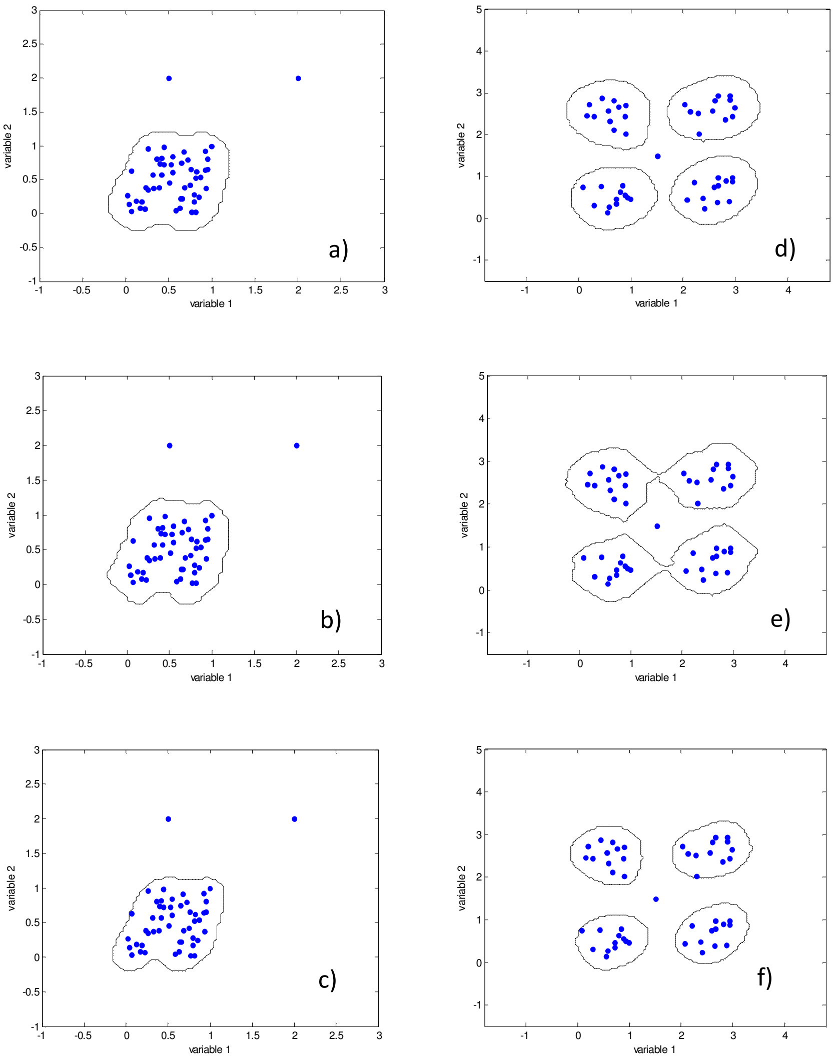

Figure 2.8 *Contour plots derived for the first simulated dataset implementing k-Nearest Neighbours based approach. First simulated dataset: Euclidean (2.8a), Manhattan (2.8b), Mahalanobis (2.8c). Second simulated dataset: Euclidean (2.8d), Manhattan (2.8e), Mahalanobis (2.8f).*

The second variant of the kNN-based AD approach is a nearest neighbour method for probability density function estimation [20]. In this approach, the choice of *k* is crucial and is usually approximately equal to $n^{1/2}$.

In a *p*-dimensional space, let $d_k(\mathbf{xt})$ be the Euclidean distance from a test molecule **xt** to its *k*-th nearest training molecule. The dimensional volume of the p-dimensional sphere having radius $d_k(\mathbf{xt})$ is given by $V_k(\mathbf{xt})$, then the nearest neighbour density estimator at the data point **xt** is given by:

$$f(\mathbf{xt}) = \frac{k/n}{V_k(\mathbf{xt})} = \frac{k/n}{c_p\left[d_k(\mathbf{xt})\right]^p} \tag{2.2}$$

Here, c_p is the volume of the unit sphere in p dimensions. In simple terms, here the probability density function estimate is defined with a window width $d_k(\mathbf{xt})$.

Being prone to the local noise, the overall estimates with this approach do not seem quite convincing. The approach suffers from the irregularities resulting due to the dependence of the resulting estimator on the $d_k(\mathbf{xt})$ function [20].

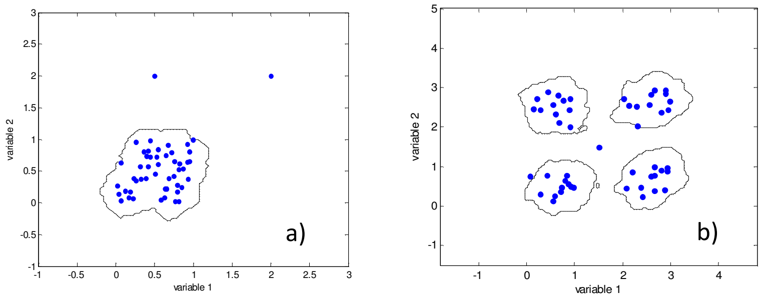

Figure 2.9 *Contour plots derived for the simulated datasets using Nearest Neighbour density estimator.*

Figure 2.9 provides with the contour plots for the simulated datasets implementing this density estimator using $k = 5$. The defined AD seems to be well localized around the data clusters excluding the isolated data samples in both the datasets.

2.4 Probability Density Function Methods

Considered as one of the most advanced approaches for defining AD, these methods are based on estimating the Probability Density Function (PDF) for the given data. This is feasible by both, parametric methods where the density function has the shape of a standard distribution (Gaussian or

Poisson distribution, for instance) and non-parametric methods which do not have any such assumptions concerning the data distribution. A main feature of these approaches is their ability to identify the internal empty regions. Moreover, if needed, the actual data distribution can be reflected by generating concave regions around the interpolation space borders [11-12]. However, there are also several drawbacks associated with this set of approaches, discussed later in this chapter.

Generally these approaches are implemented by estimating probability density of the dataset followed by identifying Highest Density Region that consists of a known fraction (given as user input) from the total probability mass [11].

Let *X* be some random quantity with PDF *f*. Based on this function which actually describes the distribution of *X*, the probabilities associated with *X* can be obtained using the relation, $P(a < X < b) = \int_a^b f(x)\,\mathrm{d}x$ for all a < b.

Consider that some observed data points, assumed to be samples from an unknown probability density function are provided, then the estimate of the density function from these observed data can be constructed using density estimators [20].

For the random variable *X* with density *f*, we can have

$$f(x) = \lim_{h \to 0} \frac{1}{2h} P(x - h < X < x + h) \tag{2.3}$$

Thus for a given *h*, based on the sample proportion falling within the interval, $P(x - h < X < x + h)$ can be easily estimated. Given a weight function *w*, the naive estimator can be written as:

$$\hat{f}(x) = \frac{1}{nh} \cdot \sum_{i=1}^{n} w\left(\frac{x - X_i}{h}\right) \tag{2.4}$$

The above equation indicates that the density estimator was derived by placing a box of width 2*h* and height $(2nh)^{-1}$ on each observation and later the summation on all the observations was performed to obtain the final estimate [20].

Replacing the weight function *w* with a kernel function *K* such that,

$\int_{-\infty}^{\infty} K(x)\,dx = 1$, the kernel estimator can be derived as:

$$\hat{f}(x) = \frac{1}{nh} \cdot \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \tag{2.5}$$

where *h* is the window width, also referred to as smoothing parameter or bandwidth.

Taking the analogy of a naive estimator being the construction of density by sum of boxes centred at different data points, a kernel estimator can be considered as sum of ‘bumps’ on different data points. Shape of such bumps is identified by the kernel function *K* while their width is decided by the window width *h* [20].

The idea of defining the kernel estimator as the summation of bumps placed on different data points can be extended to the multivariate datasets. For a multivariate data set $\mathbf{x}_1,\ldots,\mathbf{x}_n$, the resulting multivariate kernel density estimator with kernel *K* and window width *h* can be defined using the following equation:

$$\hat{f}(\mathbf{xt}) = \frac{1}{n} \cdot \sum_{i=1}^{n} K\left\{\frac{\mathbf{xt} - \mathbf{x}_i}{h}\right\} \tag{2.6}$$

where *K(*$\mathbf{xt}$, $\mathbf{x}$) is the kernel function for *p*-dimensional $\mathbf{xt}$. *K* usually is a radially symmetric unimodal PDF, for instance a standard multivariate normal density function, defined as follows:

$$K(\mathbf{xt}, \mathbf{x}_i) = \frac{1}{h^p \cdot (2\pi)^{p/2}} \cdot \exp\left(-\frac{1}{2}(\mathbf{xt} - \mathbf{x}_i)^{\mathrm{T}}(\mathbf{xt} - \mathbf{x}_i)\right) \tag{2.7}$$

As can be seen in equation 2.6, a single smoothing parameter was used indicating that the kernel placed on all the data points will be equally scaled in all the directions. Like for several other statistical procedures, in multivariate analysis pre-scaling the data could result quite useful as it will avoid getting extreme differences of spread in different coordinate directions. For the data scaling carried out, the standard kernel estimator in equation could be used without using different complicated variants usually involving more than one smoothing parameters [20].

Once the density estimation was carried out for all the training samples, the probability density value of the training sample corresponding to a cut-off percentile was considered as the threshold for AD definition. The test samples **xt** that were associated with a probability density lesser than this threshold were considered outside the model's AD [5].

2.4.1 Gaussian kernels

Among the multivariate kernel density methods which use the standard multivariate normal density function as the kernel function, the following three variants of Gaussian kernel estimators were implemented:

a) Fixed Gaussian kernel

'Fixed' indicates that with this kernel, the smoothing parameter/bandwidth *h* is constant over all training objects.

For this kernel, the optimal bandwidth was calculated as follows [20]:

$$h_{\mathrm{opt}} = A\left(K\right) n^{-1/(p+4)} \tag{2.8}$$

where the constant *A(K)* in *p* dimensions was defined as:

$$A(K) = \left\{4/\left(2p+1\right)\right\}^{1/(p+4)} \tag{2.9}$$

Finally, the kernel estimate of PDF was then derived using the equation 2.7.

There are some drawbacks associated with this kernel method. Since the smoothing is constant, there are several chances of taking spurious noise into account in the estimates. Even in case the estimates were efficiently smoothed, this could be compromised with the essential details in the distribution getting masked [20].

b) Optimized Gaussian kernel

Instead of using a constant smoothing parameter *h*, this is optimized by leave-one-out cross-validation taking into account the differences in standard deviation of the variables [21].

The kernel estimate is derived as:

$$K\left(\mathbf{xt},\mathbf{x}_i\right)=\prod_{j=1}^{p}\frac{1}{h_{opt}\cdot s_j\cdot\sqrt{2\pi}}\exp\left(-\frac{1}{2}\cdot\frac{\left(xt_j-x_{ij}\right)^2}{h_{opt}^2\cdot s_j^2}\right) \qquad (2.10)$$

where s_j is the standard deviation of the *j*th variable.

The optimization procedure requires the estimate of the parameter *h* so that:

$$\max\left[\prod_{i=1}^{n}\hat{f}\left(\mathbf{x}_i\right)\right] \qquad (2.11)$$

where $\hat{f}\left(\mathbf{x}_i\right)$ is the probability density of *i*th sample in cross-validation.

c) Variable Gaussian kernel

With this kernel, smoothing is adapted to the local density of the data. The strategy towards the construction of estimate is quite similar to that with classical kernel estimate, however, allowing the scale parameter for bumps to vary from one point to the other. Moreover, flatter kernels will be allocated to the sparse regions within the data. For all the case studies discussed in this thesis, this kernel was implemented with a bandwidth calculated as the inverse function of the Euclidean distance to *k*-th neighbour [21].

Given kernel function *K*, bandwidth *h*, a positive integer *k* and $d_k\left(\mathbf{xt}\right)$ being the Euclidean distance between the test point **xt** from its k^{th} nearest training neighbour, the variable Gaussian kernel estimate was derived as follows:

$$K\left(\mathbf{xt},\mathbf{x}_i\right)=\prod_{j=1}^{p}\frac{1}{h_{opt}\cdot d_k\left(\mathbf{xt}\right)\cdot s_j\cdot\sqrt{2\pi}}\cdot\exp\left(-\frac{1}{2}\cdot\frac{\left(xt_j-x_{ij}\right)^2}{h_{opt}^2\cdot\left[d_k\left(\mathbf{xt}\right)\right]^2\cdot s_j^2}\right) \qquad (2.12)$$

In this case, the window width on **xt** is proportional to the distance between **xt** and its k^{th} nearest neighbour; the flatter kernels will be associated with sparse data regions. The bandwidth decides the overall smoothing while its response to the very local detail will be depending upon the value of *k*. With this kernel, the estimate will inherit the local smoothing properties, like in the case of ordinary kernel estimator [20].

Figure 2.10 provides with the contour plots derived for the both the simulated datasets implementing the three variants of Gaussian Kernel. For the first dataset, the AD defined in all the three cases were very much adapted to the shape of the cluster and like the distance-based approaches, the percentile approach to define thresholds left both the isolated samples excluded from the AD. The results derived with Fixed and Variable kernels converged to a great extent showing no clearly visible differences. The AD defined with Optimized kernel was slightly more adapted to the shape of the clusters.

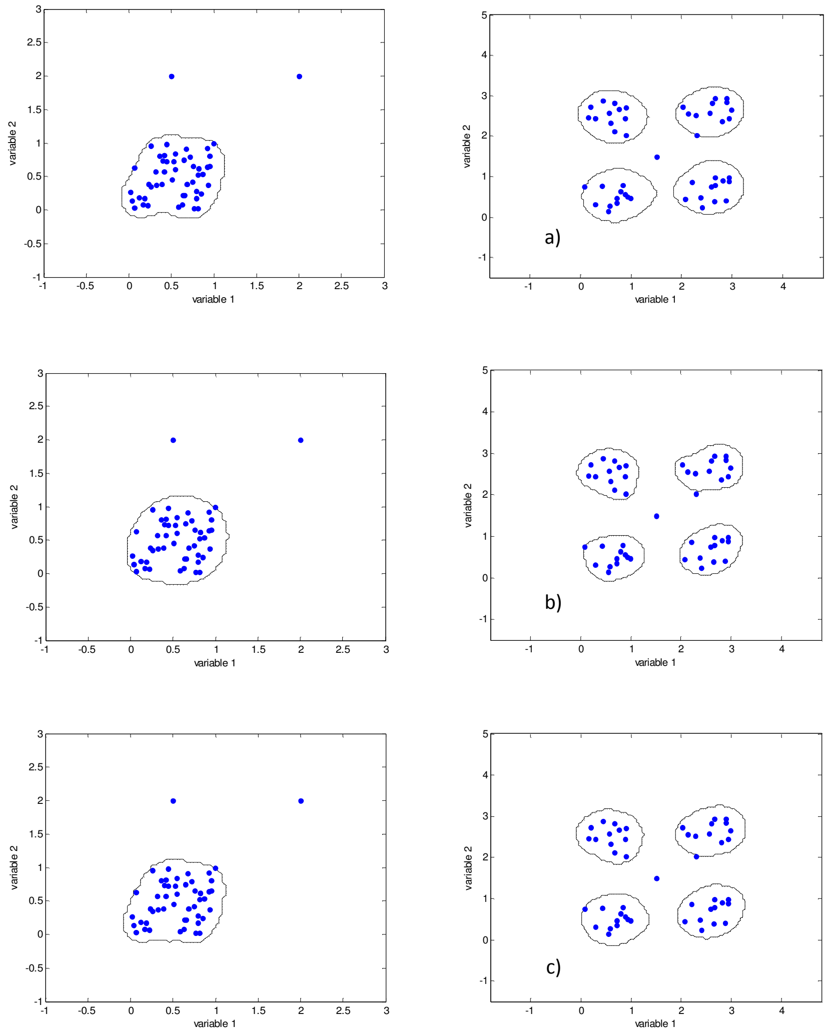

Figure 2.10 : *Contour plots derived for both the simulated datasets implementing three variants of the Gaussian kernel. First simulated dataset: Fixed Gaussian kernel (2.10a), Optimized Gaussian kernel (2.10b) and Variable Gaussian kernel (2.10c), Second simulated dataset: Fixed Gaussian kernel (2.10d), Optimized Gaussian kernel (2.10e) and Variable Gaussian kernel (2.10f).*

2.4.2 Adaptive kernel methods

Combining the features of kernel and Nearest Neighbours approach, this strategy constructs the kernel estimate at observed data points allowing the window width of kernels to vary from one point to another. There is a two stage procedure involved in determining if a given observation is associated within a lower density region [20]:

In the first stage, a pilot estimate is constructed making use of other density estimation methods. This estimate provides a rough understanding of the density and in turn provides with a pattern of bandwidths that are used to construct the adaptive estimator in the second stage.

Step 1: Pilot estimate $\hat{f}(\mathbf{x}_i)$ is found for all the i^{th} observation such that, $\hat{f}(\mathbf{x}_i) > 0$.

Step 2: Local bandwidth factors λ_i are defined as follows:

$$\lambda_i = \left\{\hat{f}(\mathbf{x}_i) / g\right\}^{-\alpha} \tag{2.13}$$

where α is called sensitivity parameter, such that $0 \leq \alpha \leq 1$ and g is the geometric mean of the $\hat{f}(\mathbf{x}_i)$.

Step 3: Adaptive kernel estimate with kernel function *K* and bandwidth *h* can be defined as:

$$\hat{f}(\mathbf{xt}) = \frac{1}{n} \cdot \sum_{i=1}^{n} \frac{1}{\lambda_i^p} \cdot K\left\{\frac{\mathbf{xt} - \mathbf{x}_i}{h \cdot \lambda_i}\right\} \tag{2.14}$$

Dependence of the bandwidth factors on the power of pilot density provides flexibility to the overall approach. When a higher power α is used, the method will be quite sensitive to the variations in the pilot density, whereas approach will be implemented as fixed width kernel approach when α is reduced to 0 [20]. For all the case studies discussed in this thesis, an adaptive kernel method was implemented with fixed Gaussian kernel as the pilot estimate and sensitivity parameter α equal to 1/2 [14,20].

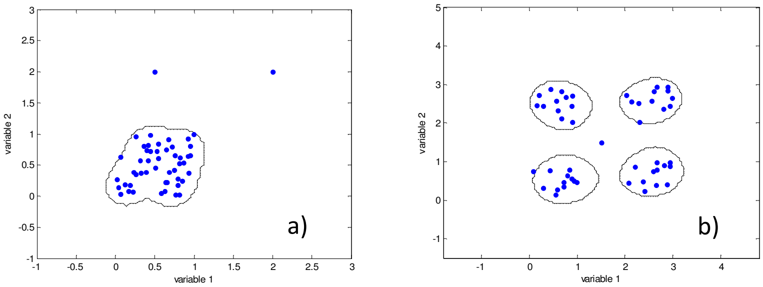

Figure 2.11 *Contour plots derived for simulated datasets implementing the Adaptive kernel. First simulated dataset (2.11a), Second simulated dataset (2.11b)*

Figure 2.11 provides with the contour plot derived for the simulated datasets implementing the Adaptive kernel. The resulting interpolation space resembled to those derived with different variants of the Gaussian kernels.

2.4.3 Triangular kernel

For an observation **xt** in multidimensional space, this kernel can be determined as follows:

$$K(\mathbf{xt},\mathbf{x}_i)=\begin{cases} 1-\left|(\mathbf{x}_i-\mathbf{xt})^{\mathrm{T}}(\mathbf{x}_i-\mathbf{xt})\right| & \text{if } \left|(\mathbf{x}_i-\mathbf{xt})^{\mathrm{T}}(\mathbf{x}_i-\mathbf{xt})\right|<1 \\ 0 & \text{otherwise} \end{cases} \qquad (2.15)$$

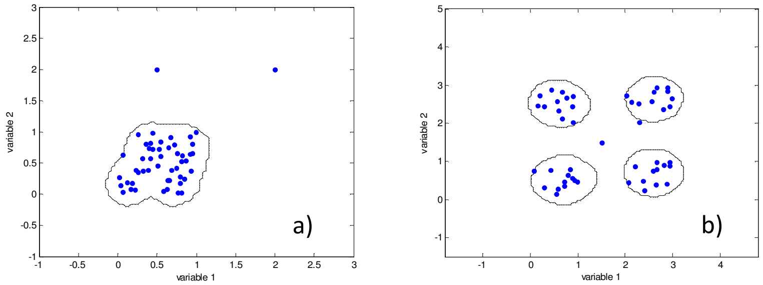

Figure 2.12 *Contour plots derived for simulated datasets implementing the Triangular kernel. First simulated dataset (2.12a), Second simulated dataset (2.12b)*

Figure 2.12 provides with the resulting contour plot for the simulate datasets implementing this kernel. Again the resulting AD for both the simulated datasets were quite similar to those defined using different variants of Gaussian kernels as well as Adaptive kernel.

2.4.4 Epanechnikov kernel

This is an optimal kernel to minimize the integrated mean errors. The multivariate Epanechnikov kernel is defined as [20]:

$$K(\mathbf{xt},\mathbf{x}_i)=\left\{\begin{array}{ll}\frac{1}{2}c_p^{-1}(p+2)\left(1-\frac{1}{h}\cdot(\mathbf{x}_i-\mathbf{xt})^{\mathrm{T}}(\mathbf{x}_i-\mathbf{xt})\right) & \text{if } \left|\frac{1}{h}\cdot(\mathbf{x}_i-\mathbf{xt})^{\mathrm{T}}(\mathbf{x}_i-\mathbf{xt})\right|<1\\ 0 & \text{otherwise}\end{array}\right\}$$

where c_p is the volume of the unit p-dimensional sphere. (2.16)

The bandwidth h has been calculated as [20]:

$$h=\left(n^{-1/(p+4)}\right)\cdot\left\{\frac{8p(p+2)(p+4)\left(2\sqrt{\pi}\right)^p}{(2p+1)c_p}\right\}^{1/(p+4)} \quad (2.17)$$

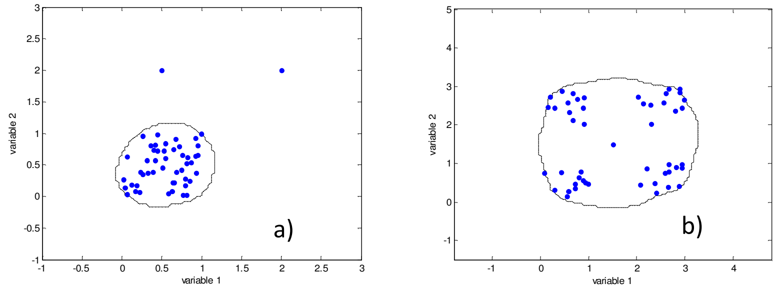

Figure 2.13 *Contour plots derived for simulated datasets implementing the Epanechnikov kernel. First simulated dataset (2.13a), Second simulated dataset (2.13b)*

The contour plot for the simulated dataset implementing this kernel is shown in Figure 2.13. For the first dataset, the defined AD remained localized around the cluster while for the second dataset, the AD enclosed the entire training space taking into account empty regions like with range and geometric based approaches.

Probability density distribution methods are advanced and but their efficiency is also associated with disadvantages of different kernels. For instance, kernel methods are usually associated with under- smoothing the tails while the nearest neighbourhood approach tries to overcome this issue however, by over-smoothing the tails. The adaptive kernel method overcomes such issues, however being adaptive to the local density.

All the classical AD methodologies discussed in this chapter will be further implemented on several QSAR models considered as case studies later in this thesis work. The results derived on these case studies will allow a further understanding of these discussed methodologies, as well as their advantages and disadvantages. It will be also interesting to see if the similarities in the approaches used to characterise the interpolation space is also evident from the common set of test molecules being excluded from the model’s AD.

Chapter 3

A novel k-Nearest Neighbours based Applicability Domain evaluation

Although existing literature discusses several approaches towards defining the Applicability Domain (AD) of QSAR models, an optimal approach has yet not been recognized. This chapter proposes a novel approach that defines the AD of QSAR models taking data distribution into account and derives a heuristic decision rule exploiting the k-Nearest Neighbours (kNN) principle. The proposed approach is a three stage procedure as a part of which, training thresholds are allocated, criterion deciding if a given test sample should be retained within the AD is defined and finally, the reliability in the derived results is reflected by taking model statistics and prediction error into account.

3.1 Background and motivation

As discussed in the previous chapter, several approaches were proposed in the past years to define the AD of QSAR models. All these approaches were associated with their own advantages and limitations [5, 11-14]. From time to time, several approaches were proposed that were aimed to be more efficient or were thought to overcome several limitations of the existing approaches.

Due to its simplicity and easy implementation, *k*-Nearest Neighbours had been a preferred choice for several proposed QSAR studies [4,9,19,22-26]. The kNN principle basically reflects upon the structural similarity of a test sample to the training samples used to build that model. In theory, the distance of a query sample is considered from its *k* closest data points in the chemical space. Lower distance values correspond to a higher similarity, while the increasing distances signify higher levels of structural mismatch. The *k* value plays a significant role in defining how constraint the approach will be and thus, it can be referred to as the smoothing parameter.

This chapter proposes a new heuristic approach towards defining the AD of QSAR models. The basis of this novel strategy is inspired from the kNN approach and adaptive kernel methods for probability density estimation (Kernel Density Estimators, KDE) [27]. In the classical kNN approach for AD evaluation [9,19], average distances of all the training samples from their *k* nearest neighbours are calculated and used to define a unique threshold to decide if a test sample is inside or outside the model's AD (for example, 95th percentile). Moreover, in the framework of the probability density function estimation, the nearest neighbour method provides density estimates depending on the Euclidean distance to the *k*-th nearest data point [20]. Following the same concept, the proposed method tries to integrate the kNN principle with the salient features of adaptive kernel methods [27], which define local bandwidth factors corresponding to the training data points and use them to build the density estimate at a given point.

The novelty of the kNN based AD approach proposed here lies in the overall strategy that is properly executed in a three-stage procedure to encapsulate and reflect upon several significant aspects towards model validation. Moreover, some features common to most of the AD approaches were dealt differently with this approach; for instance, rather than defining a general threshold as in all the distance-based approaches, each training sample in this approach was associated with its individual threshold; in order to find an optimal smoothing parameter *k*, this approach performed a *k*-optimization procedure based on Monte Carlo validation; additionally, model's statistical parameters and other relevant aspects were dealt simultaneously to reflect upon the reliability in the derived results.

3.2 Methodology

A stepwise execution of the following three stages characterises the workflow of this approach:

1) defining thresholds for training samples
2) evaluating AD for new/test samples
3) optimizing the smoothing parameter *k*

To allow a better interpretation of the proposed approach, results on both the two-dimensional simulated datasets (introduced in Figures 2.1 and 2.2 of Chapter 2) will be considered throughout the major part of this discussion and wherever applicable.

3.2.1 Defining thresholds for training samples

Thresholds have a great influence in characterising the AD for reliable predictions; a test sample that exceeds the threshold condition is associated with an unreliable prediction.

Like the adaptive kernel methods, instead of defining a general unique threshold as seen with several classical AD approaches, the proposed approach allocates a set of thresholds corresponding to the various training samples.

For a given value of *k*, threshold allocation process can be summarised as follows:

a) The distances of each training molecule from the remaining n – 1 molecules are calculated and ranked in increasing order, n being the total number of training molecules. This will result in a n x (n -1) neighbour table **D**; an entry D_{ij} of the table corresponds to the distance of the i-th molecule from its j-th nearest neighbour:

$$D_{i1} \leq D_{i2} \leq \ldots \leq D_{i,n-1}$$

b) The average distance of each i-th molecule from its k nearest neighbours is calculated considering the first k entries in i-th row of the neighbour table:

$$\bar{d}_i(k) = \frac{\sum_{j=1}^{k} D_{ij}}{k} \qquad where,\ 1 \leq k \leq n-1 \ \text{ and } \ \bar{d}_i(k) \leq \bar{d}_i(k+1) \tag{3.1}$$

A vector $\bar{\mathbf{d}}(k)$ of average distance values is then derived considering all the molecules in the training set.

c) Next, a reference value (from now on referred as *Ref Val*), $\tilde{d}(k)$ is determined as follows:

$$\tilde{d}(k) = Q3(\bar{\mathbf{d}}(k)) + 1.5 \cdot \left[Q3(\bar{\mathbf{d}}(k)) - Q1(\bar{\mathbf{d}}(k))\right] \tag{3.2}$$

where, $Q1(\bar{\mathbf{d}}(k))$ and $Q3(\bar{\mathbf{d}}(k))$ are the values corresponding to the 25th and 75th percentiles in the vector $\bar{\mathbf{d}}(k)$, respectively [28].

d) Next, the ordered distances of each i-th training sample from all other n - 1 training molecules are compared with the *Ref Val*. If the distance value of the i-th molecule from its given j-th training neighbour (where $1 \leq j \leq n-1$) is less than or equal to the *Ref Val*, then that distance value is retained, otherwise is discarded. The number K_i of neighbours satisfying this condition, minimum zero and maximum being n – 1, defines the density of the i-th sample neighbourhood:

$$K_i\colon\ \left\{D_{ij} \leq \tilde{d}(k)\right\} \quad \forall j\colon 1, n-1 \tag{3.3}$$

e) Finally, each i-th training molecule is associated with a threshold t_i which defines the width of its neighbourhood as:

$$t_i = \frac{\sum_{j=1}^{K_i} D_{ij}}{K_i} \tag{3.4}$$

If no distance value was retained for a given *i*-th training molecule ($K_i = 0$), then its threshold t_i would be theoretically settled to 0, but a pragmatic solution is to set it equal to the smallest threshold of the training set.

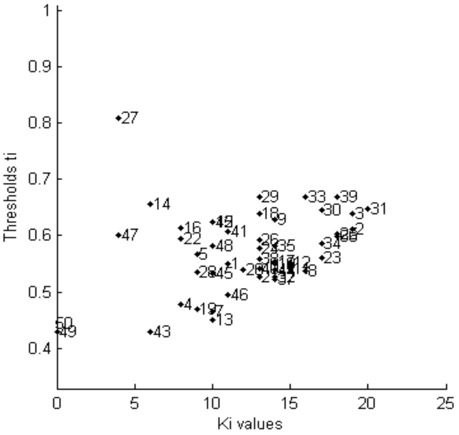

Figure 3.1 *First simulated data set. Thresholds t_i vs. number of training neighbours K_i plot (k = 12).*

The plot in Figure 3.1 provides with an overview of the thresholds for all the 50 samples in the simulated dataset. As expected, most of the training samples within the cluster (for instance, samples 2, 33 and 39) were associated with higher K_{i} values. On the other hand, obvious potential outliers (samples 49 and 50) had their thresholds equal to 0 since they couldn't satisfy the threshold criterion even for a single training neighbour (i.e. $K_i = 0$), thus no distance values contributed to their threshold calculation. Nevertheless, they were associated with the minimum threshold equal to 0.42, i.e. the threshold of sample 43.

3.2.2 Evaluating AD for new/test samples

Until this point, each training molecule was associated with its individual threshold. The next step will be to characterise the AD which usually relies upon a set of conditions that will decide if a given test molecule can be associated with a reliable prediction or not.

The criterion used by this approach to associate a given test sample to be within the domain of applicability can be summarised below.

Given a test molecule, its distance from all the *n* training molecules is calculated and simultaneously, compared to be less than or equal to the thresholds associated with each training molecule. If this condition holds true with at least one training molecule, the test molecule will be considered inside the domain of applicability for that model. Otherwise, the prediction for that test sample will be rendered unreliable.

More formally, given the training set *TR*, for each test molecule *j*, the AD decision rule is:

$$j \in AD \quad iff \quad \exists i \in TR: \quad D_{ij} \leq t_i \tag{3.5}$$

where D_{ij} is the distance between the *j*-th test molecule and the *i*-th training molecule and t_i is the individual threshold of the latter. In addition, each test/new molecule will be associated with the number K_j of nearest training neighbours for which the previous condition holds true. This number can be assumed as a measure of potential prediction reliability; indeed, high values of K_j indicate that the new molecule falls within a dense training region of the model's space, while low values of K_j denote that the new molecule still belongs to the model's space, but located in sparse training regions. K_j equal to zero rejects the molecule as it being outside the model's AD since no training neighbours are identified.

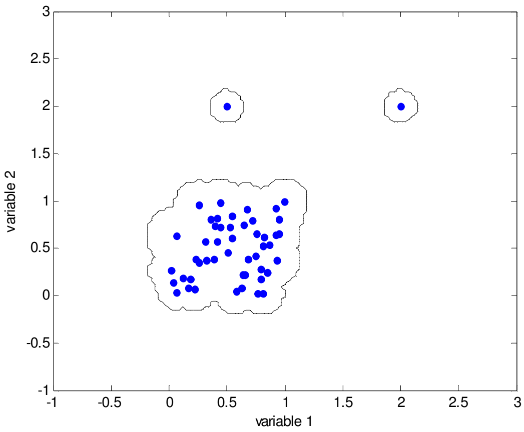

Figure 3.2 : *Contour plot to demonstrate how the AD was characterised for the first simulated dataset. Metric used: Euclidean distance; k = 12.*

Figure 3.2 provides with the contour plot for the simulated dataset derived projecting several data points enough to fill the training space. Thresholds were calculated using 12 nearest neighbours and Euclidean distance. This choice of k = 12 nearest neighbours was based on the results derived performing an internal k-optimization, discussed later in this article. The space enclosed around the cluster represented as black line indicates that all the data points within this enclosed region are inside the AD. Thus, this region reflects in a way how the AD was characterised for this two-dimensional dataset. Area of this enclosed region tends to expand or shrink depending upon the number of nearest neighbours used for threshold calculation.

As explained earlier, the extreme outliers in the training space will be associated with the number K_i of neighbours equal to zero and the lowest possible threshold in the training set. Consider the sample 49 from the simulated dataset which is an extreme outlier with its threshold equal to 0.42. If there is a test sample that seems to be quite in the vicinity of this potential outlier within the descriptor space, the test sample will be associated with an unreliable prediction since its distance from sample 49 will likely exceed the small threshold. Now, consider a case, where the descriptor values for another test sample exactly overlap or are very similar to those for this potential outlier. In this situation, the distance of that sample from the outlier will be less than the threshold and thus it will be considered within the domain of applicability. In theory, this is not wrong because the potential outlier is still a part of the training space. Practically, the approach retains all the training samples to characterize the AD but minimizing the role of potential outliers in doing so. That’s the reason why the first test sample was excluded from being reliably predicted while the second sample was not.

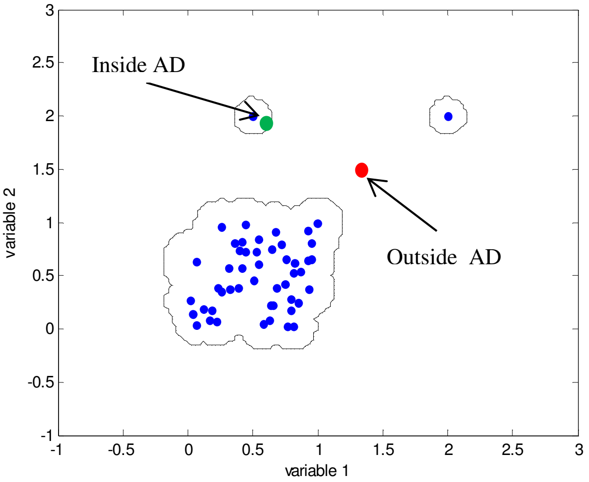

Figure 3.3 *An illustration of two test samples towards AD criterion of the proposed approach for the simulated dataset.*

However, for the latter the number K_j of nearest training neighbours will likely be equal to one indicating that its prediction has some degree of uncertainty. In conclusion, there exists a relation between the defined AD and the impact of training samples in characterising it based on their threshold values.

3.2.3 Optimizing the smoothing parameter k

Another important aspect is concerning the choice of an appropriate smoothing parameter *k*, whose theoretical range is between 1 and *n*-1. It can be seen from the AD defined for the simulated dataset using different *k* values in Figure 3.4, very low *k* values will restrict the domain of applicability in a very strict manner as compared to the AD derived opting for larger *k* values. This is because, an opted *k* value will have a direct impact on the threshold calculations which in turn can make it more rigid or easier for test samples to satisfy the threshold criterion. The strategy implemented in this thesis to select an appropriate *k* value was performed by Monte Carlo validation in '*n*' iterations, maximizing the percentage of the test samples considered within the AD, i.e. satisfying AD criterion (Equation 3.5).

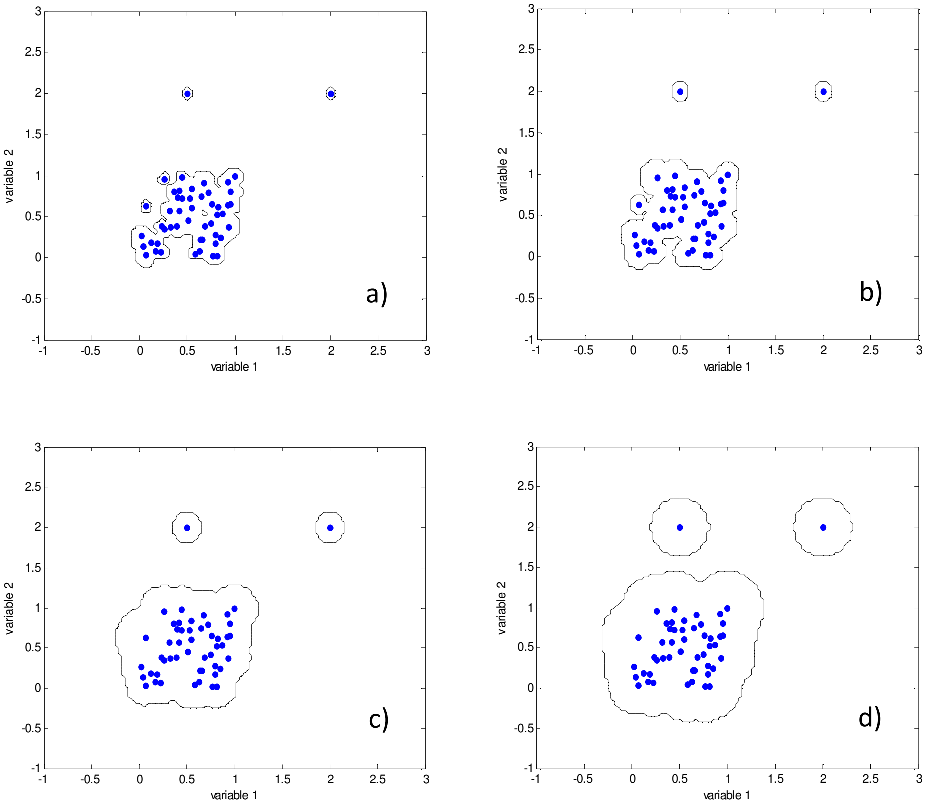

Figure 3.4 *Impact of different k values on the defined AD for simulated dataset. a) k =1, b) k =5, c) k =15 and d) k =25.*

To perform this validation, in each iteration, 20 percent of the training samples were randomly chosen as the test set and the above discussed AD procedure was executed using a range of *k* values, defined by the user. Percentage of test samples retained inside the model's AD for each *k* value in every iteration was recorded. Box-and-whisker plots (box plots) were produced to get an overview of all these derived results. For instance, consider the plot in Figure 3.3 derived for the simulated dataset showing percentage of test samples retained within the AD with different *k* values (optimization carried out with 20% of samples in the test set and 1000 iterations).

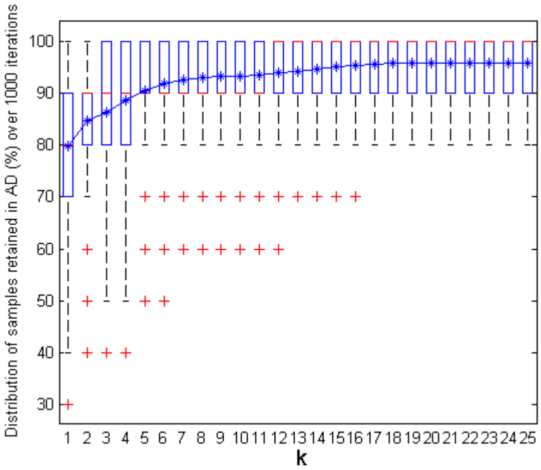

Figure 3.5 *First simulated data set. Box-and-whisker plot of test samples (%) retained within the AD for different k values during k-optimization.*

Figure 3.5 shows that the spread of the box plots for initial *k* values is quite large. This may have resulted due to the impact of restricted training thresholds that excluded several test samples from the AD. With an increase in *k* values, the spread narrowed, however the outliers were still present until *k* = 17. After this point, the box plots remained unchanged throughout the plot with no outliers. Similar observations were derived from the mean line plot which showed a significant rise initially followed by a stable curve until the first half of the *k* values. The plot didn't show any major changes for the second half of the *k* values. In order to avoid very high *k* values good enough to unnecessarily expand the defined AD, a *k* value of 12 was opted as appropriate *k* for this dataset. The plots dealt earlier (Figures 3.1 and 3.2) for this dataset were thus derived using this opted *k* value.

Median quartile in the middle of the box (marked in red) can be referred for all the *k* values to get a hint about how many test samples were retained on average during the optimization process for a given *k* value. About their usefulness in the proposed AD approach, box plots showing limited spread and allowing majority of test samples to be retained within the AD can be favoured and their corresponding range of *k* values can be considered to finally opt for the most appropriate *k*. Additionally, a line plot is integrated in the same figure indicating the mean percentage of test samples that were considered within the AD for each k value. A simultaneous interpretation of both these plots can make it easier for a user to decide upon an appropriate k value.

It was concluded that optimization of k can be a time-demanding procedure especially in the case of a huge number of samples, but it was also observed that this approach is quite insensitive to the smoothing parameter k, except for very small k values which led to the results influenced by local noise. Therefore, for many applications the optimization of the smoothing parameter can be avoided and reasonable results can instead be obtained by a fixed k value empirically calculated as $n^{1/3}$.

3.2.4 An overview of results on other simulated datasets

The simulated dataset discussed so far was used to facilitate a better understanding of how the proposed approach works. This part of the chapter provides an overview of how using the same approach the resulting AD was defined on other simulated datasets.

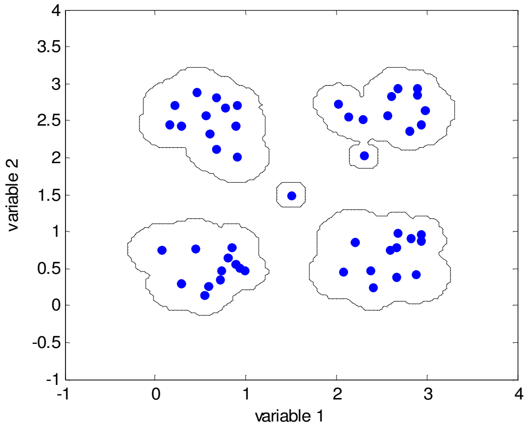

Figure 3.6 *Second simulated data set. Contour plot to demonstrate how the AD was characterised. Metric used: Euclidean distance; k = 4.*

The contour plot for the AD defined on second simulated dataset (introduced in Figure 2.2 of Chapter 2) with the new approach is shown in Figure 3.6 which was derived using k = 4. For range and geometric based approaches, the isolated sample (49) was considered inside but taking into account unnecessary descriptor space between the clusters, while for the distance and probability density distribution approaches, this sample was considered outside the AD approach due to the percentile-based threshold. With the proposed approach, all the clusters were enclosed in their own interpolation space. Since sample 49 was associated with the minimum training threshold, a small descriptor space around it was considered within the AD indicating that a test sample extremely similar to sample 49 could be considered as reliably predicted.

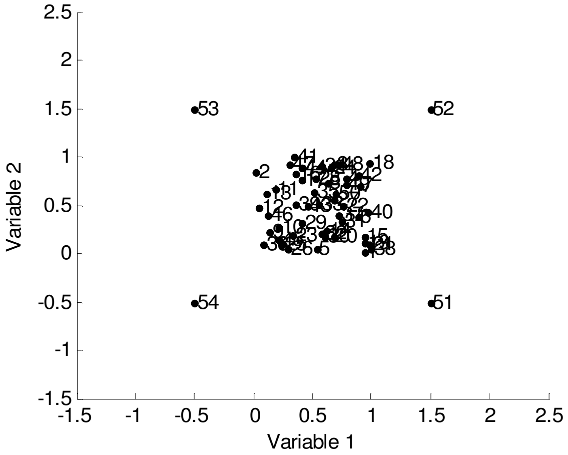

Figure 3.7 *Scatter plot for the third simulated dataset.*

Figure 3.7 provides with the scatter plot for an additional simulated dataset considered to better evaluate the proposed AD approach. As shown in the figure, this dataset has a cluster of data points in the middle and four isolated samples surrounding it. It could be easily interpreted that with several classical approaches like convex hull or bounding box, a lot of unnecessary interpolation space could be taken into account considering the four isolated samples within the model's AD.

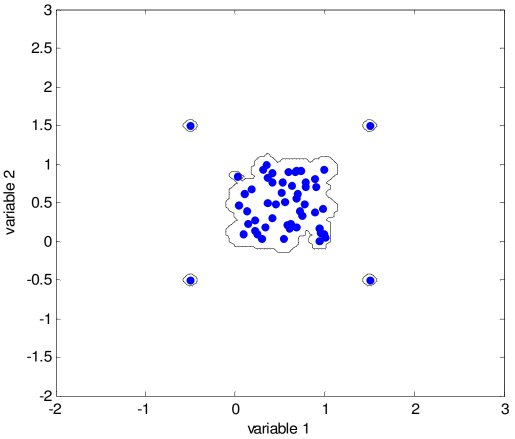

Figure 3.8 *Third simulated data set. Contour plot to demonstrate how the AD was characterised. Metric used: Euclidean distance; k = 4.*

Figure 3.8 provides with the contour plot for this simulated dataset. Since the potential outliers with this approach are associated with minimum training threshold, a small descriptor space surrounding these isolated samples was considered inside the model's AD. As expected, all the clustered data points were included within the common AD space. The above contour plot was derived using $k = 4$.

The overall strategy for this novel approach in defining the AD will be clearer when the performance of this approach will be further evaluated later in this thesis using several QSARs from the existing literature as the case studies. The results derived with this approach will be also compared with those derived using several other existing AD approaches discussed earlier in Chapter 2.

Chapter 4

Outlier detection from an Applicability Domain perspective

Presence of potential outliers in the training space can have a huge impact on characterizing the interpolation space and the resulting Applicability Domain (AD) may not be restrictive enough to exclude unreliable test molecules. On the other hand, the test molecules detected as outliers when projected on the training space can hint for their prediction being unreliable and thus can be excluded from the model's AD. This chapter introduces a novel Mahalanobis distance measure (namely, a pseudo-distance) termed as Locally-centred Mahalanobis distance and discusses its usefulness towards outlier detection. The proposed outlier detection approach hints some useful alerts towards the presence of test molecules that could be rendered as unreliable after AD evaluation. This chapter implements this newly derived distance matrix to propose the second novel approach towards evaluating for a model's AD.

4.1 Introduction and the scope of this study

Outliers represent the observations that fail to follow the general pattern of the majority of data samples [29]. Thus, it is critical to detect and appropriately treat such anomalous observations, contributing to undesired performance degradation, or, alternatively, suggesting unexpected but interesting patterns. In recent years, there had been a growing attention towards dealing with outliers since they can highly impact the variance and correlation between variables and as a result, several approaches addressing outlier detection have been proposed in the literature [30].

Several supervised and unsupervised-learning methods have been proposed to address outlier mining [31]. Most of the proposed techniques to deal with outliers were either diagnostic or robust approaches [32,33]. Several classical techniques performed well, provided the given set of data contained only a single outlier, however, their inefficiency emerged while dealing with multiple outliers [34]. Increasing dimensionality of data adds to the complexity of detecting such outliers. Lacking visual perception for data with more than two dimensions, restricted the reliable use of such classical approaches only for two-dimensional data [29]. Moreover, masking and swamping considerably restricted the usefulness of such classical approaches towards detection of multiple outliers in calibration. Many times the presence of some outliers can somehow mask the detection of other outliers. As a result, some outliers are wrongly identified as normal samples. This phenomenon is referred to as masking. On the contrary, swamping refers to the cases where the presence of a subset of observations makes normal samples being incorrectly identified as potential outliers [32, 33].

Several new and improved detection approaches emerged from time to time and were attempting to overcome major limitations of classical outlier detection techniques, however, this domain of data exploration perhaps may always leave a room for further improvement towards developing an approach that can tackle the increasing data complexity without comprising upon the quality of detection accuracy.

The outliers detected amongst molecules constituting the training space can be quite interesting from an AD perspective. The training molecules detected as outliers can have a huge impact on the interpolation space defined by different AD approaches. This impact of training outliers further depends upon the AD approach being implemented. For instance, range-based approaches are highly sensitive to such outliers and thus their defined interpolation space may be unnecessarily broadened accounting for several empty regions in the descriptor space. On the contrary, the Probability density distribution-based approaches as well as the novel kNN based AD approach (discussed earlier in Chapter 3) will try to minimize the impact of such potential outliers in defining the interpolation space. Later, test molecules considered as extreme outliers when projected on the training space could be more likely to be unreliably predicted upon AD evaluation. This implies that the outlier detection approaches can be quite useful in determining the test molecules that are extreme outliers when projected on the training space. This resulting subset of test molecules can be excluded from the model's AD, rendering them unreliably predicted in the model's descriptor space.

In this chapter, a new distance measure, called *locally-centred Mahalanobis distance*, based on the covariance matrix centred on each dataset molecule, is introduced and its salient properties are discussed. Two new parameters, remoteness and isolation degree derived from the resulting pairwise distance matrix are introduced, in order to better explore the isolation of the molecules in their local and global space. The information corresponding to these new parameters when plotted can allow the analyst to better explore several interesting features of the data, particularly, in terms of detecting those molecules that are quite diverse from the major pattern followed by the data [35]. Later, the novel distance measure can be calculated for the test molecules with respect to the training set molecules. The resulting remoteness and isolation degree values for test samples can be projected along with those for the training set molecules. Provided that the thresholds for training remoteness and isolation degree are defined, test molecules associated with values for these parameters exceeding their thresholds can be excluded from the model's AD. The performance of this new outlier detection approach towards AD evaluation is better explained taking into account the results derived on two-dimensional simulated datasets introduced earlier (Figures 2.1 and 2.2) in Chapter 2. Later, the performance of this novel outlier detection approach will be further evaluated considering several case studies later in this thesis.

4.2 Definition of the Locally-Centred Mahalanobis distance

Let the data matrix **X** be comprised of *n* molecules and *p* descriptors, defined as: $\mathbf{X} = \left(\mathbf{x}_1^{\mathrm{T}}, \mathbf{x}_2^{\mathrm{T}}, \ldots, \mathbf{x}_n^{\mathrm{T}}\right)^{\mathrm{T}}$, where $\mathbf{x}_i$ are column vectors representing the *n* observations (*i* = 1, 2,…, *n*).

The data are assumed to be independently sampled from a multivariate normal distribution $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. A general measure of squared distance from an observation $\mathbf{x}_i$ to the centroid of the *p*-dimensional space **μ**, for *i* = 1,.., *n*, can thus be written as follows:

$$d_i^2 = \left(\mathbf{x}_i - \boldsymbol{\mu}\right)^{\mathrm{T}} \cdot \mathbf{M} \cdot \left(\mathbf{x}_i - \boldsymbol{\mu}\right) \tag{4.1}$$

where **M** is a *p* x *p* symmetrical matrix. If $\mathbf{M} = \boldsymbol{\Sigma}^{-1}$ where **Σ** is the population covariance matrix, the squared Mahalanobis distance is obtained as:

$$d_i^2 = \left(\mathbf{x}_i - \boldsymbol{\mu}\right)^{\mathrm{T}} \cdot \boldsymbol{\Sigma}^{-1} \cdot \left(\mathbf{x}_i - \boldsymbol{\mu}\right) \tag{4.2}$$

These distances are distributed according to χ_{p}^2 and if the parameters **μ** and **Σ** are estimated by the arithmetic mean $\bar{\mathbf{x}}$ and the molecule's covariance matrix $\mathbf{S} = \frac{1}{n-1} \cdot \sum_{i=1}^{n} \left(\mathbf{x}_i - \bar{\mathbf{x}}\right)\left(\mathbf{x}_i - \bar{\mathbf{x}}\right)^{\mathrm{T}}$ respectively, the (estimated) squared Mahalanobis distances are:

$$MD_i^2 = \left(\mathbf{x}_i - \bar{\mathbf{x}}\right)^{\mathrm{T}} \cdot \mathbf{S}^{-1} \cdot \left(\mathbf{x}_i - \bar{\mathbf{x}}\right) \tag{4.3}$$

The distribution is given by $\frac{(n-1)^2}{n} MD_i^2 \,\square\, Beta(\frac{p}{2}, \frac{n-p-1}{2})$, (e.g., see reference [7]). If **S** and $\mathbf{x}_i$ are independent, then $\frac{n-p}{(n-1)p} MD_i^2 \,\square\, F_{p,n-p}$.

Now, if a vector $\mathbf{v} \in \boldsymbol{R}^p$ is selected in the *p*-dimensional space, the covariance matrix, centred at **v**, denoted by $\mathbf{S}_{(\mathbf{v})}$,can be calculated as:

$$\mathbf{S}_{(\mathbf{v})} = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{v})(\mathbf{x}_i - \mathbf{v})^{\mathrm{T}} \tag{4.4}$$

Then, it can be easily verified that,

$$\mathbf{S}_{(\mathbf{v})} = \mathbf{S} + \frac{n}{n-1} \cdot (\overline{\mathbf{x}} - \mathbf{v})(\overline{\mathbf{x}} - \mathbf{v})^{\mathrm{T}} \tag{4.5}$$

Finally, the squared Mahalanobis distances considering **v** as the space centre can be derived as:

$$MD^2(i, \mathbf{v}) = (\mathbf{x}_i - \mathbf{v})^{\mathrm{T}} \cdot \mathbf{S}_{(\mathbf{v})}^{-1} \cdot (\mathbf{x}_i - \mathbf{v}) \qquad i = 1, \ldots, n \tag{4.6}$$

If the above mentioned vector **v** is now replaced by an observation $\mathbf{x}_j$, for j = 1,..., n, the new locally-centred squared Mahalanobis distance between observations i and j is defined as:

$$MD_L^2(i, j) = \left(\mathbf{x}_i - \mathbf{x}_j\right)^{\mathrm{T}} \cdot \mathbf{S}_{(j)}^{-1} \cdot \left(\mathbf{x}_i - \mathbf{x}_j\right) \tag{4.7}$$

where $\mathbf{S}_{(\mathrm{j})}$ is the covariance matrix centred on the j-th observation.

It should be noted that the classical covariance matrix **S**, being centred on the arithmetic mean vector, minimizes the data variance, while, the new defined locally-centred covariance matrix encodes different information, data variance depending on the selected centre. Thus, the new distance measure is more informative than the classical Mahalanobis distance, which considers only the arithmetic mean as the data centre.

In order to obtain distances that are independent of the number of descriptors p, the distance values can be divided by p, thus obtaining locally-centred average squared Mahalanobis distances:

$$\overline{MD}_L^2(i,j) = \frac{MD_L^2(i,j)}{p} = \frac{1}{p} \cdot \left[\left(\mathbf{x}_i - \mathbf{x}_j\right) \cdot \mathbf{S}_{(j)}^{-1} \cdot \left(\mathbf{x}_i - \mathbf{x}_j\right)\right] \qquad i, j = 1, \ldots, n \tag{4.8}$$

Hereinafter these average distances will be considered, for the sake of simplicity, they will be often shortly referred to as locally-centred squared Mahalanobis distances, still using the symbol MD_L^2.

4.2.1 Salient features of the novel distance measure

There are two important key aspects related to this novel distance. Like the distances derived using the classical covariance matrix, the locally-centred

Mahalanobis distances are invariant to any sort of variable scaling. Secondly, unlike the classical Mahalanobis distance, the resulting object-centred distance is asymmetric and consequently is a pseudo-distance; indeed, the distance between two observations *i* and *j* depends on whether the selected centre is *i* or *j*:

$$MD_L^2(i,j) \neq MD_L^2(j,i) \tag{4.9}$$

This asymmetry is accounted due to the presence of all other observations and their resulting overall influence in deriving the distances, thus reflecting the significance of information retrieved from the locally-centred covariance matrix.

The asymmetry between $MD_L^2(i,j)$ and $MD_L^2(j,i)$ seems to have a significant meaning. In fact, a higher value of $MD_L^2(i,j)$ in contrast with a corresponding lower value for $MD_L^2(j,i)$ indicates that the molecule *i* belongs to a relatively denser region with respect to the molecule *j*, which appears to be more isolated. This consideration can be further supported by the fact that, when *j* is isolated being the centred object, it shows a higher variance than the case when *i* is the centred molecule, which unlike the earlier, is surrounded by several molecules in its vicinity. As seen from the way these locally-centred Mahalanobis distances are derived, the variance is calculated as the reciprocal in the distance formula and as a result, *j* tends to seem closer to *i*, while on the contrary, molecule *i* with a lower variance tends to seem comparatively further distant from *j*. Usually, the molecules with lower variance can be thought of being either located in a cluster or surrounded by several similar molecules in their vicinity.

The variable space based on Mahalanobis distances calculated using the classical covariance matrix is estimated by an ellipsoid (or hyper-ellipsoid), while in the case of locally-centred Mahanalobis distances, the variable space is defined by a family of ellipsoids (or hyper-ellipsoids) due to the multi-centred approach. Thus, a more data-driven shaped descriptor space is determined using this novel distance measure.

4.3 Remoteness and Isolation degree plot

It is quite easy to interpret the significance of columns and rows in the pair-wise distance matrix $\mathbf{MD}_L^2$ resulting from the novel average locally-centred squared Mahalanobis distances. In fact, each *j*-th column constitutes the data centre and represents how that *j*-th molecule "globally perceives" each *i*-th molecule, also taking into account the overall influence of all the other molecules, while each *i*-th row represents how that *i*-th molecule is "globally perceived" by all the other molecules.

Each *j*-th column of the $\mathbf{MD}_L^2$ matrix contains information about the distances of all other *i* molecules from the *j*-th molecule being the centre. The minimum value of a *j*-th column can be taken into account to represent the squared distance of the *j*-th molecule from its nearest neighbour; this is termed as *Isolation degree* (*Idg*):

$$Idg_j = \min_i\left(\left[\mathbf{MD}_L^2\right]_{ij}\right) \quad i \neq j \tag{4.10}$$

Similarly, each *i*-th row of the $\mathbf{MD}_L^2$ matrix contains information about the squared distances of the *i*-th molecule as it is perceived from all the other molecules. Thus, the average squared distance value for each *i*-th row is taken into account and termed as *Remoteness (Rem)*:

$$Rem_i = \frac{\sum_{j=1}^{n}\left[\mathbf{MD}_L^2\right]_{ij}}{n-1} \tag{4.11}$$

The values of remoteness can range from a minimum greater than zero and a maximum equal to (*n*-1)/*p*, while isolation degree for any given molecule remains localized between 0 and 1. It should be also noted that:

$$\frac{\sum_{i=1}^{n} Rem_i}{n} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\left[\mathbf{MD}_L^2\right]_{ij}}{n\cdot(n-1)} = 1 \tag{4.12}$$

i.e., the average value of the remoteness vector or, in other words, the average value of the matrix $\mathbf{MD}_L^2$ elements is equal to one. Then, the remoteness could be interpreted as the influence that each molecule exerts over the covariance structure of the data, i.e. the values significantly larger than one identify the most influent molecules.

4.3.1. Usefulness towards outlier detection

The remoteness highlights objects which are far from the bulk of the remaining objects, i.e. they can be considered as classical outliers in the selected variable space; the Isolation degree detects a different kind of "anomalous" objects, i.e. those objects that, although located within the variable space, are isolated from the other ones or, in other words, these objects are surrounded by objects not so near. Therefore, a scatter plot of Remoteness vs. Isolation degree, called *RI plot*, for the data set in analysis can be a useful tool for exploratory purposes.

The thresholds to detect remote and isolated samples, for the two distributions of remoteness and isolation degree, are defined as the upper "fences" in the box & whisker plots [28]:

$$threshold = Q_3 + 1.5 \cdot (Q_3 - Q_1) \tag{4.13}$$

where Q_1 and Q_3 are the first and third quartiles for remoteness and isolation degree values, respectively, and their difference is the interquartile range.

To better evaluate the role of remoteness and isolation degree towards potential outlier detection, the results for both the simulated data sets introduced in Chapter 2 were analysed. As mentioned earlier, the first dataset consists of a cluster of 48 data samples and two additional samples (49 and 50) quite distant from each other as well as from the main sample cluster (Figure 2.1) while the second dataset had its data samples roughly divided within four clusters and a single data sample (49) localized more or less between these clusters (Figure 2.2).

The locally-centred squared Mahalanobis distances were calculated for the two simulated and the object-oriented pair-wise distance matrix $\mathbf{MD}_L^2$ was derived. The average distance values from each row and the minimum distance values from each column were retrieved from this distance matrix to derive the remoteness and isolation degree vectors, respectively. The values of these two parameters were used as the point coordinates of all the data samples in the RI plot.

Thresholds for both remoteness and isolation degree were calculated according to equation 4.13 and reported in the RI plots by red lines. The data samples associated with very high values for remoteness were classified as outliers of first type being far from the variable space defined by the bulk of the data, i.e. remote samples; the data samples associated with high values of isolation degree were classified as outliers of second type, they being isolated from the other samples in spite of their position within the variable space, i.e. isolated samples.

The RI plot obtained by the locally-centred Mahalanobis distance for first simulated dataset is shown in Figure 4.1. As expected, two data samples 49 and 50 were highly isolated from the cluster and far from the bulk of the data. Both these data samples were associated with high values for remoteness and isolation degree which clearly indicated that they are quite isolated in their local and global spaces. Moreover, data sample 27 was associated with a higher value of isolation as compared to the other samples in that cluster.

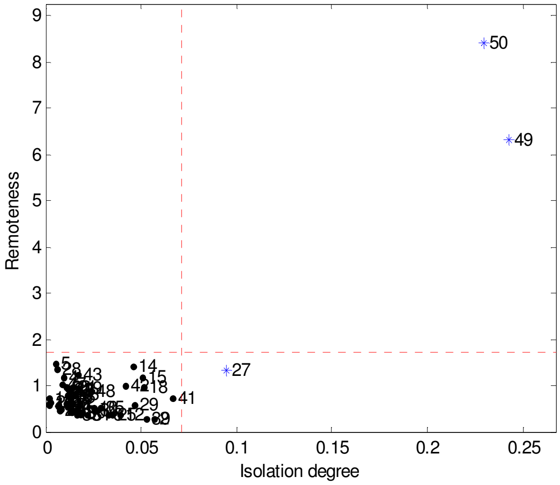

Figure 4.1 *RI plot for the first simulated data set*

A careful observation of the scatter plot in Figure 2.1 indicates that sample 27 is within the extremities of the cluster as well as no other data samples from the cluster are very closely located in its vicinity. This indicates that the new approach is quite sensitive to the isolation of the samples

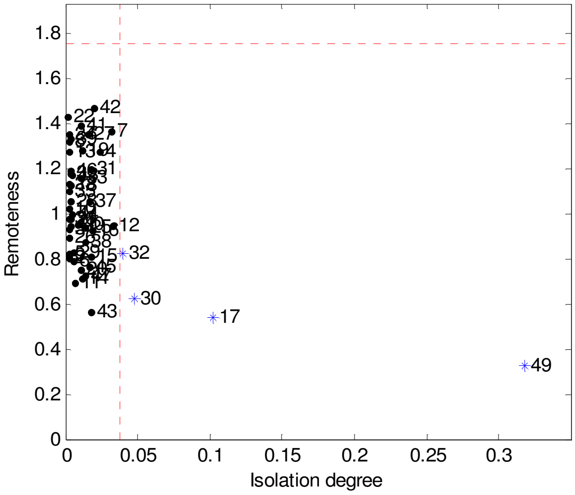

Figure 4.2 *RI plot for the second simulated data set.*

The second data set used as case study was a two–dimensional simulated data set introduced in chapter 2 with data samples roughly divided within four clusters and a single data sample (49) localized more or less between these clusters. The scatter plot of this data set (Figure 2.2) indicates this isolated sample clearly being a potential outlier; however, it was also interesting to see how the outlier detection techniques were able to analyse this data.

As shown in Figure 4.2, the novel outlier detection approach was able to clearly identify sample 49 as a second type outlier based on its extreme value for isolation degree. Remoteness for the data samples was not extremely high for any specific data sample and then no first type outliers are detected. Samples 17, 30 and 32 that were not very closely located to their nearest of the four clusters were also identified with higher values of isolation degree.

4.4 Implementing the novel approach towards AD evaluation

So far, the usefulness of remoteness and isolation degree was explored towards outlier detection. As pointed earlier, if these new matrix parameters can be calculated also for test samples and simultaneously projected with those for the training samples, the resulting plot could be quite useful to evaluate if the test samples can be reliably predicted or not. The test samples exceeding the defined training thresholds for remoteness or isolation degree or both of them can be excluded from the model's AD.

To implement this strategy for AD evaluation, remoteness and isolation degree of the test samples were determined as follows:

For a given test set **Xt** with *m* observations, the remoteness of the test sample **xt*ᵢ*** was derived by calculating its locally-centred Mahalanobis distance from each training observation $\mathbf{x}_j$ centering at $\mathbf{x}_j$, such that $j = 1,\ldots, n$. and then finding the mean of the resulting distance vector:

$$\overline{MD}_L^2(i,j) = \frac{MD_L^2(i,j)}{p} = \frac{1}{p}\cdot\left[\left(\mathbf{xt}_i - \mathbf{x}_j\right)^{\mathrm{T}} \cdot \mathbf{S}_{(j)}^{-1} \cdot \left(\mathbf{xt}_i - \mathbf{x}_j\right)\right] \qquad j = 1,\ldots,n \tag{4.14}$$

where $\mathbf{S}_{(j)} = \frac{1}{n-1}\cdot\sum_{i=1}^{n}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}}$

On the other hand, the isolation degree of the test sample $\mathbf{xt}_i$ was derived by calculating its locally-centred Mahalanobis distance from each training observation $\mathbf{x}_j$ centering at the *i*-th test object $\mathbf{xt}_i$ and then finding the minimum value from the resulting distance vector:

$$\overline{MD}_L^2(j,i) = \frac{MD_L^2(j,i)}{p} = \frac{1}{p}\cdot\left[\left(\mathbf{x}_j - \mathbf{xt}_i\right)^{\mathrm{T}} \cdot \mathbf{S}_{(i)}^{-1} \cdot \left(\mathbf{x}_j - \mathbf{xt}_i\right)\right] \qquad j = 1,\ldots,n \tag{4.15}$$

where, $\mathbf{S}_{(i)} = \frac{1}{n}\cdot\sum_{j=1}^{n}(\mathbf{x}_j - \mathbf{xt}_i)(\mathbf{x}_j - \mathbf{xt}_i)^{\mathrm{T}}$

The test molecules exceeding the remoteness and isolation degree thresholds in equation 4.13 can be excluded from the model's AD.

Considering these definitions for remoteness and isolation degree for training and test molecules, contours plot was derived by projecting several test samples enough to fill the training space of both the simulated datasets.

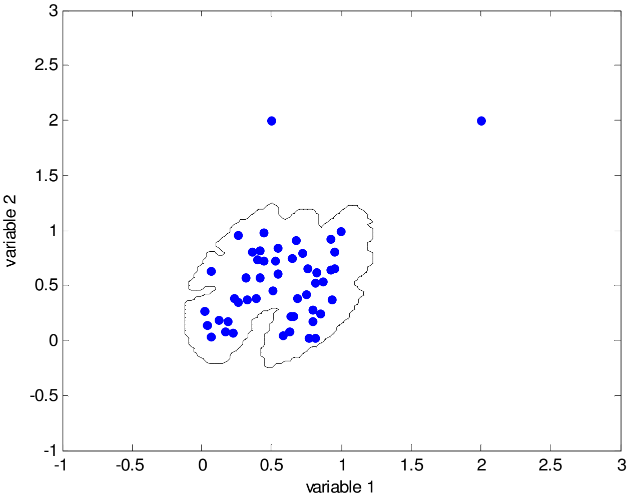

Figure 4.3 *AD contour plot for the first simulated dataset*

Figure 4.3 provides with the contour plot for the first simulated dataset. The defined interpolation space consisted of all the test samples that had their remoteness and isolation degree values below the defined thresholds. The AD seems quite adapted to the shape of the data cluster which was not so clearly interpretable with other classical approaches. Moreover, the choice of slightly higher thresholds for test samples is quite visible in the plot.

Figure 4.4 provides with the contour plot for the second simulated dataset. The defined interpolation was mainly concentrated around the four clusters. Due to the choice of thresholds, the defined AD seemed slightly extended around the cluster's extremities. Like in the case of first novel AD approach, this approach also considered the descriptor space around the isolated sample (49) within the defined AD.

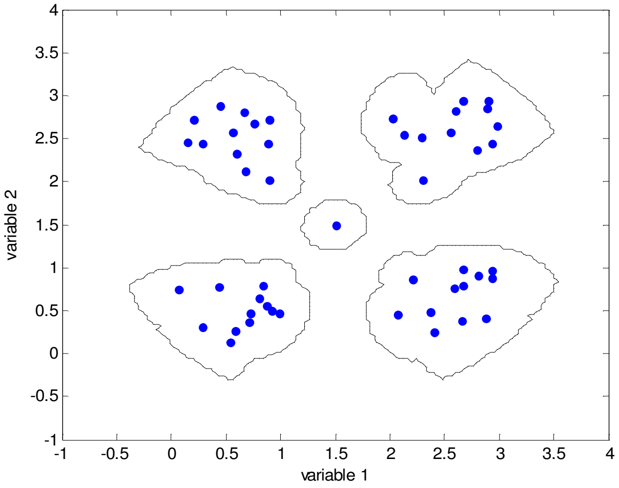

Figure 4.4 *AD contour plot for the second simulated dataset*

Both the simulated datasets were simpler in dimensions and were mainly chosen to provide with a better understanding of the proposed approach towards AD evaluation. The potential of this novel approach will be further clearer while deriving the AD for several multidimensional case studies later in this thesis.

PART III

APPLICATIONS

Chapter 5

Case studies

This section is entirely dedicated towards implementing the already discussed classical and novel AD approaches on several QSAR models from the existing literature. Each case study is initially introduced highlighting the five major OECD principles for model validation, followed by discussing the results derived evaluating for their AD using various approaches. The test samples considered as consensus outliers with different AD approaches hint the possible similarities in the underlying AD algorithms as well as higher chances of rendering those test samples being unreliably predicted. Finally, the impact on model's statistics was evaluated after excluding the test samples that were rendered outside the model's AD using all the listed approaches.

5.1 An overview of the case studies

The earlier chapters discussed several classical approaches from the existing literature towards defining the AD of QSAR models. Moreover, two new AD approaches were also introduced and illustrated using simulated datasets. In this section, several QSAR models from the existing literature will be used as case studies to evaluate their AD implementing different classical and novel approaches discussed earlier. All these models will be introduced based on the five OECD principles for model validation. This will help to better understand the validity of these models. Later, the results derived implementing all the earlier discussed AD approaches on these models will be provided. An overview of all the test samples excluded from the model's

AD with different approaches will be provided including their names, CAS numbers and their error in prediction will be provided.

Most of the case studies discussed in this chapter are of regulatory relevance. CAESAR Bioconcentration factor models [36-38] and QSAR models for ready biodegradability of chemicals [39], were clearly developed to contribute to the REACH implementation.

5.2 Assessing reliability in derived results

For all the regression models considered, before the AD evaluation was performed, an overview of the model's statistics (retaining the test set in its entirety) was provided using the following key parameters:

a) Determination coefficient R^2

$$R^2 = 1 - \frac{\sum_{i=1}^{n_{TR}} (\hat{y}_i - y_i)^2}{\sum_{i=1}^{n_{TR}} (y_i - \bar{y}_{TR})^2} \tag{5.1}$$

b) Root-Mean-Square Error *RMSE*

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n_{TR}} (\hat{y}_i - y_i)^2}{n_{TR}}} \tag{5.2}$$

c) Predictive squared correlation coefficient Q^2 [40,41]

$$Q^2 = 1 - \frac{\left[\sum_{j=1}^{n_{TS}} (\hat{y}_j - y_j)^2\right] / n_{TS}}{\left[\sum_{i=1}^{n_{TR}} (y_i - \bar{y}_{TR})^2\right] / n_{TR}} \tag{5.3}$$

d) Root-Mean-Square Error in Prediction *RMSEP*

$$RMSEP = \sqrt{\frac{\sum_{j=1}^{n_{TS}} (\hat{y}_j - y_j)^2}{n_{TS}}} \qquad (5.4)$$

where, y_i is the measured response value for the i-th training sample and $\hat{y}_i$ its predicted value; y_j is the measured response value for the j-th test sample and $\hat{y}_j$ its predicted value; n_{TR} and n_{TS} represent the total number of training and test samples, respectively, and $\overline{y}_{TR}$ is the mean response of the training set.

Later, when different AD approaches were implemented on these models, in order to reflect upon the model's predictive ability, following key parameters were evaluated:

a) Number of test samples excluded from the model's AD.

b) Q^2 calculated from the test samples retained within the AD

c) List of all the test samples (their sample IDs) considered outside the AD.

Additionally, for the novel *k*NN based AD approach discussed in Chapter 3:

a) For each *j*-th test sample, its absolute standardized error calculated as:

$$SE_j = \frac{\left| y_j - \hat{y}_j \right|}{s_Y} \qquad (5.5)$$

where, y_j is the measured value for the *j*-th test sample and $\hat{y}_j$ its predicted value; s_Y the standard error of estimate derived from the training set.

b) The information about how many times the threshold criterion (Equation 3.5) is satisfied by each test sample, that is, how many training neighbours (i.e. K_j) are located at a distance less than or equal to their threshold values, from a given test sample.

In theory, a test sample satisfying the threshold criterion several times (i.e. having high K_j) is expected to be predicted with higher accuracy. This can be desired since less distant training neighbours indicate a higher structural similarity of the test sample. On the contrary, a test sample satisfying the threshold criterion for no training neighbours ($K_j = 0$) indicates that there wasn't any training sample similar enough to reliably predict that test sample. K_j (number of training neighbours) vs. absolute standardised error plot for all the test samples derived was derived.

For all the classification models on ready biodegradability of chemicals, following key parameters were evaluated to determine their predictive ability [38]:

a) Specificity (Sp)

$$\mathrm{Sp} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{5.6}$$

where, TN (True Negatives) is the number of not ready biodegradable samples that were classified as not ready biodegradable. FP (False Positives) is the number of not ready biodegradable samples wrongly classified as ready biodegradable.

b) Sensitivity (Sn)

$$\mathrm{Sn} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{5.7}$$

where, TP (True Positives) is the number of ready biodegradable samples correctly predicted as ready biodegradable. FN (False Negatives) is the number of ready biodegradable samples wrongly predicted as not ready biodegradable.

c) Error Rate is calculated as the complement of the average of specificity and sensitivity.

It should be remembered that all the AD approaches discussed in this thesis define a model's AD in its descriptor space. However, an attempt has been made in this chapter to better understand if the observations made evaluating for a model's AD in its descriptors space can be well reflected on its response domain. To achieve this, test samples excluded from the model's AD were evaluated for their corresponding error in prediction (absolute difference in their experimental and predicted response values). It could be interesting to see if the test samples rendered as unreliable in the model's descriptor space are also associated with higher prediction error or not. In theory, this is a reasonable assumption since structurally similar chemicals can be associated with similar descriptor values which collectively are able to capture the increasing or decreasing trend of the modelled endpoint. Thus, if a query/test chemical is excluded from the model's descriptor space, it cannot be predicted reliably either. However, in practice, exceptions may arise due to several reasons for instance, defects in experimental techniques/experimental variability or even over-fitted models.

5.3 CAESAR Bioconcentration factor models

5.3.1 Model description

OECD principle 1: A defined endpoint

CAESAR hybrid model provides prediction for Bioconcentration factor (BCF) in fish. Experimental data on BCF was obtained for two fish species, Cyprinus Carpio and salmonids using the OECD 305 protocol.

From regulatory point of view, BCF is of very high significance for REACH implementation. The BCF value for a given chemical can decide if it can be identified as bioaccumulative (if BCF>2000 or logBCF>3.3) or very bioaccumulative (if BCF>5000 or logBCF>3.7).

All the experimental BCF values used for developing this model were converted to their log units [38].

OECD principle 2: An unambiguous algorithm

CAESAR BCF model is a hybrid model derived combining the outputs from two different models (model A and model B). The training set of both these models consists of 378 samples, while the validation was carried out using a test set with 95 samples.

Both these models are Radial Basis Function Neural Network (RBFNN) [42], however, the earlier used an heuristic approach while the latter implemented Genetic Algorithm for descriptor selection. Table 5.1 reports the descriptors associated with model A and B, respectively. AD evaluation will be carried out for both these models [36-38].

Table 5.1 *List of descriptors used to develop CAESAR BCF models*

Descriptor	Description	Models
MlogP	Moriguchi octanol-water partition coefficient	Models A and B
Cl-089	Cl attached to C1(sp2)	Model A
GATS5V	Geary autocrrelatin – lag 5/weighed by atomic van der Waals volumes	Model A
X0Solv	Solvation connectivity index	Model B
SsCl	Sum of all (–Cl) E-State values in molecule	Model B
AEige	Absolute eigenvalue sum from electronegativity weighted distance matrix	Model A
BEHp2	Highest eigenvalue n. 2 of Burden matrix / weighed by atomic polarizabilities.	Models A and B
MATS5V	Moan autocorrelation – lag 5/weighed by atomic van der Waals volumes	Model B

OECD principle 3: A defined domain of applicability

The CAESAR model allows a user to understand its defined domain of applicability in the following three ways:

a) By evaluating the ranges of descriptor values: If a given test sample has any of its descriptor values outside the defined ranges, the user will be provided with an alert.

b) Identifying chemical fragments not included within the training space: If a test sample contains a chemical fragment that is not included within the chemical diversity of the training set, an error message will be generated.

c) Identifying the most similar training samples: For each test sample, six most similar training samples are shown. This allows a better understanding of the structural similarity between the predicted sample and the training space. This can also provide a good basis to interpret the reliability in prediction derived for the test samples [36-38].

OECD principle 4: Appropriate measures of goodness-of-fit, robustness and predictivity

Table 5.2 provides with the default statistical parameters for model A and B, retaining all the test samples within the model's AD.

Table 5.2 Model statistics for the CAESAR BCF models.

Model	Training set		Test set	
	R^2	*RMSE*	Q^2	*RMSEP* (d)
1) Model A	0.804	0.591	0.797	0.600
2) Model B	0.810	0.581	0.774	0.634

OECD principle 5: A mechanistic interpretation, if possible

The authors provided the following a posteriori interpretation towards the model descriptors in the QSAR Model Reporting Format (QMRF) of this model: The model is significantly relying on the MlogP descriptor. This descriptor seems to work quite well with chemicals containing C, N and O atoms, while it may not be very accurate for samples containing other atoms like Cl and P [38].

5.3.2 AD evaluation for CAESAR BCF model A

Table 5.3 provides with an overview of the results derived implementing various classical and novel AD approaches. Implementing PCA Bounding Box rendered two test samples outside the AD providing the most noticeably positive impact on the resulting Q^2. These samples were retained within the AD with classical Bounding Box. Excluding 29 samples outside the model's AD, Optimized Gaussian kernel approach was associated with the most restricted AD and the highest recorded Q^2 of 0.830 but obviously due to several test samples being outside the AD. The Q^2 slightly improved with the novel kNN-based AD approach, while no positive impact was observed with the LCMD based method with 8.4 % of the test samples discarded from the model's AD.

Table 5.3 *An overview of the results for AD evaluation on CAESAR BCF model A (Test set:95 samples)*

AD method	Samples outside AD (%)	Q^2	List of samples outside AD
Bounding Box	0	0.797	None
PCA Bounding Box (Using first 2 PCs)	2.1	0.804	33 40
Convex Hull	0	0.797	None
Leverage approach	4.2	0.803	18 33 43 61
Centroid dist. (Euclidean, 95 percentile)	4.2	0.804	33 43 61 91
Centroid dist. (Manhattan, 95 percentile)	4.2	0.804	33 43 61 91
Centroid dist. (Mahalanobis, 95 percentile)	4,2	0.803	18 33 43 61
kNN general thr (Euclidean, k=5)	8.4	0.797	3 33 34 40 61 82 83 94
kNN general thr. (Manhattan, k=5)	7.4	0.799	3 33 34 61 82 83 94
kNN general thr. (Mahalanobis, k=5)	10.5	0.794	3 33 34 40 61 80 82 83 91 94
Gaussian kernel: fixed	10.5	0.794	3 24 33 34 40 61 82 83 91 94
Gaussian kernel: optimized	30.5	0.830	3 9 12 22 24 33 34 38 40 45 47 51 53 54 56 61 68 69 75 76 80 82 83 87 89 91 93 94 95
Gaussian kernel: variable	15.8	0.787	3 9 24 33 34 40 43 61 80 82 83 89 91 94 95
Adaptive kernel	7.4	0.800	3 33 43 61 82 83 91
Epanechnikov kernel	8.4	0.800	3 33 40 43 61 83 91 94
kNN kernel (k=8)	9.5	0.797	3 33 34 40 43 61 83 91 94
Triangular kernel	11.6	0.792	3 24 33 34 40 61 80 82 83 91 94
Novel kNN approach (Euclidean, k=8)	6.3	0.801	3 33 40 61 82 83
Novel kNN approach (Manhattan, k=8)	8.4	0.797	3 33 34 61 80 82 83 94
Novel kNN approach (Mahalanobis, k=8)	8.4	0.797	3 33 34 40 61 82 83 94
Novel LCMD approach	8.4	0.786	3 34 43 61 80 82 83 94

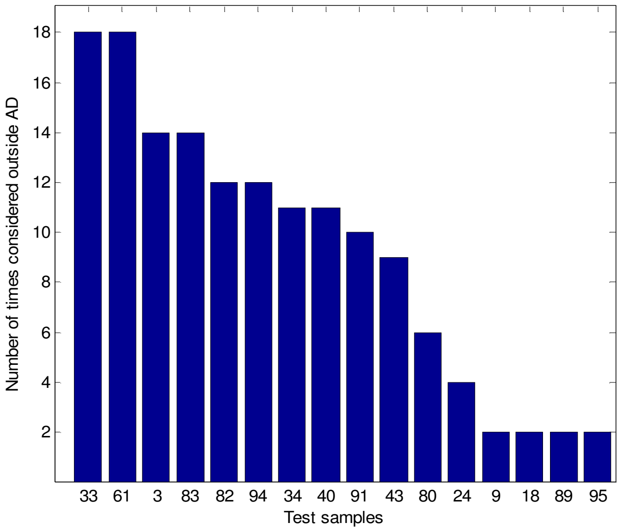

Figure 5.1 *Consensus test samples excluded from the AD of CAESAR BCF model A*

Figure 5.1 provides with the consensus test samples excluded from the model's AD implementing various classical and novel proposed AD approaches. Test samples 33 and 61 were identified as unreliable predictions implementing most of the AD approaches. Such resemblance in the final output from different approaches strengthens the decision to exclude unreliable test samples.

Table 5.4 provides with some useful information about the test samples considered outside the model's AD with different classical and novel approaches. Apart from several unreliably predicted samples, the list in this table also specifies some cases where the prediction error was quite negligible. For instance, sample 34 (tetrabromo-2-chlorotoluene) that was excluded from the model's AD with several approaches but was associated with a prediction error of log 0.01 units.

Table 5.4 *An overview of all the test samples excluded from the AD of CAESAR model A with different approaches*

Sample ID	Name	CAS	Exp. logBCF	Pred. logBCF	Abs. pred.error
3	Pentachlorophenol	87-86-5	2.50	1.84	0.66
9	3,6-Dichlorodibenzofuran	74918-40-4	3.01	3.13	0.12
12	2,2,4-Trimethyl-1,3-pentanediol	144-19-4	-1.00	0.64	1.64
18	3,4-Dichlorophenol	95-77-2	1.69	1.33	0.36
22	2,6-Dicyclohexylphenol	4821-19-6	2.89	2.10	0.79
24	2-Hydroxy-4-n-octoxybenzophenone	1843-05-6	1.90	1.94	0.04
33	Hexachlorobenzene	118-74-1	4.23	2.90	1.33
34	Tetrabromo-2-chlorotoluene	39569-21-6	3.98	3.97	0.01
38	Monochlorobenzene	108-90-7	1.13	1.61	0.48
40	Pentachlorobenzene	608-93-5	3.49	3.22	0.27
43	Trichlorometane	67-66-3	0.93	0.54	0.39
45	1,10-Dibromodecane	4101-68-2	1.78	2.68	0.90
47	Tetrachloroethylene	127-18-4	1.72	1.13	0.59
51	n-Pentadecane	629-62-9	1.22	2.68	1.46
53	2,2''-Methylenebis(6-t-buthyl-4-methylphenol)	119-47-1	1.97	2.33	0.36
54	Benzene-1,2-dicarboxylic acid bis (2-ethylhexyl) ester	117-81-7	1.19	1.47	0.28
56	Triethanolamine	102-71-6	0.59	1.01	0.42
61	2,4,6-Trichloroaniline	634-93-5	2.00	1.41	0.59
68	2,2''-Dichlorohydrazobenzene	782-74-1	3.65	3.19	0.46
69	1-(N-Phenylamino)naphthalene	90-30-2	3.23	2.74	0.49
75	Tris(1,3-dichloro-2-propyl)phosphate	13674-87-8	0.13	1.75	1.62
76	p-Phenylphenol	92-69-3	1.59	1.96	0.37
80	4-Chloro-1-nitro-2(trifluoromethyl) benzene	118-83-2	1.87	2.03	0.16
82	N-Hexamethylolmelamine hexamethylether	3089-11-0	0.28	0.06	0.22
83	Disperse Yellow 163	71767-67-4	1.56	1.16	0.40
87	O,O-Dimethyl-S-(N-methylcarbamoylmethyl) phosphorodithioate	60-51-5	-0.26	0.12	0.38
89	m-nitrobenzene sulfonic acid	98-47-5	0.70	0.33	0.37
91	Tris(p-isopropylphenyl)phosphate	26967-76-0	1.50	2.02	0.52
93	1-Amino-8-naphthol-3,6-disulfonic acid	90-20-0	0.46	0.45	0.01
94	3,3''-Dichloro-5,5''-benzidine disulfonic acid	123251-96-7	0.20	0.04	0.16
95	Disperse Yellow 64	10319-14-9	1.08	1.80	0.72

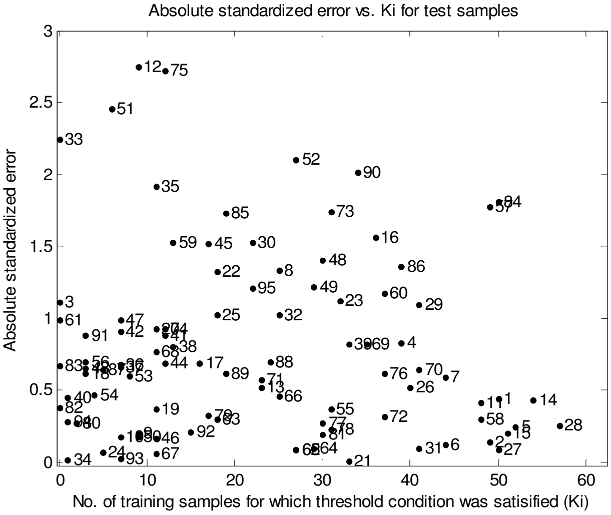

Figure 5.2 *K_j vs. Absolute standardized error plot for the test samples of CAESAR BCF model A*

Figure 5.2 provides with a plot from the novel kNN based AD approach that tries to compare the observations made in the model's descriptor space and the response domain. The test samples are clearly showing a decreasing pattern from left towards right indicating a lowering prediction error with a corresponding increase in the number of training thresholds satisfied by the test samples. This plot also tries to graphically reflect upon the observations made from the previous table and plot for this model. Samples 33 and 61 for instance, are associated with reasonably higher prediction error and were able to satisfy none of the training thresholds indicating them being unreliably predicted. On the other hand, test sample 14 and 28 were associated with very low prediction error and satisfied maximum training thresholds. Thus, higher structural similarity resulted in better predictions as evident from this plot.

5.3.3 AD evaluation for CAESAR BCF model B

Table 5.5 provides with an overview of the results derived implementing various classical and novel AD approaches on CAESAR BCF model B.

Table 5.5 *An overview of the results for AD evaluation on CAESAR BCF model B (Test set: 95 samples)*

AD method	**Samples outside AD (%)**	**Q^2**	**List of samples outside AD**
Bounding Box	0	0.774	None
PCA Bounding Box (First 2 PCs)	0	0.774	None
Convex Hull	0	0.774	None
Leverage approach	3.2	0.767	43 50 91
Centroid dist. (Euclidean, 95 percentile)	3.2	0.767	43 50 91
Centroid dist. (Manhattan, 95 percentile)	5.3	0.764	36 37 43 50 91
Centroid dist. (Mahalanobis, 95 percentile)	3.2	0.767	43 50 91
kNN general thr. (Euclidean, k=5)	1.1	0.772	82
kNN general thr. (Manhattan, k=5)	1.1	0.772	82
kNN general thr. (Mahalanobis, k=5)	4.2	0.783	75 82 87 94
Gaussian kernel: fixed	13.7	0.778	3 33 34 40 43 50 74 75 82 83 87 91 94
Gaussian kernel: optimized	29.5	0.787	3 21 33 34 40 43 44 46 47 48 50 52 54 56 61 73 74 75 80 81 82 83 87 88 91 93 94 95
Gaussian kernel: variable	22.1	0.777	3 33 34 40 43 44 47 48 50 73 74 75 80 81 82 83 87 88 91 93 94
Adaptive kernel	2.1	0.769	43 82
Epanechnikov kernel	3.2	0.769	33 43 82
kNN kernel (k=8)	4.2	0.767	33 40 43 82
Triangular kernel	10.5	0.786	3 33 34 50 74 75 82 83 87 94
Novel kNN approach (Euclidean, k=8)	3.2	0.785	33 75 82
Novel kNN approach (Manhattan, k=8)	7.4	0.779	33 40 74 75 82 83 87
Novel kNN approach (Mahalanobis, k=8)	6.3	0.782	33 74 75 82 83 87
Novel LCMD approach	5.3	0.764	43 50 82 83 91

The range and geometric-based approaches retained all the test samples inside the model’s AD. All other set of approaches associated some test samples being unreliably predicted, however, no major impacts were observed on the resulting Q^2. This includes both the novel proposed AD approaches. This parameter varied slightly even after excluding several other test samples as obvious in the case of Gaussian kernel based approaches.

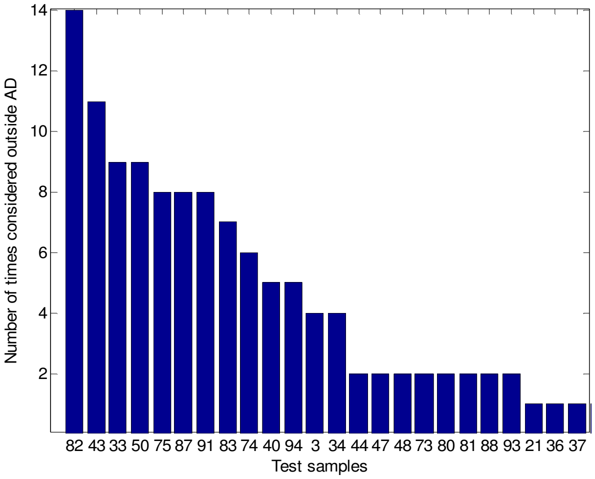

Figure 5.3 *Consensus test samples excluded from the AD of CAESAR BCF model B*

Figure 5.3 provides with an overview of the consensus test samples being excluded from the model's AD implementing different approaches. Samples 43 and 82 (Trichlorometane and N-hexamethylolmelamine hexamethylether) were associated with the maximum frequency, thus indicating them being excluded from the AD using several different algorithms independent of each other. Several other test samples that were excluded by only one AD approach were not highlighted in the figure, however, Table 5.6 provides with some useful information about all the test samples excluded by different AD approaches.

Table 5.6 *An overview of all test samples excluded from the AD of CAESAR model B with different approaches*

Sample ID	Name	CAS	Exp. logBCF	Pred. logBCF	Abs .pred.error
3	Pentachlorophenol	87-86-5	2.50	1.75	0.75
21	Cyclohexane	110-82-7	1.92	1.98	0.06
33	Hexachlorobenzene	118-74-1	4.23	3.57	0.66
34	Tetrabromo-2-chlorotoluene	39569-21-6	3.98	2.77	1.21
36	2,3,4,2'',5''-Pentachlorobiphenyl	38380-02-8	4.02	4.53	0.51
37	2,3'',4,4'',6-Pentachlorobiphenyl	56558-17-9	4.81	4.52	0.29
40	Pentachlorobenzene	608-93-5	3.49	3.48	0.01
43	Trichlorometane	67-66-3	0.93	1.03	0.10
44	1,1,2,2-Tetrachloroethane	79-34-5	0.93	0.91	0.02
46	1,1,2,2-Tetrachloro-1,2-difluoroethane	76-12-0	1.78	1.25	0.53
47	Tetrachloroethylene	127-18-4	1.72	0.66	1.06
48	Dibromoneopentylglycol	3296-90-0	-0.04	0.22	0.26
50	Heptachlor	76-44-8	3.95	4.17	0.22
52	1,3,5-Tri-tert-butylbenzene	1460-02-2	4.37	2.65	1.72
54	Benzene-1,2-dicarboxylic acid bis (2-ethylhexyl) ester	117-81-7	1.19	1.49	0.30
56	Triethanolamine	102-71-6	0.59	0.28	0.31
61	2,4,6-Trichloroaniline	634-93-5	2.00	1.45	0.55
73	2,2''-Dichlorodiethyl ether	111-44-4	-0.08	0.77	0.85
74	Trichloroacetic acid	76-03-9	-0.15	-0.22	0.07
75	Tris(1,3-dichloro-2-propyl)phosphate	13674-87-8	0.13	1.74	1.61
80	4-Chloro-1-nitro-2(trifluoromethyl) benzene	118-83-2	1.87	2.28	0.41
81	3-Nitrophthalic acid	603-11-2	0.72	0.26	0.46
82	N-Hexamethylolmelamine hexamethylether	3089-11-0	0.28	0.46	0.18
83	Disperse Yellow 163	71767-67-4	1.56	1.07	0.49
87	O,O-Dimethyl-S-(N-methylcarbamoylmethyl) phosphorodithioate	60-51-5	-0.26	0.34	0.60
88	2,2-Dichloropropionic acid	N/A	0.85	-0.01	0.86
91	Tris(p-isopropylphenyl)phosphate	26967-76-0	1.50	1.25	0.25
93	1-Amino-8-naphthol-3,6-disulfonic acid	90-20-0	0.46	0.65	0.19
94	3,3''-Dichloro-5,5''-benzidine disulfonic acid	123251-96-7	0.20	0.42	0.22
95	Disperse Yellow 64	10319-14-9	1.08	1.44	0.36

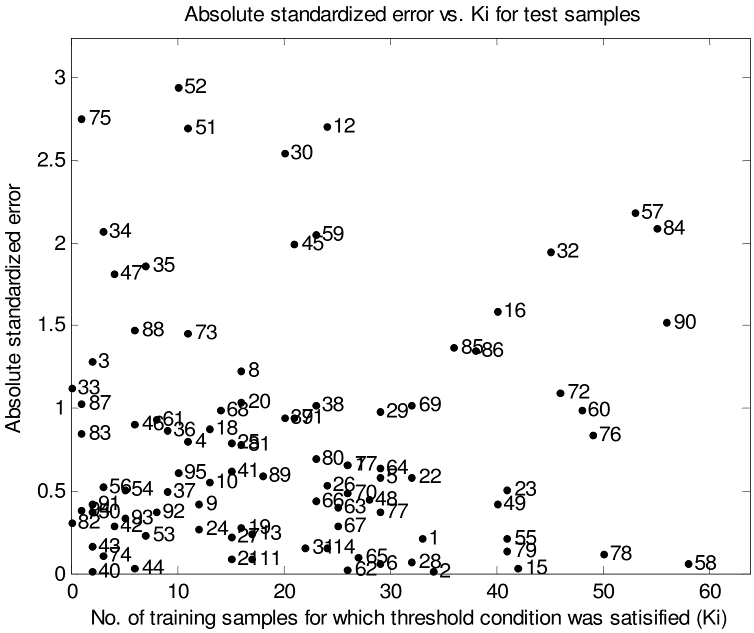

Figure 5.4 : *Kj vs. Absolute standardized error plot for the test samples of CAESAR BCF model B*

Figure 5.4 provides with the K_j vs. absolute standardized error plot derived from the novel kNN based AD approach. Several test samples like 30, 32, 57 and 84 were hindering the expected lowering pattern in prediction error with increase K_j values. Such samples indicate being associated with high predictor error despite of their higher K_j values which in theory shouldn't be the case. However, this plot tries to reflect the outcome of AD evaluation in the model's descriptor space taking into account the model's response domain and the observations in these two different spaces may not converge necessarily.

5.4 Ready biodegradability of chemicals

5.4.1 Model description

OECD principle 1: A defined endpoint

This set of classification models is aimed at evaluating the persistence of chemical substances in the environment by predicting their ready biodegradability. Since the accumulation of persistent chemicals could lead to hazardous impacts on a longer time scale, REACH regulation requires the information relevant to the ready biodegradability of chemical substances that are produced or imported in quantities greater than one ton per year. Being classification models, their resulting predictions for query chemicals are either if they are Ready Biodegradable (RB) or not ready biodegradable (NRB) [39].

OECD principle 2: An unambiguous algorithm

Three QSAR models were developed using the following different classification modelling techniques to incorporate linear, non-linear and local models: *k* Nearest Neighbours (kNN), partial least squares discriminant analysis (PLSDA) and support vector machines (SVM). Since individual models can account for different amounts of noise, two consensus models were developed in order to improve the overall quality in predictions. The first consensus model allocated the most frequent class predicted for a query chemical using the above three classification models. On the other hand, the second consensus model allocated a given query chemical to a class that was predicted the same with all the three individual models; otherwise no class was assigned. Model calibration in all the cases was carried out using a data set of 837 molecules while it was validated on a test set consisting of 218 molecules. Further, the developed models were evaluated on an external validation test set consisting of 670 molecules. Table 5.7 provides with an overview of all the molecular descriptors from the DRAGON 6 package used in developing the three classification models [43]. The descriptor selection for this set of models was performed using Genetic Algorithm (GA).

Table 5.7 *List of descriptors used to develop the biodegradability models.*

Descriptor	**Description**	**Model**
B01[C-Br]	presence/absence of C–Br at topological distance 1	PLSDA
B03[C-Cl]	presence/absence of C–Cl at topological distance 3	PLSDA
B04[C-Br]	presence/absence of C–Br at topological distance 4	PLSDA
C%	percentage of C atoms	kNN– PLSDA
C-026	R–CX–R	SVM
F01[N-N]	frequency of N–N at topological distance 1	kNN
F02[C-N]	frequency of C–N at topological distance 2	SVM
F03[C-N]	frequency of C–N at topological distance 3	kNN
F03[C-O]	frequency of C–O at topological distance 3	PLSDA
F04[C-N]	frequency of C–N at topological distance 4	kNN–PLSDA
HyWi_B(m)	hyper-Wiener-like index (log function) from Burden matrix weighted by mass	PLSDA
J_Dz(e)	Balaban-like index from Barysz matrix weighted by Sanderson electronegativity	kNN
LOC	lopping centric index	PLSDA
Me	mean atomic Sanderson electronegativity (scaled on Carbon atom)	PLSDA
Mi	mean first ionization potential (scaled on carbon atom)	PLSDA
N-073	Ar2NH/Ar3N/Ar2N–Al/R···N···R	PLSDA
nArCOOR	number of esters (aromatic)	SVM
nArNO2	number of nitro groups (aromatic)	PLSDA
nCb-	number of substituted benzene C(sp2)	kNN–SVM
nCIR	number of circuits	PLSDA
nCp	number of terminal primary C(sp3)	kNN
nCrt	number of ring tertiary C(sp3)	SVM
nCRX3	number of CRX3	PLSDA
nHDon	number of donor atoms for H-bonds (N and O)	SVM
nHM	number of heavy atoms	kNN
nN	number of nitrogen atoms	SVM
nN-N	number of N hydrazines	PLSDA–SVM
nO	number of oxygen atoms	kNN–PLSDA
NssssC	number of atoms of type ssssC	kNN–SVM
nX	number of halogen atoms	SVM
Psi_i_1d	intrinsic state pseudoconnectivity index–type 1d	PLSDA
Psi_i_A	intrinsic state pseudoconnectivity index□type S average	SVM
SdO	sum of dO E-states	PLSDA
SdssC	sum of dssC E-states	kNN
SM6_B(m)	spectral moment of order 6 from Burden matrix weighted by mass	SVM
SM6_L	spectral moment of order 6 from Laplace matrix	PLSDA
SpMax_A	leading eigenvalue from adjacency matrix (Lovasz–Pelikan index)	PLSDA
SpMax_B(m)	leading eigenvalue from Burden matrix weighted by mass	SVM
SpMax_L	leading eigenvalue from Laplace matrix	kNN–PLSDA–SVM
SpPosA_B(p)	normalized spectral positive sum from Burden matrix weighted by polarizability	PLSDA
TI2_L	second Mohar index from Laplace matrix	PLSDA

OECD principle 3: A defined domain of applicability

The article from where these models were retrieved does not provide with any direct evaluation of the model's AD. However, an article focussing exclusively on evaluating their AD is currently in preparation.

OECD principle 4: Appropriate measures of goodness-of-fit, robustness and predictivity

Table 5.8 provides with the default model statistical parameters, retaining all the test samples within the model's AD. It should be noted that for the second consensus model the not assigned molecules were not considered to evaluate the TP and TN.

OECD principle 5: A mechanistic interpretation, if possible

The authors provided following a posteriori mechanistic interpretation to relate the chosen set of descriptors to the modelled endpoint:

The usefulness of the chosen descriptors was interpreted deriving score and loading plots from the PCA study on the training set and projecting test set molecules over the training space. For the PLSDA model, the descriptors were related to biodegradability directly using the latent variables used for model development.

kNN model: Descriptors encoding information about the substituted benzenes and nitrogen (functional group counts based descriptor nCb- and 2D atom pairs based descriptors F01[N-N], F04[C-N], and F03[C-N]) differentiated the NRB from RB molecules based on the presence of cyclic and nitro groups. nHM indicated the presence of heavy atoms which may be more relevant to the NRB molecules. Since RB molecules are less branched than NRB ones, descriptors SdssC, NssssC and nCp were more oriented towards the NRB molecules indicating that increased branching molecules could lower the ready biodegradability.

Table 5.8 *Model statistics for the biodegradability models.*

Model	Desc	k/LVs/ c	Fitting			test set			validation set		
			ER	**Sn**	**Sp**	**ER**	**Sn**	**Sp**	**ER**	**Sn**	**Sp**
kNN	12	6	0.14	0.84	0.89	0.15	0.81	0.9	0.17	0.75	0.91
PLSDA	23	5	0.14	0.88	0.83	0.15	0.83	0.87	0.17	0.80	0.86
SVM	14	5	0.14	0.81	0.92	0.14	0.82	0.91	0.18	0.74	0.91
consensus 1	41		0.11	0.86	0.91	0.13	0.82	0.92	0.17	0.76	0.91
consensus 2	41		0.07	0.91	0.95	0.09	0.88	0.94	0.13	0.81	0.94
			(19% not assigned)			(15% not assigned)			(13% not assigned)		

Desc: Descriptors used, *k*/LVs/c: indicates the optimal parameters, no. of nearest neighbours (*k*) for kNN, number of latent variables (LVs) for PLSDA and the cost (c) for SVM. ER: Error Rate, Sn: Sensitivity indicating correctly predicted non ready biodegradable, Sp: Specificity indicating correctly predicted ready biodegradable

PLSDA model: Matrix based descriptors contained information about the molecular branching and based on the significant latent variables used, they were clearly oriented towards the NRB molecules, which is in agreement with the findings that lower branching favours ready biodegradation.

The descriptors containing information about cycles, nitrogen and halogens were oriented towards NRB molecules like for the kNN model. Descriptors indicating the presence of oxygen further differentiated the RB from NRB molecules, indeed functional groups with oxygen atoms assist biodegradation process.

SVM model: Several descriptors encoding information about the molecular branching, aromatic groups and halogens (including matrix-based descriptors, constitutional indices and atom-centred fragments) differentiated the RB from NRB molecules, being more oriented towards the latter ones.

To better understand the usefulness of matrix-based descriptors towards ready biodegradability, their encoded information was further explored by performing OLS regression between these targeted matrix-based descriptors and DRAGON molecular descriptors. As a result of this analysis, these matrix-based descriptors were associated with properties like molecular branching, cyclicity and molecular size which are significant parameters impacting the biodegradability.

5.4.2 AD evaluation on consensus models

One of the important aspects of considering this case study is to perform the AD evaluation on consensus models. Since consensus models are mainly relying on the output derived from the set of primary models (in this case, kNN, PLSDA and SVM models), following strategy was adopted to deal with defining the AD of the resulting two consensus models.

The AD of all the three individual models was evaluated like for the other case studies using all the different classical and novel AD approaches discussed earlier (though the results are not discussed for these models). For both the consensus models, a given test sample was considered within its AD with a given approach only if it was retained inside the AD of all the three individual models. The decision rule could be interesting since the final decision to retain or discard a test sample in the AD depends on the output from three different models-local, linear and non-linear. The decision rule adopted towards defining the AD resembles the criterion used by the second model in considering a test sample to be RB or NRB.

Tables 5.9 and 5.10 provide with an overview of the results derived with different classical and novel AD approaches on first and second consensus models, respectively. The test samples listed being outside the AD are the same since the same AD criterion was followed by both the consensus models. The difference however lies in the model statistical parameters since the predicted responses for both these models are different. Moreover, there are some test samples with unassigned class in the case of second consensus model. In both the cases, no significant impacts were observed on the resulting statistical parameters.

In theory, a test sample can only be considered within the consensus model's AD provided that it was included within the AD of three individual models which were based on very diverse algorithms towards model development. If a test sample falls inside the AD of local, linear and non-linear models, this further adds to the reliability in considering such test samples within the model's AD. However, such strict criterion may also make the defined AD more restrictive to the test samples. For both the consensus models, none of the approaches were able to significantly improve the model statistical parameters retaining reasonable number of test samples within the model's AD.

Table 5.9 *An overview of the results for AD evaluation on the first consensus model*

AD method	Samples outside AD (%)	ER	Sn	Sp	List of samples outside AD
Bounding Box	4.1	0.14	0.82	0.91	2 73 130 166 181 189 192 215 217
PCA Bounding Box	0.5	0.13	0.82	0.92	217
Convex Hull	-	-	-	-	-
Leverage approach	13.3	0.13	0.84	0.90	2 19 24 27 57 73 74 76 77 78 80 83 91 94 96 130 134 146 159 164 166 186 189 190 192 200 215 216 217
Centroid dist. (Euclidean, 95 percentile)	9.2	0.14	0.82	0.91	57 73 74 76 77 78 94 134 159 164 166 172 186 190 192 200 202 215 216
Centroid dist. (Manhattan, 95 percentile)	8.7	0.14	0.82	0.91	57 73 74 76 77 78 91 94 134 151 152 159 166 192 200 202 215 216 217
Centroid dist. (Mahalanobis, 95 percentile)	10.6	0.13	0.83	0.90	2 27 57 73 74 76 77 78 83 91 94 96 130 134 159 164 166 186 189 192 200 215 217
kNN general thr (Euclidean, k=5)	10.1	0.13	0.83	0.90	27 57 73 76 77 91 94 134 147 151 152 164 166 186 189 190 192 196 200 215 216 217
kNN general thr. (Manhattan, k=5)	8.7	0.14	0.82	0.91	73 76 77 80 91 94 134 147 151 152 166 186 189 190 192 200 215 216 217
kNN general thr. (Mahalanobis, k=5)	16.1	0.13	0.84	0.90	2 24 27 57 73 74 75 76 77 78 80 83 90 91 94 130 134 147 151 152 158 159 164 166 173 186 189 190 192 196 200 212 215 216 217

AD method	Samples outside AD (%)	ER	Sn	Sp	List of samples outside AD
Gaussian kernel: fixed	34.9	0.13	0.85	0.89	2 5 7 19 24 27 30 47 48 51 57 58 62 64 67 69 72 73 75 76 77 78 79 80 82 83 88 89 90 91 92 94 96 105 106 110 111 112 113 115 116 119 121 122 124 126 127 130 133 134 135 137 140 141 142 144 146 147 148 149 151 152 153 154 157 158 159 160 161 164 166 168 172 173 174 178
Gaussian kernel: optimized	88.5	0.00	1.00	1.00	All test samples except 9 14 15 16 17 18 22 41 46 53 74 97 103 108 109 117 128 151 152 153 154 155 156 157 158 167 171 201 202 203 205 206 208 209
Gaussian kernel: variable	14.7	0.13	0.84	0.90	24 27 57 73 74 76 77 78 83 91 94 105 134 135 147 151 152 158 159 161 164 166 186 187 189 190 192 196 200 215 216 217
Adaptive kernel	11.9	0.13	0.83	0.90	27 57 73 74 75 76 77 78 83 94 105 134 147 151 152 164 166 186 189 190 192 196 200 215 216 217
Epanechnikov kernel	20.6	0.14	0.84	0.89	2 24 27 57 73 75 76 77 78 80 83 90 91 94 96 110 112 116 130 133 134 135 144 147 151 152 154 158 159 161 164 166 172 173 185 186 187 189 190 192 196 200 215 216 217
kNN kernel	11.9	0.13	0.84	0.90	24 27 57 73 74 76 77 78 83 91 94 134 147 151 152 164 166 186 189 190 192 196 200 215 216 217
Triangular kernel	77.1	0.05		1.00	All test samples except 9 11 12 13 14 15 16 17 18 22 29 33 34 38 39 41 44 46 53 59 60 66 68 74 84 87 97 99 100 102 103 108 109 114 117 128 162 163 165 167 169 170 171 177 183 201 203 205 206 208 209
Novel kNN approach (Euclidean)	11.0	0.13	0.83	0.90	27 57 73 75 76 77 80 94 134 152 154 158 161 164 166 173 186 189 190 192 200 215 216 217
Novel kNN approach (Manhattan)	11.9	0.14	0.81	0.90	2 57 73 76 77 80 91 94 105 134 135 147 151 152 161 164 166 186 187 189 190 192 200 215 216 217
Novel kNN approach (Mahalanobis)	11.5	0.13	0.83	0.90	2 27 73 76 77 90 91 94 110 134 147 151 152 158 164 166 186 187 189 190 192 200 215 216 217
Novel LCMD approach	11.5	0.14	0.83	0.90	2 27 57 73 74 76 77 78 83 91 94 96 130 134 146 159 164 166 186 189 192 200 215 216 217

Table 5.10 *An overview of the results for AD evaluation on the second consensus model*

AD method	Samples outside AD (%)	ER	Sn	Sp	List of samples outside AD
Bounding Box	4.1	0.08	0.89	0.94	2 130 166 181 189 192 215 217
PCA Bounding Box	0.5	0.08	0.90	0.94	217
Convex Hull	-	-	-	-	-
Leverage approach	13.3	0.07	0.93	0.93	19 24 27 57 73 74 76 77 78 80 83 91 94 96 130 134 146 159 164 166 186 189 190 192 200 215 216 217
Centroid dist. (Euclidean, 95 percentile)	9.2	0.08	0.90	0.94	57 73 74 76 77 78 94 134 159 164 166 172 186 190 192 200 202 215 216
Centroid dist. (Manhattan, 95 percentile)	8.7	0.08	0.90	0.94	57 73 74 76 77 78 91 94 134 151 152 159 166 192 200 202 215 216 217
Centroid dist. (Mahalanobis, 95 percentile)	10.6	0.08	0.91	0.93	2 27 57 73 74 76 77 78 83 91 94 96 130 134 159 164 166 186 189 192 200 215 217
kNN general thr (Euclidean, k=5)	10.1	0.08	0.91	0.93	27 57 73 76 77 91 94 134 147 151 152 164 166 186 189 190 192 196 200 215 216 217
kNN general thr. (Manhattan, k=5)	8.7	0.08	0.90	0.93	73 76 77 80 91 94 134 147 151 152 166 186 189 190 192 200 215 216 217
kNN general thr. (Mahalanobis, k=5)	16.1	0.07	0.93	0.93	2 24 27 57 73 74 75 76 77 78 80 83 90 91 94 130 134 147 151 152 158 159 164 166 173 186 189 190 192 196 200 212 215 216 217
Gaussian kernel: fixed	34.9	0.07	0.93	0.92	5 7 19 24 27 30 47 48 51 57 58 62 64 67 69 72 73 75 76 77 78 79 80 82 83 88 89 90 91 92 94 96 105 106 110 111 112 113 115 116 119 121 122 124 126 127 130 133 134 135 137 140 141 142 144 146 147 148 149 151 152 153 154 157 158 159 160 161 164 166 168 172 173 174 178
Gaussian kernel: optimized	88.5	0.00	1.00	1.00	All test samples except 9 14 15 16 17 18 22 41 46 53 74 97 103 108 109 117 128 151 152 153 154 155 156 157 158 167 171 201 202 203 205 206 208 209
Gaussian kernel: variable	14.7	0.07	0.93	0.93	24 27 57 73 74 76 77 78 83 91 94 105 134 135 147 151 152 158 159 161 164 166 186 187 189 190 192 196 200 215 216 217

AD method	Test outside AD (%)	ER	Sn	Sp	List of samples outside AD
Adaptive kernel	11.9	0.08	0.91	0.93	27 57 73 74 75 76 77 78 83 94 105 134 147 151 152 164 166 186 189 190 192 196 200 215 216 217
Epanechnikov kernel	20.6	0.08	0.93	0.92	24 27 57 73 75 76 77 78 80 83 90 91 94 96 110 112 116 130 133 134 135 144 147 151 152 154 158 159 161 164 166 172 173 185 186 187 189 190 192 196 200 215 216 217
kNN kernel	11.9	0.07	0.93	0.93	24 27 57 73 74 76 77 78 83 91 94 134 147 151 152 164 166 186 189 190 192 196 200 215 216 217
Triangular kernel	77.1	0.08	0.95	1.00	All test samples except 9 11 12 13 14 15 16 17 18 22 29 33 34 38 39 41 44 46 53 59 60 66 68 74 84 87 97 99 100 102 103 108 109 114 117 128 162 163 165 167 169 170 171 177 183 201 203 205 206 208 209
Novel kNN approach (Euclidean)	11.0	0.07	0.91	0.93	27 57 73 75 76 77 80 94 134 152 154 158 161 164 166 173 186 189 190 192 200 215 216 217
Novel kNN approach (Manhattan)	11.9	0.09	0.89	0.93	57 73 76 77 80 91 94 105 134 135 147 151 152 161 164 166 186 187 189 190 192 200 215 216 217
Novel kNN approach (Mahalanobis)	11.5	0.08	0.91	0.93	27 73 76 77 90 91 94 110 134 147 151 152 158 164 166 186 187 189 190 192 200 215 216 217
Novel LCMD approach	11.5	0.08	0.91	0.93	2 27 57 73 74 76 77 78 83 91 94 96 130 134 146 159 164 166 186 189 192 200 215 216 217

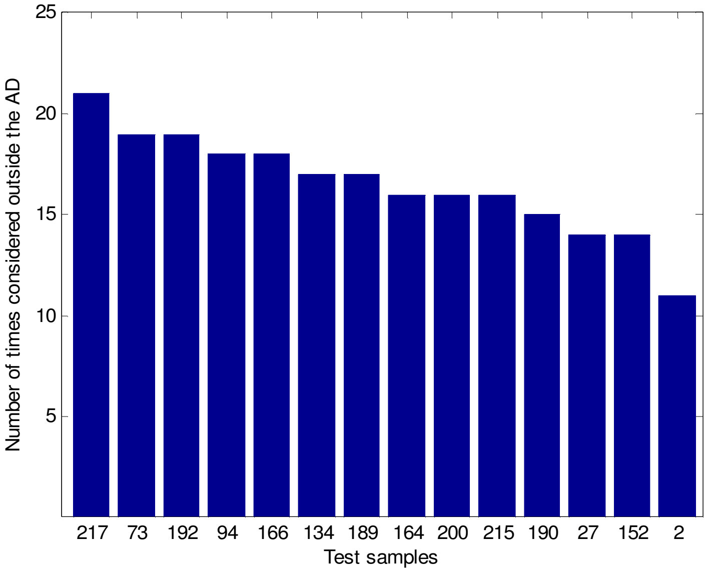

Figure 5.5 *Consensus test samples excluded from the AD of consensus models*

Figure 5.5 provides with an overview of the most commonly excluded test samples from the AD of the both consensus models.

Table 5.11 provides with an overview of all the test samples considered outside the AD with different approaches. It enlists almost entire test set since with approaches like fixed and optimized gaussian kernel as well as triangular kernel, a huge number of test samples were considered as outside the AD. For each of the test sample listed in this table, its experimental response as well as the predicted class from both the consensus models were provided. Several of the test samples listed in Figure 5.5 were predicted reliably even if they were rendered as unreliable in the model's descriptor space being excluded from the model's AD with several diverse approaches. This resembles the observation made for the regression models dealt as case studies earlier in this thesis work. This implies that the results derived in a model's descriptor space may not necessarily reflect the results derived in the response domain of that model.

Table 5.11 *An overview of all the test samples excluded from the AD of both consensus models with different approaches*

Sample ID	Name	CAS	Exp.class	consensus 1	consensus 2
1	n-heptane	142-82-5	RB	RB	RB
2	ethylene oxide	75-21-8	RB	RB	RB
3	Toluene	108-88-3	RB	RB	RB
4	di-n-butylamine	111-92-2	RB	RB	RB
5	3,7-dimethyl-1,6-octadien-3-ol	78-70-6	RB	NRB	NRB
6	3,6-dioxadecan-1-ol	112-34-5	RB	RB	RB
7	n-butyraldehyde	123-72-8	RB	RB	RB
8	4-hydroxy-4-methyl-2-pentanone	123-42-2	RB	NRB	not assigned
10	bis(2-ethylhexyl) fumarate	141-02-6	RB	RB	RB
11	12-hydroxyoctadecanoic acid	106-14-9	RB	RB	RB
12	Nonadecaneonitrile	28623-46-3	RB	RB	RB
13	(dichloromethyl)benzene	98-87-3	RB	NRB	NRB
19	bis(2-hydroxyethyl) terephthalate	959-26-2	RB	RB	RB
20	4-hydroxybenzonitrile	767-00-0	RB	NRB	not assigned
21	p-toluenesulfonic acid	104-15-4	RB	RB	not assigned
23	methyl 3-oxo-2-pentylcyclopentylacetate	24851-98-7	RB	RB	not assigned
24	Imidazole	288-32-4	RB	NRB	NRB
25	3-hydroxypyridine	109-00-2	RB	RB	RB
26	1-hexene	592-41-6	RB	RB	RB
27	isopropyl bromide	75-26-3	RB	NRB	NRB
28	n-butylamine	109-73-9	RB	RB	RB
29	hexadecan-1-ol	36653-82-4	RB	RB	RB
30	2-methoxyethanol	109-86-4	RB	RB	RB
31	propyl acetate	109-60-4	RB	RB	RB
32	13-docosenoamide	112-84-5	RB	RB	not assigned
33	adipic acid	124-04-9	RB	RB	RB
34	2-methoxyethyl acrylate	3121-61-7	RB	RB	RB
35	2-hydroxypropyl methacrylate	923-26-2	RB	RB	RB
36	2,4-hexadienic acid (synonym:sorbic acid)	110-44-1	RB	RB	RB
37	2-methylenesuccinic acid	97-65-4	RB	RB	RB
38	butyl acetoacetate	591-60-6	RB	RB	RB
39	Aniline	62-53-3	RB	NRB	not assigned
40	benzyl methacrylate	2495-37-6	RB	RB	RB
42	stylene oxide	96-09-3	RB	NRB	not assigned
43	benzoylaminoacetic acid	495-69-2	RB	RB	RB
44	2-(methylamino)benzoic acid	119-68-6	RB	RB	not assigned
45	alpha-terpineol	98-55-5	RB	NRB	NRB
47	3-acetyl-6-methyl-2,4(3H)-pyrandione (synonym:dehydroacetic acid)	520-45-6	RB	RB	RB
48	Xylitol	87-99-0	RB	RB	RB
49	Benzoin	119-53-9	RB	RB	not assigned
50	beta-alanine	107-95-9	RB	RB	RB

Sample ID	Name	CAS	Exp.class	consensus 1	consensus 2
51	1-chlorooctane	111-85-3	RB	RB	not assigned
52	2-ethoxyethanol	110-80-5	RB	RB	RB
54	methyl dodecanoate	111-82-0	RB	RB	RB
55	succinic acid	110-15-6	RB	RB	RB
56	2-hydroxyethyl acrylate	818-61-1	RB	RB	RB
57	2-hydroxy-1,2,3-propanetricarboxylic acid	77-92-9	RB	RB	not assigned
58	DL-tartaric acid	133-37-9	RB	RB	RB
59	sec-butyl alcohol	78-92-2	RB	RB	RB
60	terephthalic acid	100-21-0	RB	RB	RB
61	Phenylacetonitrile	140-29-4	RB	RB	RB
62	1-methyl-4-(1-methylvinyl)cyclohexene	138-86-3	RB	NRB	not assigned
63	cyclohexyl methacrylate	101-43-9	RB	RB	RB
64	2-(methylamino)ethanol	109-83-1	RB	RB	RB
65	1,1'-iminodi-2-propanol	110-97-4	RB	RB	RB
66	2-[2-(2-methoxyethoxy)ethoxy]ethanol	112-35-6	RB	RB	RB
67	chloroacetic acid	79-11-8	RB	RB	RB
68	dioctyl phthalate(synonym:di-n-octyl phthalate)	117-84-0	RB	RB	RB
69	dicyclohexyl benzene-1,2-dicarboxylate	84-61-7	RB	NRB	not assigned
70	beta-naphthol	135-19-3	RB	NRB	NRB
71	Pyridine	110-86-1	RB	NRB	not assigned
72	sorbitan monolaurate	1338-39-2	RB	RB	RB
73	Perfluoro(1,2-dimethylcyclohexane)	306-98-9	NRB	NRB	NRB
75	1,1,1-trichloro-2,2-bis(4-chlorophenyl)ethane (synonym:DDT)	50-29-3	NRB	NRB	NRB
76	2-(3,5-di-tert-butyl-2-hydroxyphenyl)benzotriazole	3846-71-7	NRB	NRB	NRB
77	2,4-di-tert-butyl-6-(5-chloro-2H-1,2,3-benzotriazol-2-yl)phenol	3864-99-1	NRB	NRB	NRB
78	mixture of 1,2,4,5,6,7,8,8-octachloro-2,3,3a,4,7,7a-hexahydro-4,7-methano-1H-indene, 1,4,5,6,7,8,8-heptachloro-3a,4,7,7a-tetrahydro-4,7-methano-1H-indene and their analogue compounds	76-44-8	NRB	NRB	NRB
79	1(a),2(a),3(a),4(e),5(e),6(e)-hexachlorocyclohexane (synonym:gamma-BHC)	608-73-1	NRB	NRB	NRB
80	trichloronitromethane (synonym:chloropicrine)	76-06-2	NRB	NRB	NRB
81	N,N,N',N'-Tetramethyl-2,2'-oxybis(ethylamine)	3033-62-3	NRB	NRB	not assigned
82	1,3-dimethylurea	96-31-1	NRB	RB	not assigned
83	2,2-Dibromo-2-cyanoacetamide	10222-01-2	NRB	NRB	NRB
84	2-Chloro-4-nitroaniline	121-87-9	NRB	NRB	NRB
85	4-nitro-o-anisidine	97-52-9	NRB	NRB	NRB
86	2-chloro-1,4-dimethoxybenzene	2100-42-7	NRB	NRB	not assigned
87	3-Methyl-4-(methylsulfanyl)phenol	3120-74-9	NRB	NRB	NRB
88	2-amino-5-nitrobenzonitrile	17420-30-3	NRB	NRB	NRB
89	1,2-difluorobenzene	367-11-3	NRB	NRB	not assigned

Sample ID	Name	CAS	Exp.class	consensus 1	consensus 2
90	N,N,N',N'-Tetrakis(oxiran-2-ylmethyl)-4,4'-methylenedianiline	28768-32-3	NRB	NRB	NRB
91	1,3,5-tris(epoxypropyl)triazinane-2,4,6-trione (synonym:1,3,5-tris(epoxypropyl)-1,3,5-triazine-2,4,6(1H,3H,5H)-trione)	2451-62-9	NRB	NRB	NRB
92	9-methoxy-7H-furo(3,2-g)chromen-7-one (synonym:9-methoxy-7H-furo(3,2-g)[1]benzopyran-7-one or methoxalen)	298-81-7	NRB	NRB	not assigned
93	3(or4)-methyl-4-cyclohexen-1,2-dicarboxylic anhydride	5333-84-6	NRB	NRB	not assigned
94	1,1,11-trihydroperfluoroundecanol	307-70-0	NRB	NRB	NRB
95	2-Ethylhexyl hydrogen (2-ethylhexyl)phosphonate	14802-03-0	NRB	NRB	NRB
96	dibutyltin oxide	818-08-6	NRB	NRB	not assigned
98	(2-chloroethyl)benzene	622-24-2	NRB	NRB	NRB
99	o-chlorotoluene	95-49-8	NRB	NRB	NRB
100	p-chlorotoluene	106-43-4	NRB	NRB	NRB
101	1-chloro-4-isopropenylbenzene	1712-70-5	NRB	NRB	NRB
102	N,N-diethylaniline	91-66-7	NRB	NRB	NRB
104	2,4,6-trichloroaniline	634-93-5	NRB	NRB	NRB
105	N-nitrosodiphenylamine	86-30-6	NRB	NRB	NRB
106	Dinonylphenol	1323-65-5	NRB	NRB	NRB
107	2,4-dinitrophenol	51-28-5	NRB	NRB	NRB
110	4-bromo-2,5-dichlorophenol	1940-42-7	NRB	NRB	NRB
111	dibromocresyl glycidyl ether	30171-80-3	NRB	NRB	NRB
112	1,4-bis(benzoyloxyimino)-2,5-cyclohexadiene	120-52-5	NRB	NRB	NRB
113	bis(alpha,alpha-dimethylbenzyl) peroxide	80-43-3	NRB	NRB	NRB
114	4-vinyl-1-cyclohexene	100-40-3	NRB	NRB	NRB
115	Menthol	1490-04-6	NRB	NRB	not assigned
116	tris(dimethylphenyl) phosphate	25155-23-1	NRB	NRB	NRB
118	4-(1-methyl-1-phenylethyl)phenol	599-64-4	NRB	NRB	NRB
119	1-chloronaphthalene	90-13-1	NRB	NRB	NRB
120	1-methoxynaphthalene	2216-69-5	NRB	NRB	not assigned
121	2-tert-butyl-9,10-anthraquinone	84-47-9	NRB	NRB	NRB
122	2-chloroanthraquinone	131-09-9	NRB	NRB	NRB
123	2-naphthalenethiol	91-60-1	NRB	NRB	NRB
124	Carbazole	86-74-8	NRB	NRB	NRB
125	diphenylmethyl 2-chloroethyl ether	32669-06-0	NRB	NRB	NRB
126	5-chloro-2-(2,4-dichlorophenoxy)phenol	3380-34-5	NRB	NRB	NRB
127	bis(1-methyl-2-chloroethyl) ether	108-60-1	NRB	NRB	NRB
129	N,N-diethyl-m-toluamide	134-62-3	NRB	NRB	NRB
130	7H-benzo[d,e]anthracen-7-one (synonym:benzanthrone)	82-05-3	NRB	NRB	NRB
131	12-dodecanelactam	947-04-6	NRB	RB	not assigned
132	Benzothiazole	95-16-9	NRB	NRB	NRB
133	Dichloropropane	78-87-5	NRB	NRB	NRB
134	3,3,4,4,5,5,6,6,7,7,8,8,9,9,10,10,10-heptadecafluorodecan-1-ol	678-39-7	NRB	NRB	NRB
135	1,1-dichloro-N-[(dimethylamino)sulfonyl]-1-fluoro-N-phenylmethanesulfenamide	1085-98-9	NRB	NRB	NRB

Sample ID	Name	CAS	Exp.class	consensus 1	consensus 2
136	Acenaphthylene	208-96-8	NRB	NRB	NRB
137	Chlorotriphenylmethane	76-83-5	NRB	NRB	NRB
138	Triethanolamine	102-71-6	NRB	RB	not assigned
139	ethyl carbamate	51-79-6	NRB	RB	RB
140	O,O-dimethyl S-(N-methylcarbamoylmethyl) dithio phosphate	60-51-5	NRB	NRB	NRB
141	N,N-bis(octylphenyl)amine	26603-23-6	NRB	NRB	NRB
142	p,p'-dioctyldiphenylamine	101-67-7	NRB	NRB	NRB
143	o-toluenesulfonamide	88-19-7	NRB	NRB	NRB
144	tri-p-cumenyl phosphate	26967-76-0	NRB	NRB	NRB
145	Benzenesulfonamide	98-10-2	NRB	NRB	NRB
146	5-methylbicyclo[2.2.1]hept-5-ene-2,3-dicarboxylic anhydride	25134-21-8	NRB	NRB	NRB
147	1-(2,5-dichloro-4-sulfophenyl)-3-methyl-5-pyrazolone	84-57-1	NRB	NRB	NRB
148	2-mercaptoimidazoline	96-45-7	NRB	NRB	NRB
149	pyridine-2,5-dicarboxylic acid	100-26-5	NRB	RB	RB
150	tetrahydro-1,4-oxazine	110-91-8	NRB	RB	RB
151	1,3,5-tris(2-hydroxyethyl)isocyanuric acid	839-90-7	NRB	NRB	NRB
152	4-anilino-3-nitrobenzenesulphonanilide	5124-25-4	NRB	NRB	NRB
153	2-isopropyl-6-methyl-4-pyrimidinol	2814-20-2	NRB	NRB	NRB
154	1,3-dichloropropene	542-75-6	NRB	NRB	not assigned
155	3-methoxypropylamine	5332-73-0	NRB	RB	RB
156	N,N-dimethylacrylamide	07/03/2680	NRB	RB	RB
157	3,3'-iminodipropaneonitrile	111-94-4	NRB	NRB	not assigned
158	tetramethylthiuram disulphide	137-26-8	NRB	NRB	NRB
159	tris(1,3-dichloro-2-propyl) phosphate	13674-87-8	NRB	NRB	NRB
160	1-chloro-2,3-epoxy-2-methylpropane	598-09-4	NRB	NRB	NRB
161	1,1',1'',1'''-(ethylenedinitrilo)tetrakis(propan-2-ol)	102-60-3	NRB	NRB	NRB
162	N-methylaniline	100-61-8	NRB	NRB	NRB
163	N-methylacetanilide	579-10-2	NRB	NRB	NRB
164	N,N'-diphenyl-p-phenylenediamine	74-31-7	NRB	NRB	NRB
165	N,N-dimethylbenzylamine	103-83-3	NRB	NRB	not assigned
166	1-tert-butyl-3,5-dimethyl-2,4,6-trinitrobenzene	81-15-2	NRB	NRB	NRB
168	2,4-dichlorophenyl 4'nitrophenyl ether	1836-75-5	NRB	NRB	NRB
170	2-nitro-4-methoxyaniline	96-96-8	NRB	NRB	NRB
172	bis[1-(tert-butylperoxy)-1-methylethyl]benzene	25155-25-3	NRB	NRB	NRB
173	2,6,6-trimethylbicyclo[3.1.1]heptyl-2-hydroperoxide	5405-84-5	NRB	NRB	NRB
174	3,3,5-trimethylcyclohexanone	873-94-9	NRB	NRB	NRB
175	Terphenyl	26140-60-3	NRB	NRB	NRB
176	1-methylnaphthalene	90-12-0	NRB	NRB	NRB
177	4,4'-methylenediphenol	620-92-8	NRB	NRB	NRB
178	2-[4-(diethylamino)-2-hydroxybenzoyl]benzoic acid	5809-23-4	NRB	NRB	not assigned
179	isobutyl 2-naphthyl ether	2173-57-1	NRB	NRB	NRB
180	2-Aminonaphthalene-1,5-disulfonic acid	117-62-4	NRB	NRB	NRB
181	decahydronaphthalene(mixture of cis-form and trans-form)	91-17-8	NRB	NRB	NRB

Sample ID	Name	CAS	Exp.class	consensus 1	consensus 2
182	(Tricyclo[5.2.1.0(2,6)]decane-4,8-diyl)dimethanol	26896-48-0	NRB	NRB	NRB
183	Anthracene	120-12-7	NRB	NRB	NRB
184	2-aminoanthraquinone	117-79-3	NRB	NRB	NRB
185	1,4-Bis(isopropylamino)-9,10-anthraquinone	14233-37-5	NRB	NRB	NRB
186	1H-1,2,3-benzotriazole	95-14-7	NRB	NRB	NRB
187	5,5-diphenylImidazolidine-2,4-dione (synonym:5,5-diphenyl-2,4-Imidazolidinedione)	57-41-0	NRB	NRB	NRB
188	Thioacetamide	62-55-5	NRB	NRB	not assigned
189	1,1,2,2-tetrabromoethane	79-27-6	NRB	NRB	NRB
190	2,2-dichloro-1,1,1-trifluoroethane	306-83-2	NRB	NRB	NRB
191	3,4-dichloro-1-butene	760-23-6	NRB	NRB	NRB
192	perfluoro(tributylamine)	311-89-7	NRB	NRB	NRB
193	2,2'-dichlorodiethyl ether	111-44-4	NRB	NRB	not assigned
194	2-(isopropoxy)ethanol	109-59-1	NRB	RB	RB
195	3,5,5-trimethylhexanal	5435-64-3	NRB	NRB	NRB
196	Trichloroacetaldehyde	75-87-6	NRB	NRB	NRB
197	Docosanamide	3061-75-4	NRB	RB	not assigned
198	dimethyl phosphonate	868-85-9	NRB	RB	RB
199	tri-n-pentyl phosphate	2528-38-3	NRB	NRB	NRB
200	Perfluorooctanoic acid	335-67-1	NRB	NRB	NRB
202	Pentabromotoluene	87-83-2	NRB	NRB	NRB
204	4'-aminoacetanilide	122-80-5	NRB	NRB	NRB
207	6-tert-butyl-2,4-xylenol	1879-09-0	NRB	NRB	NRB
209	4-(methylthio)phenol	1073-72-9	NRB	NRB	not assigned
210	2-methyl-3-(4-tert-butylphenyl)propionaldehyde	80-54-6	NRB	NRB	NRB
211	4,6-dinitro-o-cresol	534-52-1	NRB	NRB	NRB
212	O,O-diethyl-o-(alpha-cyanobenzylideneamino)thio phosphate	14816-18-3	NRB	NRB	NRB
213	dimethyl 2,6-naphthalenedicarboxylate	840-65-3	NRB	RB	not assigned
214	2,2,6,6-Tetramethylpiperidin-4-on	826-36-8	NRB	NRB	NRB
215	2,2',2''-(2,4,6-trioxo-1,3,5-triazinane-1,3,5-triyl)triethyl triacrylate	40220-08-4	NRB	NRB	NRB
216	2-{N-(2-cyanoethyl)-N-[4-(4-nitrophenylazo)phenyl]amino}ethyl benzoate	40690-89-9	NRB	NRB	NRB
217	chlorophthalocyaninatocopper(II)(synonym:pygmentblue-15)	12239-87-1	NRB	NRB	NRB
218	polychlorobiphenyl(number of chlorine is 2-10)	25512-42-9	NRB	NRB	NRB

General conclusions and future prospects

It is crucial to know the limitations of QSAR models for reliable predictions before they can be applied on a diverse set of test molecules. The predictive ability of QSARs is restricted in their structural and response domain which indicates that only those test samples that are structurally similar to the training set can be given as input to such trained models. With growing awareness about the use of QSARs, more sophisticated algorithms have been proposed from time to time. Availability of such state-of-the-art approaches has allowed QSAR modellers to overcome several prevailing issues in efficient and faster ways. In theory, a QSAR model can be developed based on one of the several available model development algorithms, however its applicability is always restricted since a limited pool of structural diversity is taken into account while developing such predictive models. Thus, addressing the AD of such powerful yet restrictive models can be a useful way to guide the users and keeping them from making predictions which could be unreliable due to extrapolation.

Several classical ways of addressing the AD of QSAR models were introduced and an attempt to better explore their features was made considering simulated models as well as published models from the literature. All of these classical approaches were able to partially overcome some of the prevailing issues in defining the model's AD but simultaneously, were associated with several other drawbacks. Whether it comes to inefficiency with data complexity or issues in defining an interpolation space sufficiently restricting it to reliable predictions, all the approaches had their own salient features accompanied by some disappointing drawbacks most of the times.

The range-based approaches may be the simplest, however PCA bounding box was associated with the most positive impact on model statistics in case of CAESAR model A by excluding just two test molecules outside the model's AD. On the other hand, advanced kernel approaches like optimized gaussian kernel considered a giant portion of the test set to be excluded from the model's AD, in some cases without any noticeable impact on the model statistics. Thus, being simple or advanced may not restrict the application of such approaches.

Two novel approaches were introduced and their underlying algorithms were discussed. One of them defined the interpolation space relying heavily on an opted *k* value while the other applied the salient features of Locally-centred Mahalanobis distances to identify test samples beyond the scope of a model. Both the approaches were quite diverse but their results converged in several cases with each other as well as with those derived for other classical approaches, indicating the presence of several consensus test samples to be excluded from the model's AD. Both the discussed approaches were quite efficient even in higher dimensions, the defined interpolation space was reasonably restricted and the excluded test samples in several cases were associated with higher absolute standardized errors indicating that the results derived in the model's descriptor space can converge to the observations made in the model's response domain.

There is still a lot to explore within and beyond the scope of this thesis work. Development of efficient AD strategies to deal with consensus models could be one of them. Such models are interesting since they don't exist on their own and their predictions are completely reliable on the output of several other models. In the case of biodegradability models, the resulting error rate reduced reasonably implementing consensus models. Usually, with classical and new AD approaches dealt in this thesis, the final output simply indicated if the test sample is inside or outside the model's AD. There can be several test samples that may be structurally similar but not to a sufficient extent. The predictions derived for such test samples may not be completely meaningless. So to deal with such issues, in future some approaches could be developed quantifying the reliability in predictions rather than simply deciding to include or exclude a prediction. It could be also interesting to see if combining the AD output from several approaches could help overcoming their prevailing drawbacks and allow a better reliability in AD assessment.

List of Figures

List of Tables

Bibliography

1. REACH. European Community Regulation on chemicals and their safe use. Available online: http://ec.europa.eu/environment/chemicals/reach/reach_ intro.htm

2. REACH in brief-European Commission-Europa. Available online: http://ec.europa.eu/environment/chemicals/reach/pdf/2007_02_reach_in_brief.p df

3. Chapter R.6: QSARs and grouping of chemicals –ECHA. Available online: http://echa.europa.eu/documents/10162/13632/information_requirements_r6_en .pdf

4. Tropsha A. Best Practices for QSAR Model Development, Validation, and Exploitation. *Molecular Informatics*, **2010**, *29*, 476-488.

5. Sahigara, F.; Mansouri, K.; Ballabio, D.; Mauri, A.; Consonni, V.; Todeschini, R. Comparison of Different Approaches to Define the Applicability Domain of QSAR Models. *Molecules* **2012**, *17*, 4791-4810.

6. Dimitrov, S.; Dimitrova, G.; Pavlov, T.; Dimitrova, N.; Patlewicz, G.; Niemela, J.; Mekenyan, O.A. Stepwise approach for defining the applicability domain of SAR and QSAR models. *J. Chem. Inf. Model.* **2005**, *45*, 839–849.

7. Worth, A.P.; van Leeuwen, C.J.; Hartung, T. The prospects for using (Q)SARs in a changing political environment: high expectations and a key role for the European Commission’s Joint Research Centre. *SAR QSAR Environ. Res.* **2004**, *15*, 331–343.

8. Nikolova-Jeliazkova, N.; Jaworska, J. An approach to determining applicability domains for QSAR group contribution models: an analysis of SRC KOWWIN. *Altern. Lab. Anim.* **2005**, *33*, 461–470.

9. Sheridan, R.; Feuston, R.P.; Maiorov, V.N.; Kearsley, S. Similarity to molecules in the training set is a good discriminator for prediction accuracy in QSAR. *J. Chem. Inf. Comp. Sci.* **2004**, *44*, 1912–1928.

10. OECD. Quantitative Structure-Activity Relationships Project [(Q)SARs]. Available online: http://www.oecd.org/document/23/0,3746,en_2649_34377_ 33957015_1_1_1_1,00.html

11. Netzeva, T.I.; Worth, A.; Aldenberg, T.; Benigni, R.; Cronin, M.T.D.; Gramatica, P.; Jaworska, J.S.; Kahn, S.; Klopman, G.; Marchant, C.A.; *et al.* Current status of methods for defining the applicability domain of (quantitative) structure-activity relationships. The report and recommendations of ECVAM Workshop 52. *Altern. Lab. Anim.* **2005**, *33*, 155–173.

12. Jaworska, J.; Nikolova-Jeliazkova, N.; Aldenberg, T. QSAR Applicabilty domain estimation by projection of the training set descriptor space: A review. *Altern. Lab. Anim.* **2005**, *33*, 445–459.

13. Worth, A.P.; Bassan, A.; Gallegos, A.; Netzeva, T.I.; Patlewicz, G.; Pavan, M.; Tsakovska, I.; Vracko, M. The Characterisation of (Quantitative) Structure-Activity Relationships: Preliminary Guidance. ECB Report EUR 21866 EN, European Commission, Joint Research Centre; Ispra, Italy, **2005**; p. 95.

14. Sahigara, F.; Ballabio, D; Todeschini, R.; Consonni, V. Defining a novel k-nearest neighbours approach to assess the applicability domain of a QSAR model for reliable predictions. *Journal of Cheminformatics* **2013**, **5**:27. doi:10.1186/1758-2946-5-27.

15. Wold, S.; Esbensen, K.; Geladi, P. Principal component analysis. *Chemometr. Intell. Lab.* **1987**, *2*, 37–52.

16. Preparata, F.P.; Shamos, M.I. Convex hulls: Basic Algorithms. In *Computational Geometry: An Introduction*; Preparata, F.P., Shamos, M.I., Eds.; Springer-Verlag: New York, NY, USA, **1991**; pp. 95–148.

17. Todeschini, R.; Consonni, V. Molecular Descriptors for Chemoinformatics. Wiley–VCH, Weinheim, **2009**.

18. Tropsha, A.; Gramatica, P.; Gombar, V. The importance of being earnest: validation is the absolute essential for successful application and interpretation of QSPR Models. *QSAR & Comb. Sci.* **2003**, *22*, 69–77.

19. Tetko, I.V.; Sushko, I.; Pandey, A.K.; Zhu, H.; Tropsha, A.; Papa, E.; Oberg, T.; Todeschini, R.; Fourches, D.; Varnek, A. Critical assessment of QSAR models of environmental toxicity against Tetrahymena pyriformis: Focusing on applicability domain and overfitting by variable selection. *J. Chem. Inf. Model.* **2008**, *48,* 1733–1746.

20. Silverman, B.W. Density Estimation for Statistics and Data Analysis. London, UK:Chapman and Hall; **1986**.

21. Forina, M; Lanteri, S; Armanino, C; Cerrato, Oliveros C; Casolino, C. V-PARVUS software,User manual. **2004.** http:/parvus.unigr.it.

22. Asikainen,A.H.; Ruuskanen,J.; Tuppurainen,K.A. Consensus kNN QSAR: a versatile method for predicting the estrogenic activity of organic compounds in silico. A comparative study with five estrogen receptors and a large, diverse set of ligands. *Environ Sci Technol*, **2004**, *38*, 6724-6729.

23. Cedeño, W.; Agrafiotis ,D.K. Using particle swarms for the development of QSAR models based on K-nearest neighbor and kernel regression. *J Comput Aided Mol Des*, **2003**, *17*,255-263.

24. Golbraikh, A.; Shen, M.; Xiao, Z.; Xiao, Y.D.; Lee, K.H.; Tropsha, A. Rational selection of training and test sets for the development of validated QSAR models. *J Comput Aided Mol Des*, **2003**,*17*,241-253.

25. Itskowitz, P.; Tropsha, A. k Nearest Neighbors QSAR Modeling as a Variational Problem: Theory and Applications. *J Chem Inf Model*, **2005**, *45*,777-785.

26. Nigsch, F.; Bender, A.; Van Buuren B.; Tissen, J.; Nigsch, E.; Mitchell, J.B. Melting point prediction employing k-nearest neighbor algorithms and genetic parameter optimization. *J Chem Inf Model*, **2006**, 46, 2412-2422.

27. Breiman, L.; Meisel, W.; Purcell, E. Variable kernel estimates of multivariate densities. *Technometrics*, **1977**,*19*, 135–144.

28. Box plot – MATLAB. Available online: http://www.mathworks.it/it/help/stats/boxplot.html

29. Rousseeuw,P.J.; Van Zomeren, B.C. Unmasking Multivariate Outliers and Leverage Points. *J. Amer. Statist. Assoc.* **1990**, *85*, 633-639.

30. Campbell, N.A. The Influence Function as an Aid in Outlier Detection in Discriminant Analysis. *Appl. Statist.* **1978**, *27*, 251-258.

31. Angiulli, F.; Pizzuti, C. Fast Outlier Detection in High Dimensional Spaces. Principles of Data Mining and Knowledge Discovery. Springer Berlin/ Heidelberg, **2002**. 43-78.

32. Pell, R.J. Multiple outlier detection for multivariate calibration using robust statistical techniques. *Chemom. Intell. Lab. Syst.* **2000**, *52*, 87–104.

33. Walczak, B.; Massart, D.L. Multiple outlier detection revisited. *Chemom. Intell. Lab. Syst.* **1998**, *41*, 1–15.

34. Peña, D.; Prieto, F.J. Multivariate Outlier Detection and Robust Covariance Matrix Estimation. *Technometrics*. 2001, *43*, 286-310.

35. Todeschini, R; Ballabio, D; Consonni, V; Sahigara, F; Filzmoser, P. Locally-centred Mahalanobis distance: a new distance measure with salient features towards outlier detection. *Anal Chim Acta* **2013**. doi:10.1016/j.aca.2013.04.034.

36. Zhao, C.; Boriani, E.; Chana, A.; Roncaglioni, A.; Benfenati, E. A new hybrid QSAR model for predicting bioconcentration factor (BCF). *Chemosphere* **2008**, *73*, 1701–1707.

37. Lombardo, A.; Roncaglioni, A.; Boriani, E.; Milan, C.; Benfenati, E. Assessment and validation of the CAESAR predictive model for bioconcentration factor (BCF) in fish. *Chem. Cent. J.* **2010**,*4* (Suppl 1), doi:10.1186/1752-153X-4-S1-S1.

38. BCF-CAESAR. Available online: http://www.caesar-project.eu/index.php?page=results&section=endpoint&ne=1

39. Mansouri, K; Ringsted,T; Ballabio,D; Todeschini,R; Consonni,V. Quantitative Structure–Activity Relationship Models for Ready Biodegradability of Chemicals. *J. Chem. Inf. Model.* **2013**, *53*, 867–878.

40. Consonni, V.; Ballabio, D.; Todeschini, R. Comments on the definition of the Q2 parameter for QSAR validation. *J. Chem. Inf. Model.* **2009**, *49*, 1669–1678.

41. Consonni, V.; Ballabio, D.; Todeschini, R. Evaluation of model predictive ability by external validation techniques. *J. Chemometr.* **2010**, *24*, 104–201.

42. Wan, C.; Harrington, P.B. Self-configuring radial basis function neural networks for chemical pattern recognition. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 1049–1056.

43. DRAGON (Software for Molecular Descriptor Calculations), ver. 6. Talete srl, Milano, Italy. Available online: http://www.talete.mi.it

List of papers published during the PhD Student period

Period: 07/2010 – 06/2013

1. Sahigara, F.; Mansouri, K.; Ballabio, D.; Mauri, A.; Consonni, V.; Todeschini, R. Comparison of Different Approaches to Define the Applicability Domain of QSAR Models. *Molecules* **2012**, *17*, 4791-4810.

2. Todeschini, R; Ballabio, D; Consonni, V; Sahigara, F; Filzmoser, P. Locally-centred Mahalanobis distance: a new distance measure with salient features towards outlier detection. *Analytica Chimica Acta* **2013**. doi:10.1016/j.aca.2013.04.034.

3. Sahigara, F.; Ballabio, D; Todeschini, R.; Consonni, V. Defining a novel k-nearest neighbours approach to assess the applicability domain of a QSAR model for reliable predictions. *Journal of Cheminformatics* **2013**, *5*:27. doi:10.1186/1758-2946-5-27.

The research leading to these results has received funding from the European Community's Seventh Framework Programme [FP7/2007–2013] under Grant Agreement n. 238701 of Marie Curie ITN Environmental Chemoinformatics (ECO) project.