Extending *K*-Means Clustering for Analysis of Quantitative Structure Activity Relationships (QSAR)

Robert William Stanforth

School of Computer Science and Information Systems

Birkbeck College, University of London

Submitted for the degree of Doctor of Philosophy
of the University of London

January 2008

I hereby declare that the work presented in this thesis is my own, and that it has not previously been submitted for a degree or award at this or any other academic institution.

Signed:

Robert Stanforth

January 2008

Abstract

A Quantitative Structure-Activity Relationship (QSAR) study is an attempt to model some biological activity over a collection of chemical compounds in terms of their structural properties. A QSAR model may be constructed through (typically linear) multivariate regression analysis of the biological activity data against a number of features or 'descriptors' of chemical structure. As with any regression model, there are a number of issues emerging in real applications, including (a) *domain of applicability* of the model, (b) validation of the model within its domain of applicability, and (c) possible non-linearity of the QSAR. Unfortunately the existing methods commonly used in QSAR for overcoming these issues all suffer from problems such as computational inefficiency and poor treatment of non-linearity. In practice this often results in the omission of proper analysis of them altogether.

In this thesis we develop methods for tackling the issues listed above using *K*-means clustering. Specifically, we model the shape of a dataset in terms of intelligent *K*-means clustering results and use this to develop a non-parametric estimate for the domain of applicability of a QSAR model. Next we propose a 'hybrid' variant of *K*-means, incorporating a regression-wise element, which engenders a technique for non-linear QSAR modelling. Finally we demonstrate how to partition a dataset into training and testing subsets, using the *K*-means clustering to ensure that the partitioning respects the overall distribution. Our experiments involving real QSAR data confirm the effectiveness of the methods developed in the project.

Acknowledgements

I am very pleased to be able to acknowledge the contributions made by all those who have assisted and supported me in my research.

First and foremost, I should like to thank my supervisor, Prof. Boris Mirkin, for introducing me to the world of clustering. Over the course of my research he has provided me with a steady stream of his insights into data visualisation and manipulation, and advice into research methodology in general. These ideas have frequently struck a chord with me, allowing my own ideas to develop and culminating in this thesis.

I also offer my sincere gratitude to my co-supervisor and former manager, Dr. Evgueni Kolossov, not only for originally suggesting this topic of research to me, but also for his continual support and encouragement while we worked together in QSAR. Without his investment and confidence in me, this research would not have been possible.

Finally, I am extremely thankful to my employer and sponsor, ID Business Solutions Limited. The Company has unfailingly supported my studies, providing me with all the time and resources I have needed to keep my research on track. It has also furnished me with the immensely rewarding experience of making industrial application of the methods developed in this thesis.

Contents

Tables and Figures

1. Introduction

A Quantitative Structure-Activity Relationship (QSAR) is a study of the dependence upon chemical structure of some observable property or 'activity' over a collection of chemical compounds. Modelling this dependence enables predictions to be made about the activity of previously unseen chemical compounds.

Biological activities that have been studied in a QSAR context range from toxicological effects upon an organism [Benigni 2003, Aptula *et al* 2005, Toropov *et al* 2007] to inhibitory effects on the biochemical activity of certain enzymes [Senese & Hopfinger 2003, Zernov *et al* 2003, Samanta *et al* 2006]. Physical properties of chemical compounds have also been studied, in particular aqueous solubility [Butina & Gola 2003, Yan & Gasteiger 2003], the acid dissociation coefficient pK_a [Xing *et al* 2003], and the octanol partition coefficient log*P* [Mannhold & van de Waterbeemd 2001, Roy *et al* 2007].

A compound's chemical structure refers to the physical constitution of a molecule of the compound. This molecular structure is represented, in the first instance, by its *molecular graph* – a graph in the mathematical sense comprising a collection of vertices (denoting atoms in the molecule) connected by edges (chemical bonds between the atoms), capturing the topological structure of the molecule.

In a QSAR study, modelling the dependence of activity upon chemical structure typically involves some form of regression analysis. A 'training collection' of chemical compounds (whose activity values are known) is the principal input into the modelling process, which proceeds by extracting its trends in the relationship between chemical structure and activity. The aspiration is for these trends to generalise to other chemical compounds beyond those occurring in the training set, thereby allowing predictions to be made about the activity of new (as yet unmeasured) chemical compounds based solely on knowledge of their chemical structure.

This QSAR modelling approach is underpinned by the so-called 'Fundamental Assumption of QSAR', that chemical compounds with similar chemical structures will have similar activities [McKinney *et al* 2000]. This assumption is a prerequisite both for the meaningful description of trends within the training set, and for the interpolation of those trends to encompass other compounds.

A precise application of this Fundamental Assumption – and indeed any consideration of a 'trend' involving chemical structures – requires that the notion of similarity of chemical structures be made precise. Formulating this measure of structural similarity is itself intimately bound with how the chemical structure is numerically represented from a point of view of isolating the trends. Although there do exist structural similarity measures that operate directly on the molecular graph, this thesis will be concerned only with *descriptor-based* QSAR, in which chemical structure is further represented by a collection of *chemical descriptors* – quantitative features calculated in terms of the molecular graph [Diudea 2000]. For the purposes of the ensuing regression analysis, chemical structure is thereby characterised as a point in *chemical descriptor space* – the linear feature space spanned by a collection of descriptors suitably chosen for the QSAR dataset under consideration.

The necessity of the Fundamental Assumption of QSAR for regression analysis to be valid leads to a related observation: a model based on regression analysis can only be expected to make valid predictions for chemical compounds whose structure is similar to some of those in the training set. (This is precisely the well-known problem of *extrapolation* in regression.) For example, if a QSAR model is trained using only chemical compounds from one homologous series (say, the saturated alkanes), then one would have no grounds for confidence in predictions it makes about compounds *not* in that group (unsaturated compounds, for example).

Given this lack of global applicability of a QSAR model, one is led to ask: for which chemical structures does the model apply, without having to extrapolate away from its training data? The region of chemical structure space, within which a QSAR model applies is called the *domain of applicability* of the model [Jaworska *et al* 2005]. In the context of a QSAR model based on regression analysis of a training set, this domain of applicability comprises those regions of chemical structure space that are adequately represented by training compounds of similar structure, such that the regression model can make its prediction by interpolation rather than extrapolation.

The domain of applicability must be considered a crucial and integral part of any QSAR model, for a model cannot be used with confidence without knowledge of whether its prior conditions for use are being met. Unfortunately, this aspect of QSAR modelling has often been overlooked, prompting specific attention to the matter from

regulatory authorities responsible for acceptance of QSARs for environmental and medical applications [Jaworska *et al* 2003, Gramatica 2007].

Although *necessary* for regression analysis to build predictive models, the Fundamental Assumption of QSAR is not in itself *sufficient* for this purpose, because such modelling also relies on the recovery of genuine trends from the training data. An overzealous attempt by the regression algorithm to find trends in the training data runs the risks of mistaking any artefact in the data for a trend. This situation, in which the model describes the noise rather than the genuine underlying trends, is called *overfitting*, while *parsimony* denotes the absence of overfitting [Hawkins 2004]. The predictive power of an overfitted model is compromised because prediction will attempt to interpolate fictitious trends, based only on noise in the data, that are not borne out by any underlying structure-activity relationship.

Notwithstanding some well-known heuristics (for example, the number parameters in the model approaching or exceeding the number of training entities), it is not straightforward to determine reliably *a priori* whether a regression analysis will result in an overfitted model. Following training, the model must therefore be validated to ascertain its parsimony [Tropsha *et al* 2003].

The most pragmatic test of a regression model's parsimony rests with its predictive power. To this end, it is considered good practice to perform *external validation* on a newly trained QSAR model. This consists of applying the model to an *external test set* of compounds whose activity values are known, and checking how well the predictions agree with these known values [Gramatica 2007].

There are a number of reasons, however, why external validation may not be appropriate. There may simply be no external test set available. On the other hand, an external test set may have been obtained at considerable expense, in which case there may be a stronger case for allowing it to contribute directly to the model by including its contents in the training set, rather than sidelining it to the validation phase. Yet another contraindication may be that an external test set has been provided, but (wholly or partially) lies outside the model's domain of applicability, and so would not constitute a fair test of the model's predictive power within its domain.

As an alternative to external validation, and in response to the aforementioned limitations on its use, we shall in this thesis investigate the validation of a QSAR model using an *internal test set*. In lieu of the external provision of a test set completely separate from the training data, internal test set validation involves separating all the available structure/activity data into two complementary subsets – training and test set – at the outset of the modelling, taking care that each subset remain representative of the original whole. The test set is 'held out' while the regression is being performed on the remainder of the data, and is reintroduced later for the validation phase.

A further active area of research in QSAR is the development of nonlinear modelling techniques. Traditionally descriptor-based QSAR has seen a prevalence of linear modelling through multivariate linear least-squares regression [Wold *et al* 2000] and related techniques such as "partial least squares" (PLS) [Xing *et al* 2003].

This prevalence of linear modelling methods is presumably due to their comparative simplicity, both from the point of view of implementation and in the form of the linear models they create. Indeed, the transparency of a linear model can allow it to admit a mechanistic interpretation: the linear model parameters amount to the contributions of each chemical descriptor to the activity. This is in contrast to a neural network model (for example), which can only be viewed as a 'black box' without offering insight into how an especially large or small activity may have arisen.

However, there have been many QSAR studies in which the structure-activity relationship cannot be adequately described by a linear model, and one must look beyond the well-understood linear techniques.

We have so far identified three open problems in QSAR: defining a QSAR model's *domain of applicability*, extracting an *internal test set*, and modelling *non-linear* dependence upon chemical descriptors. A common thread emerges to unite these problems: each requires a way of describing the shape of a QSAR dataset without making prior assumptions on the form that the shape may take. In the case of the *domain of applicability*, we require a description of the dataset's distribution in chemical descriptor space. In practice, QSAR datasets are not distributed in descriptor space according to simple distributions but tend to form rather irregular shapes, and so we shall seek a *non-parametric* representation of the shape. Furthermore, extraction of

an *internal test set* must reflect the distribution of the available data; the extraction will therefore also be guided by this description of the distribution. Finally, when modelling a *non-linear* structure-activity relationship, we shall once again seek a non-parametric representation, but this time incorporating the activity into the description.

In order to address the three problems described above, we shall investigate how *K*-means clustering may be employed in the description of a QSAR dataset. We shall use the clustering to partition a large, diverse dataset into constituents that individually are sufficiently small and cohesive to be described by a simple parametric representation. Collectively, however, they will assume the flexibility to model a complex dataset.

Working in chemical descriptor space for the *domain of applicability* investigation, each cluster will receive a simple spherical description, leading to the representation of a dataset's domain of applicability as an amalgamation of hyperspheres.

In order to model *non-linearity* in QSAR, we shall extend this segmented approach to incorporate activity values. Within each cluster we shall construct a separate simple linear model for the dependence of activity on the chemical descriptors. This will lead to the synthesis of a complete model with locally linear regions. Furthermore, we shall investigate how the activity variation in the training data can influence the location of the clusters, promoting their alignment with such local regions of linearity in the data. To this end we shall derive a new 'hybrid' variant of the *K*-means least-squares clustering criterion that incorporates a 'regression-wise' contribution alongside the conventional distance-wise formulation.

Finally, the *internal test set* investigation will revisit the cluster description of the dataset's shape in descriptor space, extracting the internal test set evenly amongst the clusters in order to preserve the overall data distribution.

For each of the three problems under investigation, the new methods developed according to this cluster-based paradigm will be experimentally validated using publicly available QSAR datasets. Thereby, we shall not only assess their suitability on their own merits, but also determine whether they offer any improvements over the existing approaches to tackling these problems.

The thesis will be structured as follows. In chapter 2 we shall conduct a review of the existing approaches to tackling the problems of domain of applicability, internal test

set extraction, and modelling non-linearity in QSAR. Chapter 3 will introduce *K*-means clustering, and will develop the extensions and generalisations of *K*-means that will subsequently be employed in our own investigations into these QSAR problems. Chapter 4 will develop a measure of distance to domain of applicability, and analyse the efficacy of the model of domain of applicability that it induces. Chapter 5 will be concerned with the 'hybrid' variant of *K*-means clustering and its application to modelling non-linear relationships. In chapter 6, we shall investigate the use of *K*-means clustering to guide the extraction of an internal test set, and analyse to what extent it can provide an even, representative sampling without detriment to the model training process. Chapter 7 will then conclude the thesis by pursuing a discussion of the methods developed in the course of this investigation, identifying their interrelationships and common threads, and identifying scope for further research.

2. Current Issues in QSAR and Existing Methods to Tackle Them

2.1. Overview of QSAR

Quantitative Structure-Activity Relationships (QSAR) are attempts to capture relationships between chemical structure and some observable 'activity' over a collection of chemical compounds, with a view to using models of these relationships to predict the activity of new compounds. The activity may be biological, for example toxicity [Aptula *et al* 2005] or carcinogenicity [Crettaz & Benigni 2005], or may be a physiochemical property, such as aqueous solubility [Yan & Gasteiger 2003, Lind & Maltseva 2003, Butina & Gola 2003] or the so-called 'octanol partition coefficent' log*P* [Ghose & Crippen 1986, Mannhold & van de Waterbeemd 2001, Roy *et al* 2007].

QSAR models are built according to an inductive machine learning formulation. The modelling process begins with a 'training set' of chemical compounds whose chemical structure and biological activities are known, and proceeds by performing some sort of regression analysis to construct a predictive model of activity as a function of structure.

QSAR studies are therefore underpinned by the assumption that chemical compounds with similar structures have similar activity values; we shall refer to this as the Fundamental Assumption of QSAR [McKinney *et al* 2000]. Quite how this Fundamental Assumption of QSAR is applied, however, depends on precisely how we judge what constitutes similarity of chemical structures.

Indeed, one of the first decisions to be taken in a QSAR study is how to characterise the chemical structures quantitatively, to render them in a form suitable for regression analysis [Wold *et al* 2000]. This thesis will be concerned with *descriptor-based* QSAR models, in which chemical structure is measured using a number of chemical 'descriptors' – quantities either calculated based on the topological structure of the chemical compound [Todeschini & Consonni 2002, Estrada & Uriarte 2001] or experimentally determined physical or chemical properties.

In descriptor-based QSAR, each compound's chemical structure is represented by its values of each of the descriptors in use. The modelling can then proceed by using

multivariate regression on the *descriptor space* – the linear feature space obtained by using the chemical descriptors as the features. Typically multivariate linear least-squares regression is used (see for example [Aptula *et al* 2005, Huuskonen 2000, Vanyúr *et al* 2003]), although other linear and non-linear techniques such as support vector regression [Vapnik 1995] are also frequently used [Lind & Maltseva 2003, Zernov et al 2003].

2.2. Overview of the Domain of Applicability Problem

As with any regression, a QSAR model cannot be expected to extrapolate well: it cannot be expected to give a reliable prediction for a chemical compound dissimilar in structure to those in the original dataset set that was used to train the model.

The term 'domain of applicability' of a QSAR model denotes the region of chemical structure space on which the model can in principle be expected to give reliable predictions. In light of the inductive machine learning formulation of QSAR models, the domain of applicability may be taken to be the region of chemical space that is adequately represented by similar chemical structures in the training set – in other words, regions within which predictions will not suffer from the extrapolation problem.

In order to be of practical use, it must be possible to determine whether or not a given chemical structure is inside or outside the domain of applicability [Gramatica 2007]. In the latter case, it is also desirable to be able to calculate how far outside the domain the structure is: the reliability of prediction will be only slightly impaired just outside the domain, while the prediction error is expected to worsen steadily as the distance from the domain increases. Indeed, different applications will have different tolerances for what constitutes adequate representation by chemical structures in the training set [Sheridan *et al* 2004].

There are several existing methods in use for approximating a QSAR model's domain of applicability, and for measuring the distance to that domain [Jaworska *et al* 2005, Stanforth *et al* 2005, Stanforth *et al* 2007a]. Those methods include the bounding box and convex hull models of dataset shape; the Euclidean, Mahalanobis, and city-block (Manhatten) distances; and the non-parametric methods of k nearest neighbours and probability density estimation by Parzen's window. These will be described in turn.

2.2.1. Bounding Box

The bounding box method [Sheridan *et al* 2004, Jaworska *et al* 2005] is conceptually

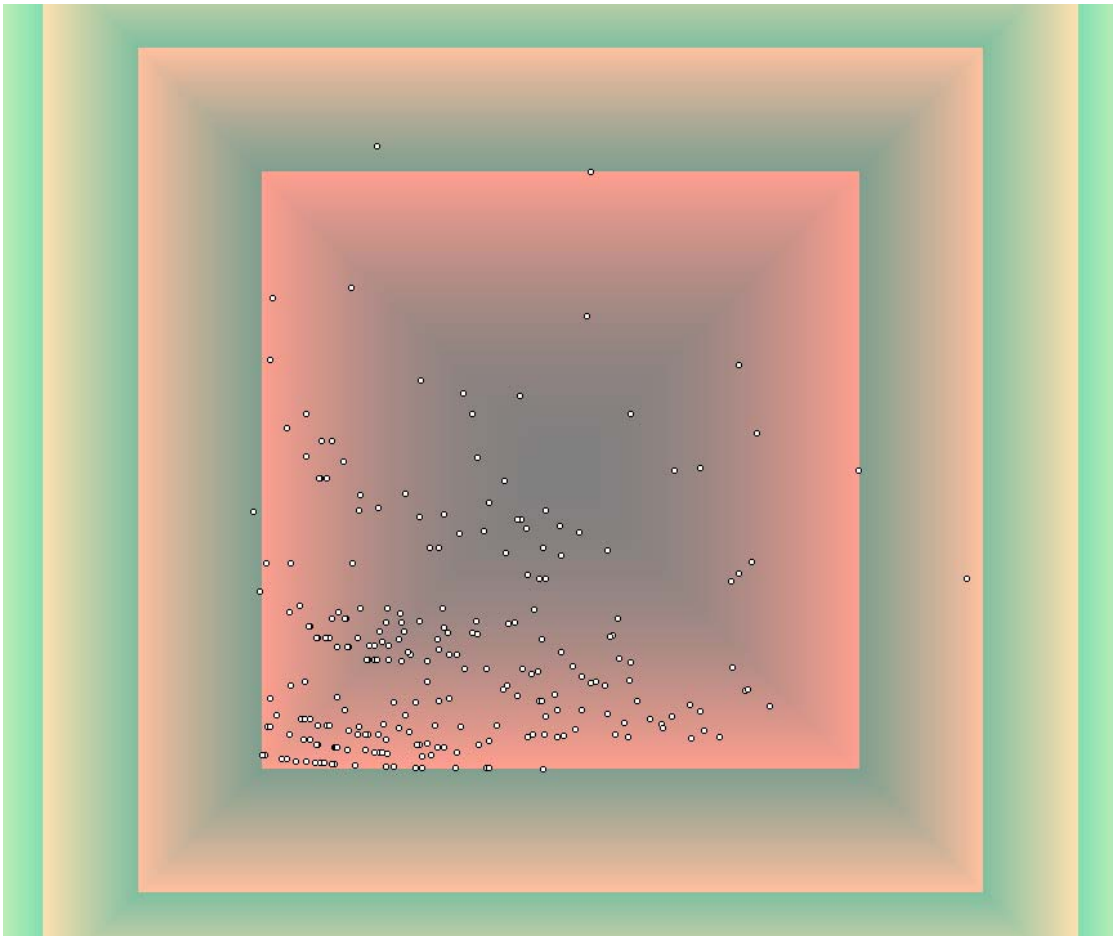

Figure 2.1: Contour Plot of the Distance to Bounding Box

The 258 points plotted here relate to chemical compounds originating from Huuskonen's dataset for investigating aqueous solubility [Huuskonen 2000]. For or ease of visualisation, the chemical structures are plotted using only two descriptors: molecular weight rendered horizontally, and Todeschini's Hydrophilicity index H_y [Todeschini & Consonni 2002] rendered vertically.

The innermost contour in the above diagram, surrounding the central red region, portrays the bounding box. It is constructed in alignment with the descriptor axes, as the region of points for which *each* descriptor value lies within the 5th to 95th percentile interval for that descriptor over the whole dataset. In other words, the bounding box is the region in which both descriptors have absolute value less than unity, assuming that each descriptor has been normalised (rescaled) to take −1 as its 5th percentile and +1 as its 95th percentile (over the dataset). Successive heavy contours correspond to the regions in which both normalised descriptors have squared absolute value less than 2, 3, etc.

the simplest, and computationally the fastest, of all the approaches considered here. The domain of applicability is taken to be the smallest axis-aligned hypercuboid in descriptor space that contains the whole training set. In other words, a chemical structure is deemed to be inside the domain if and only if, for each descriptor, the descriptor value for that structure is in the range of values taken by that descriptor over the whole training set. Alternatively, in order to overcome sensitivity to outliers, each descriptor range may be based on a quantile range of that descriptor over the training set (instead of the entire range of the descriptor over the training set); see Figure 2.1.

The bounding box method ensures that there is no extrapolation with respect to any individual descriptor. However, modelling the training set as a rectangular box is too crude to avoid extrapolation in multivariate descriptor space. In the absence of careful experimental design to ensure statistical independence of the descriptors, using the bounding box method to estimate domain of applicability will typically result in substantial regions of false positives in which chemical structures are erroneously deemed to be inside the domain even if they differ from it in essential features.

If principal components analysis (PCA) is applied as a preprocessing step, then the results of the bounding box method can be improved [Jaworska *et al* 2005]. The resulting principal components will be uncorrelated with one another over the training set and, moreover, in the (admittedly unlikely in practice) scenario that the training data is drawn from a multivariate normal distribution in descriptor space then the principal components will satisfy the stronger condition of statistical independence assumed by the bounding-box method. However, this uncorrelated nature of the principal components (as directions in descriptor space) is questionable as a sufficient assumption for applying the bounding-box method: PCA implicitly aggregates all descriptors into a single descriptor space, and therefore the domain of applicability should logically be expressed isotropically as a coherent region of descriptor space rather than as the conjunction of artificial ranges. (The most dramatic manifestation of this problem comes with a training set whose distribution in descriptor space is perfectly spherical. The choice of principal components is then arbitrary, and so any choice of mean-centred hypercube could equally well arise as the domain of applicability, resulting in substantial ambiguity.)

2.2.2. Convex Hull

The convex hull method [Preparata & Shamos 1985, Jaworska *et al* 2005, Fernández-Pierna *et al* 2002] improves on the bounding box approach by taking the domain of applicability to be the smallest convex region of descriptor space containing the whole training set. This ensures that the domain is restricted to consist of precisely those points at which the model can be applied without extrapolation.

There are still problems with the convex hull method, however. If the training set covers a non-convex region of descriptor space then false positives may still occur in regions of concavity: in such interior regions unrepresented by the training set, interpolation can suffer the same problems as extrapolation, so they are not necessarily part of the domain. Furthermore, the computational complexity of the convex hull method is prohibitive both in time and in storage as the number of dimensions rises to the order of 10 or 20, as typically occurs in QSAR studies.

2.2.3. City-Block (Manhattan) Distance

All distance-based methods for estimating domain of applicability involve some measure of distance from a chosen 'centre' of the dataset. A choice therefore has to be made over each of the following [Jaworska *et al* 2005]:

- Measure of distance (norm) between points in descriptor space
- Centre point
- Scale of the measure (or, equivalently, a threshold value defining the applicability domain's boundary)

The city-block or 'Manhattan' distance (or l_1 norm) between two points is the summary absolute difference between descriptor values:

$$d_1(\mathbf{x}, \mathbf{y}) = \Sigma_v |x_v - y_v| \qquad \textbf{(2.1)}$$

where the subscript v indexes the descriptors. (The name arises because the city-block distance is the length of the shortest path between the two points if the path is constrained to consist of segments lying parallel to descriptor axes, analogously to plotting a route between two points in New York.)

The centre point within descriptor space is generally chosen on a per-descriptor basis. For the city-block distance, one usually uses median values, inter-quartile midpoints, or other inter-quantile midpoints. Descriptor values should also be scaled so that they have a common range (e.g. inter-quartile or other inter-quantile range).

The city-block distance can be viewed as improving over the bounding-box method by penalising points that simultaneously take extreme values (only just in range) with respect to *several* descriptors. However, for a typical dataset this penalty is too harsh, with the resulting applicability domain (a diagonally aligned square in two dimensions or a multidimensional analogue of an octahedron in higher dimensions) overestimated along the descriptor axes and underestimated in directions involving several descriptors (the reverse of the situation with the bounding-box method).

This distance method may be valid for studies involving descriptors that take regularly-spaced discrete values (for example counts of occurrence of certain chemical structural features), but even then the training set must be carefully constructed to ensure full coverage of the domain.

2.2.4. Euclidean Distance

Euclidean distance (or l_2 norm) is the standard 'geometric' distance between two points in descriptor space, and is computed from the summary squared distance between descriptor values as per Pythagoras' theorem:

$$d_2(\mathbf{x}, \mathbf{y})^2 = ||\mathbf{x} - \mathbf{y}||^2 = \Sigma_v (x_v - y_v)^2 \quad \textbf{(2.2)}$$

where the subscript v indexes the descriptors. It gives rise to a spherical applicability domain.

The grand mean (centre of gravity) of the training set is usually taken as the centre point, although this is not required to be the case. It makes sense to scale the descriptors during preprocessing so that they have a common variance.

The characteristic property of the Euclidean distance is that it is *isotropic*: it is unchanged if expressed with respect to a different set of orthogonal axes. For this reason it is well suited to methods that aggregate the descriptors into a single descriptor space, including PCA preprocessing.

Applicability domain estimates based on Euclidean distance are valid provided that the training set is *isotropically* distributed about the centre point: the attenuation of training set density must be the same regardless of choice of direction in descriptor space. The *standardised* (full-dimensional) multivariate normal distribution would satisfy this assumption, as would uniform distribution within a (full-dimensional) sphere.

2.2.5. Mahalanobis Distance or Leverage

Another distance-based method is based on the concept of 'leverage' or 'Mahalanobis distance' h of a test compound's chemical structure:

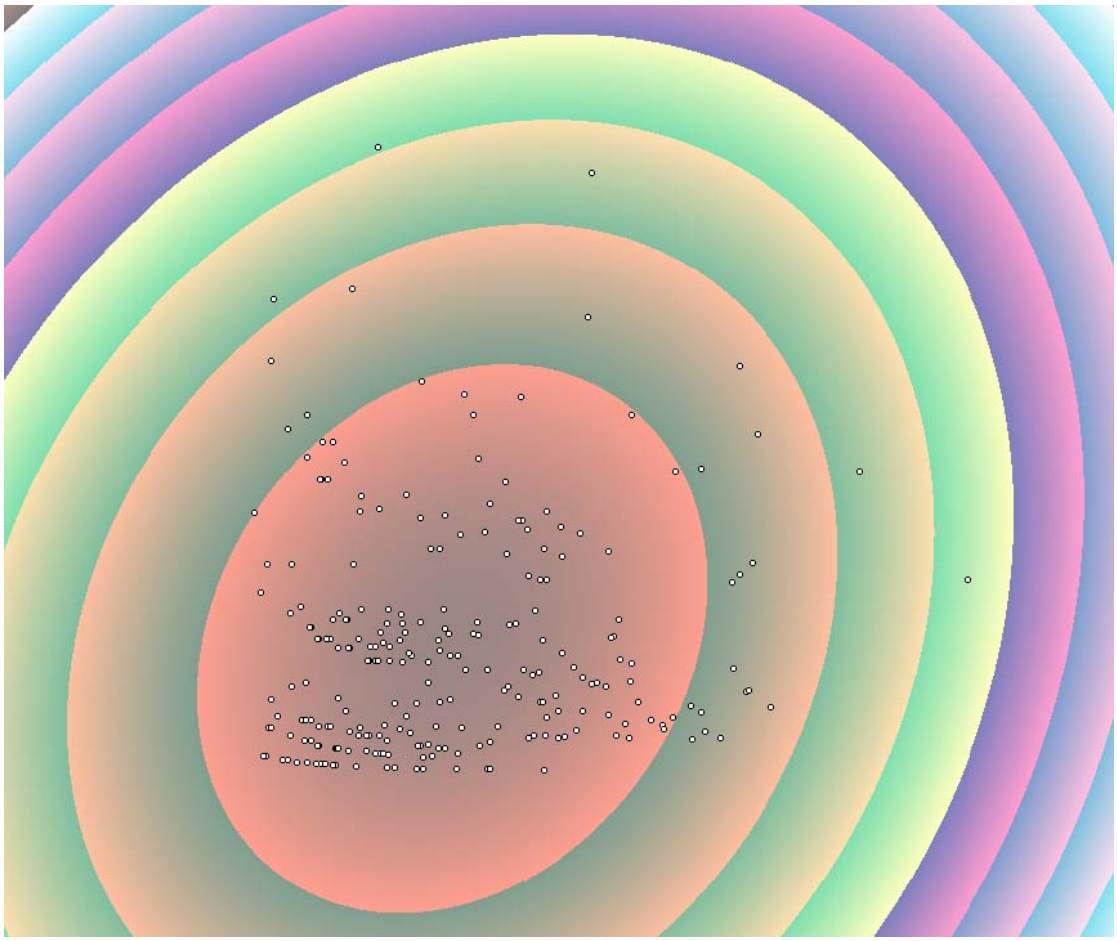

Figure 2.2: Contour Plot of Mahalanobis Distance

This plot portrays the 258 chemical compounds of the Huuskonen dataset using two descriptors (molecular weight and Todeschini hydrophilicity index H_y [Todeschini & Consonni 2002], aligned with the page). The concentric ellipses (also aligned with the principal components) are contours of equal leverage or Mahalanobis distance from the dataset's grand mean.

$$h(\mathbf{x}) = N^{-1} + \mathbf{x}^T(X^TX)^{-1}\mathbf{x} \qquad \textbf{(2.3)}$$

where the vector **x** represents the test compound's structure in centred descriptor space and X is the training data matrix whose N rows represent the training compounds' structures in the centred descriptor space [Sheridan *et al* 2004, Jaworska *et al* 2005, Eriksson *et al* 2003, Tropsha *et al* 2003]. (As with Euclidean distance, 'centred' in this context means that the grand mean of the training data is taken as the origin of descriptor space. Note that X^T denotes the transpose of the matrix X.) In the context of linear least squares regression models, the leverage of an entity is a measure related to the statistical error of its prediction, and can be viewed as a measure of extrapolation and of influence of that entity on the linear model. (Addition of the N^{-1} term in (2.3) accounts for the statistical error arising from estimation of the linear model's constant.)

The leverage is the Euclidean distance applied to data that has been preprocessed using principal components analysis (PCA), and normalising the principal components to have equal variance. Doing so neatly skirts the problems of the Euclidean distance as the principal components are guaranteed to be uncorrelated. Indeed, a training set that is distributed with concentric ellipsoidal contours (e.g. multivariate normally distributed) thus transformed will become isotropic as required.

Although there is a sound mathematical footing underlying the interpretation of leverage as the statistical error in prediction, it relies on the assumption that there is an underlying linear model applicable globally – both inside and outside the domain, with unreliability of prediction arising purely from statistical error in the least squares estimation of the model parameters rather than from limitations in the applicability of the underlying model. Not unrelated is the observation that leverage would provide only a crude estimate of the shape of the domain of applicability: contours of $h(\mathbf{x})$ are always ellipsoids in descriptor space.

2.2.6. Nearest Neighbour Methods

An altogether different approach is taken by the 'nearest neighbour' methods, most notably 'k nearest neighbours' (k-NN) [Sheridan *et al* 2004, Tropsha *et al* 2003, Xu & Agrafiotis 2003]. In this approach a distance measure is constructed by taking the average distances from the test compound's chemical structure to the k nearest

chemical structures in the training set; see Figure 2.3. More sophisticated variants include probability density estimation [Jaworska *et al* 2005], in which a 'membership-of-domain' value is estimated via Parzen's Window as the average over all training compounds of a narrow Gaussian distribution centred at each training structure in descriptor space [Parzen 1962].

These methods both give appealing results [Jaworska *et al* 2005, Golbraikh &

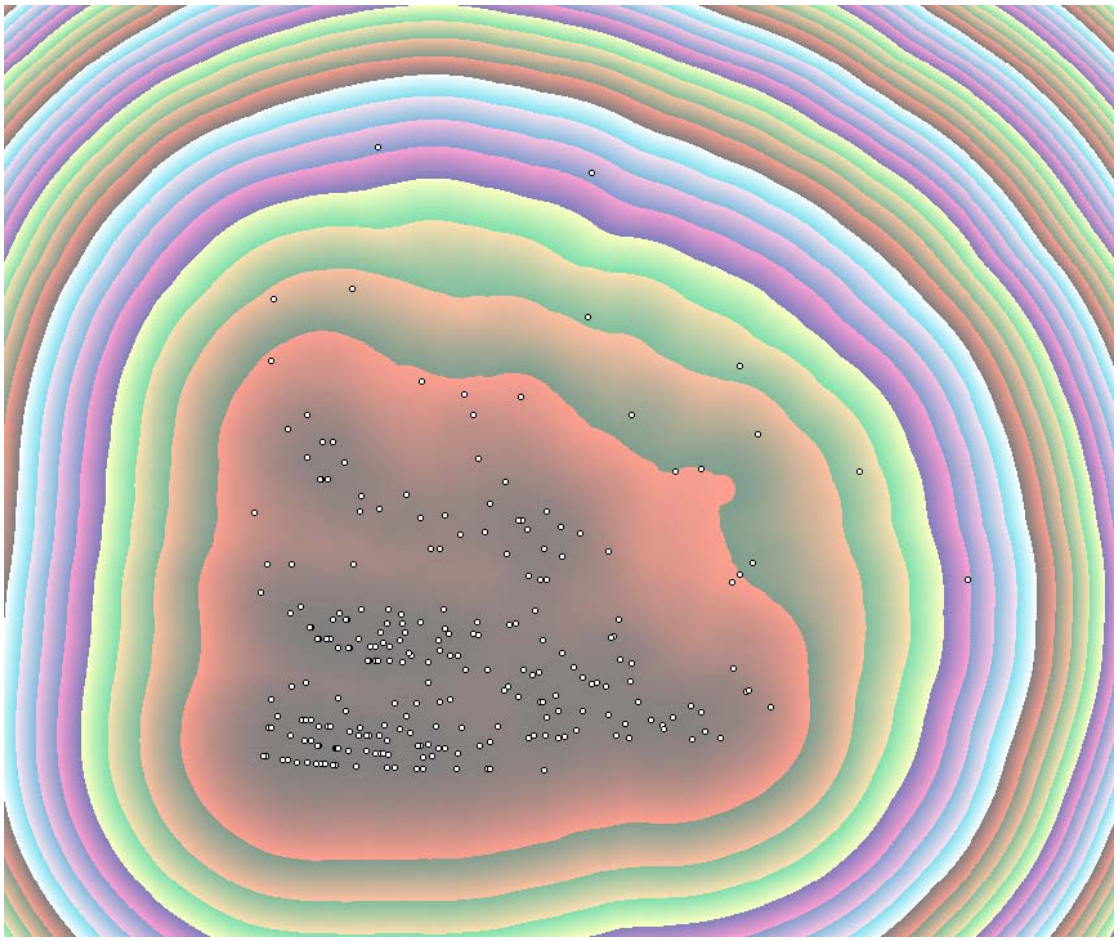

Figure 2.3: Contour Plot of Average Distance to Ten Nearest Neighbours

This plot revisits the 258 chemical compounds of the Huuskonen dataset using two descriptors, molecular weight and Todeschini's hydrophilicity index H_y [Todeschini & Consonni 2002], aligned with the page. The contours illustrate the "*k* nearest neighbour" distance to this dataset, with $k = 10$. The innermost heavy contour, surrounding the central red region, has been chosen to contain 95% of the dataset, and as such may nominally be taken as the boundary of the dataset's domain. Successive heavy contours encompass points for which the mean squared distance (to the point's 10 nearest neighbours) is within 2, 3, 4, etc. times that of the points on the innermost contour.

Tropsha 2002a], but, in much the same way as the convex hull method, their dependence on every single training compound gives rise to substantial time and storage requirements. Indeed, for both methods, the whole training set (which may number several thousand structures [Butina & Gola 2003, Clark *et al* 2003]) must be stored with the model, and in general must be processed in its entirety every time a distance-to-domain calculation is performed (although efficient representations such as *k*D trees [Murtagh 2000] can significantly reduce this computation time in the case of *k* nearest neighbours). This is in contrast to both the bounding box and the leverage-based methods. In the former, the set of ranges over all dimensions provides an easy-to-store and easy-to-use representation of the estimated domain. In the latter, once the covariance matrix $(X^T X)^{-1}$ in (2.3) has been calculated up front it can be reused for all subsequent distance-to-domain calculations without knowledge of the individual chemical structures in the training set.

Table 2.1 summarises the restrictions of the existing methods for determining the domain of applicability of a QSAR model [Kolossov & Stanforth 2007]. As can be seen in the table, every existing method for assessing the domain of applicability suffers either from unwieldy computational and storage issues or from undesirably stringent restrictions on the dataset.

It is worth noting that, in general, 'inverting' the calculation of QSAR descriptors is an impractical task: given a set of descriptor values there is not in general an algorithm for recovering the chemical structure that induced them. This makes any requirements on the distribution of a dataset in descriptor space particularly awkward for QSAR studies, as it is not practical to undertake an experimental design that will produce training structures at prescribed locations in descriptor space.

2.3. Overview of Automatic Extraction of a Test Set

An essential part of the QSAR modelling process (and indeed of any inductive machine learning endeavour) is that of *validation* of the model [Tropsha *et al* 2003, Gramatica 2007] against a test set of entities (chemical compounds) whose activity values are known, but which do not contribute to the construction of the QSAR model *per se*. This amounts to the practical verification that the trained model does indeed generalise to chemical compounds not in the training set. Successful validation is therefore a prerequisite to having confidence in the accuracy of predictions made by

Method	Assumptions on Dataset	Comments	Computational Issues
Bounding Box	descriptors statistically independent		none
Bounding Box with PCA	**seldom satisfied:** principal components statistically independent	mathematically sound, but the assumptions are seldom genuinely satisfied	requirement to store the principal components (latent variables)
Convex Hull	uniformly distributed within a convex region of descriptor space		number of facets can grow (in the worst case) expo-nentially with number of descriptors, leading to substantial complexity for creating, storing, and membership testing even with 10-20 descriptors
City-Block (Manhattan) Distance	uniformly distributed within 'hyper-octahedron'	mathematically sound, especially for count-based descriptors	none
Euclidean Distance	spherical distribution in descriptor space **OR** descriptors statistically independent		none
Mahalanobis Distance (Leverage)	ellipsoidal distribution in descriptor space **OR (seldom satisfied)** principal components statistically independent	equivalent to Euclidean Distance with PCA	requirement to store either the principal components (latent variables) or the entire covariance matrix
k-nearest-neighbours	none	applicability domain can be 'overfitted'	requirement to store all training structures
Probability Density Estimation (Parzen's Window)	none		requirement to store all training structures, and to process them all for *every* prediction

Table 2.1: Comparison of Existing Measures for Determining Domain of Applicability

the model, and is an important component in the assurance of a QSAR model's quality [Kolossov & Stanforth 2007].

Indeed, recent regulatory requirements from the Organisation for Economic Co-operation and Development (OECD) mandate that such validation be carried out before the QSAR model may be used in an environmental or medical situation [Jaworska *et al* 2003].

In an ideal situation, validation is performed against an *external* test set that is supplied along with the training set at the outset of the modelling process. There are potentially two problems with this, however. Firstly, there is no *a priori* guarantee that the chemical structures of the compounds in the test set actually lie within the domain of applicability of the model [Kolossov & Stanforth 2007]. Because the model cannot be expected to make accurate predictions for structures to which it is inapplicable, such compounds would therefore not constitute a fair test: any poor performance that they may highlight would relate to regions of chemical descriptor space in which the model laid no claim to be able to make predictions. (There is, similarly, no guarantee that an external test set would provide full *coverage* of the model's applicability domain.) In the absence of assurance provided by successful distance-to-domain calculations, an external test set experiment has to be viewed as an investigation into where the model happens to be applicable, rather than into whether it is valid.

The second problem with the provision of an external test set is that, since it contains (by assumption) accurately known activity values for a whole collection of chemical compounds, it is hard to justify leaving it out of the training data. There is the tantalising possibility that, by combining the training and external test sets and then extracting from that amalgam a new test set according to some more intelligent design, a better model with broader applicability may be trained.

This thesis will therefore investigate the intelligent, algorithmic generation of an *internal* test set: a test set comprising chemical compounds chosen from the complete original dataset. Such an algorithm will extract a test set that, by design, will afford a fair yet thorough test of the model's predictive power, while leaving behind a residual training set that remains sufficiently representative to be able to generate a model that makes the most of the original data.

We shall now review in turn a number of existing algorithms for extracting such a representative test set: random selection, sampling based on activity value, cell-based partitioning, sphere exclusion, *D*-optimal design, and optimisable *K*-dissimilarity. As was the case with existing methods for estimating a model's domain of applicability (§2.2), most of these algorithms are either rather crude or present forbidding computational requirements.

2.3.1. Random Selection

In the absence of better methods to hand, it is quite common to select a prescribed number of test compounds at random from the training set.

Since there is absolutely no design to this method of selection, however, there is no guarantee of high test coverage.

Reproducibility is a highly desirable property. Note, however, that the random selection method, and indeed any other method involving a random element, can be made to be reproducible. We simply stipulate the random number generator to use, and reset its seed to a fixed initial value every time the method is executed.

2.3.2. Selection Based on Dependent Variable (Activity)

In this method, a test set accounting for 25%, say, of the original dataset is extracted by ranking the training data points (chemical compounds) in order of their dependent variable (activity) and moving every fourth data point to the test set [Golbraikh & Tropsha 2002b].

The improvement of this method over random selection is dubious. In a one-dimensional independent variable (descriptor) space, if there is a model describing the training data well, then good coverage with respect to the dependent variable corresponds to good coverage with respect to the independent variable. However, the situation is different in multidimensional descriptor space. Although the assumption underpinning QSAR studies (and regression analysis in general) is that points nearby in descriptor space have similar activity values, the converse is not true: even in the presence of a model perfectly fitting the data, entire hypersurfaces of points in descriptor space will have equal activity values. Distribution of the test set in descriptor space, at least in directions orthogonal to the gradient of fitted activity, may as well be random.

2.3.3. Cell-Based Partitioning of Feature (Descriptor) Space

In this method, descriptor-space is partitioned into axis-aligned cuboid cells. Random selection occurs within each cell.

The difficulty with this method lies in how to select the cells. If too few cells are used then there will be an undesirably high degree of random selection. At the other extreme, if too many cells are considered then many of them will contain just one data point (or so few points that moving just one point to the test set would render that cell over-represented by test points), and the problem of how to select which cells to over-represent becomes an issue of a similar nature to the test set extraction problem itself. It is difficult to know *a priori* how fine a cellular resolution to use with respect to each descriptor.

2.3.4. Sphere Exclusion in Feature (Descriptor) Space

Sphere-exclusion [Kovatcheva *et al* 2004, Golbraikh & Tropsha 2002b] denotes a class of algorithms working to the principle that once a point has been designated as belonging to the test set, any other data points within a certain distance from it (i.e. within its sphere of exclusion) should be kept in the training set.

The main disadvantage of this method is its computational complexity.

2.3.5. *D*-Optimal Design in Feature (Descriptor) Space

In *D*-optimal design [Galil & Kiefer 1980, Hawkins *et al* 2003], a prescribed number of test compounds are extracted to maximise the determinant of their variance/covariance matrix in descriptor space.

Roughly speaking, this will tend to maximise the volume inside the convex hull of the test points in descriptor space. Unfortunately, it results in sparse test coverage near the centroid of the dataset, and can also result in outlying regions losing too many points to the test set for the residual training set to be representative.

As is the case with *K*-means clustering, a greedy algorithm is generally employed and we settle for a test set that is locally (but not necessarily globally) *D*-optimal.

2.3.6. Optimisable *K*-Dissimilarity Selection

Optimisable *K*-dissimilarity selection [Clark 1997, Clark *et al* 2003] is an algorithm for extracting a diverse, representative subset of a dataset of chemical compounds. The algorithm proceeds by iteratively adding a compound to the diverse subset until the desired target size is reached. At each iteration, a shortlist of *K* compounds (e.g. $K = 5$) is randomly chosen out of those compounds in the dataset whose chemical structures differ from all the previous shortlists' structures by a prescribed threshold distance in chemical descriptor space. From this shortlist, the compound that is most dissimilar in structure to all those included so far is selected, and added to the diverse subset.

This algorithm is successful at finding a structurally diverse subset of the dataset, but fares less well on the criterion of representation, especially for large subsets. In order to maximise the diversity, structures from the fringes of the dataset in chemical descriptor space will typically be chosen more often than structures from the 'core' of the dataset.

2.4. Overview of Modelling Activity in the Absence of an Adequate Linear Relationship

Multivariate linear regression has always been amongst the most popular modelling techniques within QSAR. In particular, certain physical properties including aqueous solubility and the octanol partition coefficient log*P* have been successfully described using simple linear QSAR models, even on large diverse datasets [Yan & Gasteiger 2003, Mannhold & van de Waterbeemd 2001].

However, it is not always possible to find a linear model for the dependence of a biological activity on chemical structures that is applicable across the entirety of a diverse dataset. There may be several reasons for this. For example, the diversity of the dataset may mean that it encompasses a number of substantially different classes of chemical structure, such that the activity arises through entirely distinct biological mechanisms in different classes. In such cases one would not expect a single linear structure-activity relationship to be globally applicable; rather, each distinct biological mechanism may individually admit a linear relationship applicable only to the region of chemical space within which that mechanism occurs.

Piecewise linear regression methods have been employed to try to model in the presence of several distinct mechanisms of activity. For example, the IDBS PredictionBase software [IDBS 2007] supports a two stage modelling paradigm in which the various mechanisms of activity are first specified (by the modeller) as corresponding to ranges of values of the activity. A separate multivariate linear least-squares regression model is then trained based on those chemical structures associated with each mechanism. Quantitative prediction for a new chemical compound then proceeds as follows:

Classification: determine, through a pattern matching algorithm, which mechanism of activity is most likely to apply to the chemical structure.

Evaluation: apply the linear regression model associated with this mechanism of activity to predict the quantitative activity for this chemical compound.

A second possible reason for the absence of a globally applicable linear relationship is that the underlying dependence of activity on chemical structure, assuming that it exists, may simply be non-linear despite arising from a single mechanism.

Some attempts have been made to fit non-linear models to such scenarios in QSAR; for example, support vector regression with a carefully tailored non-linear kernel has been used in the modelling of aqueous solubility [Lind & Maltseva 2003], and artificial neural networks have been applied in a recent study of antibacterial activity [Cherkasov 2005]. The parameter-free technique of *k*-nearest-neighbour regression, in which quantitative predictions are derived by averaging the known activity values of nearby chemical structures in the training set, has also been used [Cedeño & Agrafiotis 2004].

A piecewise multivariate linear regression approach can be applied even when the individual models' domains are not interpreted as regions of distinct mechanism of activity. In such cases, the piecewise regression constitutes another parameter-free method of accommodating an underlying dependence that is not necessarily linear. This method has been successfully applied in modelling aqueous solubility [Butina & Gola 2003].

2.5. Existing Applications of Clustering in QSAR

Clustering enjoys widespread use in QSAR, in various forms. One of its most common applications in QSAR relates to the taming of a large dataset that, considered in its entirety, would be too diverse to be amenable to QSAR modelling. Clustering can isolate a collection of chemically homogeneous subsets of the dataset, each of which is more likely to admit a predictive model applicable to its surrounding region of chemical space [Senese & Hopfinger 2003].

Clustering by *K*-means has also been used in the automatic extraction of a test set for QSAR models. In order to ensure even sampling of the available data by the test set, the original dataset is clustered by *K*-means, and then either a single chemical compound [Burden & Winkler 1999] or a proportion [González *et al* 2004, Samanta *et al* 2006] is selected at random from each cluster. A refinement of this approach will be developed and studied in detail in chapter 6.

Clustering was applied in a novel fashion in a recent QSAR study [Senese & Hopfinger 2003] of a class of chemical compounds associated with the inhibition of the activity of 'HIV-1 protease' – an enzyme involved in the replication of the HIV virus. The study developed a large number of models using a single method but varying the method's parameters, with a view to improving predictivity by being able to combine the predictions from several distinct models instead of relying solely on a single model. It was observed that few of these models were genuinely unique: almost all of the models were very similar to several of the others. Therefore, instead of clustering the chemical compounds, the models themselves were clustered according to the similarity of their residual training errors. Selecting one model from each of the best clusters (as opposed to, say, the best five models overall) ensured that predictions would be drawn from genuinely diverse models.

An implementation of the *K*-means algorithm that is suitable for QSAR datasets has been proposed [Smellie 2004], achieving substantial time savings by sacrificing the requirement for a 'perfect' *K*-means clustering. The resulting 'lossy' clustering, while only approximately satisfying the *K*-means criterion, is judged to be quite adequate for the purposes of determining a representative subset of a QSAR dataset.

Some QSAR studies have also employed fuzzy clustering, including the 'fuzzy *c*-means' algorithm, which is the fuzzy version of the *K*-means algorithm. In fuzzy

clustering, the assignment of entities to clusters is not discrete (as is the case with conventional, ‘crisp’ or ‘hard’ clustering); instead, each entity is permitted to share its membership amongst two or more, or even all, of the clusters. One such QSAR study [Feher & Schmidt 2003] uses fuzzy clustering to model the dependence of a biological activity on a biochemical compound’s *conformation*: its physical three-dimensional shape (as opposed to merely its abstract topological structure) as determined by, for example, the actual orientation of its chemical bonds. It was found that the possible conformations of the compounds under consideration were not amenable to crisp clustering because they formed a continuum lacking in natural groupings or divisions. The fuzzy approach, however, allowed the successful generation of fuzzy clusters (and hence diverse representative conformations) that effectively overlap in their coverage of the dataset, without the requirement for an underlying cluster structure as discrete clusters.

3. *K*-Means Clustering and Its Extensions

3.1. Introduction to *K*-Means

3.1.1. Representation by Clusters

A *cluster representation* of a collection of entities in a dataset describes the dataset as consisting of a number of clusters, with each entity a member of one cluster. Formally, a cluster representation comprises the following elements, for each cluster:

Content: which entities are members of the cluster

Intent: aggregate description of the cluster including, for example, cardinality, location and size in feature space, and most representative member

Disregarding the 'contents' and retaining only the 'intents' of a cluster representation leads to a synoptic model of the dataset. This simplified model captures the broad trends of the dataset, as emerges from its clusters, without retaining any information on individual entities.

Such a synoptic model of the dataset will inevitably lose some information: an aggregate description is substituted for the full enumeration of each cluster. We may therefore consider this cluster-based model to be a simplified approximation to the dataset. The natural next step is to quantify the accuracy of this approximation, or, equivalently, the degree of information loss or retention.

One advantage of considering a cluster's 'intent' as distinct from its 'content' is that it assists in the classification of new points: entities not in the dataset and not contributing to the development of the clustering. In principle, a new point can be assessed with reference to the intent-based description of each cluster, thence determining the cluster in which it best fits. Taken to its extreme, the entire feature space can be systematically classified in this way to partition it into regions each associated with one cluster.

3.1.2. The *K*-Means Loss Function

In the case of *K*-means clustering, the dataset is assumed to consist of entities drawn from some *M*-dimensional feature space. The 'intent' of a *K*-means cluster k is captured by its cardinality and a 'prototype' – a suitably chosen representative point in feature space [MacQueen 1967, Lloyd 1982, Bock 2007].

In what follows, we consider a dataset of N entities $\mathbf{x}_n$ that, according to a partition function $\pi(n)$, are each assigned membership of one of K clusters, which have sizes N_k and representative points $\mathbf{c}_k$. We write $(\pi, [\mathbf{c}_k])$ for this cluster representation.

The characteristic property of K-means clustering arises from the way in which we measure the accuracy of the cluster-based synoptic model it induces of the dataset. This model is visualised according to K-means clustering as containing, in place of each cluster, an equivalent number of points concentrated at the cluster's representative point. The approximation that this involves may therefore be further considered at the level of individual points, via an approximation of each point by its cluster's representative point. This inspires the following 'sum of squares error' information loss function [Steinley 2006]:

$$L([\mathbf{x}_n]; \pi, [\mathbf{c}_k]) = \Sigma_n \|\mathbf{x}_n - \mathbf{c}_{\pi(n)}\|^2 \qquad \textbf{(3.1)}$$

for entities $\mathbf{x}_n$ represented by the K-means clustering $(\pi, [\mathbf{c}_k])$. In the above, $\|\mathbf{x} - \mathbf{y}\|^2$ denotes the squared Euclidean distance between points $\mathbf{x}$ and $\mathbf{y}$ in the feature space.

This use of Euclidean distance in feature space immediately introduces dependence on the features' scales (and a requirement that they be expressed in the same units) [Hartigan 1975, Jain & Dubes 1988, Mirkin 2005]. Before using K-means clustering, conscious thought must therefore be given to how the features are normalised, in the absence of a prior common unit. A later section will discuss this in more detail (§4.3).

We may readily deduce from the K-means loss function (3.1) the choice (for each cluster k) of representative point $\mathbf{c}_k$ that minimises it (with the cluster contents fixed). Indeed, equation (3.1) separates additively both over clusters and over features: using superscripts to select feature components we are led to choose c^v_k to minimise $\Sigma_{n:\pi(n)=k} (x^v_n - c^v_k)^2$, which is shown by elementary calculus to yield a minimum at $N_k\, c^v_k = \Sigma_{n:\pi(n)=k}\, x^v_n$. The information loss function is therefore minimised by selecting cluster centroids (i.e. feature-wise means) as the representative points.

Taking cluster centroids as the representative points $\mathbf{c}_k$, the operation of approximating each point by its cluster's centroid assumes the form of a linear idempotent (projection) operator P_π on the dataset:

$$P_\pi([\mathbf{x}_n]) = [\mathbf{c}_{\pi(n)}] = [\frac{1}{N_{\pi(n)}} \Sigma_{m:\pi(m)=\pi(n)} \mathbf{x}_m]$$

(3.2)

This projection operator allows us to derive the well-known decomposition of data scatter into 'explained' and 'unexplained' contributions [Mirkin 2005]. The overall data scatter resolves into a component *within* the projection – the 'explained' (or 'between-cluster') data scatter of the approximated (projected) data points – plus a residual component *along* the projection – the 'unexplained' (or 'within-cluster') data scatter comprising the summary distance between original and approximated (projected) data points. This latter, residual, component is of course none other than the *K*-means loss function:

$$\text{unexplained scatter} = \text{total scatter} - \text{explained scatter}$$
$$L([\mathbf{x}_n]; \pi, [\mathbf{c}_k]) = \Sigma_n \|\mathbf{x}_n\|^2 - \Sigma_n \|\mathbf{c}_{\pi(n)}\|^2$$

(3.3)

3.2. *K*-Means Optimisation

3.2.1. Local Optimality Criterion for *K*-Means

A *K*-means clustering $(\pi, [\mathbf{c}_k])$ of a dataset $[\mathbf{x}_n]$ is said to be (locally) '*K*-means-optimal' if each point is closer to its *own* cluster's centroid than to any other cluster's centroid. ('Closeness' is judged in terms of Euclidean distance within the dataset's feature space.)

Analysis of this local *K*-means-optimality condition resolves it into two important components:

Optimality of Cluster Membership (optimal Content, for given Intent):

Each point is a member of the cluster to whose representative point it is closest.

Optimality of Representative Points (optimal Intent, for given Content):

Each cluster's representative point is the centroid of its member points.

These mutually interdependent conditions have an elegant interpretation in terms of the *K*-means loss function *L* (3.1). Firstly, given the choice of representative points $[\mathbf{c}_k]$, the required partitioning π (which assigns each point membership of the cluster whose representative point is closest) is immediately seen to be that which minimises

L. Secondly, given the partitioning π, the specified choice of representative points [$\mathbf{c}_k$] (as cluster centroids) was shown in §3.1.2 also to be that which minimises L.

The concept of 'Optimality of Cluster Membership' can be extrapolated to apply to new entities, not necessarily in the original dataset: any such point in feature space is considered a member of the cluster to whose representative point it is closest. This classification of the entire feature space into cluster-associated regions, according to nearest representative point, results in the familiar Voronoi tessellation [Voronoi 1908, Preparata & Shamos 1985] popularly associated with *K*-means clustering [Duda & Hart 1973, Hartigan 1975].

3.2.2. Alternating Optimisation Algorithm

The alternating optimisation (AO) algorithm (also referred to as the expectation maximisation (EM) algorithm) for *K*-means clustering is a greedy algorithm for finding a locally *K*-means-optimal clustering, by alternately updating the partitioning to minimise L (leaving cluster representatives fixed) and updating the cluster representatives to minimise L (leaving the partitioning fixed). When no more updates are possible, both halves of the local *K*-means-optimality condition will be satisfied.

The full Alternating Optimisation algorithm is as follows:

Algorithm 3.1: Alternating Optimisation for *K*-Means Clustering

Inputs:

- dataset of N points $\mathbf{x}_n$ in M-dimensional feature space
- number of clusters K
- initial partitioning π of $\{1, \ldots, N\}$ into K clusters

Procedure:

1. Flag all clusters as being 'dirty'.
2. Repeat, all the while there are any 'dirty' clusters:
    a. **Optimise Representative Points**
    For each 'dirty' cluster k:
        i. Calculate the centroid $\mathbf{c}_k$ based on the current membership of cluster k according to π.

ii. Calculate the squared Euclidean distance $d_{n\,k} = \|\mathbf{x}_n - \mathbf{c}_k\|^2$ from *each* point $\mathbf{x}_n$ to centroid $\mathbf{c}_k$.

iii. Remove the 'dirty' flag from cluster k.

b. **Optimise Cluster Membership**

For each data point n:

i. Determine the cluster k for which $d_{n\,k}$ is smallest.

ii. If $\pi(n)$ is not already equal to k, then flag clusters $\pi(n)$ and k as 'dirty'.

iii. Update $\pi(n)$ to be equal to k.

Outputs:

- final partitioning π of $\{1, \ldots, N\}$ into K clusters
- representative point (centroid) $\mathbf{c}_k$ of each cluster k

The 'dirty' flags in this algorithm, set whenever a cluster's content changes, serve a dual purpose. Firstly, they prevent wasteful recalculation of the centroid (and the distances to it) of any cluster that is unchanged since the previous iteration.

Secondly, and more importantly, the 'dirty' flags form the stopping condition for the algorithm. At the end of step 2b, the cluster memberships (according to π) are optimal for the current representative points $\mathbf{c}_k$. In general, however, the representative points are only optimal *according to the previous iteration's cluster memberships*. For any 'dirty' cluster k (i.e. that has gained or lost a point), $\mathbf{c}_k$ cannot be assumed to be equal to the centroid of its updated content. The clustering is therefore not necessarily locally K-means-optimal and the greedy algorithm may profitably proceed with further iterations.

If, on the other hand, there are no 'dirty' clusters at the end of step 2b, then the partitioning π was not changed at all during this iteration. In other words, the previous iteration's cluster memberships are still in effect – as, therefore, is the optimality of the representative points achieved by step 2a. The greedy algorithm terminates at this invariant point having found a (locally) *optimal* K-means clustering.

At any point during this algorithm, the value of the loss function may be obtained as $L = \Sigma_n\, d_{n\,\pi(n)}$.

The alternating optimisation algorithm is guaranteed to terminate because:

1. In each iteration, step 2b will either make no change (resulting in termination) or will effect a strict decrease in the loss function.
2. Step 2a never causes the loss function to increase.
3. At the end of step 2a, the value of the loss function is wholly dependent on the partition function (i.e. cluster memberships).
4. Because of the strict decreases to the loss function, each iteration has a distinct configuration of the partition function.
5. Since there are only finitely many configurations of the partition function, the algorithm can only proceed for finitely many iterations.

This argument places an extremely large bound (K^N) on the number of iterations that may be followed before termination. In practice, the number of iterations is much smaller than this [Duda & Hart 1973], and was empirically found to be around $O(\sqrt{N})$, although this also depends heavily on the distribution of the data to be clustered [Jain & Dubes 1988].

The most time-consuming part of the algorithm is step 2a(ii), in which the point-to-centroid distances are updated. Each distance calculation has complexity $O(M)$ (where M is the dimension of the feature space), leading to a complexity of $O(KMN)$ per iteration [Manning *et al* 2008], and an estimated overall complexity of $O(KMN\sqrt{N})$. Further methods exist for reducing the computation time; for example, a static binary tree decomposition of the dataset allows many of the candidate clusters in step 2b(i) (for each point $\mathbf{x}_n$) to be ruled out, avoiding the need to calculate the distances $d_{n\,k}$ to those clusters' centroids [Kanungo *et al* 2000].

It is worth discussing one corner case here: although highly unlikely to occur, its analysis will have some bearing on variants of *K*-means considered later. At step 2a(i), there is a small chance that the cluster C_k is empty, either because it was empty in the initial partitioning or as a result of the previous iteration transferring all its members to other clusters. Since is not possible to compute its centroid $\mathbf{c}_k$ in this case, we leave $\mathbf{c}_k$ formally undefined, and in step 2b(i) (for this and all subsequent iterations) exclude $\mathbf{c}_k$ from the available centroids in the search for the nearest centroid to each data point $\mathbf{x}_n$.

This issue of clusters becoming empty can be resolved by using the 'Exchange Algorithm' [Späth 1985, Bock 2007] instead of Alternating Optimisation, as an alternative way of achieving a locally *K*-means-optimal clustering. In the Exchange Algorithm, each iteration only considers a single data point (chosen in a round-robin fashion), instead of simultaneously reconsidering *all* data points' cluster assignment as in AO. The data point $\mathbf{x}_n$, assumed to be in cluster k, is moved to the new cluster l that maximises:

$$\frac{N_k}{N_k-1}\,\|\mathbf{x}_n - \mathbf{c}_k\|^2 \;-\; \frac{N_l}{N_l+1}\,\|\mathbf{x}_n - \mathbf{c}_l\|^2 \qquad \textbf{(3.4)}$$

(where N_k is the current number of members of cluster k) provided that there is a cluster l (other then k itself) for which this expression is positive [Steinley 2006]. The expression in (3.4) is derived from the decrease in the loss function L (see (3.1)) resulting from moving point $\mathbf{x}_n$ out of cluster k and into cluster l, taking into account the updated location of their centroids $\mathbf{c}_k$ and $\mathbf{c}_l$. In the singular case in which $N_k = 1$ (i.e. $\mathbf{x}_n$ is the only member of cluster k and thus coincides with $\mathbf{c}_k$), the first term of (3.4) is taken to be zero; this effectively ensures that no cluster can ever become empty. One possible minor disadvantage of this method is that it introduces dependence upon the order in which the points in the dataset are presented.

3.2.3. Initialisation with Anomalous Pattern Clustering

So far we have discussed in detail iteration and termination of the alternating optimisation algorithm. However, the initial partitioning into clusters (including the value of *K*) was specified as an external input to the algorithm instead of being performed as an integral part of it. Alternating optimisation is therefore incomplete as an *ab initio K*-means algorithm; to merit that status it will require a preprocessing step to generate the initial partitioning.

The Intelligent *K*-means Algorithm [Mirkin 2005] uses so-called Anomalous Pattern Clustering (APC) to procure the initial partitioning required by alternating optimisation. APC is a procedure to extract from the dataset a coherent subset that is in some sense deviant from the bulk of the (original) dataset. Applied iteratively, APC results in a sequence of clusters, each on the periphery of the remainder of the dataset,

that are suitable for use as the initial partitioning in the alternating optimisation algorithm.

An attractive feature of Anomalous Pattern Clustering is that it provides greater resolution at the centre of the dataset in feature space, a region that is typically denser in data points than the periphery.

Anomalous Pattern Clustering uses a constrained form of 2-means clustering to determine where to place the boundary between the new 'anomalous pattern' cluster and the remainder of the dataset. The full Intelligent *K*-means Algorithm is presented below:

Algorithm 3.2: Intelligent *K*-Means

Inputs:

- dataset of N points $\mathbf{x}_n$ in M-dimensional feature space

Procedure:

1. Calculate the centroid (feature-wise mean) $\mathbf{g}$ of the entire dataset.
2. For each point $\mathbf{x}_n$, calculate its *deviance* d_n as the squared Euclidean distance $||\mathbf{x}_n - \mathbf{g}||^2$ from $\mathbf{g}$ to $\mathbf{x}_n$.
3. **Anomalous Pattern Clustering**
    a. Initialise $K = 0$.
    b. Flag each point in the dataset as being 'available'.
    c. Repeat, until there are no 'available' points remaining:
        i. Increase K by 1.
        ii. Determine the most deviant 'available' point, i.e. the 'available' point $\mathbf{x}_p$ for which d_p is greatest.
        iii. Prepare a partitioning function ρ of the 'available' points in the dataset into 2 clusters: the 'anomalous' cluster, containing the single point $\mathbf{x}_p$, and the 'remaining' cluster, containing all other 'available' points. In other words, set $\rho(p) = 2$, and $\rho(n) = 1$ for all other 'available' points (with $n \neq p$).
        iv. Invoke the Alternating Optimisation algorithm, applied only to the 'available' points in the dataset, to optimise the partitioning

ρ. However, during this AO invocation, constrain the representative point $\mathbf{c}_1$ of the 'remaining' cluster to be equal to $\mathbf{g}$ at all times.

v. For each 'available' point $\mathbf{x}_n$ in the 'anomalous' cluster (i.e. for which $\rho(n) = 2$) according to the final partitioning ρ resulting from the AO invocation in step (iv), set $\pi(n) = K$ and remove its 'available' flag.

4. **Alternating Optimisation**
   Invoke the Alternating Optimisation algorithm on the full dataset, initialised with (and updating) the current partitioning π into K clusters.

Outputs:

- final partitioning π
- representative point (centroid) $\mathbf{c}_k$ of each cluster k

The invocation of Alternating Optimisation in step 3c(iv) is guaranteed to terminate, by exactly the same arguments as were given in §3.2.2, in spite of the stipulation that the 'remaining' cluster's representative point $\mathbf{c}_1$ is constrained to be fixed at $\mathbf{g}$. Whether $\mathbf{c}_1$ is optimised or fixed, step 2a of Alternating Optimisation (see Algorithm 3.1) never allows the remaining cluster's contribution to the loss function to increase.

It can also be seen that the Alternating Optimisation invocation in step 3c(iv) never results in an empty 'anomalous' cluster (and therefore the loop at step 3c will eventually terminate). At each 2-means AO iteration, step 2b of Algorithm 3.1 assigns to (or keeps in) the 'anomalous' cluster any point that is closer to $\mathbf{c}_2$ than to $\mathbf{g}$. However, because $\mathbf{c}_2$ is at this stage the centroid of the 'anomalous' cluster's previous members, considering their orthogonal projection onto the straight line through $\mathbf{g}$ and $\mathbf{c}_2$ demonstrates that at least some of them must lie beyond $\mathbf{c}_2$ and so will be kept in the 'anomalous' cluster. (The special case in which $\mathbf{c}_2 = \mathbf{g}$ can be easily overcome by specifying that ties are resolved in favour of the 'anomalous' cluster in step 2b(i).)

This Intelligent *K*-Means Algorithm provides scope for parametric control of the number of clusters in the initial partitioning. At the end of the APC step (step 3), vestigial clusters that are not sufficiently 'significant' may be 'dissolved': their points n are left unclustered with $\pi(n)$ formally undefined, not contributing to any centroid in

step 2a of the first iteration of Alternating Optimisation, and first assigned in step 2b of that iteration. (The labelling of the clusters is then compacted, and their number K reduced, accordingly.) Example criteria for 'significance' of a proposed initial cluster are:

- Accept unconditionally:
  iterate APC as above
- Reject if K exceeds a threshold:
  parameter explicitly specifies (maximum) number of clusters
- Accept only if the size (cardinality) of the cluster exceeds a threshold
- Accept only if the contribution of the cluster to the 'explained' data scatter (see (3.3)) exceeds a threshold proportion of the dataset's 'total' scatter centred about **g**.
- Reject if the cumulative contribution of all previously extracted clusters to the 'explained' data scatter has exceeded a threshold proportion of the dataset's 'total' data scatter about **g**:
  parameter dictates (target) initial value of loss function

It is important to recognise that, although the alternating optimisation algorithm guarantees *existence* of a K-means-optimal clustering, such a clustering is not in general *unique*. The K-means-optimality condition is a form of *local* optimality, in the sense that it requires that (a certain class of) adjustments to the clustering cannot reduce the loss function any further. However, it does not guarantee *global* minimality of the loss function: there may be other K-means clusterings with a lower loss value, leading (via the K-means algorithm) to other (locally) K-means-optimal clusterings with lower loss values still.

A criticism commonly levelled against K-means is this failure to guarantee finding a *global* minimum of the loss function [Jain & Dubes 1988]. However, this is mitigated by the use of a deterministic method such as Anomalous Pattern Clustering to initialise the partitioning. Whilst still not guaranteeing an absolute global minimum, APC does provide a considered starting point leading to a sensible, reproducible locally optimal clustering.

3.3. Fuzzy Extensions to K-Means

3.3.1. Crisp and Fuzzy Clustering

Traditionally, a partitioning of a dataset into clusters is a 'crisp' arrangement: each point in the dataset (or in the space in which it is embedded) is classified as a member of one, and only one, cluster.

Such crisp cluster membership may be described in terms of an indicator function z_k for each cluster k.

$$z_k(\mathbf{x}) = 1 \quad \text{if } \mathbf{x} \text{ is in cluster } k$$
$$z_k(\mathbf{x}) = 0 \quad \text{otherwise} \tag{3.5}$$

The fact that the collection of K clusters provides a disjoint cover of feature space, i.e. with each point in feature space belonging to precisely one cluster, can be stated as follows:

$$\Sigma_k \, z_k(\mathbf{x}) = 1 \tag{3.6}$$

This identifies the cluster membership (for a point $\mathbf{x}$ in feature space) as a *distribution* over the K clusters. Of course, the membership distribution is constrained at this stage to be atomic: unanimously concentrated on one cluster.

This formulation is amenable to substitution of fuzzy sets for crisp clusters. In such a so-called fuzzy clustering, the distribution of membership (for a single point $\mathbf{x}$) over the K clusters is no longer constrained to be unanimous, but is free to be shared amongst several or all clusters to varying degrees. The cluster memberships $z_k(\mathbf{x})$ are now *fuzzy* indicator functions: they take values in range [0, 1] with 1 denoting full exclusive membership of cluster k, 0 denoting no membership of cluster k, and values in between representing degrees of partial membership [Ruspini 1969].

3.3.2. Optimising Fuzzy Cluster Membership

During the K-means alternating optimisation algorithm, and during subsequent classification of new entities, a point $\mathbf{x}$ in feature space is assigned membership of (crisp) clusters on the basis of nearest centroid. In other words, full membership is awarded to the cluster k that minimises $d_k(\mathbf{x}) = \|\mathbf{x} - \mathbf{c}_k\|^2$.

A fuzzy generalisation of this cluster assignment, that is still dependent only on distance-to-cluster measures d_k, can be formulated as an optimisation problem [Bezdek 1973, Bezdek 1981, Bezdek & Pal 1992]:

Fuzzy Cluster Membership Optimisation

minimise:	$f([z_k]) = \Sigma_k \, z_k^{\,\alpha} \, d_k$
with respect to:	$z_k \;\; (k = 1, 2, \ldots, K)$
subject to:	$z_k \geq 0$
	$\Sigma_k \, z_k \;=\; 1$

(3.7)

This is solved for any given point **x** to obtain fuzzy memberships $z_k(\mathbf{x})$ of the clusters k. The parameter $\alpha \geq 1$ controls the degree of fuzziness of the resulting cluster membership distribution, with $\alpha = 1$ reproducing the crisp nearest-centroid (minimal d_k) cluster assignment employed in traditional K-means. In the limit as α increases to infinity, the membership functions converge (pointwise, whenever all d_k are non-zero) to the common fixed value K^{-1}: the clusters have been blurred to the extreme that there is no distinction between them. The value $\alpha = 2$ is often taken for computational expedience [Mirkin 2005].

For $\alpha > 1$, a Lagrange multiplier may be used to obtain an expression for the optimal membership assignment, applicable wherever all $d_k(\mathbf{x})$ are non-zero:

$$z_k(\mathbf{x}) \;=\; \frac{d_k(\mathbf{x})^{-1/(\alpha-1)}}{\Sigma_l \, d_l(\mathbf{x})^{-1/(\alpha-1)}}$$

(3.8)

For any valid value of α, if any $d_k(\mathbf{x})$ is zero, then optimal membership for **x** is achieved by sharing it in an arbitrary fashion amongst those clusters k for which $d_k(\mathbf{x})$ is zero. Assuming that there is only ever one such cluster, this is consistent with (3.8) in preserving the continuity (and indeed smoothness) of the membership functions.

3.3.3. Fuzzy Membership in *K*-Means Clustering

The K-means loss function was presented earlier (3.1) in terms of a partition function π. The partitioning into clusters may equivalently be described in terms of cluster memberships exactly as in (3.5), as follows:

$$z_{n\,k} = 1 \quad \text{if } \pi(n) = k$$
$$z_{n\,k} = 0 \quad \text{otherwise}$$

(3.9)

It is worth explicitly stating the following constraints on such membership functions – constraints which are necessary and sufficient for their correspondence with partitioning functions:

Unity: $\Sigma_k\, z_{n\,k} = 1$
Atomicity: $z_{n\,k} \in \{0, 1\}$

(3.10)

The equivalent formulation of the loss function in terms of cluster memberships is as follows:

$$L([\mathbf{x}_n]; [z_{n\,k}], [\mathbf{c}_k]) = \Sigma_n \Sigma_k\, z_{n\,k}\, ||\mathbf{x}_n - \mathbf{c}_k||^2$$

(3.11)

This immediately identifies the 'Cluster Membership' aspect of *K*-means-optimisation (for each point, given fixed centroids) as an instance of fuzzy cluster membership optimisation (3.7) with $\alpha = 1$. Of the two constraints (3.10) implicitly imposed by crisp *K*-means, the 'unity' constraint occurs verbatim in (3.7), and the 'atomicity' constraint has already been noted in §3.3.2 as a consequence of fuzzy cluster membership optimisation with $\alpha = 1$.

Working by analogy, we can construct a loss function in the spirit of *K*-means for fuzzy clustering [Dunn 1973, Nascimento *et al* 2003, Nascimento 2005]:

$$L([\mathbf{x}_n]; [z_{n\,k}], [\mathbf{c}_k]) = \Sigma_n \Sigma_k\, {z_{n\,k}}^{\alpha}\, ||\mathbf{x}_n - \mathbf{c}_k||^2$$

(3.12)

The fuzzy cluster membership optimisation in (3.7) and (3.8) demonstrates how to obtain those cluster memberships that minimise this new loss function, given representative points $\mathbf{c}_k$. Conversely, given fuzzy cluster memberships, it is easily verified that the loss function is minimised using the following 'fuzzy centroids' for the clusters' representative points [Bezdek 1981, Nascimento *et al* 2003]:

$$\mathbf{c}_k = \frac{\Sigma_n\, {z_{n\,k}}^{\alpha}\, \mathbf{x}_n}{\Sigma_n\, {z_{n\,k}}^{\alpha}}$$

(3.13)

Fuzzy clustering according to this loss function is called 'fuzzy *c*-means'.

The Alternating Optimisation algorithm can be readily adapted to optimise fuzzy *c*-means [Dunn 1973]. With reference to the formulation in Algorithm 3.1, the 'optimise memberships' step 2b is modified to update all fuzzy cluster memberships $z_{n\,k}$. The concept of 'dirty' clusters becomes redundant simply because *all* clusters are dirtied as a result of the fuzzy cluster membership updates; this, together with the fact that the fuzzy cluster memberships can now take on an infinite continuum of configurations, means that AO applied to fuzzy *c*-means will not in general converge in finitely many iterations. Implementations of the algorithm must be arranged to terminate when, for example, the per-iteration decrease in the loss function falls below some tolerance threshold.

As an alternative to using full-blown fuzzy *c*-means, it is possible to obtain a clustering of a dataset using crisp *K*-means, but then use fuzzy cluster membership to classify new entities in feature space. This trade-off offers a compromise, incorporating the discrete Alternating Optimisation algorithm to generate the clustering in the first instance (without having to worry about stopping conditions), alongside the smoothly varying fuzzy cluster memberships in feature space (based on the original crisp clusters' centroids).

3.3.4. Significance of Objective Function

In Bezdek's original formulation of fuzzy membership, the objective function $f([z_k]; \mathbf{x}) = \Sigma_k \, z_k^{\alpha} \, d_k(\mathbf{x})$ was used purely as a tool to guide the fuzzy cluster memberships towards a suitable optimum. However, we consider here a deeper interpretation of f as an aggregate measure of distance from $\mathbf{x}$ to the dataset (modelled as a collection of clusters). Informally speaking, f is an average distance-to-cluster measure weighted according to the fuzzy cluster memberships in consideration, with these weights transformed by the power of α to attenuate the reported distance according to their degree of fuzziness. Fuzzy membership for $\mathbf{x}$ is distributed over the clusters in such a way as to minimise this distance to dataset – in other words, to maximise its belongingness to or *affinity* with the dataset.

With $\alpha = 1$ (crisp clustering), the optimal value of the objective function f is simply the smallest value of d_k: the distance to the nearest cluster. For $\alpha > 1$, the optimal objective value is determined by substitution of (3.8):

$$f([z_k(\mathbf{x})]) \;=\; \Sigma_k\, z_k(\mathbf{x})^{\alpha}\, d_k(\mathbf{x}) \;=\; [\Sigma_k\, d_k(\mathbf{x})^{-1/(\alpha-1)}]^{-(\alpha-1)} \tag{3.14}$$

It is illustrative to compare this expression with a similar average distance-to-cluster, weighted according to the fuzzy cluster memberships in their original form (i.e. *not* transformed by power of α):

$$\Sigma_k\, z_k(\mathbf{x})\, d_k(\mathbf{x}) \;=\; \frac{\Sigma_k\, d_k(\mathbf{x})^{(\alpha-2)/(\alpha-1)}}{\Sigma_k\, d_k(\mathbf{x})^{-1/(\alpha-1)}} \tag{3.15}$$

Comparing these expressions reveals a remarkable significance to the value $\alpha = 2$:

$$\begin{aligned} f([z_k(\mathbf{x})]) \;&=\; \Sigma_k\, z_k(\mathbf{x})^2\, d_k(\mathbf{x}) \;=\; [\Sigma_k\, d_k(\mathbf{x})^{-1}]^{-1} \\ \Sigma_k\, z_k(\mathbf{x})\, d_k(\mathbf{x}) \;&=\; K\,[\Sigma_k\, d_k(\mathbf{x})^{-1}]^{-1} \end{aligned} \tag{3.16}$$

In other words, apart from a constant multiple K, these expressions (where $\alpha = 2$) are equal to one another, and to the harmonic mean of the distance-to-cluster measures.

3.4. Variations of the *K*-Means Criterion

3.4.1. Kernel-Based *K*-Means

Crisp *K*-means clustering, and the Alternating Optimisation algorithm for generating it, can be generalised by substituting different 'cluster intent' representations along with corresponding entity-to-cluster distance measures.

Let us consider a dataset of N points $\mathbf{x}_n$ in M-dimensional feature space U, clustered according to the partition function $\pi(n)$ into K crisp clusters. Now suppose that the 'intent' $\mathbf{c}_k$ of a cluster k is described by a vector in some associated M'-dimensional property space P. A general class of *K*-means-related clustering formulations can be isolated by using expressions of the following form to measure the distance from a point $\mathbf{x}$ in feature space to a cluster characterised by the vector $\mathbf{c}$ in property space:

$$d(\mathbf{x}, \mathbf{c}) \;=\; D([\mathbf{x}; \mathbf{c}], [\mathbf{x}; \mathbf{c}]) \tag{3.17}$$

where D is a symmetric 'kernel' function acting upon the combined $M+M'$-dimensional space $W=U\times P$ of [point in feature space, vector in property space] pairs. The induced loss function for the clustering is:

$$L_D([\mathbf{x}_n];\, \pi,\, [\mathbf{c}_k]) \;=\; \Sigma_n\, D([\mathbf{x}_n;\, \mathbf{c}_{\pi(n)}],\, [\mathbf{x}_n;\, \mathbf{c}_{\pi(n)}]) \qquad \textbf{(3.18)}$$

In addition, we require D to satisfy Mercer's condition [Mercer 1909, Courant & Hilbert 1962], a form of positive semi-definiteness, which may be equivalently stated as the existence of a (non necessarily finite-dimensional) inner product space V_D and a measurable (but not necessarily computable – efficiently or indeed at all) function $G_D : W \to V_D$ such that $D(\mathbf{u}_1, \mathbf{u}_2) = G_D(\mathbf{u}_1) \cdot G_D(\mathbf{u}_2)$.

Kernel functions are 'additive', in the sense that a linear combination with positive coefficients of kernel functions will itself be a kernel function satisfying Mercer's condition. (This is immediately verified by considering, for example, $V_{D1+D2} = V_{D1} \times V_{D2}$, $G_{D1+D2} : W \to V_{D1+D2}$, and $G_{D1+D2}(\mathbf{w}) = [G_{D1}(\mathbf{w}), G_{D2}(\mathbf{w})]$.)

One interesting class of kernels, in which the property space P (of which $\mathbf{c}$ is a member) is the same space as the feature space U, is those of the following form:

$$D([\mathbf{x}_1;\, \mathbf{c}_1],\, [\mathbf{x}_2;\, \mathbf{c}_2]) \;=\; K(\mathbf{x}_1, \mathbf{x}_2) - K(\mathbf{x}_1, \mathbf{c}_2) - K(\mathbf{c}_1, \mathbf{x}_2) + K(\mathbf{c}_1, \mathbf{c}_2) \qquad \textbf{(3.19)}$$

where K is itself a kernel on U satisfying Mercer's condition, say with $G_K : U \to V_K$ and $K(\mathbf{u}_1, \mathbf{u}_2) = G_K(\mathbf{u}_1) \cdot G_K(\mathbf{u}_2)$. Then $D([\mathbf{x}_1;\, \mathbf{c}_1],\, [\mathbf{x}_2;\, \mathbf{c}_2]) = G_D([\mathbf{x}_1;\, \mathbf{c}_1]) \cdot G_D([\mathbf{x}_2;\, \mathbf{c}_2])$ with $V_D = V_K$, $G_D : W \to V_D$, and $G_D([\mathbf{x};\, \mathbf{c}]) = G_K(\mathbf{x}) - G_K(\mathbf{c})$.

Traditional 'distance-wise' K-means, as we shall hereafter refer to the variety of K-means introduced in §3.1.2, can be identified as a particular instance of this formulation by taking V_K to be the feature space U itself and G_K to be the identity function:

$$d_{\text{dist}}(\mathbf{x}, \mathbf{c}) \;=\; D_{\text{dist}}([\mathbf{x};\, \mathbf{c}],\, [\mathbf{x};\, \mathbf{c}]) \;=\; \mathbf{x}\cdot\mathbf{x} - 2\,\mathbf{x}\cdot\mathbf{c} + \mathbf{c}\cdot\mathbf{c} \;=\; \|\mathbf{x} - \mathbf{c}\|^2 \qquad \textbf{(3.20)}$$

Using in this fashion a non-trivial kernel function K satisfying Mercer's condition is a way of effectively introducing non-linearity into the K-means clustering [Dhillon *et al* 2004]. This is an application of the well-known 'kernel trick', applicable to any data mining algorithm that can be formulated within the feature space in terms of an inner product alone, without requiring the implementation to perform addition or scalar multiplication within the feature space directly. Such an algorithm performed with

respect to such a kernel will be equivalent to the original algorithm being performed on the transformed dataset [$G_K(\mathbf{x}_n)$] in the space V_K.

One of the most common applications of this 'kernel trick' is in Support Vector Machines (SVM) [Vapnik 1995]; other applications include principal components analysis (PCA) [Schölkopf *et al* 1998]. In the case of a *K*-means-optimal clustering, the flat boundaries of the Voronoi tessellation of the clustering in V_K correspond to (in general) curved boundaries in the original feature space.

In these cases – in which the property space (of which **c** is a member) is the same space as the feature space and the kernel is of the form in equation (3.19) – the formulation of *K*-means retains the interpretation of the centroid $\mathbf{c}_k$ of a cluster *k* as a representative point for that cluster. All that has changed is that distances are effectively calculated within the transformed feature space $G_K(U)$ (embedded in the space V_K). Note that 'centroids' are constrained to lie in the image $G_K(U)$ of the feature space: they must be associated with some point in the original feature space *U*. Because of this, and because the transformation function G_K is not necessarily available for efficient computation, calculation of the centroid $\mathbf{c}_k$ for a cluster *k* must be performed in *U* in a fashion specific to the particular kernel in use, by solving the minimisation of equation (3.19) summed over all members of the cluster.

Other classes of kernels have also been investigated, in which the property space (of allowable values of **c**) is *not* the same as the feature space. One specific case is *regression-wise K*-means, in which each cluster is represented by a multivariate linear regression model [Späth 1979, Diday *et al* 1979, Diday *et al* 1989, Hathaway & Bezdek 1993] . This case is of particular interest to us, and is discussed in detail in the following section. For a further example, Diday has also studied the representation of each cluster by a 'core' or 'multi-centre' – a sample of representative elements of the cluster – rather than a single centroid **c** [Diday 1974].

3.4.2. Regression-Wise *K*-Means

Let us consider a dataset of *N* entities [$\mathbf{x}_n$] in some *M*-dimensional feature space *U*, and suppose that with each entity $\mathbf{x}_n$ there is associated some numerical measure y_n of an observable behaviour or 'activity' of the entity. We treat *y* as an 'output' variable, in that we hypothesise that it approximately depends through some underlying mechanism on the feature variables **x**.

On such a dataset, or indeed on any subset of it, we may perform a multivariate regression analysis to model the variation of y against the 'input' variables $\mathbf{x}$. Clustering in which the 'intent' of a cluster is portrayed by its regression model of y against $\mathbf{x}$ on the cluster is called regression-wise clustering.

Focusing on K-means clustering with *linear* regression models representing the cluster intents leads to a kernel-based formulation, as follows. We extend the feature space U to include activity values as $U^+ = U \times \mathbf{R}$; within this M+1-dimensional extended feature space points in the dataset are expressed as $[\mathbf{x}_n, y_n]$. On the other hand, the *property* space $P = U^* \times \mathbf{R} = \{[\mathbf{a}, b]\}$ is the M+1-dimensional space of gradient vectors $\mathbf{a}$ with constants b representing the linear models $y \approx \mathbf{a} \cdot \mathbf{x} + b$.

The underlying approximation in regression-wise clustering is the approximation of a point's 'activity' value by the value 'read off' from its cluster's linear regression model. Using a least square error criterion for the regression, we arrive at the following measure of distance from point to cluster [Späth 1979]:

$$d_{\text{regr}}([\mathbf{x}, y], [\mathbf{a}, b]) \;=\; (y - (\mathbf{a} \cdot \mathbf{x} + b))^2 \qquad \textbf{(3.21)}$$

This is consistent as an instance of equation (3.17) in which the kernel is as follows:

$$D_{\text{regr}}([\mathbf{x}_1, y_1; \mathbf{a}_1, b_1], [\mathbf{x}_2, y_2; \mathbf{a}_2, b_2]) \;=\; (y_1 - (\mathbf{a}_1 \cdot \mathbf{x}_1 + b_1))\,(y_2 - (\mathbf{a}_2 \cdot \mathbf{x}_2 + b_2)) \qquad \textbf{(3.22)}$$

Thus Mercer's condition can be seen to be satisfied for D_{regr} via the transformed space $V_{\text{regr}} = \mathbf{R}$ and the transformation G_{regr} into V_{regr} given by
$G_{\text{regr}}([\mathbf{x}, y; \mathbf{a}, b]) = y - (\mathbf{a} \cdot \mathbf{x} + b)$.

The resulting loss function for regression-wise K-means clustering is as follows:

$$L_{\text{regr}}([\,[\mathbf{x}_n, y_n]\,],\, \pi,\, [\,[\mathbf{a}_k, b_k]\,]) \;=\; \Sigma_n\, (y_n - (\mathbf{a}_{\pi(n)} \cdot \mathbf{x}_n + b_{\pi(n)}))^2 \qquad \textbf{(3.23)}$$

As always for kernel-based K-means formulations, this loss function separates additively over the clusters. Standard calculus techniques lead to the following equations for the optimal linear regression model $[\mathbf{a}_k, b_k]$ for cluster k:

$$\begin{array}{lll} \Sigma_{n:\pi(n)=k}\, \mathbf{x}_n \mathbf{x}_n^{\,T}\, \mathbf{a}_k & +\; \Sigma_{n:\pi(n)=k}\, \mathbf{x}_n\, b_k & =\; \Sigma_{n:\pi(n)=k}\, \mathbf{x}_n y_n \\ \Sigma_{n:\pi(n)=k}\, \mathbf{x}_n^{\,T}\, \mathbf{a}_k & +\; N_k\, b_k & =\; \Sigma_{n:\pi(n)=k}\, y_n \end{array} \qquad \textbf{(3.24)}$$

(In the above, $\mathbf{u}^T$ denotes the transpose of, in this case, a column vector $\mathbf{u}$ into a row vector.) These are immediately recognisable as the so-called normal equations for the linear least-squares regression of y_n onto $\mathbf{x}_n$ over the contents of cluster k [Tabachnik & Fidell 2006].

Because calculating the solution of the normal equations (3.24) for a cluster k involves the inversion of an $M \times M$ matrix (the matrix of variances of and covariances between features x^v over the cluster), special treatment must be given to the case in which the matrix is singular or ill-conditioned. This occurs when the cluster's extent in feature space $\{\mathbf{x}_n : \pi(n)=k\}$ is confined or approximately confined to a hyperplane (or even to a 'plane' of yet smaller rank), and will inevitably occur whenever a cluster's size N_k falls below $M+1$. By analogy with the corresponding (although in that context much less likely) scenario in distance-wise K-means in which a cluster becomes empty, our strategy for dealing with this contingency is to leave the regression model $[\mathbf{a}_k, b_k]$ formally undefined, effectively dropping the cluster from the current and subsequent iterations of the alternating optimisation algorithm.

An especially catastrophic instance of this ill-conditioning occurs when all clusters happen to become ill-conditioned *in the same iteration* of the alternating optimisation algorithm. In this case, the algorithm has no alternative but to stop in a 'failure' state. Although in general unlikely, note that this case will inevitably occur if the entire dataset's extent in feature space is confined to a hyperplane.

Finally, it is worth noting that the fuzzy K-means variation described in §3.3.3 can perfectly well be applied to regression-wise K-means, resulting in the following fuzzy regression-wise loss function [Hathaway & Bezdek 1993]:

$$L_{\text{regr}}([\mathbf{x}_n, y_n]; [z_{n\,k}], [\mathbf{a}_k, b_k]) \;=\; \Sigma_n \Sigma_k \, z_{n\,k}{}^{\alpha} \, (y_n - (\mathbf{a}_k \cdot \mathbf{x}_n - b_k))^2 \qquad \textbf{(3.25)}$$

Solving the linear least-squares regression on each fuzzy cluster k has the following normal equations:

$$\begin{aligned} \Sigma_n \, z_{n\,k}{}^{\alpha} \, \mathbf{x}_n \mathbf{x}_n{}^T \, \mathbf{a}_k \;\; &+ \;\; \Sigma_n \, z_{n\,k}{}^{\alpha} \, \mathbf{x}_n \, b_k \;\; &= \;\; \Sigma_n \, z_{n\,k}{}^{\alpha} \, \mathbf{x}_n y_n \\ \Sigma_n \, z_{n\,k}{}^{\alpha} \, \mathbf{x}_n{}^T \, \mathbf{a}_k \;\; &+ \;\; \Sigma_n \, z_{n\,k}{}^{\alpha} \, b_k \;\; &= \;\; \Sigma_n \, z_{n\,k}{}^{\alpha} \, y_n \end{aligned} \qquad \textbf{(3.26)}$$

These offer the immediate interpretation as a linear least-squares regression in which each entity, instead of having its influence on the model weighted equally over the whole dataset as in the traditional case, has its influence weighted in proportion with its fuzzy membership of cluster *k*.

3.5. Conclusions

In this chapter we have reviewed the theory behind the alternating optimisation (or expectation maximisation – EM) algorithm for *K*-means clustering, and studied extensions to it along three specific lines of development: initialisation, alternative notions of affinity of entity to cluster, and fuzzy formulations.

Providing the *K*-means algorithm with an initial partitioning was identified as an essential step. Traditionally, the goal of this initialisation step has been to find a starting point from which the alternating optimisation algorithm is more likely to end up at (or near) *the* global optimum [Steinley 2006]. In this work, however, we are less anxious about finding this absolute global optimum, and are quite content with a reproducible initialisation method that leads to a reasonable local optimum. (Indeed, later sections will conclude that, for our applications, we are more concerned with the overall area covered by the clusters rather than with the details of their partitioning.) The Intelligent *K*-Means Algorithm described in §3.2.3 fits this bill precisely, by basing initial partitions on 'Anomalous Patterns' in the data.

Fuzzy clustering was introduced, following Bezdek's formulation as an optimisation problem [Bezdek 1973], leading to the well known Fuzzy *c*-Means algorithm [Dunn 1973]. In addition, a deeper analysis in §3.3.4 led to a surprising new interpretation of this optimisation: the fuzzy membership values (for an entity) are those which, in a certain sense, maximise the entity's overall *belongingness* to the clusters. Furthermore, particular significance was found to be attributable to the specific value of 2 for the 'fuzziness' parameter α (a value traditionally used purely for convenience). In this case, the maximal 'belongingness' attained by the optimal fuzzy membership for an entity is related to the harmonic mean of the distances from the entity to the cluster centroids. This interpretation will be pursued in chapter 4, where it will form the basis for the measurement of distance to dataset.

Finally, we considered different representations of clusters, and varying the measurement of distance from entity to cluster, in *K*-means clustering (§3.4). It was shown that a form of the 'Kernel' trick may be used to construct non-linear variants of the *K*-means clustering criterion, and a general framework was developed to encompass both this and least-squares regression-wise clustering. This framework will be applied in chapter 5 in the development of a form of model-based clustering amenable to the construction of piecewise linear models.

It should be noted, however, that regression-wise clustering is just one example of the broader notion of 'model-based clustering', in which the various clusters are modelled as being drawn from some parameterised statistical distribution with some or all of the parameters varying amongst the clusters, and 'optimisation' of the clustering consists of recovering maximal likelihood estimators for these parameters. Even the 'standard' distance-wise *K*-means criterion itself emerges from a model-based formulation, modelling the clusters as spherical multivariate normal (Gaussian) distributions with common variance and parameterised by their centroids (means). Other succinct minimisation criteria can be derived, corresponding to, for example, allowing the clusters to follow ellipsoidal Gaussian distributions with varying orientations [Murtagh & Raftery 1984] and with varying sizes and shapes [Banfield & Raftery 1993]. This issue of modelling an elongated part of a dataset arises in the study of extracting an representative test set in §6.2, and although we do not pursue these model-based methods there in this thesis, there may be some benefit in doing so in future work.

4. A Method for Estimating Domain of Applicability

4.1. Clustering to Model Dataset Shape

Clustering in general, including *K*-means, is traditionally used in data mining for *classification*. The emphasis has been on the isolation of separate clusters, such that within each cluster the data entities share some common characteristics, but between clusters different characteristics are exhibited. The synoptic cluster-based model described in §3.1.1, in which a cluster's 'intent' captures the characteristics common to its 'content', may then be used subsequently to decide to *which* cluster a new data entity should belong.

K-means clustering is amenable to an entirely different usage. Instead of classifying into separate, individually meaningful clusters, we use the aggregate collection of *K*-means clusters of a dataset to model the dataset's *shape*. Taken together, the *K*-means clusters collectively cover the region of feature space occupied by the dataset.

This application of *K*-means further mitigates the criticisms of the Alternating Optimisation algorithm concerning its failure to find a *global* minimum of the *K*-means loss function: see §3.2.3. Two distinct *K*-means clusterings may indeed both be (locally) *K*-means-optimal, and may have completely unrelated partitionings, leading to serious ambiguity and instability in their use as a classification tool. However, considered as models of the shape of the dataset, the two sets of clusters will, in spite of their differences, cover much the same region of feature space.

Comparison of these two *modi operandi* of a *K*-means clustering – 'classification into clusters' versus 'modelling shape of dataset' – draws attention to the different kinds of dataset to which they are applicable. In the case of classification, a stable *K*-means representation can only be expected to be found if the dataset actually has an underlying structure comprising at least *K* isolated clusters, separated by empty or low-density 'no-man's-land' regions. Notwithstanding the existence of contrived counterexamples, the Alternating Optimisation algorithm has empirically been found to be generally successful in recovering this underlying cluster structure of such datasets, with a greater degree of success the more pronounced the separation [Jain & Dubes 1988]. However, where such an underlying structure as separate clusters is absent, the Voronoi tessellation of the *K*-means clustering will be forced to have its

boundaries positioned somewhat arbitrarily. There is a risk of instability arising from this arbitrariness: a potentially quite different local *K*-means-optimum is yielded either by using a different initial partitioning [Jain & Dubes 1988], or by holding out a proportion of the dataset for an internal cross-validation experiment.

In the case of using *K*-means clustering for modelling a dataset's shape, on the other hand, no such constraints are placed on the underlying structure of the dataset. Indeed, the fact that the clusters abut without separation is a positive advantage in that it allows the set of clusters to form a cover of the region of feature space occupied by the dataset. The *K*-means clustering amounts to a parameter-free model of the dataset that places no prior assumptions on its shape, including its connectivity or convexity.

4.2. Cluster-Based Distance-To-Domain

When using a dataset as the training set for constructing a model by some machine learning method, the model's 'domain of applicability' is nothing more than the region that the dataset occupies in feature space. A crisp *K*-means clustering for the dataset can therefore model the shape of this applicability domain as being the union of the regions covered by each cluster: an entity, characterised as a point in feature space, is a member of the applicability domain if, and only if, it is inside one of the clusters. Considering the boundary of this applicability domain, it becomes natural to measure the *distance* to domain (from a point outside the boundary) in terms of the point's distance to the nearby clusters.

There is still an element missing from the *K*-means cluster-based model of a dataset's domain of applicability. The segment of feature space associated with a cluster k is a Voronoi cell – the cell containing the cluster's centroid $\mathbf{c}_k$. However, because the Voronoi tessellation covers the entire feature space rather than just the vicinity of (the centroids of the clusters of) the dataset, it will inevitably be the case that at least some of the Voronoi cells are unbounded, extending to infinity in some directions in the feature space. To achieve a tiling of the domain of applicability *only*, it is necessary, for each such exterior Voronoi cell, to judge which part of it is occupied by entities in the dataset.

In this assessment of the region of occupation of a cluster's Voronoi cell, we make the assumption that the occupied region is approximately spherical, centred at the

corresponding cluster's centroid. This was empirically found generally to be a fair approximation for a *K*-means-optimal clustering, and is a reasonable assumption to make in light of the fact that distance-wise *K*-means is based on the isotropic Euclidean distance in feature space [Hartigan 1975].

Although this assumption constrains a cluster's occupied size in feature space to have little dependence on direction, it stops short of requiring the clusters to have the same size as each other. Indeed, it has been empirically observed that *K*-means clusters' Voronoi cells tend to be smaller in the interior of high-density regions of a dataset.

In making use of this spherical cluster approximation in the model of the shape of the dataset, it becomes necessary to augment a cluster's 'intent' with a *radius* for its approximating sphere, expressing the size of its extent in feature space. A cluster k is therefore now described by the following elements:

- Cardinality: N_k
- Centroid: $\mathbf{c}_k$
- Radius: R_k

According to these cluster descriptions, the synoptic model of the dataset (of N points $[\mathbf{x}_n]$ in an M-dimensional feature space) is now as a union of K clusters, with each cluster k consisting of N_k points occupying an M-dimensional sphere of (squared) radius R_k centred at $\mathbf{c}_k$ in feature space.

For the actual numerical value of the (squared) radius R_k of a cluster k, we shall adopt the 95$^{\text{th}}$ percentile of the (squared) distance to centroid $d_{n\,k} = ||\mathbf{x}_n - \mathbf{c}_k||^2$, over all members $\mathbf{x}_n$ of the cluster k. This ensures that 95% of the members of the cluster are inside its approximating sphere. Most of the remaining 5% are typically only just outside it, but if there are any outlying points in this band at a vastly greater distance from the centroid then they will not unduly influence the estimation of cluster radius.

For a point $\mathbf{x}$ in feature space lying outside (the approximating sphere of) a cluster k, we shall measure its (squared) distance to cluster $d_k(\mathbf{x})$ to be the (squared) distance from the centre of the approximating sphere, in multiples of the sphere's radius:

$$d_k(\mathbf{x}) = \frac{||\mathbf{x} - \mathbf{c}_k||^2}{R_k} \qquad \textbf{(4.1)}$$

The contour $d_k(\mathbf{x}) = 1$ of this distance-to-cluster measure is precisely the boundary of the approximating sphere for cluster k.

These distance-to-cluster measures suggest the following measure of distance-to-domain, being simply the distance to the closest cluster (where distances and 'closeness' are measured in cluster radii):

$$D^{(1)}{}_c(\mathbf{x}) = \min_k d_k(\mathbf{x}) = \min_k \frac{||\mathbf{x} - \mathbf{c}_k||^2}{R_k} \qquad \textbf{(4.2)}$$

We shall use the term 'region of influence' of cluster k to denote the region of feature space for which k is the cluster minimising d_k. (This is not in general equal to the Voronoi cell for cluster k, because of the cluster-dependent R_k factor in d_k. The boundaries of each region of influence are in general piecewise hyperspherical surfaces rather than piecewise hyperplanar.)

The contour $D^{(1)}{}_c(\mathbf{x}) = 1$ may in the first instance be taken as defining the boundary of the domain of applicability.

It can be seen from the contour plot of $D^{(1)}{}_c(\mathbf{x})$ (Figure 4.1) that, despite the attempt to use the clusters collectively to tile the applicability domain without undue emphasis on their individuality, there nevertheless remain artefacts wherever the domain's boundary (or indeed any other contour of distance to domain) crosses from one cluster's region of influence to that of an adjacent cluster. The contours remain continuous at such points but are no longer differentiable.

As a means of restoring smoothness of the distance-to-domain contours at the boundaries between regions of influence, we seek to allow several nearby clusters to contribute to a point's distance-to-domain, instead of the naïve 'winner takes all' approach of considering the nearest cluster only. This is analogous to the technique employed in the k-nearest-neighbours measure of distance to a dataset, in which the impact of the lack of smoothness is partially dispersed by considering the k nearest points instead of pinning sole dependence on the outright winner.

Fully analytical smoothness, and an algebraically simpler expression for distance-to-domain, can be achieved by following this through to the extreme of allowing *all*

clusters to contribute to the distance-to-domain measure, with each cluster being granted a proportionally greater influence the nearer it is to the point in question.

Fuzzy cluster membership introduced in §3.3.2 provides precisely the tools required for this strategy: the fuzzy membership $z_k(\mathbf{x})$ for a point $\mathbf{x}$ of cluster k is ideally suited

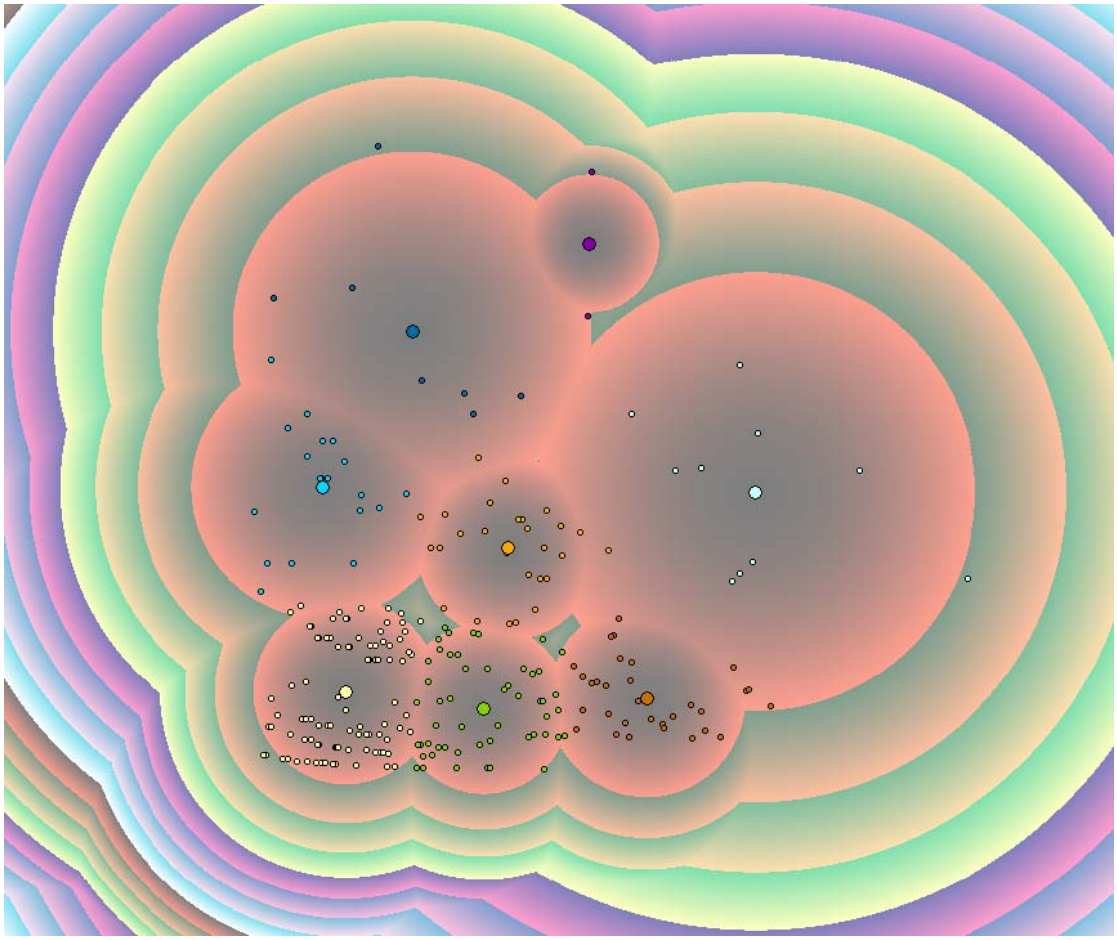

Figure 4.1: Contour Plot of Smallest Distance to Centroid

This plot portrays the 258 chemical compounds of the Huuskonen dataset [Huuskonen 2000] as used for the plots in §2.2, using the same two descriptors (molecular weight and Todeschini's hydrophilicity index H_y [Todeschini & Consonni 2002]).

The Intelligent *K*-Means algorithm (Algorithm 3.1) was applied to cluster the dataset, resulting in eight clusters (excluding one singleton). This is indicated on the above plot by the colouring of the points, while the larger coloured glyphs mark the locations of the clusters' centroids.

The contours of $D^{(1)}{}_c(\mathbf{x})$ according to this clustering are drawn. These illustrate loci of points that share a common value of the minimum (over all clusters k) value of the distance to the centroid of k measured in multiples of cluster radius. The heavy contours in the plot occur where the square of this minimum cluster-radius-tempered distance takes integer values.

to control the degree to which that cluster should contribute to the distance-to-domain measure. Recall from §3.3.4 that the fuzzy cluster memberships $z_k(\mathbf{x})$ were chosen to maximise the *affinity* of $\mathbf{x}$ with the collection of clusters, and that, using the popular and mathematically significant fuzziness value $\alpha = 2$, this optimal affinity (or, more correctly, the minimal *non-affinity* or distance) can be expressed as the harmonic mean of the distances to individual clusters (see equation (3.16)). Substituting the radius-tempered distance (equation (4.1)) for d_k in equation (3.16) gives the following measure of distance to dataset:

$$D(\mathbf{x}) \;=\; \Sigma_k \, z_k(\mathbf{x}) \frac{||\mathbf{x} - \mathbf{c}_k||^2}{R_k} \;=\; \frac{K}{\Sigma_k \, (R_k \,/\, ||\mathbf{x} - \mathbf{c}_k||^2)} \qquad \textbf{(4.3)}$$

This expression is, as desired, a weighted average of distances to clusters, with the weights distributed according to the proprietary share that each cluster can claim over the point in question.

One impact of using fuzzy membership to apportion the influence of nearby clusters is that, because the influence is dispersed over clusters beyond just the nearest, the value $D(\mathbf{x})$ yielded by (4.3) slightly overestimates the distance-to-domain. The dual perspective of this observation is that the domain of applicability is slightly underestimated by (4.3), as points (particularly those near the boundary of exterior clusters) must have a spread of proximity to several clusters in order to be accepted.

Quantifying this overestimation of distance is hard: it depends (among other factors) on the surface area of the domain of applicability. In the interests of pragmatism it was decided to normalise the distance-to-domain measure empirically, by introducing a multiplicative constant factor chosen to render the 95$^{\text{th}}$ percentile of the distance-to-domain measure (over the N entities in the dataset) equal to unity.

The fully assembled distance-to-domain measure is as follows:

$$D^{(2)}{}_{\text{c}}(\mathbf{x}) \;=\; \frac{A}{\Sigma_k \, (R_k \,/\, ||\mathbf{x} - \mathbf{c}_k||^2)} \qquad \textbf{(4.4)}$$

where A is the 5$^{\text{th}}$ percentile value of $\Sigma_k \, (R_k \,/\, ||\mathbf{x} - \mathbf{c}_k||^2)$ over the original dataset.

See Figure 4.2 for a contour plot of $D^{(2)}{}_{\text{c}}(\mathbf{x})$. In comparing this contour plot with the previous one for $D^{(1)}{}_{\text{c}}(\mathbf{x})$ (Figure 4.1), it is observed that, in addition to the contours

having been successfully smoothed out, their shape is also less rigidly bound to the partitioning into clusters. This suggests that $D^{(2)}_c(\mathbf{x})$ has greater stability, more robustly dealing with a different partitioning arising from an alternative initialisation method or from leave-group-out internal validation.

It is illustrative at this stage to compare these distance-to-domain measures, based on distance to nearby cluster(s), to other existing measures. Firstly, a similarity with Nearest Neighbour (*k*-NN) measures, which are based on 'distance to nearest

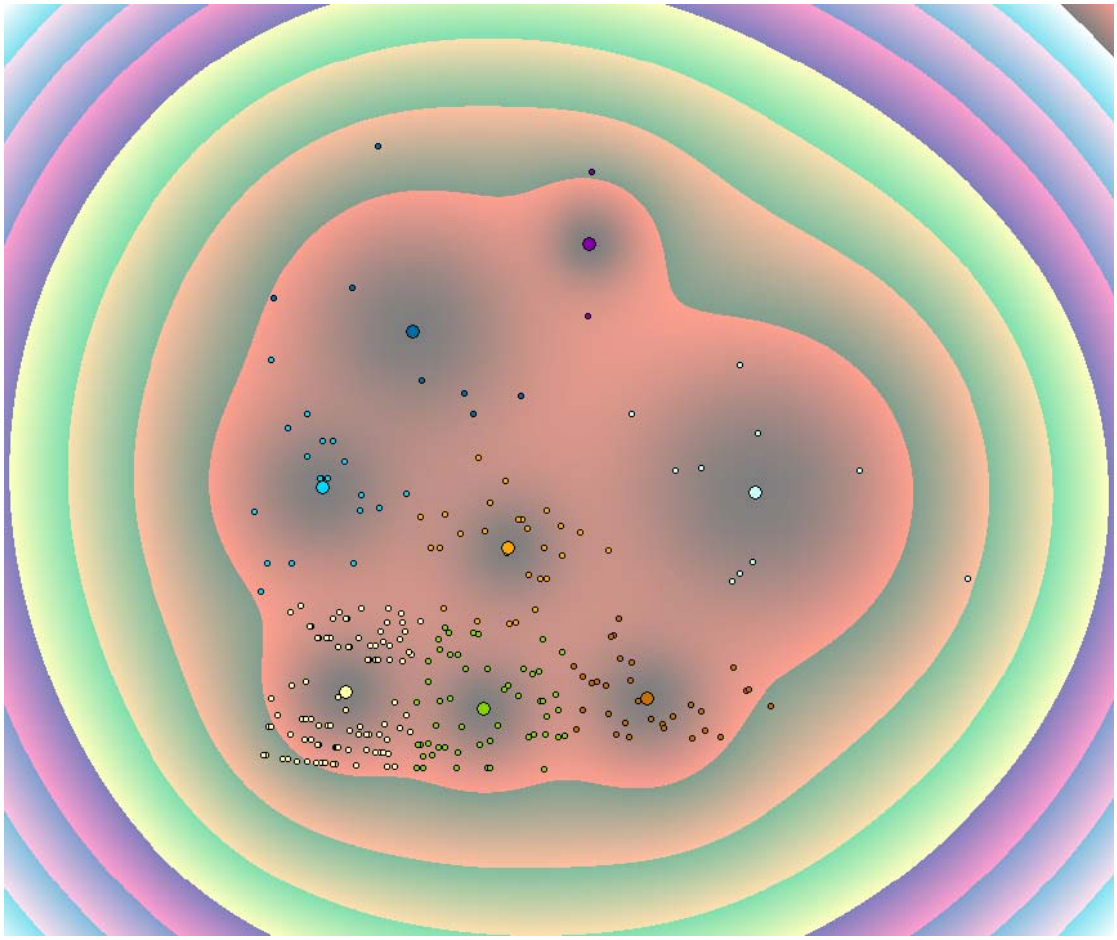

Figure 4.2: Contour Plot of Fuzzy-Weighted Average Distance to Centroid

As with the previous plot, this diagram illustrates the *K*-means clustering of 258 chemical compounds of the Huuskonen dataset using two descriptors: molecular weight and Todeschini's hydrophilicity index H_y [Todeschini & Consonni 2002].

The contours illustrate the cluster-based distance to dataset measure $D^{(2)}_c(\mathbf{x})$. The innermost heavy contour is that for which $D^{(2)}_c(\mathbf{x}) = 1$, and the central red region that it surrounds contains (by choice of the *A* parameter) 95% of the points in the dataset. The successive heavy contours correspond to $D^{(2)}_c(\mathbf{x}) = 2, 3, 4$, etc.

point(s)', has already been noted. From a point of view of modelling, however, the representation of the dataset used by Nearest Neighbour measures is not so much a model of the dataset as a full replica of it.

This explicit dependence of a *k*-NN measure on each and every point in the original dataset has two problems. Firstly, the whole dataset must be published as part of the model purely to facilitate the distance-to-domain computations. Secondly, this detailed dependence on each individual data point is overkill in the sense that, instead of capturing the essential *trends* of the dataset's distribution in feature space, it preserves the *noise* in the data. The fact that a few spurious points strongly impact the *k*-NN measure in their locality suggests that the additional computational burden serves only to reduce the stability of the measure. (This will be experimentally tested in a later section.)

At the other extreme, these cluster-based measures have an element in common with Mahalanobis distances and ellipsoid approximation: specifically, Euclidean distance to centroid and sphere approximation are used in the individual distance-to-cluster measures. The cluster-based distance measures presented here are amalgams of individual cluster-specific Euclidean distance measures. Although approximation by hypersphere or ellipsoid is too crude for typical QSAR datasets, it is a suitable to make an approximation by hypersphere for such a dataset's individual *clusters*, which collectively accommodate the irregularity of the shape of the dataset through their spatial arrangement rather than through their individual shapes.

These observations locate the cluster-based distance-to-domain measures in the middle of a spectrum, with the overfitted *k*-NN measure at one extreme and the underfitted Euclidean and Mahalanobis distances at the other.

4.3. Experimentation

As observed in §3.1.2, the features must be normalised to have commensurate scales prior to applying *K*-means clustering. Although in principle any of the various methods commonly used in data mining for normalising features (and thereby rendering them dimensionless) may be used, we note that scaling to give unit variance is not necessarily the most appropriate for clustering. For a given scale (i.e. range of values), the variance will be greater the more platykurtic the distribution [Chissom

1970], and the greatest variance is attained when the distribution has two modes concentrated at the endpoints of the range [Mirkin 2005]. Dividing out by the variance will therefore penalise precisely the case in which the values taken by the feature are distributed in a fashion amenable to clustering.

Instead, we shall normalise by applying an affine transformation that maps the 5^{th} and 95^{th} percentiles to −1 and +1 respectively; this standardises the bulk of the range without introducing sensitivity to outliers.

Two specific experiments were performed for assessment of the practical utility of the cluster-based distance-to-domain measure. (They were first presented, along with the basis for the following discussion, in [Stanforth *et al* 2005] and [Stanforth *et al* 2007a].) Firstly, an 'internal validation' experiment was used to test whether retraining the distance-to-domain on a subset yields a comparable measure. Secondly, an 'external validation' experiment was performed to investigate correlation between distance to domain and prediction error in a multivariate regression model.

Two QSAR datasets were used for the experimentation. Internal validation was based on a dataset, consisting of 13066 chemical compounds, compiled within IDBS for training a QSAR model of the octanol partition constant log*P* [Ghose & Crippen 1986, Roy *et al* 2007]. The IDBS PredictionBase software [IDBS 2007] was used to identify chemical descriptors having high (individual) correlation with experimental log*P*, and having acceptable Shannon entropy values [Godden *et al* 2000], over that training set. On this basis, ten topological and information-content descriptors [Devillers & Balaban 1999] were selected.

For the external validation, a model for toxicity of phenols described in [Aptula *et al* 2005] was used. This was based on a training set of 185 chemical compounds and 12 descriptors, with a further 50 compounds used to form the external validation set. Predicted activity values were derived from the model as trained using multivariate least-squares fitting in all 12 descriptors, yielding a coefficient of multiple correlation of r^2=0.83. The IDBS PredictionBase software [IDBS 2007] was used to verify the suitability of this model in terms of stability and predictivity. Although the model is stable (with an average leave-one-out cross-validated coefficient of multiple correlation of q^2=0.83), there is some variation in the quality of predictions over the external validation set: almost half of the validation compounds have predictions

correct to within a standard deviation of the model, but 10% yield prediction errors in excess of three standard deviations. (The root mean square prediction error over the whole validation set is 1.73 model standard deviations.) We shall investigate whether the poorer predictions are associated with a greater distance-to-domain.

4.3.1. Internal Validation

10-fold cross-validation [Tropsha *et al* 2003, Kohavi 1995] was used to assess the stability of the cluster-based distance-to-domain measure, retraining the measure on a subset of the original training set and recalculating distances-to-domain for the remaining points according to the retrained measure. The measures were then checked for concordance.

Recall that we are interested in whether a chemical structure is outside the domain and, if so, by how far. However, we are less concerned with the quantitative distance-to-domain value of a structure that is *inside* the domain. We reflect this in our validation by applying a 'clamping' function $g_t(d) = \max \{ d, t \}$ to the distance-to-domain values d to force them to be at least as great as some minimum threshold value t, below which variation in distance-to-domain is considered irrelevant. A range of values of t from 0.7 to 1.3 was used to investigate the influence of the structures near the nominal boundary of $D(\mathbf{x}) = 1$.

Formally, the cross validation procedure can be described as follows:

Algorithm 4.1: *V*-Fold Cross-Validation of Distance-to-Domain Measures

Inputs:

- dataset T of N points $\mathbf{x}_n$ in M-dimensional feature space U
- method for training a distance-to-domain measure D on U based on any given dataset in U
- clamping threshold t, below which variation in distance-to-domain is considered irrelevant
- number V of groups (or 'folds')

Procedure:

1. Train the distance-to-domain measure D on the entire dataset T.
2. Record $D(\mathbf{x}_i)$ for each point $\mathbf{x}_i$ in T.

3. Randomly partition T into V equal-sized groups $T_1 \ldots T_V$.
4. For each of the V groups T_j:
    a. Retrain the distance-to-domain measure $D^{(j)}$ on the depleted dataset $T \backslash T_j$ formed be leaving out group T_j.
    b. Compute Δ_j, the root mean square value of the relative differences $[\, g_t(D^{(j)}(\mathbf{x}_i)) - g_t(D(\mathbf{x}_i)) \,] \,/\, g_t(D(\mathbf{x}_i))$ over T.

Outputs:

- root mean square relative deviation Δ_j for each group j.

This procedure was applied to the distance measure $D(\mathbf{x})=D^{(2)}{}_c(\mathbf{x})$ in equation (4.4), and to the following four additional distance-to-domain measures (in normalised descriptor space) for comparison:

Bounding Box: variant in which the distance-to-domain is taken to be the maximum squared normalised descriptor value:

$D(\mathbf{x}) = D([x_1, \ldots, x_M]) = \max \{x_1^2, \ldots, x_M^2\}$

Leverage: using normalised value $N\, h(\mathbf{x}) \,/\, 3(M+1)$ where M is the dimension of the feature (descriptor) space [Tropsha *et al* 2003]

Nearest Neighbours *k*-NN: mean squared distance to nearest k=10 training points, normalised analogously to equation (4.4) such that 95th percentile value of this measure over the training set is 1

Cluster-Based: exactly as derived in §4.2 except without taking cluster radius into account: i.e. taking R_k=1 in equation (4.3)

A training set of 13066 chemical structures and 10 descriptors was used. A value of V=10 was taken, and the same 10-fold partitioning was used in step 3 for each of the five measures. This yielded the results displayed in Table 4.1.

Table 4.1 shows that the cluster-based distance-to-domain measures have stabilities that compare well with the existing measures. Indeed, outside the nominal boundary of $D(\mathbf{x})$=1, our cluster-based distance-to-domain measure $D^{(2)}{}_c(\mathbf{x})$ is the most stable of those analysed. The minor loss of instability inside that boundary is due to the fact that, close to a centroid, the cluster-based distance-to-domain measures start to approximate the distance to the nearest centroid, and therefore become sensitive to the details of the clustering. The k-NN method is unsurprisingly the least stable: its

dependence on each individual point in the training set understandably gives rise to significantly altered measures on a depleted training set.

Although the Δ_j values in step 4b were based on relative differences (chosen because they render Δ_j invariant under rescaling the distance-to-domain measure), similar qualitative results were obtained for four of the five methods when the experiment was rerun using absolute differences. The exception was the leverage method, which became noticeably less stable than the other methods. This can be attributed to near-singularity of the variance/covariance matrix Σ_i $(\mathbf{x}_i-\boldsymbol{\mu})(\mathbf{x}_i-\boldsymbol{\mu})^T$ in certain directions in

Threshold distance t[(c)]	Bounding Box[(a)]	Leverage[(a)]	k-NN[(a)]	Cluster-Based[(a,b)] (fixed radius)	$D^{(2)}{}_c(\mathbf{x})$[(a,b)]
0.70	0.0502	0.0388	0.0498	0.0491	0.0472
0.75	0.0458	0.0366	0.0471	0.0454	0.0413
0.80	0.0422	0.0346	0.0446	0.0423	0.0366
0.85	0.0392	0.0329	0.0424	0.0395	0.0328
0.90	0.0368	0.0313	0.0404	0.0371	0.0296
0.95	0.0347	0.0299	0.0386	0.0351	0.0269
1.00	**0.0330**	**0.0287**	**0.0369**	**0.0333**	**0.0245**
1.05	0.0316	0.0275	0.0355	0.0316	0.0223
1.10	0.0303	0.0265	0.0346	0.0301	0.0203
1.15	0.0291	0.0255	0.0329	0.0288	0.0185
1.20	0.0281	0.0246	0.0318	0.0275	0.0170
1.25	0.0271	0.0237	0.0309	0.0264	0.0156
1.30	0.0262	0.0229	0.0300	0.0254	0.0145

Table 4.1: Internal Validation of Distance-to-Domain Measures

[(a)] These figures relate to the IDBS log*P* dataset consisting of 13066 chemical compounds, described by 10 descriptors. 10-fold cross-validation was performed on this dataset: each distance measure was trained on the whole training set and compared with the same measure retrained on the remaining compounds.

[(b)] Clustering this dataset resulted in 17 clusters being generated.

[(c)] Each row tabulates root mean square relative deviations of distance-to-domain, subject to a minimum threshold distance, averaged over the 10 cross-validation iterations.

feature space giving rise to sensitivity to small changes; the corresponding absolute deviations in leverage along those directions will then grow quadratically with **x**. Still, with the relative differences, the leverage method fared the second best.

4.3.2. External validation

The original motivation for distance-to-domain analysis is to identify where reliability of a prediction may be compromised due to lack of similar chemical structures in the training set. In order to verify this relationship between domain of applicability and reliability of prediction, the distance-to-domain measure was applied to an extra

	Bounding Box[(c)]	Leverage[(c)]	*k*-NN[(c)]	Cluster-Based[(b,c)] (fixed radius)	$D^{(2)}_c(\mathbf{x})^{(b,c)}$
Entire validation set[(a)]	1.733 (50)	1.733 (50)	1.733 (50)	1.733 (50)	1.733 (50)
in domain: D<0.8	1.018 (2)	1.701 (44)	1.667 (45)	1.646 (44)	1.667 (45)
in domain: D<0.9	1.338 (8)	1.707 (47)	1.667 (45)	1.667 (45)	1.667 (45)
in domain: D<1.0	1.754 (27)	1.707 (47)	1.667 (45)	1.657 (46)	1.657 (46)
in domain: D<1.1	1.686 (40)	1.707 (47)	1.705 (47)	1.649 (47)	1.657 (46)
in domain: D<1.2	1.669 (41)	1.707 (47)	1.705 (47)	1.649 (47)	1.657 (46)

Table 4.2: External Validation of Distance-to-Domain Measures

[(a)] The figures in this table are based on a toxicity model in 12 descriptors trained using 185 chemical compounds and validated using 50 further compounds. The model has coefficient of multiple correlation r^2=0.830.

[(b)] Clustering this training set resulted in six clusters being generated.

[(c)] The 'prediction error' statistics tabulated here are root mean squared prediction errors expressed as a multiple of the model's standard deviation. They were calculated over the whole validation set (first row), and, for each distance-to-domain measure, over only those compounds inside the domain according to various thresholds of that measure (second and subsequent rows). In each case the number of validation compounds involved is displayed in parentheses.

validation set of chemical compounds with known biological activity values. The root mean square error in prediction over the whole validation set was compared with the corresponding value obtained by considering only those compounds inside the domain.

The linear least squares fitting model of phenol toxicity [Aptula *et al* 2005] was used. The means of squared prediction errors were first calculated over the whole validation set. Then, for each distance-to-domain measure and for a number of threshold values, the mean squared errors were recalculated over those chemical structures with distance-to-domain less than the threshold. The results are shown in Table 4.2.

Once again the results come out in favour of the new cluster-based methods derived in §4.2. Both these new methods are successful in defining domains with improved model predictivity, even slightly outperforming the *k*-NN method: it would appear that the extra degree to which *k*-NN produces a fine-grained model fitting the training data is not capturing any extra information on the domain from which the dataset is drawn as compared with the cluster-based model.

Considering the 'D<1.0' row, we can form an F-statistic from the ratio of the mean-square error 1.657^2 (of the 46 compounds inside the domain) to the corresponding mean-square error 2.444^2 of the 4 compounds outside the domain. This F-statistic of 0.459 (with 46 and 4 degrees of freedom) is 91% significant, against the null hypothesis of prediction errors being independent of distance to domain.

The leverage measure does not perform so well, with only a marginal enrichment in predictivity on disregarding the high-leverage validation structures. Its corresponding F-statistic for the 'D<1.0' case is only 78% significant. These results do not invalidate the statistical theory behind leverage, but rather highlight the different assumptions made. In deriving the cluster-based method we assume that an approximately linear model holds in the neighbourhood of the training data. The statistical analysis of leverage, on the other hand, assumes that some linear model is applicable globally, and that prediction errors far from the dataset arise solely from errors in estimating the model parameters.

4.4. Conclusions

A review of the existing approaches to measuring the domain of applicability of a QSAR model showed that the existing methods are either too crude and therefore underfitted (e.g. Mahalanobis distance), or are excessively concerned with individual points and hence overfitted (e.g. *k*-NN distance and convex hull).

By modelling the dataset (and hence the domain of applicability) as a collection of *K*-means clusters, characterised by hyperspheres, we arrive at a compromise between these two extremes. The *K*-means clustering provides a non-parametric model of the dataset, capturing broad trends in the shape of the dataset, including non-convex and disconnected regions.

The *K*-means clustering essentially stipulates the level of detail – the resolution or 'granularity' – of this model of the dataset's shape: within-cluster fluctuations are set aside, effectively removing the noise in the data scatter from the description. The information loss resulting from this removal is directly related to the objective minimised by the *K*-means algorithm.

Having used the intelligent *K*-means algorithm to model the domain of applicability at an appropriate level of detail as a collection of hyperspheres, the *distance* to domain can then be assembled by aggregating the individual distances to each cluster. This aggregation is achieved using the well-established technique of fuzzy partitioning, and in doing so imbues it with a new interpretation: the optimal fuzzy partitioning is that which maximises each test point's belongingness to, or *affinity* with, the domain of applicability.

It was demonstrated that this use of fuzzy partitioning results in a smoother measure than a naïve 'distance to nearest representative point'. It also reinforces the view of our distance-to-domain measure as an informed compromise between Mahalanobis distance and *k*-NN distance.

Using *K*-means clustering for modelling the shape of the applicability domain (instead of for unsupervised classification of the dataset) is novel, although recent discussions in [Djorgovski *et al* 2002] suggest a similar paradigm for the cluster-based description of complex morphologies. It turns out that this usage circumvents a common difficulty with *K*-means: that of instability when the dataset lacks a pre-eminent

strong underlying cluster structure. In such cases, different initial partitionings and different subsamplings can lead to dramatically different clusterings. However, much the same region of feature space is collectively covered by the clusters' hyperspheres, even in those unstable cases. The potential instability in *partitioning* therefore does not carry over to instability in *distance measure*. Indeed, the stability of the cluster-based distance to domain measure was experimentally verified using 10-fold cross-validation on a large QSAR dataset, wherein it fared better than existing measures including *k*-NN and Mahalanobis distance.

The ultimate test of the efficacy of a distance-to-domain measure rests in its ability to identify when a prediction (using a model trained on the dataset) is at risk of being unreliable due to its lying outside the domain of the model. An experiment using an external test set for a QSAR model vindicated the cluster-based distance to domain measure by demonstrating the following: restricting attention to those chemical compounds in the test set that are *inside* the model's domain of applicability gives a significant reduction in average prediction error. Moreover, this reduction was greater than that yielded by any of the existing methods. In other words, filtering using the cluster-based distance-to-domain measure provides the greatest enrichment in predictive ability.

5. A Segmentation Method for Local Modelling

5.1. Overview

In previous chapters, we have discussed the use of clustering to describe a dataset of entities in feature space alone, without any attempt to model or otherwise take account of their observable behaviour. In §4, for example, the cluster-based distance-to-domain measure developed does not affect the construction of any model of observable activities in terms of the chemical structure. It merely contributes to the preconditions for such models, specifying in which regions of chemical descriptor (feature) space a QSAR model may be expected to apply.

In this chapter we shall augment each entity with a numerical value measuring some aspect of the entity's observable behaviour – in other words, an 'output' variable – with a view to modelling the dependence of the output variable on the feature variables. In the terminology of QSAR modelling, in which the entities are chemical compounds, the output variable is the activity under study; it may denote some biological response such as toxicity, or some physical property such as solubility. (In principle, several output variables could be considered simultaneously in this fashion: all the methods in this chapter will apply equally well to such cases. However, in practice, QSAR studies tend to consider only one observable activity at a time, partly to avoid having to make the assumption that the same modelling techniques will be equally applicable to all activities under consideration.)

Even before we begin to use clusters to influence QSAR models, it is worth studying how knowledge of the entities' observable activity values can help to improve the clustering in feature space. Although we use *K*-means clusters to tile feature space without requiring the clusters to exhibit any strong degree of separation, it is not unreasonable to expect that, if a dataset does actually have some form of underlying cluster structure, then recovering that underlying structure with the *K*-means clusters will lead to a better model of the dataset and hence distance-to-domain measure. If part of what distinguishes this underlying cluster structure is the way in which the activity values vary, then it will be of benefit to allow them to guide the clustering, even if it is ultimately treated as a clustering in feature space alone.

This approach of using activity values to guide a clustering in feature space is particularly appropriate when it is suspected that the observable activity depends on the chemical features via two or more biological mechanisms of activity. In this case, the quantitative dependence of the activity on the chemical features will vary over different parts of the feature space, according to which mechanism of activity has the upper hand in a given region. In guiding a clustering by trying to assimilate regions of feature space with consistent activity dependence, an individual cluster may take on a further interpretation as being a region in which one biological mechanism of dependence of activity on chemical structure is likely to be in effect.

Interpreting the clusters as mechanisms of activity dependence inspires a means of proceeding to use the clustering to contribute directly to the construction of a model for activity. Within the segment of feature space in which each individual mechanism is effective, a model *for that mechanism alone* may be trained with the mechanism's segment of effectiveness as its domain of applicability. These 'elementary' models, as we shall call them, may be aggregated into a single 'composite' model, with a domain of applicability that is itself a composition of the segments of effectiveness of the constituent mechanisms.

The composite model makes predictions through a two-stage operation:

Classification: determine the mechanism of activity in effect for this chemical compound.

Evaluation: apply the elementary model for this mechanism of activity to predict the activity for this chemical compound.

Translating from mechanisms of activity to clusters, this would suggest basing the segmentation directly on the clustering in this composite modelling approach. We construct an elementary model on each cluster's locality in isolation, and use their composition as the full model.

Regression-wise clustering provides the tools for both of these approaches: mechanism-oriented guidance of clustering in feature space, and segmentation-based composite modelling of activity. The 'content' of a regression-wise cluster dictates its extent in chemical space, corresponding to a region in feature space that is coherent in its activity dependence. The 'intent' of a regression-wise cluster immediately yields an elementary model applicable to that cluster's segment of feature space.

In §4.2, it was observed that the cluster-based distance-to-dataset measures are a trade-off between the overfitted *k*-nearest-neighbour (*k*-NN) distances and the underfitted measure given by Mahalanobis distance. Similarly, local modelling based on regression-wise clustering occupies a middle ground in an analogous spectrum for regression models of activity in terms of chemical descriptors. A *k*-NN regression model makes a prediction by averaging the known activity values of the nearest *k* training points. This strongly non-parametric approach shares with the *k*-NN distance measure the burdensome requirement that all training chemical structures must be known at prediction time, yielding a 'model' (of activity dependence in the vicinity of the training structures) that is simply an augmentation of the training data with an interpolation rule, rather than an attempt to distil the training set down to its essential trends.

At the other end of the spectrum, a global linear regression model is rigid in its assumption that the activity depends on the chemical structure according to a single linear relation on the chemical descriptors. This is not always a valid assumption for an entire dataset, but may hold (to an acceptable degree of accuracy) on individual clusters; this is especially so on regression-wise clusters, which are guided with that assumption in mind.

Extending the analogy, we shall use weighted averaging based on fuzzy cluster membership to blend the elementary models together smoothly, exactly as was done with the distance-to-domain measure in (4.4).

5.2. Methodology for Local Modelling

5.2.1. Hybrid *K*-Means Clustering

Recall from §3.4.2 that (in QSAR terminology) regression-wise *K*-means clustering represents each cluster by a linear regression model for activity against the chemical descriptors, and classifies a chemical compound according to the cluster whose regression model provides the closest approximation to the compound's activity value.

Prediction of a cluster-based composite model then entails the following two-stage (classification, evaluation) algorithm:

Algorithm 5.1: Composite Model Prediction

Inputs:

- Partitioning of feature space into *K* clusters
- Elementary models: one predictive model applicable on each cluster
- Chemical structure, characterised in chemical descriptor space by **x**

Procedure:

1. **Classification**
   Determine the cluster *k* to which **x** belongs.
2. **Evaluation**
   Predict the activity *y* for **x** according to the elementary model on cluster *k*.

Outputs:

- Predicted activity *y*

In order to use the model to make predictions about the activity of a chemical compound, it is necessary first to know the cluster to which it belongs. Determining the regression-wise cluster of a previously unseen chemical compound presents a problem: the 'classification' step above requires knowledge of the compound's activity value, but this value is unknown, and indeed is not even estimated until the subsequent 'evaluation' step.

This inherent circularity means that regression-wise *K*-means clustering is not, in its pure form, suitable for segment-based composite modelling. The regression-wise clusters are focused principally on the entities' activity values, to the detriment of their ability to evoke a locality in chemical space. When we do examine a regression-wise cluster's extent projected onto chemical space, we find that the clusters may overlap substantially: the activity dimension is pivotal in the hard separation of regression-wise clusters.

This issue of overlapping clusters is partially dependent on the quality of the dataset, and in particular the degree to which *some* (not necessarily linear) relation between activity and feature values is exhibited. If chemical compounds that are structurally similar (in terms of their chemical descriptor values) do indeed have similar activities – the Fundamental Assumption of QSAR – then there is little scope for clusters to

have overlapping projections in chemical space. Compounds in the overlap are constrained by the assumption to have nearby activity values. On the other hand, if a poor quality dataset contains compounds with similar chemical structures but significantly different activity values, then such compounds would tend to be placed into different regression-wise clusters as the only way to account for their activity variation.

The additivity property of kernels described in §3.4.1 allows us to remedy the situation. Including a predominant contribution of the conventional (distance-wise) *K*-means criterion (3.19) (solely in chemical descriptor space) in the kernel causes a distance-wise element to be retained in the clustering, promoting separation of the clusters in chemical space alone. The result is a 'hybrid' *K*-means formulation in which a cluster's intent is represented by a centroid $\mathbf{c}$ *and* a linear model $[\mathbf{a}, b]$. Membership of a 'hybrid' *K*-means cluster is governed by the composite approximation $\mathbf{x} \approx \mathbf{c}$ AND $y \approx \mathbf{a}\cdot\mathbf{x} + b$, as can be seen from its measure of distance from point to cluster as follows:

$$\begin{aligned} d_{\text{hydrid};\,p}([\mathbf{x}, y], [\mathbf{c}, \mathbf{a}, b]) &= (1-p)\, d_{\text{dist}}(\mathbf{x}, \mathbf{c}) + p\, d_{\text{regr}}([\mathbf{x}, y], [\mathbf{a}, b]) \\ &= (1-p)\, \|\mathbf{x} - \mathbf{c}\|^2 + p\, (y - (\mathbf{a}\cdot\mathbf{x} + b))^2 \end{aligned} \qquad \textbf{(5.1)}$$

The dimensionless parameter p specifies the proportion by which the regression-wise element contributes to the distance-to-cluster measure and loss function.

The hybrid kernel from which the hybrid distance-to-cluster measure results (via equation (3.17)) is stated explicitly thus:

$$\begin{aligned} &D_{\text{hybrid};\,p}([\mathbf{x}_1, y_1; \mathbf{c}_1, \mathbf{a}_1, b_1], [\mathbf{x}_2, y_2; \mathbf{c}_2, \mathbf{a}_2, b_2]) \\ &\quad= (1-p)\, D_{\text{dist}}([\mathbf{x}_1; \mathbf{c}_1], [\mathbf{x}_2; \mathbf{c}_2]) + p\, D_{\text{regr}}([\mathbf{x}_1, y_1; \mathbf{a}_1, b_1], [\mathbf{x}_2, y_2; \mathbf{a}_2, b_2]) \\ &\quad= (1-p)\, (\mathbf{x}_1 - \mathbf{c}_1)\cdot(\mathbf{x}_2 - \mathbf{c}_2) + p\, (y_1 - (\mathbf{a}_1\cdot\mathbf{x}_1 + b_1))\, (y_2 - (\mathbf{a}_2\cdot\mathbf{x}_2 + b_2)) \end{aligned} \qquad \textbf{(5.2)}$$

It satisfies Mercer's condition via, in the notation of §3.4.1, a transformed space $V_{\text{hybrid};\,p} = U \times \mathbf{R}$, which is mapped into by the transformation $G_{\text{hybrid};\,p}$ given by:

$$\begin{aligned} G_{\text{hybrid};\,p}([\mathbf{x}, y; \mathbf{c}, \mathbf{a}, b]) &= [\sqrt{1-p}\, G_{dist}([\mathbf{x}; \mathbf{c}]), \sqrt{p}\, G_{regr}([\mathbf{x}, y; \mathbf{a}, b])] \\ &= [\sqrt{1-p}\, (\mathbf{x} - \mathbf{c}), \sqrt{p}\, (y - (\mathbf{a}\cdot\mathbf{x} + b))] \end{aligned} \qquad \textbf{(5.3)}$$

The corresponding hybrid *K*-means loss function, as applies to a dataset of *N* chemical compounds $[\mathbf{x}_n, y_n]$ in some *M*-dimensional chemical descriptor space *U* augmented with a scalar activity dimension, is as follows:

$$\begin{aligned} L_{\text{hybrid};\,p}([\,[\mathbf{x}_n, y_n]\,],\, \pi,\, [\,[\mathbf{c}_k, \mathbf{a}_k, b_k]\,]) \\ &= (1-p)\, L_{\text{dist}}([\mathbf{x}_n],\, \pi,\, [\mathbf{c}_k]) \;+\; p\, L_{\text{regr}}([\,[\mathbf{x}_n, y_n]\,],\, \pi,\, [\,[\mathbf{a}_k, b_k]\,]) \\ &= (1-p)\, \Sigma_n \, \|\mathbf{x}_n - \mathbf{c}_{\pi(n)}\|^2 \;+\; p\, \Sigma_n \, (y_n - (\mathbf{a}_{\pi(n)} \cdot \mathbf{x}_n + b_{\pi(n)}))^2 \end{aligned} \qquad \textbf{(5.4)}$$

There is a slight issue of dimension in the above equations. In (5.1), for example, the distance-wise term (with coefficient $1-p$) has units of chemical descriptor (squared), while the regression-wise term (with coefficient p) has units of activity (squared). Prior to feature (descriptor) normalisation, this unit mismatch also occurred within the Euclidean distances $\|\mathbf{x} - \mathbf{c}\|^2$, which involve adding a contribution from each feature. It was for the benefit of such distance calculations, and to allow a sensible interpretation of 'isotropic' within feature space, that it was necessary to normalise the features to comparable scales. Now, in order to make sense of the hybrid expressions in this section, it is also necessary to normalise the activity values in the same way, effectively removing their units.

It is clear from equation (5.4) that the hybrid *K*-means loss function separates into a centroid-dependent (distance-wise) component and a linear-model-dependent (regression-wise) component, implying that the optimal [centroid, linear model] representation of a cluster consists simply of its traditional distance-wise centroid and its pure regression-wise linear model.

5.2.2. Composing the Model

The distance-wise element in the hybrid *K*-means criterion derived in §5.2.1 serves to promote a cleaner separation of clusters in chemical descriptor space than would be achieved with pure regression-wise *K*-means. Crucially, it also provides cluster centroids. So, although hybrid *K*-means clusters may still have some residual overlap in their projections onto feature space, the final centroids may be used as the basis for a pure distance-wise Voronoi tessellation. This will allow the 'Classification' step (step 1) of Algorithm 5.1 to proceed.

In resolving the hybrid clustering into a purely distance-wise partitioning of feature space, we effectively applied a single supplementary iteration of the Alternating

Optimisation algorithm (Algorithm 3.1), using the distance-wise formulation, to update the cluster contents accordingly.

An alternative to this single supplementary iteration is to apply as many supplementary iterations of distance-wise Alternating Optimisation as are required for it to converge. Under this alternative, the resulting clustering will be *K*-means-optimal (in the distance-wise sense). To achieve this we have of course foregone optimality in the hybrid *K*-means sense, although we shall nevertheless recalculate the clusters' regression models – the 'elementary models' in the 'Evaluation step' of Algorithm 5.1 – so as to be optimal with respect to the content of the final clustering.

In this latter case, in which distance-wise Alternating Optimisation was applied in full (i.e. until reconvergence), a shift of perspective allows us to view the hybrid *K*-means computation as a preprocessing step for the distance-wise *K*-means computation: the hybrid *K*-means clustering provides the distance-wise Alternating Optimisation algorithm with initial clusters that are aligned with regions of linearity. Relegating the hybrid *K*-means to a mere initialisation phase will of course reduce its impact, as compared with the 'single supplementary iteration' approach in which it had an immediate bearing on the final clustering. We shall experimentally investigate the difference between these two approaches in a later section.

Both regression-wise *K*-means and hybrid *K*-means share with standard distance-wise *K*-means clustering the requirement for an initial cluster assignment (and indeed determination of the number *K* of clusters to use). We propose that this initialisation be achieved using Anomalous Pattern Clustering, exactly as was incorporated into the distance-wise Intelligent *K*-Means Algorithm (Algorithm 3.2) and used elsewhere in this thesis. Note that the variant of 2-means used by this Anomalous Pattern Clustering to extract the initial clusters should be applied using the standard distance-only *K*-means criterion given by equation (3.1). It would be inappropriate and infeasible to use any form of regression-wise clustering during Anomalous Pattern Clustering's constrained 2-means because the 'anomalous' cluster is initialised to be a single point, which expresses a *location* in feature space but is insufficient to define a *regression* model.

A further enhancement, affecting both steps of Algorithm 5.1, is to use fuzzy cluster memberships to ensure that the elementary models are blended together smoothly.

The final composite model $f(\mathbf{x})$ for the activity of a chemical compound characterised in chemical descriptor space by $\mathbf{x}$ then takes the following form:

$$f(\mathbf{x}) = \Sigma_k \, z_k(\mathbf{x}) \, f_k(\mathbf{x}) \qquad \textbf{(5.5)}$$

where $z_k(\mathbf{x})$ is the fuzzy membership for $\mathbf{x}$ of cluster k, and $f_k(\mathbf{x})$ is the elementary model for cluster k.

In the same spirit, we shall use the fuzzy cluster memberships to influence the elementary models f_k themselves, and train them according to the fuzzy regression-wise *K*-means formulation in equations (3.25) and (3.26). This helps to go beyond the (crisp) regression-wise *K*-means formulation's rigid concern with fitting the training data, by taking more account of continuity of trends into neighbouring clusters.

In spite of these uses of fuzzy cluster membership, and as was the case with the distance-to-domain measure derived in §4.2, the fuzzy memberships do not affect the generation of the clustering: they are merely involved with how the final clustering is applied.

5.3. Experimentation

Two set of experiments were performed. Firstly, the methods were applied to a collection of randomly generated datasets in order to prove the principle.

Secondly, a real QSAR dataset for aqueous solubility was used to investigate the practical applicability of the method.

5.3.1. Experimentation on Randomly Generated Datasets

The hybrid *K*-means method was first applied to a collection of artificially generated datasets, each consisting of entities drawn from a randomly chosen mixture of distributions in some *M*-dimensional linear feature space. A single 'output' variable is also generated (for each entity), according to a different (randomly generated) approximately linear model on each distribution. The intent was to measure how well these 'mixed model' datasets may be fitted using cluster-based composite regression models, and to what degree this fit may be improved by guiding the clustering with a regression-wise contribution. The following is a modified form of a discussion first presented in [Stanforth *et al* 2007b].

Ten datasets, each with 5000 points in ten-dimensional feature space augmented with one activity component, were generated randomly. Each dataset was generated with an underlying structure of five clusters, with the clusters' sizes chosen uniformly at random within the simplex of their possible relative sizes. Each cluster was assigned a randomly generated mean and spread tensor, on the basis of which the cluster contents were generated according to the multivariate normal distribution. Each cluster was also randomly assigned a linear activity model and an activity error variance; activity values for the points in the cluster were generated according to this linear model with random perturbations according to the error variance.

Each dataset was clustered according to the hybrid *K*-means algorithm using the criterion derived in section §5.2.1, the clustering having first been initialised according to Anomalous Pattern Clustering. Results were output at this stage, and again after one supplementary iteration of *K*-means in which the minimum distance assignment was performed with no regression-wise contribution. The *K*-means algorithm was then allowed to proceed with no regression-wise contribution until convergence was achieved again, after which the results were output for a third and final time.

This procedure was repeated (for each dataset) for several values of p, the relative proportion of the regression-wise contribution.

At each stage, the following results were generated:

Regression-wise criterion: value of the regression-wise loss function, expressed as an explained proportion: $1 - L_{\text{regr};\, p} / L_{\text{regr};\, p}(\text{worst})$

Hybrid criterion: value of the hybrid loss function, expressed as an explained proportion: $1 - L_{\text{hybrid};\, p} / L_{\text{hybrid};\, p}(\text{worst})$

Distance-wise criterion: value of the distance-wise loss function, expressed as an explained proportion: $1 - L_{\text{dist};\, p} / L_{\text{dist};\, p}(\text{worst})$

Mean relative prediction error: the mean value of $|y_{n;\, \text{predicted}} - y_n| / |y_n|$ over all entities n, where the predicted value is according to the regression model of the entity's cluster in the current configuration.

In the above, the 'worst' configuration (used for normalising the loss function values) is that obtained using a single cluster and a constant (flat) regression model, leading to the maximum (worst) possible value of the criterion.

Table 5.1 below presents the mean relative errors of prediction for all datasets at all three stages, for the various values of p under consideration. Mean values over all ten

Dataset	$p = 0$	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$	$p = 0.5$
1	2.0865 2.0865 2.0865	1.5636 2.0125 2.0839	1.3256 1.9633 2.0534	0.6849 1.9708 2.0534	0.5302 2.0743 2.0560	0.5673 2.1780 2.0560
2	0.8609 0.8609 0.8609	0.5667 0.9819 0.8909	0.5353 0.8698 0.8902	0.5582 1.0281 0.7512	0.5423 0.9231 0.7509	0.5381 0.9405 0.9148
3	0.4919 0.4919 0.4919	0.4072 0.5639 0.4919	0.4075 0.5516 0.4919	0.4104 0.5514 0.4919	0.4080 0.5533 0.4919	0.3762 0.5389 0.4919
4	0.5529 0.5529 0.5529	0.4333 0.5528 0.5528	0.4219 0.5628 0.5628	0.4185 0.5153 0.5327	0.4197 0.5456 0.5330	0.4200 0.5564 0.5328
5	0.3150 0.3150 0.3150	0.1817 0.3199 0.3200	0.1864 0.3152 0.3200	0.1747 0.3202 0.3201	0.1545 0.3149 0.3201	0.1539 0.3157 0.3200
6	0.8154 0.8154 0.8154	0.5500 0.8012 0.8214	0.3196 0.5979 0.5770	0.3512 0.5677 0.4339	0.3738 0.5954 0.4048	0.3651 0.5942 0.5913
7	0.7504 0.7504 0.7504	0.4859 0.5733 0.5743	0.3332 0.5748 0.5539	0.4152 0.5082 0.5583	0.5957 0.5879 0.5814	0.5252 0.6339 0.5814
8	0.3790 0.3790 0.3790	0.2697 0.4102 0.3964	0.2257 0.4276 0.3955	0.2110 0.4605 0.3997	0.2088 0.4666 0.3960	0.2026 0.4322 0.3584
9	0.1625 0.1625 0.1625	0.1607 0.1625 0.1625	0.1624 0.1625 0.1625	0.1628 0.1625 0.1625	0.1633 0.1605 0.1625	0.2831 0.2176 0.1625
10	1.5186 1.5186 1.5186	0.5604 0.9123 0.8755	0.5875 0.9341 0.9003	0.4545 0.8754 0.9253	0.4115 0.9154 0.9155	0.5007 1.0462 0.9140
Mean	0.7933 0.7933 0.7933	0.5179 0.7290 0.7170	0.4505 0.6959 0.6908	0.3841 0.6960 0.6629	0.3808 0.7137 0.6612	0.3932 0.7453 0.6923

Table 5.1: Mean Relative Prediction Error

The three quantities in each cell present: the error of the composite model based on the original hybrid *K*-means clustering (top), the error based on the clustering after one distance-wise iteration (middle), and the error of the composite model based on a clustering post-processed with the distance-wise *K*-means until convergence (bottom).

datasets are also included.

The prediction results for the 'original' hybrid *K*-means (top value in each cell) show a strong decreasing trend (i.e. improvement) as *p* starts to increase from zero. This is unsurprising, as the relative prediction error closely corresponds to the regression-wise *K*-means criterion [Diday *et al* 1989]. Note that this stage's 'prediction' results have a somewhat artificial advantage as they are based on a cluster assignment that in turn depends on prior knowledge of the activity values. Even so, as *p* continues to increase towards 50%, the decreasing trend in prediction errors is not maintained (and is even reversed for several datasets), suggesting that a relentlessly large regression-wise contribution is not aiding the modelling, and that retaining a distance-wise contribution is significantly beneficial in divining the underlying structure of the dataset.

As we would expect, performing the supplementary distance-only iteration of *K*-means causes the predictive results (centre value in each cell in Table 5.1) to worsen. This is because we are now effectively forgoing our 'unfair' prior knowledge of the activity values and basing the cluster selection on feature values and cluster centroids alone. Here we observe, for most of the datasets and also for the mean, a trend in which the predictive power improves as *p* starts to increase from zero then worsens again as *p* becomes too large. For any dataset, a value of *p* specific to that dataset should then be chosen to minimise the prediction errors, expressing the optimal trade-off between regression-wise guidance and distance-wise cluster separation.

The alternative scheme of carrying through the supplementary distance-only *K*-means until convergence is achieved again yields similar, even slightly better, results. (See the bottom value in each cell in Table 5.1.) The point at which continuing to increase the proportion *p* of regression-wise contribution starts to have a detrimental effect tends to occur later than it did with only a single supplementary distance-only iteration (around 0.4 rather than 0.3). This can be explained by the fact that performing a greater amount of distance-based post-processing is better able to overcome a heavier regression-wise bias in the initial processing.

Overall, the following conclusions can be made from these experiments:

- The proposed hybrid-based method does indeed allow for a significant reduction in the relative prediction error, of the order of 10%-20%. On average, the error decreases from 79% (using pure distance-wise *K*-means) to 66% (using hybrid *K*-means with the optimal value of the compromise coefficient *p*).
- On average, the option of post-processing with the conventional distance-wise *K*-means works better. However, when the error of the hybrid model is high, as is the case with datasets 1 and 10, the option of post-processing with a single application of the minimal distance rule (i.e. one supplementary Alternating Optimisation iteration) leads to better results.
- The best reduction of the error is achieved with the value of the compromise coefficient *p* at around 0.3.

Table 5.2 presents the values of the regression-wise, hybrid, and distance-wise *K*-

Criterion	$p = 0$	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$	$p = 0.5$
regr	0.9777	0.9954	0.9962	0.9965	0.9967	0.9967
	0.9777	0.9806	0.9798	0.9792	0.9783	0.9783
	0.9777	0.9812	0.9828	0.9816	0.9817	0.9825
hybrid; *p*	0.6533	0.9672	0.9824	0.9880	0.9910	0.9927
	0.6533	0.9548	0.9676	0.9719	0.9736	0.9751
	0.6533	0.9555	0.9707	0.9744	0.9771	0.9794
dist	0.6533	0.6388	0.6239	0.6108	0.5979	0.5814
	0.6533	0.6512	0.6494	0.6461	0.6455	0.6442
	0.6533	0.6531	0.6529	0.6514	0.6540	0.6540
Mean Prediction Error	0.7933	0.5179	0.4505	0.3841	0.3808	0.3932
	0.7933	0.7290	0.6959	0.6960	0.7137	0.7453
	0.7933	0.7170	0.6908	0.6629	0.6612	0.6923

Table 5.2: Average Values of Each *K*-Means Criterion

Values of the criterion (expressed as an explained proportion) of each of the three considered cluster-based models – based on distance-wise, regression-wise and the hybrid *K*-means clustering – at different values of *p*. The three quantities in each cell present: the criterion value after running the hybrid *K*-Means to convergence then stopping (top), that after one supplementary distance-wise iteration (middle), and the value obtained by post-processing with distance-wise *K*-means until convergence (bottom).

means criteria (averaged over the ten datasets) at the three stages of analysis. The values in this table demonstrate the degree to which the distance-wise criterion is boosted, to the detriment of the regression-wise criterion, as the supplementary distance-only *K*-means iterations are performed.

5.3.2. Experimentation on QSAR Data

A practical consideration that arises, and must be addressed before cluster-based composite modelling may be applied to a real-world dataset, is that of *overfitting* [Hawkins 2004]. While training a model, overfitting is the introduction of more complexity than is appropriate for the training data, with the result that the model will overzealously fit the 'noise' in the data instead of isolating the essential trends.

Overfitting typically manifests itself as the presence of too many parameters to fit in the model. In an extreme case, if the number of parameters in the model exceeds the number of training cases, then there are liable to be sufficiently many degrees of freedom for the 'model' to mimic the training data perfectly. Even before we reach this extreme case, if the number of model parameters is not much smaller than the number of training cases then the model is unlikely to be stable under internal cross-validation.

Multivariate linear regression involves fitting $M + 1$ model parameters (where M is the dimension of the feature space): one linear coefficient for each feature, together with the constant term. With composite modelling this number of parameters arises on each of the K clusters, resulting in $K(M + 1)$ parameters in total in the model. It is common to build QSAR models in which the number of training chemical compounds N is of the order of merely five or six times the number of descriptors M; see for example [Jiang *et al* 2003, Varnek *et al* 2004, Duchowicz *et al* 2007, Toropov *et al* 2007]. Even with a modest number of clusters K, using cluster-based composite modelling in such scenarios will therefore have a strong tendency to overfit the training data.

Having established that it is only appropriate to investigate cluster-based composite modelling using a comparatively small number of descriptors, we seek a rationale for selecting a manageable number of descriptors from the several hundred made available by software tools in common use [Talete 2007, Molconn-Z 2006].

The specific criterion that we used for selecting a descriptor is that it be invariant under *tautomerism*, which we shall define shortly.

Recall that calculation of a chemical descriptor is based purely on the molecular *connectivity graph* of the subject compound's chemical structure. This is a graph in the mathematical graph theoretical sense, and provides an abstract representation of molecular structure, with one vertex for each atom (annotated with atom type and other properties such as number of attached hydrogens[1]) and one edge for each molecular bond (annotated with the bond's so-called 'type': single, double, triple or aromatic).

This abstract characterisation of a chemical structure by its molecular connectivity graph is a somewhat idealised representation. In reality, a large class of organic chemical compounds never exist in a single, pure form, but will spontaneously change their structure between two (or more) distinct forms. If one of the forms were isolated in the laboratory it would, inevitably, rapidly revert to an equilibrium mixture of the multiple forms. When the distinct forms occurring in this equilibrium have different molecular connectivity graphs, calculation of a chemical descriptor can involve arbitration to select (artificially) one form as being the canonical one on whose molecular connectivity graph the descriptor calculation is to be based.

Tautomerism is a particularly common class of such cases of equilibrium between multiple forms [Vollhardt 1987]. In tautomerism, the molecular connectivity graphs of the forms ('tautomeric forms' or 'tautomers') occurring in the equilibrium all share the same topological structure, and differ only in atom properties (specifically, number of attached hydrogens) and bond types (usually single versus double). Descriptors that do not depend directly on atom properties or bond types (i.e. that only depend on atom types and abstract connectivity) can therefore be said to be invariant

---

[1] A hydrogen atom is only ever bonded to one other atom. It is therefore normal practice to work with the so-called 'hydrogen-depleted' molecular connectivity graph, in which the hydrogen atoms are not represented in the graph as separate vertices in their own right. Instead, each non-hydrogen atom is annotated with a count of the number hydrogen atoms attached to it.

In practice, an atom's annotated hydrogen count is often implicit, as it can almost always be inferred from its neighbourhood in the hydrogen-depleted graph.

under tautomerism: their computed values will be independent of selection of tautomeric form.

It was therefore decided to restrict attention to tautomer-invariant descriptors for the purposes of the experimental validation of the cluster-based local modelling. A model for aqueous solubility based on 1026 training compounds was used [Huuskonen 2000]. 46 tautomer-invariant topological descriptors [Devillers & Balaban 1999, Todeschini & Consonni 2002] were identified, from which a sequence of eleven was identified using the IDBS PredictionBase software [IDBS 2007] as statistically significant in the dependence relationship of activity (solubility). This sequence of descriptors was constructed in decreasing order of significance to facilitate study of the most significant ten, nine, eight, etc. descriptors. They are listed in Appendix A.

Table 5.3 presents the results of fitting composite multivariate linear regression models based on hybrid *K*-means clustering to the Huuskonen dataset as described with this nested sequence of descriptor sets.

Because solubility is measured on a logarithmic scale, it made sense to measure the *absolute* prediction errors (as opposed to the *relative* prediction errors used with the randomly generated datasets in §5.3.1). Root mean square prediction errors were also calculated; they exhibited much the same trends as those shown by the mean absolute errors in Table 5.3.

Examining the values in Table 5.3 row by row, we first observe that, unsurprisingly, for each selection of descriptors, the composite model based on clustering in descriptor-space alone (the '$p = 0$' column) achieves a significant reduction in error over the usage of a single linear model (the 'global model' column). Note that fitting a single, global linear model may be viewed as the special case of cluster-based composite modelling in which $K = 1$: the entire dataset belongs to a single supercluster. We would expect this trend to continue: the greater the number of clusters, the better the composite model is able to fit the training data (and the greater is the risk of overfitting). Indeed, some of the greatest improvements in mean absolute error in the '$p = 0$' column over the 'global model' column in Table 5.3 occur in those rows in which the number of clusters K is large; (for example, see the row in which $M = 6$).

The improvement given by the distance-only *K*-means ($p = 0$) over non-composite modelling may be interpreted as a benchmark for measuring improvement in composite modelling. The remaining columns ($p > 0$) indicate the degree to which this improvement is further enhanced (if at all) by the use of hybrid regression-wise clustering to incorporate the effects of the activity variation.

We examine Table 5.3 row by row once again, studying the trends in each row as *p* increases. For between 4 and 8 descriptors, a distinct trend emerges, in which increasing *p* from zero initially gives rise to a greater improvement in fit (reducing the

M[a]	K[b]	global model[c]	$p = 0$	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$	$p = 0.5$	$p = 0.6$
2	10	1.279	1.189	1.185	1.186	1.183	1.169	1.188	1.181
3	12	1.270	1.164	1.126	1.160	1.160	1.168	1.164	1.164
4	28	1.272	1.220	1.199	1.196	1.199	1.183	1.181	1.188
5	14	1.275	1.156	1.152	1.141	1.128	1.127	1.147	1.162
6	31	1.275	1.090	1.085	1.089	1.079	1.049	1.083	1.098
7	27	1.274	1.121	1.136	1.122	1.114	1.098	1.101	1.362
8	37	1.274	1.148	1.323	1.028	0.979	1.096	1.102	1.226
9	24	0.878	0.643	0.637	0.635	0.639	0.644	0.642	0.639
10	21	0.851	0.738	0.767	0.744	0.752	0.752	0.754	0.750
11	25	0.777	0.678	0.705	0.656	0.658	0.671	0.678	0.686

Table 5.3: Mean Absolute Error of Composite Models of Aqueous Solubility

This table presents the absolute prediction errors (averaged over all 1026 chemical compounds in the dataset) arising from composite linear models based on hybrid regression-wise *K*-means clustering for a range of values of *p*, the proportion of contribution of the regression-wise element. A single supplementary iteration (only) of distance-wise *K*-means was applied in each case.

[a] The number of descriptors. A nested sequence of descriptor sets was used, with each row in the table relating to the inclusion of one additional descriptor over the set used in the previous row, culminating in M=11 descriptors in the final row.

[b] Number of clusters.

[c] The 'global model' column contains the corresponding results for the *non-composite* (i.e. single, global) linear model.

mean absolute error). As was the case with the randomly generated datasets studied in §5.3.1, however, this improvement in fit is not maintained as p continues to increase. The greatest improvement in fit is yielded at around $p = 0.4$, where the 'baseline' improvement (given by $p = 0$ compared with the single global model) is amplified by around 25%. In these cases (for these numbers of descriptors), the regression-wise component of the hybrid K-means clustering appears to be successfully aligning the clusters with regions of linearity in the data.

When the number of descriptors M is small (at 2 or 3), there is no discernible trend in the mean absolute error as p varies. This may be accounted for by the fact that three descriptors are simply insufficient to characterise a chemical structure, for the purposes of determining its solubility: given a value for each of these three descriptors, there may exist in general several chemical compounds whose structures share these descriptor values, but which have completely different solubilities. No model – not even a non-linear model – can be expected to fit well in such circumstances. As a result, the regression-wise clustering is no more likely to find regions of locally linear dependence than distance-wise clustering, as there are none to be found.

At the other extreme, when the number of descriptors M exceeds about 9, the relationship between p and mean absolute error is also absent. This does not appear to by caused by the onset of overfitting (which will eventually trump the effects of composite modelling for sufficiently large M), for the number of clusters for these values of M is actually smaller than the $M = 8$ case, which demonstrated a strong trend. (The number of model parameters to be fitted for $M = 8$, 9, 10 & 11 are $K(M+1) = 333$, 240, 231 & 300 respectively, compared with the number of datapoints of 1026.)

Instead, we postulate that the failure to maintain the trends above $M = 9$ is caused by the comparative instability of the multivariate linear regressions (being performed for each cluster in each iteration of the hybrid K-means algorithm) in higher-dimensional feature spaces. The fact that this intolerance of higher dimensions appears to occur sooner than with the randomly generated datasets used in §5.3.1 (in which trends were discernible despite working within a 10-dimensional feature space) can be explained

by the greater number of data points (5000) in the random datasets, and the fact that the random data was contrived to contain strong underlying linear trends.

5.4. Conclusions

The *K*-means cluster-based measure of distance to domain, developed in §4, achieved a successful compromise between the overfitted nearest-neighbour approach on the one hand, and the underfitted global ellipsoidal model of dataset shape on the other hand. It managed this by using the clustering to define an intermediate level of detail that was just sufficient to capture the essential elements of the dataset's shape.

In this chapter we have given an analogous treatment to the modelling of an output variable (activity). Local linear trends are identified at the granularity (level of detail) of the *K*-means clusters; these are then aggregated into a non-linear, non-parametric model of the dependence of activity on chemical structure whose domain of applicability encompasses the entire dataset. This model provides an equally successful compromise between the overfitted *k*-nearest-neighbour regression method at one extreme, and rigid global linear regression at the other extreme.

The concept of cluster-wise modelling immediately evokes regression-wise clustering, which simultaneously optimises clusters and cluster-wise models. Regression-wise *K*-means clustering was honed into a form suitable for cluster-based modelling: using the additivity property of *K*-means criteria, a 'hybrid' form of *K*-means clustering was constructed by combining regression-wise and distance-wise *K*-means. This hybridisation of the regression-wise and distance-wise criteria manages the two potentially conflicting goals of, respectively, aligning clusters with regions of linearity, and separating the clusters in feature space.

One possible interpretation of this hybrid *K*-means clustering is as a form of preprocessing, supplying the standard distance-wise *K*-means algorithm with an initial partitioning that is aligned with local regions of linearity (of the dependence of activity on the chemical descriptors) in the dataset.

The regression-wise element in hybrid *K*-means also allows the resulting cluster-based composite model to have a deeper interpretation. Recall that in §4 the distance-wise *K*-means clusters were merely a device to tile the dataset without making any assumptions on its shape: no significance was attributed the individual clusters, and

indeed this was seen as a positive advantage as it circumvented the problem of instability of the clustering. However, upon incorporating a regression-wise contribution, the individual clusters can once again assume significance: they potentially correspond to regions in which distinct mechanisms of activity apply.

Experimentation, both on randomly generated data and on a real QSAR dataset, demonstrated that composite modelling based on hybrid *K*-means clustering can indeed give improvements, over and above those obtained using composite modelling based on conventional distance-wise *K*-means clustering alone. One observes a modest improvement that is initially roughly proportional to the weight of the regression-wise contribution. However, these improvements are reversed and eventually annulled if the regression-wise contribution is too heavy, indicating that the underlying distance-wise contribution is crucial in the development of a clustering that is amenable to composite modelling. For any applicable dataset, the optimal trade-off between distance-wise and regression-wise contributions can be identified for best results.

It remains a limitation of the method that the optimal value of this trade-off parameter *p* is not procured by the clustering. This is in contrast with other approaches to model-based clustering, for example [Murtagh & Raftery 1984], in which all parameters concerning the clusters' location, shape and orientation emerge from the expectation maximisation. It would have to be the subject of future research to determine whether the trade-off between distance-wise and regression-wise elements may be optimised, possibly on a cluster-specific basis, as an integral part of the clustering algorithm itself.

The experimentation on real QSAR data identified those scenarios to which composite modelling based on hybrid *K*-means is suited. The method is applicable to a dataset involving a comparatively small number of chemical descriptors (features) such that, while there may be no globally applicable linear model, nevertheless *dependence* (albeit non-linear) of activity on descriptor values can be observed. (This is merely the statement that the descriptor set must be sufficient to satisfy the Fundamental Assumption of QSAR: that compounds whose chemical structures have similar descriptor values will have similar activity values.) Hybrid *K*-means exhibited promising improvements (for an aqueous solubility dataset of 1000 chemical

compounds) with between four and eight descriptors. Too few descriptors and dependence of activity on chemical descriptors was lost. On the other hand, with too many descriptors, the excessive dimensionality in the regression-wise clustering led to the introduction of instability in the algorithm and eventually to overfitting. Composite modelling therefore emerges as a technique suitable for the capture of complex, localised phenomena described using a very modest number of features.

6. Test Set Extraction Using Clustering

6.1. Overview

When considering a model of some behavioural aspect of entities of a certain type, a test set is a collection of entities (of that type) whose behaviour is known, but which are not used to influence the construction of the model *per se*. A test set is used to validate a model by checking, against the known behaviour of the entities in the test set, the predictions made by the model about the behaviour of those entities.

The purpose of a test set is to validate that the model can accurately predict the behaviour of previously unseen applicable entities. This isolation of its purpose immediately identifies an important interplay between test set validation and the model's domain of applicability: in general, we would by definition not expect a model to make accurate predictions for entities outside its domain. Validation using a test set containing such inapplicable entities would therefore not constitute a fair evaluation of the model because it may lead to a model that is valid (within its domain of applicability) being reported as invalid.

In this chapter we shall consider models that are based on a 'training' dataset of entities whose behaviour is known, sampling the space of entities to which the model will ultimately apply. (Such inductive 'machine learning' modelling is in contrast to other forms of modelling, in which, for example, the model is constructed based on prior human expert knowledge or understanding of the mechanism by which the behaviour of an entity arises.) In such cases a test set will be a separate collection from the training set, not contributing to the model training (machine learning) process.

Training a model in this 'machine learning' sense should (it is hoped) result in a model that *fits* the training data to some high degree: 'resubstituting' the training entities into the trained model should approximately reproduce their known behaviour. The purpose of a test set in this context is to validate that, instead of merely just *fitting* the training data well, the model can accurately *predict* the behaviour of unseen entities. In other words, the test set is used to validate that the model *generalises* beyond sole applicability to the individual entities in the training set.

In this form of model training, the domain of applicability is determined by the content of the training set. (This is in contrast with modelling that is not based on a training set: in expert systems, for example, the domain of applicability is the region of entity space in which all the assumptions made by the human expert hold.) The 'fairness criterion' of test set validation, that proscribed from the test set any entity outside the domain of applicability, can then be rephrased as requiring that each test set entity be *similar* to some entities in the training set.

An 'external' test set is a test set that has been supplied as a separate collection from the training set. In contrast, an 'internal' test set, the subject of this chapter, is one that has been *extracted* from the supplied 'original' training set at the outset of the modelling process. The 'residual training set', comprising only those training entities that have *not* been extracted to the internal test set, thereafter adopts the role of training dataset in forming the basis for actually building the model.

When an internal test set has been extracted, the domain of applicability of the resulting model is determined only by this *residual* training set. There are two important consequences of this fact. Firstly, the domain of applicability will vary, depending on precisely which entities have been extracted to the test set. For any choice of internal test set, whether algorithmic or by manual selection, care must therefore be taken to ensure that the extracted entities remain within the domain determined by the remaining entities.

Secondly, it is strongly desirable for a model's domain of applicability to be as large as it can possibly be, given the originally supplied collection of training entities. The internal test set must be chosen so that, as far as possible, it results in a domain of applicability that is just as large as that which would have arisen if no test set had been extracted.

These consequences are of course related: a depletion of the domain of applicability as a result of test set extraction occurs precisely when extraction of all test entities in the affected area leaves them outside the depleted domain. This issue of applicability, in which we require each test entity to have nearby entities in the residual training set, will be a guiding principle when we develop an algorithm for internal test set extraction.

A concern that is in some sense dual to the applicability issue is that of *coverage*, in which we desire each *training* entity to have nearby representatives in the *test* set. A test set exhibiting poor coverage will give rise to validation that, while not necessarily being unfair, will be incomplete because some regions of the training set remain untested.

Poor test set coverage effectively means that the model will be only partially validated, possibly leading to invalid models being reported as valid (a type II error in detecting invalidity). Although this is a principal concern, higher priority will be given to avoiding the applicability issue, in which 'unfair' validation (using inapplicable test entities) may lead to the more serious problem of valid models being reported as invalid (type I error in detecting invalidity).

These dual concerns of 'applicability' and 'coverage' may be viewed as the test set representing no more and no less than the residual training set, so that they both capture the essence of the original dataset [Golbraikh & Tropsha 2002b].

6.2. Cluster-Based Test Set Extraction

The interrelationship of the concepts of domain of applicability and internal test set was described in §6.1. For any test set extraction algorithm, therefore, it makes sense to specify the distance-to-domain measure in use. In this section, we shall continue to employ the *K*-means cluster-based distance-to-domain measure derived in §4.2, and construct an algorithm for test set extraction that makes use of the same *K*-means clusters.

We seek an algorithm for extracting a test set that satisfies the following criteria:

Target Size: the extracted test set accounts for a specified proportion of the original dataset.

Reproducibility: running the algorithm twice on the same dataset gives identical results.

Applicability: any point in the extracted test set has nearby representatives in the residual training set.

Coverage: any point in the residual training set has nearby representatives in the test set.

The approach we shall take in the algorithm is to work on one cluster at a time, extracting the required proportion of entities from each cluster separately. Amalgamating the individual clusters' extracted portions into a single test set ensures that the 'target size' criterion is satisfied.

Stipulating that the same proportion of entities is extracted from each cluster ensures that, at least at the level of the cluster-based model of the dataset, the test set is evenly distributed over the training set. The residual training set (when clustered with the same partitioning as the original dataset) therefore has much the same cluster-based model as the original: the cluster sizes will have been consistently reduced, and the cluster centroids may have migrated, but statistically speaking their expected position coincides with their original position, and they will in any case still be inside the cluster's original Voronoi cell. The domain of applicability may therefore be taken to be the same, satisfying the 'applicability' criterion.

Similarly, the possibility of failing to meet the 'coverage' criterion is slim: the test set covers the dataset uniformly amongst all the clusters. The worst that can happen is that an individual cluster is only *partially* covered, if all the test points in that cluster happen to be confined to part of it. Even in that statistically unlikely case, a part of the cluster can still only be considered 'not covered' at the fine scale of within-cluster distances. In the broader scale of between-cluster distances, which is more relevant for the dataset as a whole and is precisely the level of detail retained by the cluster-based synoptic model, a cluster may be deemed adequately covered by any selection of test points in it.

Within each cluster, the choice of test entities to extract is made simply by random selection. To ensure that the 'reproducibility' criterion is satisfied, the random element is supplied by pseudo-random numbers generated according to a specified algorithm that is always reseeded to a fixed value at the start of the test set extraction. We shall use the following linear congruential sequence for pseudo-random number generation [Knuth 1981]:

$$r_0 = 0$$
$$r_{i+1} = [1664525 r_i + 1013904223] \text{ (reduced modulo } 2^{32})$$

(6.1)

This is efficient and trivial to implement in 32-bit unsigned integer arithmetic, where the reduction modulo 2^{32} occurs 'for free'. We can then reduce any of these pseudo-random numbers into the range (0, 1, …, C−1) by applying the following calculation:

$$R_C(r) = \left\lfloor \frac{r}{\lfloor (2^{32}-1)/C \rfloor + 1} \right\rfloor \tag{6.2}$$

where the 'floor' brackets in $\lfloor x \rfloor$ mean that x is rounded down to the nearest integer below. This calculation can be performed easily in 32-bit unsigned integer arithmetic. The calculation also ensures that the reduced value $R_C(r)$ depends on the *most* significant bits of (the 32-bit representation of) r, rather than on the *least* significant bits. This is important because it is the most significant bits of terms of the sequence in (6.1) that exhibit pseudo-random behaviour; the n least significant bits are periodic with period at most 2^n.

The full algorithm is as follows:

Algorithm 6.1: Cluster-Based Test Set Extraction

Inputs:

- dataset of N points $\mathbf{x}_n$ in M-dimensional feature space
- number of clusters K
- partitioning π of $\{1, \ldots, N\}$ into K clusters
- target proportion p ($0<p<1$) to extract to test set (corresponds to target size pN)

Procedure:

1. Reset the pseudo-random number generator, so that r_1 (6.1) is the next pseudo-random number to be used.
2. Initialise S to be the empty set, and initialise $u = N$ and $t = pN$.
3. For each (non-empty) cluster C_k, selected in order of increasing size N_k:
    a. Let $T_k = \min \{ \lfloor (t/u)N_k \rfloor, N_k - 1 \}$.
    b. Do T_k times:
        i. Let $j = R_{N_k}(r)$ as defined in (6.2) where r is the next random number in the sequence.

ii. Select the entity $\mathbf{x}_n$ from C_k whose index n is smallest-but-j in C_k, i.e. for which there are exactly j entities $\mathbf{x}_m$ in C_k with $m < n$.

iii. If the index n of the selected entity is already in S then return to step i.

iv. Add n to S.

c. Update $u := u - N_k$ and $t := t - T_k$.

Outputs:

- subset S of $\{1, \ldots, N\}$ indicating the extracted test set

At each iteration, u is the number of entities in all clusters remaining to be treated, and t is the number of entities that still need to be extracted in order to meet the 'target size' criterion. Note that t can never go negative because, in step 3a, the fact that $N_k \leq u$ ensures that $T_k \leq t$.

Algorithm 6.1 aims to achieve the target size of the extracted test set by extracting the same proportion of entities from each cluster. There are two constraints on doing this. Firstly, because the number of entities extracted must be a whole number, some small variation in this proportion will inevitably be discernible across the clusters due to the discretisation, especially for very small clusters. Secondly, also affecting in particular clusters of low cardinality, we expressly forbid any cluster to lose all its members to the test set. This is necessary to satisfy the 'applicability' criterion, which requires that, for any test entities in a cluster, there must be some residual training entities in that cluster in order to maintain the domain of applicability.

During the test set extraction, these constraints may cause some deviation in the actual proportion of entities extracted so far (as a proportion of the members of the clusters thus far treated). Algorithm 6.1 attempts in each iteration (i.e. for each new cluster treated) to put this proportion back on course. It does this by setting out to use the updated proportion t/u that, if followed for all remaining clusters, would achieve the required overall proportion p, thereby putting the algorithm back on track to satisfy the 'target size' criterion.

For these corrections to be most effective, the algorithm processes the clusters in order of increasing size. It is much easier to maintain to the revised proportions t/u when the clusters are large, as that is when the constraints (that the proportion actually

extracted from cluster k must be a whole multiple of $1/N_k$ and must be strictly less than 1) have the least impact. If small clusters were left to the end then there would be little freedom remaining to meet the target size. For example, in a dataset of 150 entities, if the last three clusters to be considered have two elements each, such an attempt to extract two thirds of the entities into the test set may succeed in extracting 96 from 144, thus far meeting the target of two thirds, but then be constrained to extract only one entity from each of the remaining clusters. The resulting test set of 99 entities will have missed its target size.

It was mentioned above that the scale of within-cluster-distances, related to the clusters' radii, determines the resolution at which cluster-based test set extraction preserves the original dataset's distribution in feature space. For an additional angle of comparison, the experiments described in the following subsections will therefore be repeated using a larger number of clusters.

The increase in cluster count K for this repetition is achieved by analysing the original clustering (arising from Algorithm 3.2) and reclustering any of those clusters that exceed an 'elongation' threshold. Anomalous Pattern Clustering is used for this reclustering of 'elongated' clusters, and after all 'elongated' clusters have been thus processed Algorithm 3.1 is reapplied to converge on a K-means optimal partitioning.

The 'elongation' λ_k of a cluster k is defined in this context as the cluster's variance along the axis connecting the dataset's overall centroid $\mathbf{g}$ with the cluster's centroid $\mathbf{c}_k$, expressed as a fraction of the sum of the cluster's variances over any complete set of M orthogonal directions in the M-dimensional feature space. The elongation may be computed according to the following expression, which assumes for ease of presentation that the feature data has been centred such that $\mathbf{g} = \mathbf{0}$:

$$\lambda_k = \frac{\sum_{n:\pi(n)=k} ((\mathbf{x}_n - \mathbf{c}_k)\cdot\mathbf{c}_k)^2}{(\mathbf{c}_k\cdot\mathbf{c}_k)\sum_{n:\pi(n)=k} (\mathbf{x}_n - \mathbf{c}_k)\cdot(\mathbf{x}_n - \mathbf{c}_k)} = \frac{\sum_{n:\pi(n)=k} ((\mathbf{x}_n\cdot\mathbf{c}_k)^2 - (\mathbf{c}_k\cdot\mathbf{c}_k)^2)}{\sum_{n:\pi(n)=k} (\mathbf{x}_n\cdot\mathbf{x}_n\ \mathbf{c}_k\cdot\mathbf{c}_k - (\mathbf{c}_k\cdot\mathbf{c}_k)^2)} \quad \textbf{(6.3)}$$

This elongation λ_k attains its maximum value of unity when all points in the cluster lie on a common line through the dataset's overall centroid $\mathbf{g}$. On the other hand, a cluster

whose variance/covariance tensor is isotropic (i.e. a cluster whose distribution in feature space has no directional bias and appears spherical) has elongation $\lambda_k = 1/M$.

Although, as was discussed in §4.2, *K*-means clusters tend to be approximately spherical, this tendency is not always perfectly borne out. Elongation values substantially greater than $1/M$ are indicative of this case, and we use a threshold criterion of $\lambda_k \geq 2/(1 + M)$ for distinguishing clusters as 'elongated'. (This corresponds to $\lambda_k \geq 2\ (M-1)^{-1}(1 - \lambda_k)$, i.e. the cluster's variance in the $\mathbf{c}_k - \mathbf{g}$ direction being at least double its average variance over the orthogonal directions.)

The complete algorithm is as follows:

Algorithm 6.2: Recluster Elongated Clusters

Inputs:

- dataset of N points $\mathbf{x}_n$ in M-dimensional feature space
- initial number of clusters K
- initial K-means-optimal clustering $(\pi, [\mathbf{c}_k])$ of $[\mathbf{x}_n]$ into K clusters

Procedure:

1. For each cluster k:
    a. Calculate the elongation λ_k according to equation (6.3).
    b. If $\lambda_k \geq 2/(1 + M)$ (i.e. the cluster is 'elongated'):
        i. Apply the Anomalous Pattern Clustering initialisation phase of the Intelligent *K*-Means Algorithm (steps 1-3 of Algorithm 3.2) to the entities in cluster k.
        ii. Add the new clusters resulting from step i to the collection of clusters. Increase K and update the partition function π accordingly.
2. Invoke the Alternating Optimisation algorithm on the full dataset, initialised with (and updating) the current partitioning π into K clusters.

Outputs:

- final number of clusters K
- final K-means-optimal clustering $(\pi, [\mathbf{c}_k])$ of $[\mathbf{x}_n]$ into K clusters

6.3. Measures of Quality of a Test Set

6.3.1. Linkage Measures

A number of different measures will be used to measure the quality of an extracted test set.

Intrinsic measure of test set quality will be provided by so-called 'linkage' measures [Bolshakova & Azuaje 2003] providing measures of how close, in aggregate, the test set and residual training set are to one another in feature space. (These measures are 'intrinsic' in the sense that they do not incorporate activity values or consider any test set validation results. They are defined purely in terms of the test set entities' feature values.) Linkage measures (defined with respect to a common distance measure in feature space) differ over how the aggregation is performed over each of the two sets in question. The specific linkage measures that we shall use are the Hausdorff distances [Preparata & Shamos 1985, Panchenko & Madej 2005, Bolshakova & Azuaje 2003] defined as follows:

Test linkage: the distance in feature space from the worst represented test entity to its nearest representative in the residual training set

Training linkage: the distance in feature space from the worst represented residual training entity to its nearest representative in the test set

Formally, these are defined respectively as follows, for a subset S of $\{1, \ldots, N\}$ indicating a test set extracted from N entities $\mathbf{x}_n$ in M-dimensional feature space U:

$$lk_{\text{test}}(S) = \max_{m \in S} \min_{n \notin S} d(\mathbf{x}_m, \mathbf{x}_n)$$
$$lk_{\text{train}}(S) = \max_{m \notin S} \min_{n \in S} d(\mathbf{x}_m, \mathbf{x}_n) \quad \textbf{(6.4)}$$

These linkage measures correspond closely to the 'applicability' and 'coverage' criteria respectively, that were stipulated in §6.2. 'Applicability' (of each test entity to models trained on the residual training set) demands that each test entity have a nearby representative in the residual training set; the test linkage value states just how close that representative is (in the worst case). Similarly, 'Coverage' (of the residual training set by the test set) requires each residual training entity to have a nearby test entity; the proximity of this representative is measured (in the worst case) by the training linkage.

This correspondence between test linkage and applicability, and between training linkage and coverage, mean that the arguments given in §6.2 in favour of cluster-based test set selection (Algorithm 6.1) also demonstrate the theoretical tendency of the algorithm to pursue fair linkage scores. This can of course be seen directly: ensuring wherever possible that each cluster contains both training and test entities means that, at worst, any test entity has a nearby training representative *inside the same cluster* and vice versa.

6.3.2. Model Validation Measures

In addition to these linkage measures, a separate class of 'extrinsic' measures will be employed. These take into account the known activity values associated with the training and test entities, by training a regression model on the dataset and measuring how well the model fares on the training and test sets individually. These extrinsic measures are therefore more closely related to the intended use of the test set as a validation tool.

Specifically, we shall train the linear least-squares regression model on the residual training set that remains after the test set has been extracted, and consider the root-mean-square error of 'prediction' (or more accurately, in certain cases, training residues) over the various sets listed in Table 6.1.

The first measure, $E_{\text{test}}(S)$, presents the result of the validation experiment for which the test set is ultimately intended. Having first trained the linear least-squares regression model $[\mathbf{a}(S), b(S)]$ on the residual training set $\{ [\mathbf{x}_n, y_n] \ : \ n \notin S \}$, the measure $E_{\text{test}}(S)$ calculates the root-mean-square prediction error in applying this regression model to the extracted test set S. It bears a simple relationship with the 'Prediction Error Sum of Squares' (PRESS) value often associated with such validation experiments:

$$\begin{aligned} E_{\text{test}}(S)^2 &= \frac{1}{|S|} \Sigma_{n \in S} \ (\mathbf{a}(S) \cdot \mathbf{x}_n + b(S) - y_n)^2 \\ &= \frac{PRESS(S)}{|S|} \end{aligned} \qquad \textbf{(6.5)}$$

Secondly, the measure $E_{\text{train}}(S)$ measures how well the linear least-squares regression model fits its training data (the residual training set), and is related to the 'Residual Sum of Squares' (RSS) of the model:

$$E_{\text{train}}(S)^2 = \frac{1}{N-|S|} \Sigma_{n \notin S} \ (\mathbf{a}(S) \cdot \mathbf{x}_n + b(S) - y_n)^2$$
$$= \frac{RSS(S)}{N-|S|}$$

(6.6)

Note that the linear least-squares regression model on this residual training set is by definition precisely that which minimises $RSS(S)$ and hence $E_{\text{train}}(S)$.

The next measure $E_{\text{all}}(S)$ captures the accuracy of the model over the entire dataset (covering both training and test subsets):

$$E_{\text{all}}(S)^2 = \frac{1}{N} \Sigma_n \ (\mathbf{a}(S) \cdot \mathbf{x}_n + b(S) - y_n)^2$$
$$= \frac{RSS(S) + PRESS(S)}{N}$$

RSS Measure	**Model trained over:**	**RMS error calculated over:**	**Sense**	**Purpose**
$E_{\text{test}}(S)$	residual training set	test set	low values are best	measures how well the test set conforms to the linear trend exhibited by the residual training set
$E_{\text{train}}(S)$	residual training set	residual training set	low values are best	measures the strength of linear trend in the residual training set
$E_{\text{all}}(S)$	residual training set	entire dataset	low values are best	compare with E_{orig} to measure the detriment to the model incurred by extracting the test set
E_{orig}	entire dataset	entire dataset	low values are best	measures, as a control case, the strength of linear trend in the original dataset

Table 6.1: Validation-Based Measures of Quality of Extracted Test Set

(6.7)

The reference score E_{orig} is simply the root-mean-square training residue of the linear least-squares regression model trained over the *entire* dataset. It has its own interpretation as the 'control' value of both E_{train} and E_{all} associated with an empty test set:

$$\begin{aligned} E_{\text{orig}}^2 &= \frac{1}{N} \Sigma_n \left(\mathbf{a}(\varnothing) \cdot \mathbf{x}_n + b(\varnothing) - y_n\right)^2 \\ &= \frac{RSS(\varnothing)}{N} \\ &= E_{\text{train}}(\varnothing)^2 = E_{\text{all}}(\varnothing)^2 \end{aligned}$$

(6.8)

When compared with the reference score E_{orig} (associated with no test set having been extracted), observe that the measure $E_{\text{all}}(S)$ calculates the root-mean-square error over the *same* dataset (the original dataset in its entirety) but from prediction using a *different* model (that with the test set excluded from its training data). The difference $E_{\text{all}}(S) - E_{\text{orig}}$ is therefore a measure of *stability*, in the sense that it detects by how much the regression model is perturbed upon exclusion of the test set. (Note that, for any test set S, $E_{\text{all}}(S) \geq E_{\text{orig}}$ by the least-squares-optimality of the linear regression model $[\mathbf{a}(\varnothing), b(\varnothing)]$ over the whole dataset.)

$E_{\text{all}}(S)$ therefore indicates the detriment to the model incurred by excluding the test set entities from the dataset. This detriment is indirectly related to the 'applicability' criterion. Test set entities that bear no similarity to any residual training entities occur precisely when the dataset's domain of applicability has been depleted in those vicinities, due to insufficient training entities remaining. Such cases, corresponding to a tangibly different sampling of feature space by the residual training set, cannot be expected to recover a model that applies as well to the depleted parts of the dataset.

More generally, the comparison between $E_{\text{all}}(S)$ and $E_{\text{orig}} = E_{\text{all}}(\varnothing)$ can be viewed as an attempt to validate that the test set (and hence also the residual training set) preserves the dataset's sampling distribution in feature space, and is not limited to detecting cases involving full depletion of certain localities of feature space. Under the Fundamental Assumption of QSAR – that compounds that are similar in chemical space (as characterised in the feature space of chemical descriptors) have similar activity values, an even subsampling of the dataset that honours its distribution in

chemical space will give rise to a similar model and hence a similar value of $E_{all}(S)$ to E_{orig}. A value of $E_{all}(S)$ that is substantially worse (greater) than E_{orig} is, therefore, subject to the Fundamental Assumption of QSAR, indicative of an uneven sampling in the test set extraction in which the residual training set fails to capture the essence of the original dataset.

6.4. Experimental Results

A descriptor-based linear least-squares regression model for solubility was used, trained on the publicly available Huuskonen dataset of aqueous solubilities of 1026 chemical compounds [Huuskonen 2000]. The IDBS PredictionBase software [IDBS 2007] was used to identify topological [Devillers & Balaban 1999, Todeschini & Consonni 2002] and electrotopological [Kier & Hall 1999] descriptors that are statistically significant in the dependence relationship of activity (solubility) on chemical structure according to stepwise linear least-squares regression. Two sets of descriptors were adopted for comparison, one a subset of the other, corresponding to two different stopping points in the stepwise regression: 12 descriptors gave rise to a coefficient of multiple correlation of $R^2 = 0.750$, while 26 descriptors correspond to the higher value of $R^2 = 0.826$. These are listed in Appendix A.

The Intelligent *K*-means algorithm (Algorithm 3.2) was applied (using the standard distance-wise *K*-means loss function), with the result that 5 clusters (excluding 1 singleton) emerged for the 12 descriptors, and 7 clusters (excluding 2 singletons) arose in the case of 26 descriptors.

In order to investigate how the results depend on the resolution yielded by the scale of within-cluster distances, the experiments that follow were repeated using a larger number of clusters. Algorithm 6.2 was used to boost the number of clusters. Using the method of detection of elongated clusters described in 6.2, reclustering them, and reoptimising the *K*-means clustering, gave rise to the following cluster counts (ignoring singletons in each case):

Number of Descriptors	Original Number of Clusters	Number of Elongated Clusters	Final Number of Clusters
12	5	3	39
26	7	5	47

Table 6.2: Numbers of *K*-Means Clusters, Before and After Reclustering Elongated Clusters

Test sets of various sizes T were extracted from the original dataset (of size N) according to each of the following methods:

Random: T compounds are selected completely at random from the original dataset.

Activity-Based: The original dataset is sorted so as to be indexed in increasing order of activity ($m \geq n \Rightarrow y_m \geq y_n$ and $m \leq n \Rightarrow y_m \leq y_n$), and compounds are selected as regularly as possible from this sorted list to achieve the target size T. Specifically, the extracted test set is given by $S = \{ n : \lfloor s + nT/N \rfloor > \lfloor s + (n-1)T/N \rfloor \}$ where $0 \leq s < 1$ is a random seed.

Cluster-Based: Algorithm 6.1 applied to the original Intelligent *K*-means clustering given by Algorithm 3.2.

Cluster-Based with Reclustering: Algorithm 6.1 applied to the Intelligent *K*-means clustering with its elongated clusters reclustered according to Algorithm 6.2.

For each combination of method, descriptor set, and test set size, ten test sets were extracted, with the random number sequence (6.1) reseeded to the first value only at the start of the run of ten. The results for each combination were averaged over the ten runs; this allows the investigation to incorporate the susceptibility of each method to the chance occurrence of atypical uneven samplings.

The results of calculating the linkage and validation measures for test sets extracted according to these methods are presented in Table 6.3. Test set sizes T of 10%, 25%, and 50% of the dataset size N were studied, alongside the 'control' configuration of $T = 0$.

Several trends were observed in the results in Table 6.3:

- On extracting 10% and using 26 descriptors, the cluster-based test set extraction (Algorithm 6.1) demonstrates an improvement in the E_{test} value. However, this improvement is not mirrored by the linkage scores.
- When extracting larger test set sizes, the improvements are less pronounced.

	Test Set Size	**Random**	**Activity-Based**	**Cluster-Based**	**Cluster-Based with Reclustering**
12 (*26*) Descriptors				5 (*7*) Clusters	39 (*47*) Clusters
E_{test}	**10%**	1.03 (*0.92*)	1.05 (*0.94*)	1.07 (*0.86*)	1.00 (*0.84*)
	25%	1.03 (*0.89*)	1.01 (*0.87*)	1.04 (*0.89*)	1.04 (*0.88*)
	50%	1.05 (*0.90*)	1.03 (*0.89*)	1.06 (*0.89*)	1.04 (*0.87*)
(E_{orig})	**none**	1.02 (*0.85*)	1.02 (*0.85*)	1.02 (*0.85*)	1.02 (*0.85*)
E_{train}	**10%**	1.02 (*0.85*)	1.02 (*0.84*)	1.02 (*0.85*)	1.02 (*0.85*)
	25%	1.02 (*0.85*)	1.03 (*0.85*)	1.02 (*0.85*)	1.02 (*0.85*)
	50%	1.01 (*0.84*)	1.02 (*0.84*)	1.00 (*0.84*)	1.02 (*0.85*)
(E_{orig})	**none**	1.02 (*0.85*)	1.02 (*0.85*)	1.02 (*0.85*)	1.02 (*0.85*)
E_{all}	**10%**	1.02 (*0.86*)	1.02 (*0.86*)	1.02 (*0.85*)	1.02 (*0.85*)
	25%	1.02 (*0.86*)	1.02 (*0.86*)	1.02 (*0.86*)	1.02 (*0.86*)
	50%	1.03 (*0.87*)	1.03 (*0.87*)	1.03 (*0.87*)	1.03 (*0.86*)
lk_{test}	**10%**	0.28 (*1.18*)	0.33 (*1.36*)	0.31 (*1.18*)	0.30 (*0.98*)
	25%	0.30 (*1.39*)	0.31 (*1.27*)	0.35 (*1.18*)	0.32 (*1.26*)
	50%	0.41 (*1.66*)	0.40 (*1.53*)	0.43 (*1.55*)	0.39 (*1.50*)
lk_{train}	**10%**	0.95 (*3.19*)	0.87 (*3.04*)	0.94 (*3.35*)	0.78 (*2.77*)
	25%	0.57 (*2.06*)	0.63 (*2.06*)	0.57 (*2.11*)	0.54 (*1.93*)
	50%	0.40 (*1.52*)	0.41 (*1.55*)	0.40 (*1.63*)	0.40 (*1.51*)

Table 6.3: Extracted Test Set Scores

The entries in this table relate to the solubility dataset of 1026 chemical compounds described in [Huuskonen 2000]. The first entry in each cell relates to the analysis using 12 descriptors, while the second (*parenthesised and italicised*) entry is associated with the 26 descriptor analysis. The *E* measures are the root mean square residuals described in Table 6.1, while the *lk* measures are the Hausdorff distance linkages defined in equation (6.4). All values are averaged over the ten test sets generated for the combination of method, descriptor count, and test set size.

- The E_{train} and E_{all} values are hardly affected by the choice of method, and in all cases remain close to E_{orig}.
- Significant improvements are yielded across the board by cluster-based test set extraction when using the larger number of clusters offered by Algorithm 6.2.

When using the cluster-based test set extraction (Algorithm 6.1), and extracting 10% of the overall dataset into the test set, reasonable results are obtained for the case with 26 descriptors: the model validation E_{test} measure is slightly improved in comparison with the existing methods.

In this instance there is no improvement in the linkage scores. (However, neither are they substantially worse than with the random or activity-based methods.) This ostensible paradox – that a test set extraction with much the same linkage traits (as compared with the random method) appears to give a noticeably better sampling of the original dataset, as judged empirically by validation on the test set – can be explained by the fact that the linkage measures are sensitive to the *worst case* of nearest *single* chemical structure in the complementary set. The apparent evenness of sampling obtained with the cluster-based test set extraction method suggests that on the whole, *most* training structures have *several* nearby test structures, and vice versa, notwithstanding the mediocrity of the linkage scores.

With 12 descriptors, no such improvements in the E_{test} measure offered by the cluster-based methods (at 10%) were observed. The higher dimensionality and intrinsically lower stability of 26-dimensional regression suggest that there is a greater importance in a carefully designed test set extraction (such as is offered by the cluster-based methods) in the 26 descriptor case than there is in the presence of fewer descriptors.

When reclustering the elongated clusters using Algorithm 6.2, there are consistent improvements over Algorithm 6.1 in almost all configurations. Observing that the random selection method is equivalent to the ‘corner case’ of cluster-based extraction with only one cluster (Algorithm 6.1 with $K = 1$), then boosting the number of clusters would indeed be expected to continue the trends, amplifying any improvements that the cluster-based method offers. Increasing the number of clusters serves to refine the level of detail at which the test set’s and residual training set’s distributions are constrained to match those of the original dataset.

The experiments involving 25% and 50% test sets gave rise to more marginal results. Nevertheless, it is reassuring that the cluster-based method has demonstrated its ability to equal the performance of existing method in this case. Note that extracting the test set by random selection will *in a typical case* give results close to the optimum, especially when the test set and residual training sets are both large. It is therefore to be expected that the improvement should be most pronounced at the more extreme cases involving a small test set (e.g. 10%) or a small residual training set (e.g. that complementary to a 90% test set), where an *atypical*, unrepresentative sampling distribution may be more likely to occur by chance.

It is unsurprising that the E_{train} and E_{all} measures are barely affected in the 10% cases. With only 10% of compounds removed from the training set, a substantial impact on the regression model would be indicative of the more serious problem of model instability. With greater proportions, approaching 50%, of compounds being removed to the test set, it is indeed reassuring that E_{train} and E_{all} remain close to E_{orig} (and to one another) for the cluster-based methods (as they did for the random selection and activity-based methods). This is evidence that the 'applicability' criterion of the test set extraction has been satisfied, as the model trained on the residual training set has demonstrably remained very close to the original, implying adequate coverage by residual *training* compounds.

6.5. Conclusions

An 'internal' test set, extracted from the training set of a QSAR study, may be viewed as a subsample of the originally supplied training dataset. In order to allow a fair and complete test of the QSAR model, and to avoid attenuation of the domain of applicability of the trained model, we require this subsampling to be even over the training set and so properly representative of it. In other words, for a test set to be successful, both it and its remainder in the training set must capture the essence of the original dataset.

The problem of internal test set extraction is inherently connected with the concept of domain of applicability, which has a direct influence on the aforementioned 'evenness of sampling' criterion. When extracting the test set, oversampling within a region would lead to that region dropping out the domain of applicability. This compromises not only model quality but also test fairness, because the test chemical compounds in

the oversampled region will now be liable to be outside the domain of the model trained on the remainder.

The *K*-means cluster-based measure of domain of applicability therefore inspires an algorithm for the automatic extraction of an internal test set. This algorithm uses the same cluster-based description of the distribution of the dataset in chemical descriptor space to ensure that the 'evenness of sampling' criterion is met, by ensuring that the sampling rate is uniform across the clusters.

In the case of a small test set of 10% of the original dataset being extracted, experimental validation of this cluster-based test set extraction algorithm demonstrated that it offers immediate gains, in the form of reduced average prediction errors, over the existing methods (including random selection). Furthermore, in no case did the reduction of the training set produce any significant adverse effect on the quality of the trained QSAR model.

This experimentation also revealed that it is the small test sets that stand to benefit from the more intelligent approach to test set extraction. With larger test sets (e.g. 25% and higher), naïve random selection was found to be a sufficient approach, as the law of large numbers [Grimmett & Stirzaker 1982] provides a greater confidence in the stability of the subsampling in those cases.

The *K*-means cluster-based description of the dataset's distribution effectively defines a granularity or 'level of detail' – a scale of interest at which the dataset's shape is considered and below which the data scatter is effectively treated as noise. It was found experimentally that refining this granularity (by boosting the number of *K*-means clusters in regions of descriptor space in which the dataset's distribution does not closely follow the model based on hyperspherical clusters) significantly improves the degree to which the test set evenly samples the original dataset. The improvement was discernible according to several criteria: the Hausdorff test set linkage (measuring retention of applicability domain), the Hausdorff training set linkage (measuring test coverage), and the average prediction error over the test set were all enhanced without detriment to the stability of the QSAR model.

7. Discussion and Conclusions

7.1. New Interpretations of *K*-Means Clustering

Cluster analysis in general, and *K*-means clustering in particular, have been widely used in QSAR studies in the past as an unsupervised classification tool. In this work, however, we have instead applied *K*-means cluster analysis to model the shape of a QSAR dataset's distribution within its feature space of chemical descriptors. This has not only led to some novel practical solutions to three key problems in QSAR, but has also furnished us with some new theoretical insights into *K*-means and fuzzy clustering.

It is unsurprising that cluster analysis is often associated with classification. After all, a cluster may be defined as a group of entities that is cohesive in the sense of sharing some characteristics not exhibited by entities in other clusters; this suggests that a cluster may be identified with the class defined by those characteristics. In other words, the clusters' partitioning of the *dataset* induces a classification rule for partitioning the surrounding *feature space*. Although this cluster-based classification is entirely empirical (rather than being specified *a priori*), its emergence as a natural structure in the data may nevertheless suggest an underlying significance that is worth investigating.

The shift in perspective (on clustering) from classification to the modelling of data distribution arises from the clusters losing their individual significance, in place of which they assume a collective role, working in aggregate to form a (not necessarily unique) cover of the dataset in its feature space. Thus, rather than identifying characteristic *parts* of the dataset, this new perspective is concerned with describing the dataset as a *whole*.

Disregarding the clusters' individual significance in this fashion helps to overcome an issue of *instability* in the *K*-means algorithm – in particular, the algorithm's sensitivity both to subsampling and to the choice of initial partitioning. Indeed, stability is measured in a different way in this case: instead of checking that a similar *partitioning* of the dataset is obtained (as we would do if we were concerned with the individual clusters), it suffices to perform the less stringent check that a similar *region* of chemical space is covered by the collection of clusters. It was shown in §4.3.1

through direct experimentation that *K*-means clustering is indeed stable under 10-fold cross-validation with respect to this weaker stability criterion, even when the underlying dataset lacks a strong underlying cluster structure.

The most immediate purpose served by a description of the shape of a dataset's distribution is to form a criterion for whether an arbitrary point in feature space *belongs* to the region occupied by the dataset. Such a criterion constitutes exactly the test required for whether a predictive model trained on the dataset can make a prediction at that point without having to extrapolate from its training data; in other words, this region of occupation should be taken as the 'domain of applicability' of such a predictive model.

When using clustering to provide this description of the shape of a dataset, the region of occupation of the dataset comprises the union of the regions of occupation of its clusters; correspondingly, the affinity of a new point with the dataset amounts to affinity with any one of the clusters. Note that a *K*-means cluster's region of occupation is *not* the same as its Voronoi cell: the latter (which is potentially unbounded) consists of all points that have a greater claim to membership of *that* cluster than to membership of any other cluster, regardless of whether it actually has any genuine affinity with the cluster in an absolute sense.

These distinct concepts of (relative) *membership* of a cluster's Voronoi cell and (absolute) *affinity* to a cluster were found to occur in Bezdek's formulation of fuzzy membership as an optimisation problem [Bezdek 1981, Bezdek & Pal 1992]. In that formulation, a given point's membership amongst a collection of fuzzy clusters is distributed over all clusters so as to minimise a membership-weighted average of the distances to, or *non-affinities* with, each individual cluster. Historically, this weighted average – the objective under optimisation – has been used only as a device for calculating the fuzzy membership distribution. In this work, however, we have taken the novel approach of imbuing the attained optimal objective value with significance in its own right: being an average distance to (i.e. non-affinity with) the clusters, we interpret it as a measure of collective non-affinity with the dataset as a whole. Furthermore, this offers the new interpretation of the optimal fuzzy membership values as being those which *maximise the affinity* of the given point with the dataset.

We have therefore, in this work (§4), defined the 'cluster-based distance to domain of applicability' of a QSAR model to be this distance to (i.e. non-affinity with) the dataset, that emerges from the fuzzy cluster memberships. In doing so, we are less concerned with the quantified fuzzy membership of the individual clusters; this is analagous to – and consistent with – our collective (as opposed to individualistic) perspective on the clusters.

This cluster-based measure of distance to domain was experimentally assessed in §4.3.2 with reference to a QSAR model. The experiments demonstrated that this measure is successful – indeed, more so than the existing measures such as Mahalanobis distance – in detecting those chemical compounds for which reliability of the model's predictions is compromised due to extrapolation.

The cluster-based model of domain of applicability has a compact representation – an important consideration if it is to augment a QSAR model that is itself representable by a few tens of model parameters. Indeed, the description of the region of chemical descriptor space occupied by the dataset has been distilled down to comprise only the centroids and sizes of the clusters. This resonates with the principle of *K*-means clustering as a data reduction technique in which each point is approximated by its cluster's centroid.

In our cluster-based approach, we are therefore modelling the domain of applicability as a collection of approximately hyperspherical regions. The form of the model makes no prior assumption or constraint on where these hyperspheres lie. It is this non-parametric aspect that gives our approach the freedom to describe datasets of any shape, including non-convex, disconnected, and even multiply connected cases.

These observations illustrate the relationship between the cluster-based model of the domain of applicability and other, existing estimates of the domain. At one extreme, the strongly non-parametric methods of *k* nearest neighbours and Parzen's window [Parzen 1962] can accommodate arbitrary dataset shapes regardless of their convexity or connectivity, retaining the fine-grained detail, but they make no attempt whatsoever at a compact representation. At the other extreme, Euclidean distance relies on the rigid assumption that the dataset's distribution conforms to a very specific shape (a hypersphere). Although admitting an especially compact representation, it does so at

the expense of the freedom to describe adequately the irregular shapes taken by real QSAR datasets.

It was noted that our cluster-based domain of applicability model contains, upon varying the number of clusters, both of these methods as extreme cases. With the entire dataset merged into a single supercluster, our approach reduces to the hypersphere model of the Euclidean distance method. However, as we approach the case in which each point in the dataset resides in its own distinct cluster, the cluster-based distance to dataset is nothing more than an average distance to nearby points, echoing the *k* nearest neighbour measure.

This identifies an interpretation of the number of clusters, as determining the *granularity* or *level of detail* at which the dataset is considered. The relative spatial arrangement of the clusters is retained and constitutes the description of the dataset's shape, while the distribution *within* each cluster is disregarded. The description of the dataset therefore accommodates only those details at the scale of the size of a cluster or greater. This reflects the well-known decomposition of data scatter, associated with *K*-means clustering, into 'explained' between-cluster scatter and 'unexplained' within-cluster scatter; (see equation (3.3)).

In §6, this *granularity* interpretation of *K*-means clustering of a dataset was applied to the problem of automatic extraction of a suitable subset to be held out for testing QSAR models trained on the dataset. It was argued that this test set must be representative of the distribution of the original dataset, thereby capturing the essence of its shape. Constructing a sampling that was uniform *between* the clusters but random *within* them ensures that the resulting test is representative of the original dataset at the level of detail of the clusters. Experiments on QSAR data in §6.4 demonstrated that such cluster-based sampling (uniform amongst the clusters) yields a more representative and less intrusive test set than random selection across the dataset (which corresponds to the coarsest granularity as given by a single supercluster). Furthermore, an algorithm was proposed (Algorithm 6.2) for obtaining a *K*-means clustering with a greater number of clusters – and hence a finer granularity – than that yielded by the Intelligent *K*-means algorithm initialised with Anomalous Pattern Clustering [Mirkin 2005] used hitherto in this work. The experiments showed that this

refinement – of the granularity at which the test set is designed – further enhanced the previously observed improvement in the test set's representative quality.

Interpreting the number – and the spatial size – of *K*-means clusters as a *granularity* has emerged as a common thread uniting the QSAR problems and solutions tackled in this study. In §5, we applied this interpretation to the problem of modelling an activity dependence that lacks a global linear explanation. In exact analogy with the description of domain of applicability developed in §4, we took the approach of constructing a simple parametric description of the structure-activity dependence locally within each cluster, and aggregated these into a description of the dependence over the whole dataset – without making any prior assumption on the global form taken by this dependence. So, this ensemble of simple linear cluster-specific models repeats the compromise between the *k* nearest neighbour method (for regression in this case) and a single naïve global description (here a global linear regression model).

In this segmented approach to modelling a non-linear dependence of activity on chemical descriptors, the *K*-means clusters define the scale at which fluctuations in activity are *explained* by the model instead of being considered as noise. This distinction between 'explanation' and 'noise' in the activity dependence motivated the inclusion of a contribution of the regression-wise *K*-means clustering criterion, in which not only the model but also the location of the cluster itself is chosen to fit the activity data in the region most closely. It was experimentally verified, both using randomly generated mixed-model data (§5.3.1) and using a real QSAR dataset (§**Error! Reference source not found.**), that including this regression-wise contribution in the clustering yields a segmented piecewise linear model that even more closely describes the observed activity dependence.

These experimental results indicate that the regression-wise contribution to the *K*-means clustering has the ability to align the clusters with regions of the dataset that are most amenable to a (local) linear model. This suggests yet another interpretation for the regression-wise *K*-means clusters: that a regression-wise cluster occupies a region in chemical descriptor space in which a distinct (linear) mode of dependence of activity on chemical structure is in effect. This may even further suggest that the cluster corresponds to a distinct underlying chemical or biological *mechanism of activity*. This viewpoint is a return to the more traditional 'classification' perspective

on clustering in which each cluster has its own individual significance. Furthermore, it is an instance of the situation discussed at the start of this chapter in which a cluster (here identified as a mode of linear activity dependence), despite arising through an entirely empirical process of *K*-means optimisation, nevertheless gives us cause to suspect that it may have a deeper physical significance (here as a mechanism of activity) worthy of further investigation.

7.2. Future Work

This work has proposed several new approaches to tackling the QSAR problems of: the estimation of a model's domain of applicability, the description of a structure-activity relationship that does not admit a linear model, and the extraction of a representative test set such that an unbiased model may be trained on the chemical compounds that remain. In addition to the experimental investigation performed in this study, all these new methods have been implemented in the commercially available IDBS PredictionBase software [IDBS 2007], from where they are exposed to wider experimental validation within the QSAR community.

The method of boosting the number of *K*-means clusters (Algorithm 6.2) was developed and used within the discourse on the cluster-based extraction of a test set, as a tool for refining the 'level of detail' retained by the cluster description of the dataset. Having observed in the concluding discussion that this cluster-determined level of detail (granularity) is a common theme shared by all the cluster-based approaches to the QSAR problems studied in this work, it would make sense to investigate the effect that boosting the number of clusters in this fashion would have elsewhere. In particular, future work may incorporate experimentation to investigate whether using a greater number of finer clusters results in a measure of distance to domain of applicability (of a QSAR model) that is better able to discern chemical compounds for which the model is at risk of making unreliable predictions through extrapolation. Another important consideration would be the extent to which instability emerges in this process of refining the granularity of the estimated domain of applicability.

There are a number of possible extensions to the study of segmented local modelling in §5. For example, the measurement of quality of a cluster-based composite model was based purely on its approximation of the training data. Since overfitting was

identified as a particular risk to which this method is susceptible, there would be some merit in a future study of the *predictive* power of these cluster-based composite models, assessed either using internal cross-validation or using a test set extracted at the outset.

Although the investigation of segmented local modelling was restricted to least-squares regression on each cluster (in line with the formulation of regression-wise *K*-means), in principle any modelling technique may be used for the elementary cluster models. In particular, the dimensionality reduction technique of partial least squares, in which the space of chemical descriptors is reduced to a much smaller number of latent variables that express strong (but mutually independent) correlation with activity, is popular in QSAR. This, or a similar technique for reduction in dimensionality such as principal components analysis, may help to overcome the tendency of the segmented local modelling to become overfitted rapidly when the number of training compounds is only a few times the number of chemical descriptors. Further investigation would be required to ascertain this.

In §5.2.2, it was proposed that the hybrid *K*-means criterion (incorporating a specified contribution of the regression-wise criterion) be supplemented by an application of pure distance-wise *K*-means until convergence. This involves an abrupt transition from hybrid to distance-wise *K*-means, in which the regression-wise element is simply switched off. It may be worth investigating whether significantly different results would be obtained by a steady 'cooling' of the regression-wise contribution, continuously reducing its proportion towards zero in the hybrid criterion as the alternating optimisation approaches its final, optimal (with respect to the distance-wise *K*-means criterion) clustering.

Several of the methods developed in this work can be extended to apply to other forms of clustering than *K*-means. For example, the distance to domain of applicability given by equation (4.4) can be reformulated as:

$$D^{(2)}{}_{\mathrm{c}}(\mathbf{x}) \;=\; \frac{A}{\Sigma_k \left(\mathrm{percentile}_{95}\{d_k(\mathbf{x}_n) : \pi(n) = k\} \,/\, d_k(\mathbf{x})\right)} \qquad \textbf{(7.1)}$$

where d_k is the distance-to-cluster measure $d_k(\mathbf{x}) = \|\mathbf{x} - \mathbf{c}_k\|^2$. (Recall that $\pi(n)$ is the cluster to which data point $\mathbf{x}_n$ belongs, and A is a normalisation factor.) In order to

accommodate a different form of clustering, we need only construct an appropriate distance-to-cluster measure specific to that clustering and substitute it for d_k. In particular, this would allow the distance-to-domain measure to be adapted to hierarchical clustering methods, for example Ward divisive clustering [Mirkin 2005], which are more amenable to tuning the number of clusters (and hence, in our applications, the granularity) than *K*-means. In addition, clustering based on a dissimilarity measure, for example the agglomerative methods by single, average, or complete linkage [Murtagh 1983], may be used in the case QSAR studies that consider the chemical structure directly instead of employing chemical descriptors.

Appendix A. Chemical Descriptors

This appendix tabulates the chemical descriptors used in the experimental aqueous solubility data for segmented linear modelling in §**Error! Reference source not found.** and automatic test set extraction in §6.4. The phenol toxicity dataset used to validate the distance-to-domain measure in §4.3.2 supplied its own descriptors tailored to its original study [Aptula *et al* 2005].

Table A.1 lists 11 chemical descriptors, calculated by the IDBS PredictionBase software [IDBS 2007], that are invariant under tautomerism [Vollhardt 1987]. They were selected according to the significance of their correlation with aqueous solubility, using (global) stepwise multivariate linear least-squares regression. They are listed below in decreasing order of significance. Each iteration of the segmented linear modelling experiment in §**Error! Reference source not found.** worked with a feature space of different dimension M (up to 11), based on the first M rows of the table.

Table A.2 and Table A.3 together list those 26 descriptors, calculated by the IDBS PredictionBase software [IDBS 2007], most significant to aqueous solubility according to stepwise multivariate linear least-squares regression, as used in the experimentation in §6.4. For those iterations of the experiment that worked with a 12-dimensional feature space, only the 12 descriptors from Table A.2 (the most significant ones) were used.

Rank	Descriptor	Name
1	$^{1}\chi_{path}$	Kier and Hall connectivity index on 1-bond paths [Todeschini & Consonni 2002]
2	J	Balaban average distance connectivity index [Estrada & Uriarte 2001]
3	$^{3}\chi_{cluster}$	Kier and Hall connectivity index on 3-bond clusters [Todeschini & Consonni 2002]
4	Q'	binormalised quadratic index [Todeschini & Consonni 2002, Balaban 1979]
5	Q	normalised quadratic index [Todeschini & Consonni 2002, Balaban 1979]
6	J_t	Balaban connectivity index [Todeschini & Consonni 2002]
7	M_1	first Zagreb group index [Estrada & Uriarte 2001]
8	M_2	second Zagreb group index [Estrada & Uriarte 2001]
9	N_{HBa}	number of hydrogen-bond acceptors [Todeschini & Consonni 2002]
10	N_{Ab}	number of aromatic bonds [Vollhardt 1987]
11	MW	molecular weight

Table A.1: Chemical Descriptors Used in Segmented Linear Modelling of Aqueous Solubility Data

Rank	Descriptor	Name
1	J_2	mean Galvez topological charge index of order 2 [de Julián-Ortiz *et al* 1998]
2	N_{nonH}	number of non-hydrogen atoms
3	SIC_1	structural information content of neighbourhoods of range 1 [Devillers & Balaban 1999]
4	W_{n}	normalised Wiener index [Estrada & Uriarte 2001]
5	${}^{0}\chi_{\mathrm{atom}}$	Kier and Hall connectivity index on atoms [Todeschini & Consonni 2002]
6	${}^{3}\varepsilon^{\mathrm{W}}{}_{\mathrm{cluster}}$	weighted edge connectivity index on 3-bond clusters [Devillers & Balaban 1999]
7	${}^{\mathrm{e}}B_{\mathrm{H1}}$	highest eigenvalue of Burden matrix weighted by atomic Sanderson electronegativity [Burden 1989]
8	Q'	binormalised quadratic index [Todeschini & Consonni 2002, Balaban 1979]
9	${}^{1}\varepsilon^{\mathrm{w}}{}_{\mathrm{bond}}$	weighted edge connectivity index on bonds [Devillers & Balaban 1999]
10	Q_{v}	Hall Polarity Index [Kier & Hall 1999]
11	${}^{0}\chi^{\mathrm{v}}{}_{\mathrm{atom}}$	Kier and Hall valence-weighted connectivity index on atoms [Todeschini & Consonni 2002]
12	G_2	Galvez topological charge index of order 2 [de Julián-Ortiz *et al* 1998]

Table A.2: Chemical Descriptors Used in Automatic Extraction of a Test Set in 12 Dimensions

Rank	Descriptor	Name
13	W_{rms}	root mean square Wiener index [Todeschini & Consonni 2002, Estrada & Uriarte 2001]
14	${}^{e}B_{L1}$	lowest eigenvalue of Burden matrix weighted by atomic Sanderson electronegativity [Burden 1989]
15	${}^{1}\varepsilon_{bond}$	edge connectivity index on bonds [Devillers & Balaban 1999]
16	${}^{e}B_{L3}$	third-lowest eigenvalue of Burden matrix weighted by atomic Sanderson electronegativity [Burden 1989]
17	N_{HBa}	number of hydrogen-bond acceptors [Todeschini & Consonni 2002]
18	N_{C}	number of carbon atoms
19	N_{Br}	number of bromine atoms
20	ΣBI	sum of bond electrotopological state [Kier & Hall 1999]
21	N_{O}	number of oxygen atoms
22	${}^{4}\chi^{v}_{pc}$	Kier and Hall valence-weighted connectivity index on 4-bond branched paths [Todeschini & Consonni 2002]
23	N_{wHBa}	number of weak hydrogen-bond acceptors [Todeschini & Consonni 2002]
24	E_{min}	minimum electrotopological state [Kier & Hall 1999]
25	${}^{v}B_{H2}$	second highest eigenvalue of Burden matrix weighted by van der Waals volume [Burden 1989]
26	ATS^{e}_{1}	autocorrelation of lag 1 weighted by Sanderson electronegativity [Todeschini & Consonni 2002]

Table A.3: Additional Chemical Descriptors Used in Automatic Extraction of a Test Set in 26 Dimensions

Bibliography

Aptula, A.O., Jeliazkova, N., Schultz, T.W., Cronin, M.T.D. (2005). The Better Predictive Model: High q^2 for the Training Set or Low Root Mean Square Error of Prediction for the Test Set? *QSAR Comb. Sci.* **24**, 385-396. ..... 9, 15, 16, 62, 67, 116

Balaban, A.T. (1979). Chemical Graphs XXXIV. Five New Topological Indices for the Branching of Tree-Like Graphs, *Theor. Chim. Acta.* **53**, 355-375. ..... 117, 118

Banfield, J.D., Raftery, A.E. (1993). Model-Based Gaussian and Non-Gaussian Clustering, *Biometrics* **49**, 803-821. 53

Benigni, R. (Ed.) (2003). *Models of Mutagens and Carcinogens*, CRC Press, Boca Raton, Florida. 9

Bezdek, J.C. (1973). Cluster Validity with Fuzzy State, *J. Cybernetics* **3**, 3, 58-73. 44, 52

Bezdek, J.C. (1981). *Pattern Recognition with Fuzzy Objective Algorithms*, Plenum Press, New York. 44, 45, 109

Bezdek, J.C., Pal, S.K. (Eds.) (1992). *Fuzzy Models for Pattern Recognition*, IEEE Press, New York. 44, 109

Bock, H.-H. (2007). Clustering Methods: A History of *k*-Means Algorithms, *Selected Contributions in Data Analysis and Classification*, Springer-Verlag, Berlin, 161-172. ..... 33, 39

Bolshakova, N., Azuaje, F. (2003). Cluster Validation Techniques for Genome Expression Data, *Signal Process.* **83**, 825-833. 98

Burden, F.R. (1989). Molecular identification number for substructure searches, *J. Chem. Inf. Comput.. Sci.* **29**, 227-229. ..... 118, 119

Burden, F.R., Winkler, D.A. (1999). Robust QSAR Models Using Bayesian Regularised Artificial Neural Networks, *J. Med. Chem.* **42**, 3183-3187. 31

Butina, D., Gola, J.M.R. (2003). Modeling Aqueous Solubility, *J. Chem. Inf. Comput. Sci.* **43**, 837-841. ..... 9, 15, 24, 30

Cedeño, W., Agrafiotis, D.K. (2004). Combining particle swarms and *K*-nearest neighbors for the development of quantitative structure-activity relationships, *Biocomputing*, Novo Science, New York, 43-53. 30

Cherkasov, A. (2005). Inductive QSAR Descriptors. Distinguishing Compounds with Antibacterial Activity by Artificial Neural Networks, *Int. J. Mol. Sci.* **6**, 63-86. ... 30

Chissom, B.S. (1970). Interpretation of the Kurtosis Statistic, *Am. Stat.* **24**, 19-23. .. 62

Clark, R.D. (1997). OptiSim: An Extended Dissimilarity Selection Method for Finding Diverse Representative Subsets, *J. Chem. Inf. Comput. Sci.* **37**, 1181-1188. 29

Clark, R.D., Kar, J., Akella, L., Soltanshahi, F. (2003). OptDesign: Extending Optimizable *k*-Dissimilarity Selection to Combinatorial Library Design, *J. Chem. Inf. Comput. Sci.* **43**, 829-836. ..... 24, 29

Courant, R., Hilbert, D. (1962). *Mathematical Physics, Volume 2*, Interscience, New York. 48

Crettaz, P., Benigni, R. (2005). Prediction of the Rodent Carcinogenicity of 60 Pesticides by the DEREKfW Expert System, *J. Chem. Inf. Model.* **45**, 1864-1873. 15

de Julián-Ortiz, J.V., de Gregorio Alapont, C., Ríos-Santamarina, I., García-Doménech, R., Gálvez, J. (1998). Prediction of properties of chiral compounds by molecular topology, *J. Mol. Graph. Model.* **16**, 14-18. 118

Devillers, J., Balaban, A.T. (Eds.) (1999). *Topological Indices and Related Descriptors in QSAR and QSPR*, Gordon and Breach Science Publishers, Amsterdam. 62, 84, 102, 118, 119

Dhillon, I., Guan, Y., Kulis, B. (2004). Kernel *k*-means, Spectral Clustering, and Normalized Cuts, *Proc. 10th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*..... 48

Diday, E. (1974). Optimisation in non-hierarchical clustering, *Pattern Recognition* **6**, 17-33. 49

Diday, E. et collaborateurs (1979). *Optimisation en Classification Automatique*, INRIA, Le Chesnay, France. 49

Diday, E., Celeux, G., Govaert, G., Lechevallier, Y., Ralambondrainy, H. (1989). *Classification Automatique des Données*, Dunod, Paris. 49, 80

Diudea, M.V. (Ed.) (2000). QSPR/QSAR Studies by Molecular Descriptors, Nova Science Publishers Inc., New York. 10

Djorgovski, S.G., Brunner, R., Mahabal, A., Williams, R., Granat, R., Stolorz, P. (2002). Challenges for Cluster Analysis in a Virtual Observatory, *Statistical Challenges in Modern Astronomy III*, Springer Verlag, New York, 125-135. 68

Duchowicz, P.R., Garro, J.C.M., Andrada, M.F., Castro, E.A., Fernández, F.M. (2007). QSPR Modeling of Heats of Combustion for Carboxylic Acids, *QSAR Comb. Sci.* **26**, 647-652. 82

Duda, R.O., Hart, P.E. (1973). *Pattern Classification and Scene Analysis*, Wiley, New York. 36, 38

Dunn, J.C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, *J. Cybernetics* **3**, 3, 32-57... 45, 46, 52

Eriksson, L., Jaworska, J., Worth, A.P., Cronin, M.T.D., McDowell, R.M. (2003). Methods for Reliability and Uncertainty Assessment and for Applicability Evaluations of Classification- and Regression-Based QSARs, *Environ. Health Perspect*. **111**, 1361-1375. 22

Estrada, E., Uriarte, E. (2001). Recent Advances on the Role of Topological Indices in Drug Discovery Research, *Curr. Med. Chem.* **8**, 1573-1588. ..... 15, 117, 118, 119

Feher, M., Schmidt, J.M. (2003). Fuzzy Clustering as a Means of Selecting Representative Conformers and Molecular Alignments, *J. Chem. Inf. Comput. Sci.* **43,** 810-818. 32

Fernández Pierna, J.A., Wahl, F., de Noord, O.E., Massart, D.L. (2002). Methods for Outlier Detection in Prediction, *Chemom. Intell. Lab. Syst.*, **63**, 27-39. 19

Galil, Z., Kiefer, J. (1980). Time- and space-saving computer methods, related to Mitchell's DETMAX, for finding D-optimum designs, *Technometrics* **22**, 301-313. 28

Ghose, A.K. & Crippen, G.M. (1986). Atomic Physicochemical Parameters for Three-Dimensional Structure-Directed Quantitative Structure-Activity Relationships I. Partition Coefficients as a Measure of Hydrophobicity, *J. Comput. Chem.* **7**, 565-577. ..... 15, 62

Godden, J.W., Stahura, F.L., Bajorath, J. (2000). Variability of Molecular Descriptors in Compound Databases Revealed by Shannon Entropy Calculations, *J. Chem. Inf. Comput. Sci.* **40**, 796-800. 62

Golbraikh, A., Tropsha, A. (2002a). Beware of q^2! *J. Mol. Graph. Model.* **20**, 269-276. 24

Golbraikh, A., Tropsha, A. (2002b). Predictive QSAR modeling based on diversity sampling of experimental datasets for the training and test set selection, *J. Comput. Aid. Mol. Des.* **16**, 357-369..... 27, 28, 92

González, M.P., Dias, L.C., Helguera, A.M., Rodríguez, Y.M., de Oliveira, L.G., Gomez, L.T., Diaz, H.G. (2004). TOPS-MODE based QSARs derived from heterogeneous series of compounds. Applications to the design of new anti-inflammatory compounds, *Bioorgan. Med. Chem.* **12**, 4467-4475. 31

Gramatica, P. (2007). Principles of QSAR model validation: internal and external, *QSAR Comb. Sci.* **26**, 694-701..... 11, 16, 24

Grimmett, G.R., Stirzaker, D.R. (1982). *Probability and Random Processes*, Oxford University Press, Oxford. 107

Hartigan, J.A. (1975). *Clustering Algorithms*, Wiley, New York..... 34, 36, 56

Hathaway, R., Bezdek, J. (1993). Switching regression models and fuzzy clustering, *IEEE Trans. Fuzzy Syst.* **1**, 98-110. ..... 49, 51

Hawkins, D.M. (2004). The Problem of Overfitting, *J. Chem. Inf. Comput. Sci.* **44**, 1-12. ..... 11, 82

Hawkins, D.M., Basak, S.C., Mills, D. (2003). Assessing Model Fit by Cross-Validation, *J. Chem. Inf. Comput. Sci.* **43**, 579-586. 28

Huuskonen, J. (2000). Estimation of Aqueous Solubility for a Diverse Set of Organic Compounds Based on Molecular Topology, *J. Chem. Inf. Comput. Sci.* **40**, 773-777. ..... 16, 17, 58, 84, 102, 104

ID Business Solutions (2007). *PredictionBase*, http://www.idbs.com/PredictionBase/. 30, 62, 84, 102, 113, 116

Jain, A.K., Dubes, R.C. (1988). *Algorithms for Clustering Data*, Prentice Hall, New Jersey. 34, 38, 42, 54, 55

Jaworska, J., Nikolova-Jeliazkova, N., Aldenberg, T. (2005). QSAR Applicability Domain Estimation by Projection of the Training Set into Descriptor Space: A Review, *ATLA* **33**, 445-459. ..... 10, 16, 17, 18, 19, 22, 23, 24

Jaworska, J.S., Comber, M., Auer, C., Van Leeuwen, C.J. (2003). Summary of a Workshop on Regulatory Acceptance of (Q)SARs for Human Health and Environmental Endpoints, *Environ. Health Persp.* **111**, 1358-1360..... 11, 26

Jiang, C., Li, Y., Tian, Q., You, T. (2003). QSAR Study of Catalytic Asymmetric Reactions with Topological Indices, *J. Chem. Inf. Comput. Sci.* **43**, 1876-1881.... 82

Kanungo, T., Mount, D.M., Netanyahu, N.S., Piatko, C., Silverman, R., Wu, A.Y. (2000). An Efficient *k*-Means Clustering Algorithm: Analysis and Implementation, *Proc. 16th ACM Symp. on Computational Geometry*. 38

Kier, L.B., Hall, L.H. (1999). *Molecular Structure Description: The Electrotopological State*, Academic Press, California. 102, 118, 119

Knuth, D.E. (1981). *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*, 2nd Edition, Addison Wesley Professional, Boston. 93

Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Morgan Kaufmann, San Mateo, California, 1137-1143. 63

Kolossov, E., Stanforth, R.W. (2007). The quality of QSAR models: problems and solutions, *SAR QSAR Environ. Res.* **18**, 89-100. ..... 24, 26

Kovatcheva, A., Golbraikh, A., Oloff, S., Xiao, Y-D., Zheng, W., Wolschann, P., Buchbauer, G., Tropsha, A. (2004). Combinatorial QSAR of Ambergris Fragrance Compounds, *J. Chem. Inf. Comput. Sci.* **44**, 582-595. 28

Lind, P., Maltseva, T. (2003). Support Vector Machines for the Estimation of Aqueous Solubility, *J. Chem. Inf. Comput. Sci.* **43**, 1855-1859. ..... 15, 16, 30

Lloyd, S.P. (1982). Least Squares Quantization in PCM, *IEEE Trans. Inf. Theory* **28**, 129-137. 33

MacQueen, J.B. (1967). Some Methods for Classification and Analysis of Multivariate Observations, *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, Vol. 1, 281-297. 33

Mannhold, R., van de Waterbeemd, H. (2001). Substructure and whole molecule approaches for calculating log*P*, *J. Comput. Aid. Mol. Des.* **15**, 337-354. ... 9, 15, 29

Manning, C.D., Raghavan, P., Schütze, H. (2008). *Introduction to Information Retrieval*, Cambridge University Press, in preparation. 38

McKinney, J.D., Richard, A., Waller, C., Newman, M.C., Gerberick, F. (2000). The Practice of Structure Activity Relationships (SAR) in Toxicology,*Toxicological Sciences* **56**, 8-17. ..... 9, 15

Mercer, J. (1909). Functions of positive and negative type and their connection with the theory of integral equations, *Philos. Trans. Roy. Soc. London Ser. A* **209**, 415-446. 48

Mirkin, B. (2005). *Clustering for Data Mining: A Data Recovery Approach*, Chapman & Hall/CRC, London. 34, 35, 39, 44, 62, 111, 115

Molconn-Z (2006). *Molconn-Z Developers' Toolkit*, http://www.edusoft-lc.com/molconn/. 82

Murtagh, F. (1983). A Survey of Recent Advances in Hierarchical Clustering Algorithms, *Comput. J.* **4**, 354-359. 115

Murtagh, F. (2000). Clustering in Massive Data Sets, *Handbook of Massive Data Sets*, Kluwer, Norwell, 501-543. 24

Murtagh, F., Raftery, A.E. (1984). Fitting straight lines to point patterns, *Pattern Recognition* **17**, 479-483. ..... 53, 88

Nascimento, S. (2005). *Fuzzy Clustering Via Proportional Membership Model*, IOS Press, Amsterdam. 45

Nascimento, S., Mirkin, B., Moura-Pires, F. (2003). Modeling Proportional Membership in Fuzzy Clustering, *IEEE Trans. Fuzzy Syst.* **11**, 173-186. 45

Panchenko, A.R. & Madej, T. (2005). Structural Similarity of Loops in Protein Families: Toward the Understanding of Protein Evolution, *BMC Evol. Biol.* **5**, 10. 98

Parzen, E. (1962). On estimation of a probability density function and mode, *Ann. Math. Stat.* **33**, 1065-1076. ..... 23, 110

Preparata, F.P., Shamos, M.I. (1985). *Computational Geometry*, Springer-Verlag, New York. 19, 36, 98

Roy, K., Sanyal, I., Ghosh, G. (2007). QSPR of *n*-Octanol/Water Partition Coefficient of Nonionic Organic Compounds Using Extended Topochemical Atom (ETA) Indices, *QSAR Comb. Sci.* **26**, 629-646. ..... 9, 15, 62

Ruspini, E.H. (1969). A New Approach to Clustering, *Inform. Control* **15**, 22-32. ... 43

Samanta, S., Debnath, B., Basu, A., Gayen, S., Srikanth, K., Jha, T. (2006). Exploring QSAR on 3-aminopyrazoles as antitumor agents for their inhibitory activity of CDK2/cyclin A, *Eur. J. Med. Chem.* **41**, 1190-1195. ..... 9, 31

Schölkopf, B., Smola, A., Müller, K.-R. (1998). Non-linear component analysis as a kernel eigenvalue problem, *Neural Computation* **10**, 1299-1319. 49

Senese, C.L., Hopfinger, A.J. (2003). A Simple Clustering Technique To Improve QSAR Model Selection and Predictivity: Application to a Receptor Independent 4D-QSAR Analysis of Cyclic Urea Derived Inhibitors of HIV-1 Protease, *J. Chem. Inf. Comput. Sci.* **43**, 2180-2193. ..... 9, 31

Sheridan, R.P., Feuston, B.P., Maiorov, V.N., Kearsley, S.K. (2004). Similarity to Molecules in the Training Set is a Good Discriminator for Prediction Accuracy in QSAR, *J. Chem. Inf. Comput. Sci.* **44**, 1912-1928. ..... 16, 17, 22

Smellie, A. (2004). Accelerated *K*-Means Clustering in Metric Spaces, *J. Chem. Inf. Comput. Sci.* **44**, 1929-1935. 31

Späth, H. (1979). Algorithm 39. Clusterwise linear regression, *Computing* **22**, 367-373. ..... 49, 50

Späth, H. (1985). *Cluster Dissection and Analysis: Theory, FORTRAN Programs, Examples*, Ellis Horwood, London. 39

Stanforth, R.W., Kolossov, E., Mirkin, B. (2005). A Cluster-Based Measure of Domain of Applicability of a QSAR Model, *Proceedings of the 2005 UK Workshop on Computational Intelligence*, 176-181. ..... 16, 62

Stanforth, R.W., Kolossov, E., Mirkin, B. (2007a). A Measure of Domain of Applicability for QSAR Modelling Based on Intelligent *K*-Means Clustering, *QSAR Comb. Sci.* **26**, 837-844. ..... 16, 62

Stanforth, R.W., Kolossov, E., Mirkin, B. (2007b). Hybrid *K*-Means: Combining Regression-Wise and Centroid-Based Criteria for QSAR, *Selected Contributions in Data Analysis and Classification*, Springer-Verlag, Berlin, 225-233. 77

Steinley, D. (2006). *K*-means clustering: A half-century synthesis, Brit. J. Math. Stat. Psy. **59**, 1-34. ..... 34, 39, 52

Tabachnik, B.G., Fidell, L.S. (2006). *Using Multivariate Statistics (5th Edition)*, Allyn & Bacon, Boston. 51

Talete S.r.l. (2007). *Dragon*, http://www.talete.mi.it/dragon_exp.htm. 82

Todeschini, R. & Consonni, V. (2002). *Handbook of Molecular Descriptors*, Wiley-VCH, Weinheim, Germany. 15, 17, 21, 23, 58, 60, 84, 102, 117, 118, 119

Toropov, A.A., Bakhtiyor, F.R., Leszczynski, J. (2007). QSAR Modeling of Acute Toxicity for Nitrobenzene Derivatives Towards Rats: Comparative Analysis by MLRA and Optimal Descriptors, *QSAR Comb. Sci.* **26**, 686-693. ..... 9, 82

Tropsha, A., Gramatica, P., Gombar, V.K.. (2003). The Importance of Being Earnest: Validation is the Absolute Essential for Successful Application and Interpretation of QSPR Models, *QSAR Comb. Sci.* **22**, 69-77. ..... 11, 22, 24, 63, 64

Vanyúr, R., Héberger, K., Jakus, J. (2003). Prediction of Anti-HIV-1 Activity of a Series of Tetrapyrrole Molecules, *J. Chem. Inf. Comput. Sci.* **43**, 1829-1836. 16

Vapnik, V.N. (1995). *The Nature of Statistical Learning Theory*, Springer, New York. 16, 49

Varnek, A., Fourches, D., Solov'ev, V.P., Baulin, V.E., Turanov, A.N., Karandashev, V.K., Fara, D., Katritzky, A.R. (2004). In Silico Design of New Uranyl Extractants Based on Phosphoryl-Containing Podands: OSPR Studies, Generation and

Screening of Virtual Combinatorial Library, and Experimental Tests, *J. Chem. Inf. Comput. Sci.* **44**, 1365-1382. 82

Vollhardt, K.P.C. (1987). *Organic Chemistry*, W.H. Freeman and Company, New York. 83, 116, 117

Voronoi, G. (1908). Nouvelles applications des paramètres continus à la théorie des formes quadratiques, deuxième memoire, recherche sur les parallelοèdres primitifs, *Journal für die Reine und Angewandte Mathematik* **134**, 198-287. 36

Wold, S., Sjöström, M., Andersson, P.M., Linusson, A., Edman, M., Lundstedt, T., Nordén, B., Sandberg, M., Uppgård, L. (2000). Multivariate Design and Modelling in QSAR, Combinatorial Chemistry, and Bioinformatics, *Molecular Modeling and Prediction of Bioactivity*, Kluwer Academic / Plenum Publishers, New York. 12, 15

Xing, L., Glen, R.C., Clark, R.D. (2003). Predicting pK_a by Molecular Tree Structured Fingerprints and PLS, *J. Chem. Inf. Comput. Sci.* **43**, 870-879. ..... 9, 12

Xu, H., Agrafiotis, D.K. (2003). Nearest Neighbour Search in General Metric Spaces Using a Tree Data Structure with a Simple Heuristic, *J. Chem. Inf. Comput. Sci.* **43**, 1933-1941. 22

Yan, A., Gasteiger, J. (2003). Prediction of Aqueous Solubility of Organic Compounds Based on a 3D Structure Representation, *J. Chem. Inf. Comput. Sci.* **43**, 429-434. ..... 9, 15, 29

Zernov, V.V., Balakin, K.V., Ivaschenko, A.A., Savchuk, N.P., Pletnev, I.V. (2003). Drug Discovery Using Support Vector Machines. The Case Studies of Drug-likeness, Agrochemical-likeness, and Enzyme Inhibition Predictions, *J. Chem. Inf. Comput. Sci.* **43**, 2048-2056. ..... 9, 16