JOHANNES GUTENBERG
UNIVERSITÄT MAINZ

VISUALIZATION AND VALIDATION OF (Q)SAR MODELS

Dissertation

to obtain the academic degree Dissertation of Science submitted
to the Faculty Physics, Mathematics and Computer Science
of the Johannes Gutenberg-University Mainz

on 6. July 2015 by

Martin Gütlein

born in Bamberg

Submission date: 6. July 2015
PhD defense: 22. July 2015

First reviewer:

Second reviewer:

ABSTRACT

Analyzing and modeling relationships between the structure of chemical compounds, their physico-chemical properties, and biological or toxic effects in chemical datasets is a challenging task for scientific researchers in the field of cheminformatics. Therefore, (Q)SAR model validation is essential to ensure future model predictivity on unseen compounds. Proper validation is also one of the requirements of regulatory authorities in order to approve its use in real-world scenarios as an alternative testing method. However, at the same time, the question of how to validate a (Q)SAR model is still under discussion. In this work, we empirically compare a k-fold cross-validation with external test set validation. The introduced workflow allows to apply the built and validated models to large amounts of unseen data, and to compare the performance of the different validation approaches. Our experimental results indicate that cross-validation produces (Q)SAR models with higher predictivity than external test set validation and reduces the variance of the results.

Statistical validation is important to evaluate the performance of (Q)SAR models, but does not support the user in better understanding the properties of the model or the underlying correlations. We present the 3D molecular viewer CheS-Mapper (Chemical Space Mapper) that arranges compounds in 3D space, such that their spatial proximity reflects their similarity. The user can indirectly determine similarity, by selecting which features to employ in the process. The tool can use and calculate different kinds of features, like structural fragments as well as quantitative chemical descriptors. Comprehensive functionalities including clustering, alignment of compounds according to their 3D structure, and feature highlighting aid the chemist to better understand patterns and regularities and relate the observations to established scientific knowledge.

Even though visualization tools for analyzing (Q)SAR information in small molecule datasets exist, integrated visualization methods that allows for the investigation of model validation results are still lacking. We propose visual validation, as an approach for the graphical inspection of (Q)SAR model validation results. New functionalities in CheS-Mapper 2.0 facilitate the analysis of (Q)SAR information and allow the visual validation of (Q)SAR models. The tool enables the comparison of model predictions to the actual activity in feature space. Our approach reveals if the endpoint is modeled too specific or too generic and highlights common properties of misclassified compounds. Moreover, the researcher can use CheS-Mapper to

inspect how the (Q)SAR model predicts activity cliffs. The CheS-Mapper software is freely available at `http://ches-mapper.org`.

ZUSAMMENFASSUNG

Zusammenhänge zwischen der Struktur von chemischen Verbindungen und biologischen oder toxischen Effekten zu analysieren und zu modellieren ist eine wissenschaftliche Herausforderung im Bereich der Chemieinformatik. Deshalb ist die sorgfältige Validierung von (Q)SAR Modellen entscheidend um die Vorhersage-Genauigkeit eines Modells bei ungesehenen Verbindungen zu gewährleisten. Ordnungsgemäße Validierung ist auch eine der Voraussetzungen der Regulierungsbehörden, um den Einsatz von (Q)SAR Modellen als alternative Test-Methode von Chemikalien zu genehmigen. Allerdings wird immer noch aktiv diskutiert, welches die korrekte Validierungsmethode von (Q)SAR Modellen ist. Diese Arbeit vergleicht empirisch k-fache Kreuzvalidierung mit einer externen Validierung anhand eines Test-Datensatzes. Mit der vorgestellten Methodik werden die validierten Modelle auf große Mengen ungesehener Verbindungen angewendet, und die Genauigkeit der verschiedenen Validierungsmethoden verglichen. Unsere experimentellen Ergebnisse legen nahe, dass kreuzvalidierte (Q)SAR Modelle eine höhere Vorhersage-Genauigkeit aufweisen, als solche, die mit einem externen Testdatensatz validiert worden sind. Des weiteren ist die Varianz der Kreuzvalidierung geringer.

Statistische Validierung ist zwingend notwendig, um die Vorhersage-Genauigkeit von (Q)SAR Modellen zu ermitteln. Diese Validierung ist aber nur eingeschränkt hilfreich um die Eigenschaften des Modells oder der zugrunde liegenden Beziehungen zu verstehen. In diesem Zusammenhang stellen wir den molekularen 3D-Viewer CheS-Mapper (Chemical Space Mapper) vor. Diese Computer-Anwendung ordnet chemische Verbindungen im 3D-Raum an, so dass die räumliche Distanz die Ähnlichkeit der Verbindungen widerspiegelt. Durch die Wahl der chemischen Deskriptoren kann der Benutzer die Ähnlichkeit festlegen. CheS-Mapper kann diverse Deskriptoren-Typen, zum Beispiel strukturelle Fragmente oder numerische Kennzahlen, berechnen. Des weiteren erlaubt CheS-Mapper das Clustern von Verbindungen, das Ausrichten und Übereinanderlegen der Strukturen im 3D-Raum, wie auch die farbliche Hervorhebung von Verbindungen anhand ihrer Deskriptor-Werte. Das Programm erleichtert es daher, Chemikern Muster und Zusammenhänge in den Daten zu erkennen und bekanntes wissenschaftliches Wissen zu veranschaulichen.

Zwar existieren bereits einige Visualisierungs-Werkzeuge für (Q)SAR Informationen in chemischen Datensätzen, allerdings fehlt eine ganzheitliche Visualisierungs-

Methode für Validierungs-Ergebnisse. Wir präsentieren *visuelle Validierung*, eine graphische Analyse-Methode für die Validierung eines (Q)SAR Modells mit Hilfe neuer Funktionen in CheS-Mapper 2.0. Vorhergesagte Werte für die Aktivität chemischer Verbindungen können mit tatsächlichen Aktivitäten durch die Visualisierung im 3D-Raum verglichen werden. Unser Ansatz zeigt, ob der Endpunkt zu generisch oder zu spezifisch modelliert wurde, und hebt gemeinsame Eigenschaften von falsch vorhergesagten Verbindungen hervor. Darüber hinaus können Forscher untersuchen, wie *Activity Cliffs* von einem Modell vorhergesagt werden. Die CheS-Mapper Software ist frei verfügbar unter `http://ches-mapper.org`.

ACKNOWLEDGEMENTS

Acknowledgements have been removed from the online version.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AD	Applicability domain
BBRC	Backbone Refinement Classes
CAS	Chemical Abstracts Service
CDK	Chemistry Development Kit
CheS-Mapper	Chemical Space Mapper
ECFPs	Extended-Connectivity Fingerprints
EPA	(US) Environmental Protection Agency
GUI	Graphical User Interface
IC_{50}	Half maximal (50%) inhibitory concentration
InChI	International Chemical Identifier
IUPAC	International Union of Pure and Applied Chemistry
k-NN	k-Nearest Neighbor
LC_{50}	Half (50%) lethal concentration
LOO-CV	Leave-one-out cross-validation
MCS	Maximum common sub-graph
OECD	Organisation for Economic Co-operation and Development
(Q)SAR	(Quantitative) structure activity relationship
(Q)SPR	(Quantitative) structure property relationship
PCA	Principal Components Analyis
REACH	Registration, Evaluation, Authorisation and Restriction of Chemicals
RDF	Resource Description Framework
SDF	Structure-data file
SMACOF	Multidimensional Scaling Using Majorization
SMARTS	Smiles arbitrary target specification
SMILES	Simplified molecular-input line-entry system
SOMs	Self-organizing maps

LIST OF PAPERS

MAIN PAPERS FOR THIS THESIS

- M Gütlein, A Karwath, and S Kramer. CheS-Mapper - Chemical space mapping and visualization in 3D. *Journal of Cheminformatics*, 4(1):7, 2012. PMID: 22424447, Impact Factor 2012: 3.59
- M Gütlein, C Helma, A Karwath, and S Kramer. A large-scale empirical evaluation of cross-validation and external test set validation in (Q)SAR. *Molecular Informatics*, 32(5-6):516–528, 2013, Impact Factor 2013: 4.07.
- M Gütlein, A Karwath, and S Kramer. CheS-Mapper 2.0 for Visual Validation of (Q)SAR models. *Journal of Cheminformatics*, 6:41, 2014. doi: 10.1186/s13321-014-0041-7, Impact Factor 2014: 4.54

OTHER PAPERS

- M Gütlein, E Frank, M Hall, and A Karwath. Large-scale attribute selection using wrappers. In *IEEE Symposium on Computational Intelligence and Data Mining, 2009. CIDM '09*, pages 332–339, 2009.
- B Hardy, N Douglas, C Helma, M Rautenberg, N Jeliazkova, V Jeliazkov, I Nikolova, R Benigni, O Tcheremenskaia, S Kramer, T Girschick, F Buchwald, J Wicker, A Karwath, M Gütlein, A Maunz, H Sarimveis, G Melagraki, A Afantitis, P Sopasakis, D Gallagher, V Poroikov, D Filimonov, A Zakharov, A Lagunin, T Gloriozova, S Novikov, N Skvortsova, D Druzhilovsky, S Chawla, I Ghosh, S Ray, H Patel, and S Escher. Collaborative development of predictive toxicology applications. *Journal of Cheminformatics*, 2(1):7, August 2010. PMID: 20807436.
- A Maunz, M Gütlein, M Rautenberg, D Vorgrimmler, D Gebele, and C Helma. lazar: a modular predictive toxicology framework. *Predictive Toxicology*, 4:38, 2013.

1 Introduction

(Quantitative) structure activity relationship ((Q)SAR) models predict the biological or chemical activity of small molecules. (Q)SAR modeling is a computer based (i.e., *in-silico*) approach that has the advantage of usually being much cheaper and faster than experimental *in-vitro* or *in-vivo* studies [2]. Therefore, (Q)SARs could play an important role as an alternative testing method and aid reducing the need for animal testing. At the same time, (Q)SAR modeling is still facing many challenges, e.g., the lack of sufficient data of good quality [3]. For creating predictive machine learning models, which is the most common approach to (Q)SAR modeling, a consistent dataset of sufficient size is required [4]. (Q)SAR datasets are commonly not only relatively small, but often also noisy, as the activity values have been measured in experiments that have a high experimental error. Moreover, the size of the chemical space is vast and various modes of action exist to cause compound activities, e.g. carcinogenicity. This makes learning demanding and limits the applicability of (Q)SAR models [5]. Additionally, the often incomprehensive reasoning of (Q)SAR models predictions makes them a less attractive tool for many researchers. Another issue, that is still under discussion in the (Q)SAR community, is how to correctly validate the predictivity of a model [6]. It might be due to these challenges that (Q)SAR modeling is still only hesitantly accepted as an alternative testing method.

In this thesis, we present work on validation and visualization of (Q)SAR models. We introduce a study comparing existing validation methods, and provide a visualization tool for investigating small molecule datasets. Moreover, our visualization tool is employed to inspect (Q)SAR validation results.

1.1 Contributions and Findings

This thesis can be seperated into three main parts, each part has been published in a scientific journal.

The first part is a study on validation of (Q)SAR models. Validation provides a predictivity estimate of a (Q)SAR model, and allows comparing various approaches in order to determine the most predictive approach. Hence, validation is essential to demonstrate the reliability of predictions on unseen data, especially when the (Q)SAR model is applied to risk assessment of potentially toxic chemicals. For

the validation of (Q)SAR models, regulatory authorities often refer to guidelines of the Organisation for Economic Co-operation and Development (OECD). However, as mentioned above, (Q)SAR researchers are still discussing the best validation method. In particular, some researchers propose to sacrifice training data for external test set validation instead of estimating the predictivity with cross-validation and employing the complete data for model building. Our study compares both methods to determine the predictivity of the resulting (Q)SAR models and it measures which validation estimate is more accurate [7].

Our experimental results indicate that cross-validation should be preferred to external test set validation. Reducing the amount of training data, by keeping aside an external test set, degrades the final model performance. Moreover, when deciding on the external test dataset size, the researcher faces a trade-off between model performance and variance of the validation estimate. In contrast, cross-validation produces a superior final model as the entire dataset is used for training, while at the same time, exhibiting a lower variance. Furthermore, cross-validation slightly underestimates the final model performance. We have additionally investigated the effect of applying a model to compounds from a different feature distribution. In this scenario, we have demonstrated the unreliability of validation estimates and the importance of applicability domains.

The second part of this thesis is about visualization. We have designed and implemented CheS-Mapper (Chemical Space Mapper), a 3D viewer for datasets of chemical compounds that is freely available at `http://ches-mapper.org` [8]. The software assigns a dedicated position to each compound in the dataset, such that compounds with similar feature values are close to each other in 3D space. This mapping process can be controlled by the user, as CheS-Mapper allows the calculation of various kinds of compound descriptors and offers a range of embedding algorithms. Further pre-processing functionalities include the integration of 3D structure builders, a number of clustering algorithms, and the alignment of compounds in three-dimensional space according to their structure. The 3D viewer provides zooming and rotating functionalities as well as filtering and highlighting of compounds according to their feature values.

We demonstrate in several use cases that visualizing a small molecule dataset is important to get an overview of the included compounds and their properties. This is especially valuable when a (Q)SAR model is created for this dataset, to detect possible inconsistencies within the data (data curation). Moreover, visualization can help understanding (Q)SAR information in the dataset. Preferably, the visualization tool enables researchers to investigate how physico-chemical descriptors or structural features are related to the activity.

Both validation and visualization are combined in the third part of this thesis. As mentioned above, the rationale of (Q)SAR model predictions is hard to understand for the human researcher, as (Q)SARs often are statistical machine learning models that resemble black boxes. We introduce visual validation, as an approach for the graphical analysis of (Q)SAR validation results [9]. To this end, CheS-Mapper 2.0 is employed using the same features for embedding as used in (Q)SAR model building and validation, and to highlight and inspect the endpoint values and the prediction results of the model. We have added dedicated functionalities to CheS-Mapper like, e.g., the computation of activity cliffs, the calculation of embedding stress for the entire dataset as well as single compounds, and the easy determination of common feature values of arbitrary groups of compounds.

We apply visual validation to real-world data sets and explore strengths and weaknesses of (Q)SAR models. Common properties of correctly or incorrectly predicted compounds, or groups of compounds, can be investigated. Thus, visual validation might even help in mechanistically interpreting the (Q)SAR prediction, and could therefore facilitate the acceptance of (Q)SAR models as alternative testing method.

1.2 Structure

We provide the background for the work presented in this thesis in the Chapter 2. The chapter gives an initial introduction into cheminformatics and to (Q)SAR modeling. It further includes a specific introduction to (Q)SAR model validation and to cross-validation in general. Moreover, existing visualization approaches for small molecule datasets and machine learning models are introduced.

The first part of this thesis on validation of (Q)SARs is presented in Chapter 3. We outline the experimental workflow for comparing cross-validation to external test set validation. We analyze the results, and investigate the effect of changing several parameters of the workflow (e.g., dataset size or sampling method). This chapter also includes a discussion whether cross-validation can be regarded as an external validation method.

We introduce the 3D viewer CheS-Mapper in Chapter 4. The workflow, integrated algorithms, and the graphical user interface is described. This chapter further includes technical implementation details and presents the application of CheS-Mapper to real-world datasets.

Our approach on visual validation is provided in Chapter 5. We motivate visual validation before presenting the new functionalities in CheS-Mapper 2.0 that have been added for visual validation. Subsequently, we describe the method in detail

and outline various use cases to demonstrate the graphical inspection of (Q)SAR model validation results on real-world datasets.

In the final Chapter 6, we briefly summarize the presented work and show that the parts of this thesis are complementary building blocks within the (Q)SAR modeling workflow. The last section outlines future work.

2 Background

(Q)SAR modeling belongs to the large field of cheminformatics that will be introduced in Section 2.1. Section 2.2 provides a description of (Q)SAR modeling, including various (Q)SAR approaches and challenges. Next, we outline current practices of (Q)SAR model validation in general and introduce the cross-validation method in Section 2.3. Finally, we will describe visualization techniques to analyze (Q)SAR information and to visualize machine learning models in Section 2.4.

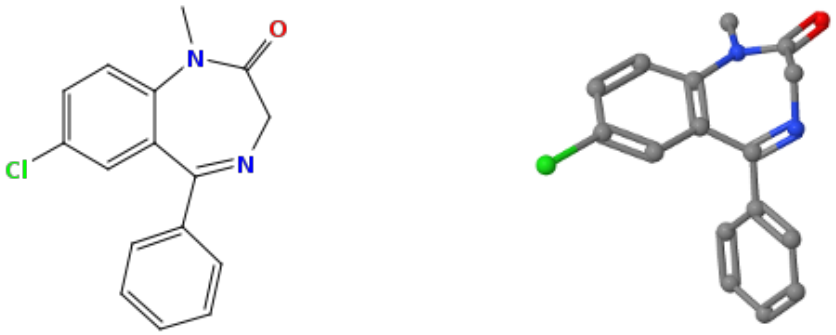
2.1 Cheminformatics

Cheminformatics can be defined as the application of informatics methods to solve chemical problems [10]. Its history goes back until 1946 [11], only shortly after the invention of the computer. However, the name cheminformatics (or chemoinformatics) was not common until 1998 [12] and it was previously denoted as chemometrics, computer chemistry, or computational chemistry [13]. Since its beginnings, the available information in chemistry has vastly accumulated and became only scientifically manageable in electronic form. Hence, techniques were required to store, process, and manipulate chemical data [13, 14]. In contrast to the related field of bioinformatics, which focuses on genes, proteins and larger chemical compounds, cheminformatics focuses on small molecules [14]. One of the main application of cheminformatics is drug design, with the goal of finding new structures that could be drug candidates [14, 15].

This chapter gives an introduction to the areas within cheminformatics related to (Q)SAR modeling and visualization, and therefore relevant to the work presented in this thesis. A comprehensive introduction to cheminformatics is provided by dedicated surveys [13–16] and textbooks [10]. Hence, we outline how to represent chemical structures (Section 2.1.1), show different types of compound descriptors (Section 2.1.2), and briefly introduce chemical databases and how they can be searched in Sections 2.1.3 and 2.1.4.

2.1.1 Representing Chemical Structures

Chemical compounds are small molecules that consist of at least two atoms, and are held together by bonds. A traditional way to describe a compound is its chemical formula that simply denotes the number of atoms in the compound (e.g., the



(a) 2D structure created with the CDK (Chemistry Development Kit)

(b) 3D structure build with Open Babel and rendered with Jmol

Figure 2.1: 2D and 3D structure of the compound *diazepam*, created with its SMILES (simplified molecular-input line-entry system) string.

chemical formula of the drug *diazepam* is $C_{16}H_{13}ClN_{20}$). However, this format does not define the chemical structure of the compound as it neglects its bonds, and is therefore ambiguous. Preferably, a representation should be unique, i.e., different compounds should have different representations. Another desirable property is that the representation supports searching for compounds in large databases (see Section 2.1.4).

2.1.1.1 Two-dimensional (Graph) Structure

Probably the most popular representation of a chemical compound is the two dimensional picture of its chemical structure: atoms are drawn using their chemical symbol (e.g., O for Oxygen; C for Carbon is usually omitted), and connected with lines that represent the bonds between atoms (see Figure 2.1a).

When referring to the (2D) structure of a molecule, one refers to its graph representation. The introduction of graph theory to chemistry goes back to 1864 [17]. The vertices of the graph are atoms, connected by un-directed, weighted edges that represent bonds between atoms. The edge weights define the bond type (e.g., single bond, double bond, triple bond, aromatic bond). Aromatic bonds describe de-localized electrons that are shared between bonds in ring systems. Aromatic rings are often denoted with alternating double and single bonds (kekulized) (as shown in Figure 2.1). Hydrogens are normally discarded (to make the representation more compact), and are only represented implicitly by filling unused valences. Many problems that arise when processing compound structures can be solved with graph theory [18]: to find out whether two compounds are structurally identical, one has to determine whether their two graphs are isomorphic. Another

common application is to detect the maximum common sub-graph (MCS) of two compounds.

Automatically creating the two-dimensional structure is termed Structure Diagram Generation or 2D Structure Layout. An early algorithmic approach was described in 1977 [19]. Recent approaches have improved the layout of non-planar compounds and aim to create aesthetically pleasing layouts [20].

Free tools to create and paint 2D structures are Modular Chemical Descriptor Language (MCDL) [21], Open Babel [22], the Chemistry Development Kit (CDK) [23], or the Rational Discovery Kit (RDKit)[1]. IDEAconsult Ltd [24] provides an online depiction service[2] that wraps the compound drawing functions of CDK and Open Babel and includes the depiction result of other online services (PubChem, Daylight, and Cactvs).

2.1.1.2 Three-dimensional Conformation

The 3D structure of a compound is not necessarily well-defined by its 2D graph: stereo-isomeric compounds have multiple different 3D conformations (e.g., mirrored images). Moreover, the structural arrangement of a compound is flexible and changes according to its molecular environment. In particular, the bio-active conformation is often significantly different from the most stable one at room temperature [25]. 3D structure builders usually predict the most stable, low-energy conformation of a compound based on its 2D structure. This is a computationally extensive task as the search space grows exponentially with the compound size. Numerous different approaches exist to build 3D structures of compounds automatically [26, 27]. As an example, the freely available 3D builder in Open Babel [22] uses a stochastic search to minimize the energy of a structure using the Merck Molecular Force Field (MMFF94) [28]. See Figure 2.1b for the image of a 3D compound structure built with Open Babel. Other free tools are, e.g., FRee On line druG conformation generation (Frog2) [29] and Balloon [30]. Probably the most popular commercial tool CORINA [31] is a rule-based system employing crystallographic data.

Freely available standalone tools to render 3D structures of compounds are Jmol[3], BALLView [32], PyMOL[4], or Visual Molecular Dynamics (VMD)[33]. Additionally,

1 http://www.rdkit.org
2 http://apps.ideaconsult.net:8080/ambit2/depict
3 http://www.jmol.org
4 https://www.pymol.org

Name	Diazepam
IUPAC name	7-chloro-1,3-dihydro-1-methyl-5-phenyl-1,4-benzodiazepin-2(3H)-one
Formula	$C_{16}H_{13}ClN_{20}$
SMILES	CN1c2c(C(=NCC1=O)c1ccccc1)cc(Cl)cc2
InChI	InChI=1S/C16H13ClN2O/c1-19-14-8-7-12(17)9-13(14)16(18-10-15(19)20)11-5-3-2-4-6-11/h2-9H,10H2,1H3
InChIKey	AAOVKJBEBIDNHE-UHFFFAOYSA-N
CAS	439-14-5
PubChem	CID 3016
ChEMBL	CHEMBL 12
ChemSpider	2908

Table 2.1: Identifiers and line notations of the drug *diazepam*.

online services to render and compute the 3D conformation of single compounds exist[5,6].

2.1.1.3 Identifiers

Chemical identifiers are assigned to a chemical and can be utilized to look up the compound in a database. Commonly, identifiers are sequentially increased, unique numbers or strings that are not related to chemistry. They are often proprietary, like the most widely used CAS registry number, assigned by the Chemical Abstracts Service (CAS) [34,35]. Other chemical IDs are provided by PubChem, ChemSpider, or ChEMBL (see Table 2.1).

2.1.1.4 Line Notations

Line notations encode the 2D structure of a compound (including stereo-chemical information) in a single line of compact and user readable text. When storing a chemical dataset including compounds and their properties in tabular format, there is usually a column that defines the compound structure using a line notation. The first approach was proposed in 1949 (Wiswesser line notation, WLN) [36], and commonly used in the 60s and 70s [13]. Another notation is the SYBYL Line Notation (SLN) [37].

A very common line notation is the simplified molecular-input line-entry system (SMILES). This proprietary format was introduced by Weininger [38]. It is popu-

5 http://www.molecular-networks.com/online_demos/corina_demo_interactive
6 http://web.chemdoodle.com/demos/2d-to-3d-coordinates

lar due to its good readability, as it is closely related to the native understanding of organic chemists [14]. Atoms are denoted by their chemical symbol (only non-aromatic atoms have to be enclosed in square brackets). Writing the character in lower case indicates aromatic rings. Alternatively, atoms can be encoded with their atomic number, e.g., [#6] instead of C. Bonds are symbolized using "-" (single), "=" (double), "#" (triple), and ":" (aromatic) characters. For simplicity, single and aromatic bonds can be omitted and are implied by adjacent atoms. Ring systems are indicated with numbers, e.g., c1ccccn1 denotes Pyridine, an aromatic ring including a nitrogen that is connected to the first carbon atom. Branches are described with parentheses, an example of a simple non-linear compound is acetic acid CC(=O)O. The SMILES representation of *diazepam* is given in Table 2.1. SMILES codes can be ambiguous, as the same structure can be described by multiple SMILES strings (e.g., by reversing the order). This ambiguity has been resolved by canonical serialization of the molecular structure. Hence, each compound has a unique canonical SMILES representation. However, as SMILES is a proprietary format, various chemical libraries have a differing implementation of a canonical SMILES serialization algorithm.

A non-proprietary line notation is the International Chemical Identifier (InChI) [39]. It was developed by the International Union of Pure and Applied Chemistry (IUPAC) as open standard. Similar to canonical SMILES, the algorithm assigns unique numbers to the atoms of a structure. A lot of open-source InChI Software converters exist (Open Babel, CDK, ChemSpider). The InChI code consists of different layers separated by slashes ("/"). The three main layers are chemical formula, atom-connections, and hydrogen layer (see Table 2.1 for the InChI representation of *diazepam*). Moreover, charges and stereo-chemical information can be configured.

The InChIKey [40] is the hashed code of the InChI (again see Table 2.1 for an example). It was designed as an identifier for databases or web searches, as the InChI for larger molecules is long and can be cut off by search engines. Hashing is a one way transformation, therefore, it is not possible to derive structural information from the InChIKey. Moreover, the key is not guaranteed to be unique, occasionally two compounds can be mapped to the same InChIKey (collision).

2.1.1.5 File Formats for 3D Structure Information

Formats to store the three-dimensional conformation of compounds are based on the connection-table format that was introduced as early as 1965 [41] (and improved by Morgan [42]). A widely used format that is especially suited for macromolecules, is the protein data bank (PDB) file [43].

```
Diazepam
OpenBabel02101417063D

 20 22  0  0  0  0  0  0  0  0999 V2000
    0.9227   -0.0816    0.0166 C   0  0  0  0  0  0  0  0  0  0  0  0
    2.3713   -0.0785    0.1723 N   0  0  0  0  0  0  0  0  0  0  0  0
    3.0674    1.1708    0.2036 C   0  0  0  0  0  0  0  0  0  0  0  0
    4.2924    1.3578    0.8756 C   0  0  0  0  0  0  0  0  0  0  0  0
    4.8923    0.3057    1.7474 C   0  0  0  0  0  0  0  0  0  0  0  0
    5.0393   -0.9262    1.3926 N   0  0  0  0  0  0  0  0  0  0  0  0
    4.5315   -1.3269    0.0847 C   0  0  0  0  0  0  0  0  0  0  0  0
    3.0092   -1.3243    0.1139 C   0  0  0  0  0  0  0  0  0  0  0  0
    2.3805   -2.3817    0.0067 O   0  0  0  0  0  0  0  0  0  0  0  0
    5.3641    0.6858    3.1186 C   0  0  0  0  0  0  0  0  0  0  0  0
    6.4851    0.0402    3.6543 C   0  0  0  0  0  0  0  0  0  0  0  0
    6.9439    0.3735    4.9297 C   0  0  0  0  0  0  0  0  0  0  0  0
    6.2794    1.3446    5.6779 C   0  0  0  0  0  0  0  0  0  0  0  0
    5.1506    1.9772    5.1575 C   0  0  0  0  0  0  0  0  0  0  0  0
    4.6873    1.6460    3.8822 C   0  0  0  0  0  0  0  0  0  0  0  0
    4.9734    2.5873    0.7978 C   0  0  0  0  0  0  0  0  0  0  0  0
    4.4153    3.6518    0.1004 C   0  0  0  0  0  0  0  0  0  0  0  0
    5.2439    5.1555    0.0225 Cl  0  0  0  0  0  0  0  0  0  0  0  0
    3.1835    3.5071   -0.5231 C   0  0  0  0  0  0  0  0  0  0  0  0
    2.5167    2.2785   -0.4698 C   0  0  0  0  0  0  0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  1  0  0  0  0
  2  8  1  0  0  0  0
  3  4  2  0  0  0  0
  3 20  1  0  0  0  0
  4  5  1  0  0  0  0
  4 16  1  0  0  0  0
  5  6  2  0  0  0  0
  5 10  1  0  0  0  0
  6  7  1  0  0  0  0
  7  8  1  0  0  0  0
  8  9  2  0  0  0  0
 10 11  2  0  0  0  0
 10 15  1  0  0  0  0
 11 12  1  0  0  0  0
 12 13  2  0  0  0  0
 13 14  1  0  0  0  0
 14 15  2  0  0  0  0
 16 17  2  0  0  0  0
 17 18  1  0  0  0  0
 17 19  1  0  0  0  0
 19 20  2  0  0  0  0
M  END
```

Figure 2.2: 3D structure of diazepam in MDL molfile format

```
<?xml version="1.0"?>
<molecule xmlns="http://www.xml-cml.org/schema">
 <atomArray>
  <atom id="a1" elementType="C" x3="0.934309" y3="0.099663" z3="-0.043885"/>
  <atom id="a2" elementType="N" x3="2.382697" y3="0.101191" z3="0.113990"/>
  <atom id="a3" elementType="C" x3="3.079979" y3="1.349842" z3="0.146144"/>
  ...
 </atomArray>
 <bondArray>
  <bond atomRefs2="a1 a2" order="1"/>
  <bond atomRefs2="a2 a3" order="1"/>
  <bond atomRefs2="a3 a4" order="2"/>
  ...
 </bondArray>
</molecule>
```

Figure 2.3: 3D structure of diazepam in chemical markup language (CML), shortened

The MDL molfile format (Molecular Design Limited) is commonly used for small molecules and is often regarded as the standard exchange format [14]. It is a proprietary format, created by the company MDL Information Systems. An example of the molfile representation of *diazepam* can be found in Figure 2.2. Its main elements are an atom block (lines 5-24), including 3D coordinates and atom symbols, and a bond block (lines 25-46) defining the connected atoms and bond types. Unfortunately, the molfile is lacking support for aromaticity and compounds can only be stored in kekulized format.

The structure-data file (SDF) is an extension of the molfile. This proprietary format is also provided by MDL. It can include multiple compounds separated by the string $$$$. Moreover, additional properties can be defined following the structural definition of each compound.

An alternative, open format is the chemical markup language (CML) [44]. CML is based on the extended markup language (XML). It is developed since 1995 and was designed to describe not only molecules, but other chemical concepts like reactions and spectra. An example of the compound *diazepam* in CML format is given in Figure 2.3.

2.1.2 Chemical Descriptors and Structural Fragments

In order to build machine learning models to predict the activity of compounds, features (or attributes) are required that describe the molecular structure. There are two different categories of features of chemical compounds: physico-chemical descriptors and structural fragments. The latter are often represented by structural fingerprints.

2.1.2.1 Molecular Descriptors

Molecular descriptors express physico-chemical properties of compound structures with numeric values. An example is the logarithm of the partition coefficient (*logP*) that describes how a compound is distributed in a water-octanol mixture, and is therefore a measure of lipophilicity or hydrophilicity. Its importance is due to the fact that compounds with higher lipophilicity tend to permeate better across biological membranes [45].

Over 1500 descriptors are available [14]. The most simple descriptors are calculated based only on the atoms of the compound (e.g., molecular weight), many are based on the 2D graph of the compound (e.g., number of hydrogen bond donors), and some complex descriptors are based on the 3D structure and surface of the

compound (e.g., van der Waals volume). A comprehensive overview of molecular descriptors is given in the book by Todeschini [46].

Some descriptors can be calculated directly (deductively) using quantum chemical models [14]. Others are predicted with (quantitative) structure property relationship ((Q)SPR) models that are based on experimentally derived data.

2.1.2.2 Structural Fragments

Structural fragments are sub-graphs that either occur in the (2D) graph of a compound (the fragment *matches* the compound), or do not occur in a compound. Hence, structural fragment features are nominal binary features as they have two possible feature values. The presence or absence of a particular structural fragment may play an important role, as the mode of action of a compound often depends on its structure. These fragments are called structural alerts.

A common syntax to describe fragments is smiles arbitrary target specification (SMARTS)[7]. This language is an extension of the SMILES notation and was designed by the developers of SMILES [38]. The major enhancements are wildcards for atoms (any atom: "*") and bonds (any bond: "~"), and the introduction of logical operators (*and, or, not*). SMARTS fragments are very expressive, yet often hardly human readable. SMARTSviewer[8] is a freely available visualization tool that automatically creates images of fragments [47].

Fingerprints are a commonly used, compact representation of a set of substructures. A fingerprint is a bit-wise string, that includes only 1s and 0s, encoding presences and absences of fragments. Often, hashed fingerprints are employed, i.e., the number of bits is lower than the number of fragments. Hence, multiple fragments are mapped to the same bit. Hashing is applied to limit the size of the fingerprint, as the number of tested structural fragments can be very high and matches are often sparse (much more 0s than 1s). Nonetheless, compression of the data causes loss of information and makes the fingerprints harder to interpret. Fingerprints can be used to compute the pairwise distance or similarity between two compounds, using a suitable (dis)similarity measure. Often, the Tanimoto similarity measure is employed, as it ignores common absences (i.e., fragments that do not match both compounds) when computing similarity. Fingerprints are well suited for searching databases (see next section). Moreover, fingerprints are commonly used for (Q)SAR modeling, e.g., using artificial neural networks [48] or support vector machines [49].

Structural fragments for a dataset can either be created dynamically, or by matching predefined lists of fragments that include functional groups. Examples of pre-

7 http://www.daylight.com/dayhtml/doc/theory/theory.smarts.html
8 http://smartsview.de

defined lists are the 166 MACCS keys [50], or the Klekota-Roth fingerprint [51] that comprises 4860 fragments relevant to bio-activity. Alternatively, structural fragments can be mined by enumerating frequent common sub-graphs. An early approach to mine frequent patterns in the compound dataset was published in 1998 and based on inductive logic programming (WARMR [52]). Succeeding approaches like gSpan [53] or GASTON [54] improved the performance of graph mining by using effcient canditate generation and unique canonical representations. The tool fminer [55], that is build upon GASTON, detects backbone refinement classes (BBRCs), which are classes of tree-shaped sub-graphs. Due to the exponential growth of the number of existing tree fragments, the number of fragments is reduced by setting a minimum-frequency threshold and by enumerating only non-redundant features that are correlated to the endpoint activity value. Hence, fminer has a supervised (endpoint dependent) feature-selection mechanism included, instead of using a hashing mechanism that is commonly applied when creating fingerprints.

Another popular method are Extended-Connectivity Fingerprints (ECFPs) [56]. ECFPs detect circular neighborhood fragments up to a user configured diameter (a common value is 6). The method assigns integer identifiers to atoms (that encode atom type and connectivity), and combines these identifiers to fragment identifiers when including neighboring atoms. The fragment identifiers are used as hash-keys for the resulting fingerprint. Open-source implementations of ECFPs are available in RDKit, and in the development version of CDK (since version 1.5.6).

2.1.3 Chemical Databases

Various databases exist for diverse categories of chemical data, like crystallographic 3D data of macro molecules (Protein Data Bank (PDB) [57]), chemical reactions (BioPath [58], Reaxis[9]), or mass spectrometry data (Mass Spectral Database (MSDC) [59]). As the focus of this work lies on small molecule data, we introduce four prominent databases including chemical compounds. These databases provide compound structure, some physico-chemical properties, and biological activities (if available).

PUBCHEM[10] is a freely accessible database, hosted by the US National Institute of Health (NIH) [60, 61]. It includes more than 49 million compounds, 129 million substances, and 739,000 bio-assays (May, 2014). It provides various search functionalities (including similarity and sub-structure search), an up-

9 http://reaxys.org
10 http://pubchem.ncbi.nlm.nih.gov

load functionality and semantic web support since January 2014 (PubChem-RDF).

CHEMBL[11] is maintained by the European Bioinformatics Institute (EBI) and freely accessible [62]. It includes more than 1.5 million of drug-like bio-active compounds that are linked with more than 12 million activities and more than 53 million publications (May, 2014). The ChEMBL database is curated manually, data is added from publications. Semantic access using an Resource Description Framework (RDF) protocol is supported [63].

REAXYS[12] is hosted by Elsevier Properties SA, its access is limited. It was published 2009 and succeeds and unifies the Beilstein database [64], the Gmelin database and the Patent Chemistry Database. Reaxys includes organic, inorganic and organo-metallic compounds, as well as reactions.

CAS REGISTRY contains 87 million organic and inorganic substances (May, 2014) and is provided by the Chemical Abstracts Service (CAS) [34, 35] that belongs to the American Chemical Society (ACS). The access is restricted. The database is especially known for its widely used identifier, the CAS (Registry) Number.

2.1.4 Searching in Datasets and Databases

Efficiently searching through databases and large datasets requires a well-suited compound representation.

A full structure search looks up a query compound that was specified by its structure (without having a chemical identifier available). The structure can, e.g., be given with its SMILES string or by using a molecular editor [65]. The search is usually executed by computing a feature-representation (like a fingerprint) or by utilizing a canonical line notation. For example, the search in the CAS database [35] is based on a hash key representation (referred to as *Augmented Connectivity Molecular Formula*). If multiple structures match the feature representation, a refined graph isomorphism comparison can be applied.

A sub-structure search determines all available compounds that include a user-specified fragment. This requires an examination of whether the query fragment is isomorphic to a sub-graph included in the target graph. This problem from graph-theory is known to be a NP-complete problem that requires a large computational effort. To search large databases efficiently, the overwhelming set of compounds in

11 `https://www.ebi.ac.uk/chembl`
12 `http://reaxys.org`

the database is filtered out using fingerprint matching (similar to the full structure search). Hence, a fingerprint of the query substructure is created and compounds that do not match this fingerprint are discarded [66]. An initial approach for substructure matching was already developed in 1957 [67].

Similarity search yields compounds that are similar to the query compound, without matching specific sub-structures. This search can be performed very efficiently by computing the (Tanimoto) similarity between fingerprints (as described in Section 2.1.2), using a user-defined similarity threshold. The results are usually sorted by similarity in descending order.

2.2 (Q)SAR Modeling

(Quantitative) structure activity relationship ((Q)SAR) models predict the impact of chemical compounds on health and environment. The underlying (Q)SAR assumption is that the biological or chemical activity of a compound depends on its structure, and that compounds with similar structures have similar biological or chemical activities (as discussed in more detail below, in Section 2.4.3).

Commonly, the distinction between QSAR and SAR models depends on the data type of the predicted endpoint. SAR models are classification models that predict distinctive nominal activities (e.g., *active* or *inactive*), i.e., SAR models make qualitative predictions. Quantitative QSAR models resemble regression models that predict numeric endpoints. However, this distinction is not applied consistently in the literature, and many researchers always apply the entire term QSAR. When modeling chemical properties of compounds (e.g., *logP*) instead of activities, the term (quantitative) structure property relationship (Q)SPR model is employed.

2.2.1 Milestones in (Q)SAR History

In 1962, Hansch initially employed (Q)SAR modeling for predicting the activity of plant growth regulators [68].

An early example of (Q)SPR modeling was the fragment-additive data-based approach to predict *logP* in 1975 [69]. The first attempt to predict biological activity with structural fragments was presented by Klopman in 1984 [70].

In 1988, Cramer *et al.* introduced the first (Q)SAR model that was based on the 3D conformation of compounds using comparative molecular field analysis (CoMFA) [71].

2.2.2 Types of (Q)SAR Modeling Approaches

Data driven machine learning models are probably the most commonly applied (Q)SAR modeling approach. An introduction to machine learning is given in the following section. Subsequently, we introduce expert systems, molecular modeling and read across.

2.2.2.1 Machine Learning

Machine learning algorithms learn a target function to predict the endpoint value of untested compounds[13]. Training a machine learning algorithm belongs to supervised learning (i.e., endpoint-dependent learning). The learner requires a training dataset as input, with known endpoint values and a set of features that describe each compound. Commonly, the feature-vectors are referred to as x-values and the endpoints as y-values. Feature values for the untested query compound have to be provided in order to predict the outcome of the compound (i.e., to apply the target function).

The application of a machine learning algorithm to a training dataset can also be regarded as search in the hypothesis space to find the hypothesis that fits the provided data best [4, 72]. The hypothesis space depends on the representation of the target function by the learner and on the feature representation of the data. Hence, each learner has an inductive bias [4] that affects how the learner will generalize to predict compounds with unknown activity that are not present in the training data. Accordingly, the optimal learner and feature representation depend on the particular data and predicted endpoint.

Often, the hypothesis space is ordered and searched from general to specific hypotheses. Very general hypotheses are overly simple descriptions of the target function. Too specific hypotheses, however, are very complex and are overly exact descriptions of the training data. As a result, an algorithm that has learned an extremely complex target function might predict the training data very well, but its ability to generalize is poor, and its predictivity on unseen data is bad. This concept is called overfitting. Proper validation of machine learning models is necessary to detect and prevent overfitting, as described in the Section 2.3.

Numerous types of machine learning algorithms exist. As an example, we will describe the decision-tree based method *random forests* below. Other types of classification algorithms are probabilistic learners like *naïve Bayes* [73] or *support vector machines* [74], a kernel-based binary classifier that can also be used for regression.

13 As we focus on (Q)SAR modeling, we refer to the examples or instances within the modeled dataset as compounds, and to the predicted target or class value as endpoint or activity value.

Linear regression [72] is one of the simplest approaches to regression and is often used as building block by more complex models like *regression model trees* [75,76].

A widely used, free toolbox for data mining and machine learning is WEKA [77] (also used in this thesis). Other applications are the statistical tool R [78], or the software suite Orange [79].

Random Forests

The random forest [80] algorithm is a predictive classifier that combines single decision trees. It is relatively resistant to over-fitting and has been applied in this work for (Q)SAR modeling in Sections 3.1, 5.3.2, and 5.3.3.

In general, a decision tree predicts a compound by evaluating a test for a particular feature at each node of the tree [4]. Accordingly, the compound is sorted through the tree until it is assigned to a leaf node that corresponds to the predicted class value. The tree is initially built by sequentially adding the *best* feature as new node. The *best* feature is selected by testing how well it separates the training compounds in this node with respect to the endpoint value. A common test criterion, that is also used in random forests, is the information gain.

Random forests combine single decision trees by employing bagging (also called bootstrap aggregation), a technique that can improve the performance of a predictor by building various predictors on sampled subsets on the data. Additionally, when building each decision tree, only a randomly sampled subset of features is available to create a new node. The prediction results of the decision trees are combined by consensus voting. The WEKA implementation that is applied in this work builds by default 100 decision trees.

2.2.2.2 Expert Systems

Expert systems are manually crafted predictors, that do not require a training set. Commonly, predictions are made based on explicit rules collected by human experts. As an example, the commercial expert system Derek [81] provided by LHASA Ltd makes qualitative predictions based on toxicophores, which are fragments that are known to occur in toxic compounds. Derek makes predictions for a range of endpoints including among others mutagenicity, carcinogenicity, and skin sensitization. Another expert system is OncoLogic [82], provided by the US Environmental Protection Agency (EPA). OncoLogic uses manually created decision trees to predict 6 levels of concern for carcinogenicity (*low*, *marginal*, ..., *high*). Moreover, it supports predicting different categories of chemicals like organic chemicals, polymers, metals/metalloids and fibrous substances. To evaluate the reliability of its predictions, OncoLogic was peer-reviewed by experts.

In general, expert systems can only be validated with external test sets, but not using re-sampling techniques like cross-validation. Moreover, these tools often suffer from a narrow applicability domain.

2.2.2.3 Molecular Modeling or 3D-(Q)SAR Modeling

Molecular modeling can be employed to predict the binding affinity of structurally similar compounds (ligands) to the active site of a protein. This might e.g., be useful to predict how well a compound is suited to inhibit the enzymatic activity of a particular protein. It is essential for molecular modeling to take the flexibility of possible 3D conformations of the ligands into account, as the bio-active conformation of ligands largely depends on the active site of the enzyme.

The first 3D-(Q)SAR approach, CoMFA, was published in 1988 [71]. CoMFA superimposes the 3D structure of the test compound with the training compounds, and calculates the force-field energy of the protein-ligand interaction at various data points. The calculated energies are employed as input features for a Partial Least Squares (PLS) regression model to predict the binding affinity.

Molecular modeling models are limited to a very specific mode of action. These models are therefore not suited to model complex endpoints, like carcinogenicity. Complex endpoints are probably affected by interactions with more than a single enzyme. Moreover, the activity of a compound might depend on its bio-availability or its transformation products.

2.2.2.4 Read Across

Read across is a manual approach by an expert chemist that estimates the endpoint of a compound by comparing it to similar compounds with known activity [83]. Similarity should take structural similarity and physico-chemical properties into account. Read across relies on "judgment evaluation", as it assumes that the analogues (compounds that are similar to the query compound) behave similarly to the query compound. A tool to support read across is the (Q)SAR toolbox [84] provided by the OECD. Read across is a very flexible approach that relies on the knowledge of the expert. It is time-consuming and cannot be validated statistically.

2.2.3 Data Availability and Quality

Measuring the toxicity of chemical compounds often costs animal lives, is expensive and time-consuming. Hence, the number of tested compounds is often small. Moreover, *in-vivo* (and *in-vitro*) experiments tend to have a high error rate, and produce noisy data that is difficult to model. Increasing the dataset size by mixing results from different studies should be done with care: the measurements of different experiments are often not comparable due to differences in the experimental setup. In particular, data from different species should not be combined [85]. There-

fore, one of the main challenges for (Q)SAR modeling is that datasets are often too small and may contain experimental errors.

2.2.4 Applicability Domain of (Q)SAR Models

The applicability domain (AD) of a (Q)SAR model describes the feature space of compounds that can reliably be predicted. Whenever a compound is predicted by the model, it should be checked whether the compound lies within the AD. This is especially important due to the vast size of chemical space. Compounds that are dissimilar to the training dataset compounds could still effect the modeled activity with an entirely different mode of action. Hence, chemicals that have differing chemical properties should not be predicted by the (Q)SAR model. In general, AD methods compare the feature values of a query compound to the feature values of the training dataset compounds. Many reviews of AD approaches exist [5, 86].

A common method of computing the AD is the distance based approach. This method computes the distance of the test compound to the center (centroid) of the training dataset. Alternatively, the pair-wise distance to all compounds of the dataset can be computed. The compound will be excluded from the AD if the distance is too high. The threshold for excluding the query compound has to be defined manually (one could for example choose the maximum distance within the training data). Moreover, the employed (dis)similarity measure has to be selected (e.g., Mahalanobis, Euclidean, City Block distance). In order to take into account correlations between feature values, distance based AD methods should be applied to PCA transformed data. Alternatively, a popular distance based approach is the leverage method [87], that computes the distance using the diagonal elements of the hat matrix.

Probability density distribution methods [88] allow identifying test compounds that fall into empty regions inside (the convex hull of) the feature space. A non-parametric approach that makes no assumptions about the data distribution is described in [86]. Nevertheless, the user has to define a manual threshold to exclude compounds from the AD (e.g., compounds below 0.05% probability).

There is no overall best AD method. AD methods should be as similar to the model as possible. Hence, the AD algorithm should be based on to the same descriptors and/or (dis)similarity measure. An integrated approach for defining the AD is performed by the *lazar* framework as described in Section 2.2.5.

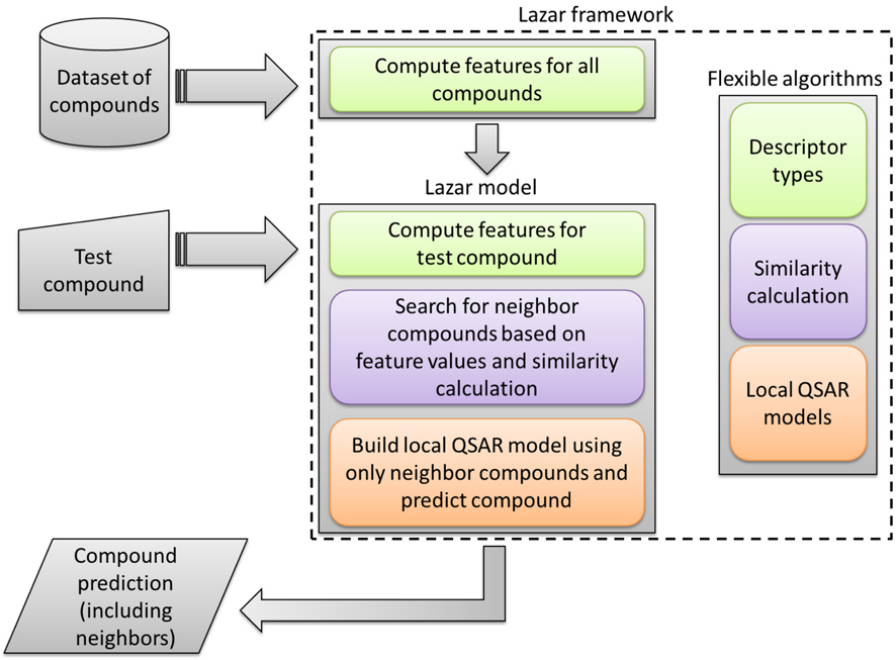


Figure 2.4: The workflow of the lazar framework, with regard to the configurable algorithms for descriptor calculation, chemical similarity calculation, and local (Q)SAR models.

2.2.5 An Application Example: lazar

The *lazar* (lazy structure–activity relationships) framework [89] is a (Q)SAR modeling approach that is comparable to read across, as it predicts query compounds according to similar compounds (neighbors) in the dataset. Therefore, local (Q)SAR models are build with neighbor compounds only. The *lazar* framework can flexibly be adjusted to the training dataset (see Figure 2.4): structural fragments or physicochemical descriptors can be selected as descriptors. Moreover, the (dis-)similarity measure to compute the neighbor compounds based on the selected descriptors is configurable. Finally, the user decides which algorithm is applied to create local (Q)SAR models.

The framework has an AD definition integrated and thus additionally returns a *confidence* value when predicting an unseen compound. The *confidence* does depend on the number of detected neighbors and their similarity score. Moreover, *lazar* extends the classical AD definition by also taking the coherence of neighbor endpoint values into account for the calculation of the *confidence* value.

2.2.6 Acceptance of (Q)SARs as Alternative Testing Method

Since 2007, the Registration, Evaluation, Authorisation and Restriction of Chemicals (REACH) [90] regulation enforces a more strict risk assessment of chemicals in the European Union, with the goal to protect human health and the environment. To avoid an increase of animal testing, the European Chemicals Agency (ECHA) (founded in 2007 to manage the implementation of REACH) also encourages the use of alternative testing methods, including *valid* (Q)SAR models. The ECHA refers to the OECD guidelines [91] for the definition of a scientifically *valid* (Q)SAR model:

"*To facilitate the consideration of a (Q)SAR model for regulatory purposes, it should be associated with the following information:*

1. *a defined endpoint*
2. *an unambiguous algorithm*
3. *a defined domain of applicability*
4. *appropriate measures of goodness-of–fit, robustness and predictivity*
5. *a mechanistic interpretation, if possible*"

According to the annual report on REACH from 2013 that is published by the ECHA, the number of reported dossiers including (Q)SAR models is rising, but modeling is still much less used than read across [92]. The rare employment of (Q)SARs as alternative testing method might be due to the fact that the above listed requirements are challenging.

Even though the OECD guidelines [91] give advice for the validation of (Q)SARs and on how to evaluate goodness-of-fit measures (point 4 above), the best method to validate (Q)SAR models is still under discussion in the (Q)SAR community (see below in Section 2.3).

Another demanding requirement for valid (Q)SARs is to provide a mechanistic interpretation (point 5 above). Preferably, (Q)SAR models should give an explanation for the prediction and relate the prediction to the structure of the query compound and its mode of action. As mentioned above, some (Q)SAR models resemble black boxes that do not provide this information. To this end, we introduce visual validation, an approach that could help to mechanistically interpret (Q)SAR modeling results (see Chapter 5).

2.3 Validation of (Q)SAR Models

There are different types of (Q)SAR models, like the above mentioned rule-based expert systems that do not require a training step. Most (Q)SAR models, however, are the result of statistical machine learning algorithms that are trained on a dataset with a known target endpoint, and automatically learn a function to predict the endpoint value of novel compounds [93]. As previously noted, optimizing the predictive performance of the (Q)SAR model on the training dataset would lead to a model that is too specific and does not perform well on unseen data (overfitting) [94]. Therefore, the model has to be validated on unseen instances, thus the data, in principle, has to be split into a training and test set.

In this work we will focus on validation for performance evaluation rather than for model selection. Model selection aims at choosing a model from a set of models that performs best on a particular dataset. It is therefore vital for the validation method to be sensitive to estimate differences between the various approaches, and yet to have a small type I error rate [95]. It is not critical if the method has a bias in computing the actual model predictivity, as long as this systematic error affects all approaches equally. In contrast, the objective of this work is performance evaluation: validation methods should preferably give an accurate predictivity estimation with low variance.

The choice of the most suitable validation method depends on the dataset size. In cases of unlimited amounts of data, a single training-test-split would be sufficient [93,96]. For small datasets, however, it is recommended to repeat the partition of the data into training and test sets. Whether enough data is available for a single split depends also on the complexity of the problem and the applied learning scheme [96, 97]. It can be noted that in most real-world (Q)SAR applications so far the datasets are rather small as the number of (*in-vivo*) tested compounds with known endpoint values is scarce [98].

Another important aspect for the validation of (Q)SARs is the data distribution. Machine learning models are usually built under the assumption that they are applied to unseen data from the same feature range and distribution as the training data. A model is likely to perform worse if applied to data from a different distribution [99, 100]. Consequently, validation estimates will only hold for unseen data from the same distribution as the validation data. It is therefore vital for the validation method to use an unbiased sample as test set. Sample selection bias can be reduced by avoiding small test sets, by repeating the training test split, or by using stratified splitting (as described below).

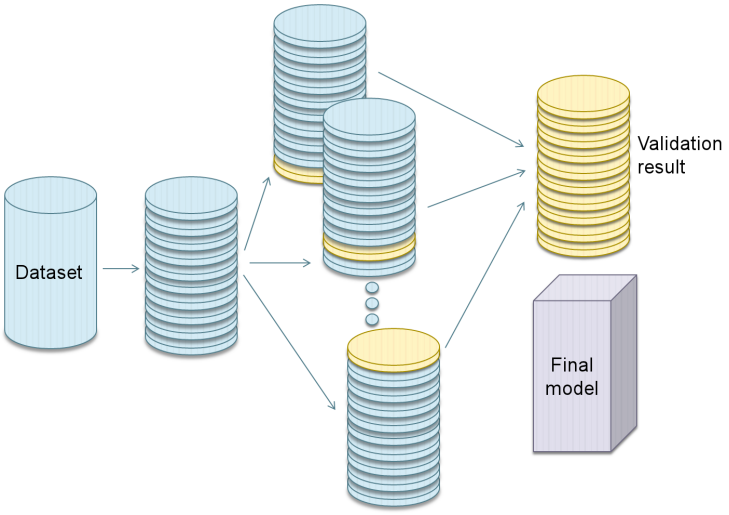


Figure 2.5: 10-fold cross-validation

2.3.1 Cross-Validation and its Variants

Cross-validation is a common and popular technique to validate learning algorithms. There exist many versions, the one employed here is k-fold cross-validation [96, 101]. As illustrated in Figure 2.5, the dataset is randomly split into k equally sized parts (10 is a common number for k). Subsequently, k-1 parts are used as training set for model building, and the left out k-th part is used as test set. The overall validation performance is computed by accumulating the predictions of all k test sets. If feasible, this procedure is repeated multiple times, to avoid any bias originating from the initial random split. A possible variant is to use stratified cross-validation: the data is split into folds that have an equal distribution of endpoint values, as the original dataset.

Hold-out validation randomly splits a dataset into training and test set according to a fixed ratio. Usually, one leaves 10%, 30% or 50% of the dataset aside as test set. This procedure is repeated a fixed number of times (30 times is a common value), and the results are then averaged to obtain the final validation performance. This method is sometimes also referred to as leave-many-out cross-validation (LMO-CV) [98, 102]. The term LMO-CV is however not consistently used in the literature, as sometimes it is also used for k-fold cross-validation [103]. To this end, we do not use this term to avoid confusion.

A special case of k-fold cross-validation is leave-one-out cross-validation (LOO-CV), where the number of folds is equal to the number of compounds in the dataset. This procedure is well suited for small datasets, where as many data as possible should be used for building the models. Bootstrap validation applies sampling with

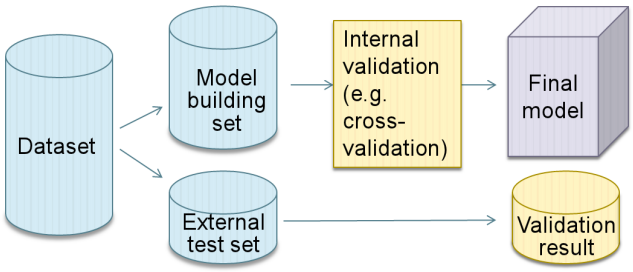


Figure 2.6: External validation using a single test set

replacement to split the data into training and test dataset. Similar to the previous methods, this process is repeated, and the validation performance of the models, built on a training set sample and applied to the corresponding test set, is averaged.[14]

The history of cross-validation is stretching over more than 80 years now. Researchers discovered in the early 1930s that the training error is overoptimistic, and that the input data for algorithms has to be split into a training and a validation set [94]. It took a while until this idea developed into cross-validation as we know it today: an early application of LOO-CV for classification was that of Lachenbruch in 1968 [104], and it was first used for multiple regression by Allen in 1974 [105]. It began to become popular in the machine learning field in the 1980s and early 1990s, when Breiman *et al.* [106] and Quinlan [107] applied cross-validation to evaluate decision tree algorithms. Later in the 1990s, many articles on theoretical and experimental comparisons were published [95, 101, 108, 109]. Nowadays, it is a popular and widely used validation technique.

2.3.2 Internal and External Validation

In the (Q)SAR community, the above listed validation methods are often used for "internal validation". According to some researchers, a model should additionally be validated by external validation with a single test set [98, 102, 103, 110]. Hence, a single split is performed at the beginning of the external validation procedure, into model building data and external test set (see Figure 2.6). A common test set size is between 10% and 30% of the data, yet in some applications test sets much larger than the model building set have been used [111, 112]. It is often recommended to employ the test set split in a strategic fashion to ensure that descriptor and endpoint values are distributed evenly in the test and in the model building set (i.e. to reduce

14 LOO-CV and bootstrap validation have not been applied in this work due to time and computational constraints.

the sample selection bias) [98,103,113]. There are numerous different approaches to apply this stratified splitting, for a comprehensive list see the guidance document of the European Commission [103]. After splitting the data, internal validation (via e.g. cross-validation) is applied on the model building set to select the best model. The selected model is reported as the final model of the external validation. However, the predictivity estimate of the reported final model is not the internal validation score, but the performance of the model applied to the external test set. The external test set has not been touched throughout the model building process. Thus, external validation does validate the model instead of a learning method.

In contrast, the predictivity estimate of plain cross-validation procedures is the cumulative or average score of the models on the repeated splits. The final model of this method is built on the complete dataset. This model has not been used in the actual validation process. Hence, a plain cross-validation approach evaluates the whole learning method (instead of the final model).

Please refer to the Section 3.3 whether cross-validation can be employed as an external validation method.

2.3.3 Current Concerns about Cross-Validation

Hawkins *et al.* [6,114] recommend cross-validation without additional external test set validation. They show that LOO-CV assesses the model fit of (Q)SPR regression models more accurately and less variably than external validation with small external test sets ($\leqslant 50$).

Despite this work, many researchers do recommend external test set validation, and consider that internal validation performance (evaluated with e.g. cross-validation) only estimates model robustness, but not model predictivity [98, 103, 115]. It is assumed that internal cross-validation often gives an overoptimistic score, and that external validation is more demanding. It is frequently argued that the solely cross-validated model is never verified on unseen compounds that have never been used during training [116], and that the models that are built and validated on the folds are different from the finally reported model [98].

Before presenting our experiments that could allay some of these concerns, we give a few possible explanations for the bad reputation of using cross-validation without external test set validation:

- The wrong use of cross-validation can lead to overoptimistic validation scores. The most common error is probably information leakage, i.e., information about the test compounds is available in the training step. An example for information leakage is given when supervised feature selection (based on the

endpoint) is applied on the complete dataset before cross-validation [117]. This is not valid, as the feature selection step is part of the model building process, and therefore should be applied within each fold.[15]

- Another frequent error when applying cross-validation is sometimes referred to as "parameter fiddling". The same single cross-validation split is used extensively within a (grid) search using lots of different parameter settings. This results in overfitting: it is likely that a parameter setting will be found that just by chance fits this particular cross-validation split very well, but will work less well on unseen data. To avoid this, multiple restarts of cross-validation should be applied (often a 10x10-fold cross-validation is recommended), and/or a nested level of cross-validation should be used for parameter optimization [119].

- A rather psychological argument of some (Q)SAR developers or users is that they prefer external test set validation, because they feel more comfortable having the final model validated (external validation), compared to a validation of the learning scheme (where the final model is built on the entire dataset). However, a statistical (Q)SAR model always implicates uncertainty. One cannot be sure when applying the model to unseen compounds, regardless if it was validated externally or not.

- There exist studies where the internal cross-validated score is much higher or badly correlated to the external score [110, 120, 121]. Often the conclusion is drawn that cross-validation is overoptimistic, and that real model predictivity can only be estimated by external validation. One might conjecture that in such a case the test set was too small and/or from a different distribution than the training set. It is also conceivable that extensive model selection with internal validation overfitted the training set.

2.4 Visualization of Chemical Datasets

The probably most basic visualization task for chemical data is layout and drawing of single chemical compounds, as described above in Section 2.1.1. When working with datasets that include multiple compounds with numerous properties, tabular viewers can be employed to provide an overview of the structures and their features (see Section 2.4.1). Moreover, visualization tools should allow to investi-

15 It is often recommended to use a nested cross-validation for feature selection, with an inner cross-validation loop on each training fold [118, 119]. This is appropriate when various feature selection methods are evaluated.

gate possible inter-dependencies between molecular properties. In particular, researchers commonly want to analyze relationships between physico-chemical or structural features of compounds and their biological or toxic effects. Dimensionality reduction can be applied to visualize these correlations (see Section 2.4.2). We further discuss aims and challenges of integrated visualization tools for chemical datasets in Section 2.4.3, and introduce numerous tools. The final section 2.4.4 highlights existing approaches for the visualization of validation results.

2.4.1 Chemical Spreadsheets

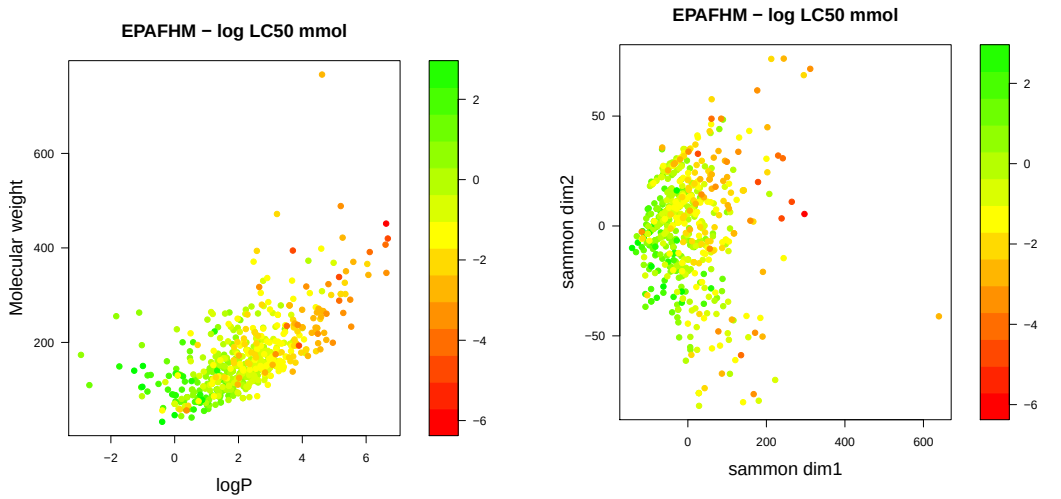
Different file formats for compound datasets have been introduced in section 2.1.1. These files store not only the actual structure, but also measured or calculated properties for each compound. Hence, the dataset follows a tabular format including compounds as rows and properties as columns. Chemical spreadsheets resemble or extend generic spreadsheet programs (like e.g. LibreOffice Calc or Microsoft Excel). These programs can read chemical files and usually render a 2D depiction of the compound in each row. The tools are frequently incorperated into larger frameworks[16][17][122] and often have a plugin functionallity to show scatterplots of selected properties. A free chemical spreadsheet is integrated into Bioclispe [123]. LICSS [124] is a free extension for Microsoft Excel.

2.4.2 Dimensionality Reduction Techniques

The most important requirement for visualization approaches is to help detecting correlations in multivariate and structure data. A possible method to analyze correlations is to apply dimensionality reduction. Scatter plots can visualize how and if the endpoint values are correlated to the selected properties or features. An example using compound data can be found in Figure 2.7a: each dot corresponds to a structure and is located in the plot according to its *logP* and molecular weight. As exemplified in this figure, a third property can be highlighted using colors. In this example the endpoint value is selected (LC_{50}), and it can immediately be seen that activity is correlated to the *logP* value. Alternatively to coloring, different symbols could be used to depict a third property. Additionally, a z-axis could be added to show one more property and arrange the dataset in three dimensions.

The scatter plot is helpful as it visualizes how and if the endpoint values are correlated to the selected properties or features. To overcome the limitation to

16 http://https://www.chemaxon.com/products/marvin/marvinview
17 http://www.schrodinger.com/Seurat



(a) Scatterplot using 2 physico-chemcial descriptors.

(b) Scatterplot using 14 physico-chemical descriptors and Sammon's Non-Linear Mapping for dimensionality reduction

Figure 2.7: Scatterplots of the fish toxicity dataset (Fathead Minnow Acute Toxicity) [1]. The color gradient indicates active compounds (with low LC_{50} values) in orange and red, and less active compounds in green. The half lethal concentration LC_{50} corresponds to the amount of the compound that is sufficient to kill half of the population.

two or three features, dimensionality reduction is applied to transform the multi-dimensional feature space to two or three numeric features while trying to preserve the data structure [125]. Hence, two compounds that have similar feature values in the original feature space should have similar values in the transformed feature space, and are therefore located close to each other in the data plot. An example for a scatter plot using Sammon's mapping is given in Figure 2.7b.

Various variants of dimensionality reduction are applied in cheminformatics [126, 127]. We introduce some common methods:

PRINCIPAL COMPONENTS ANALYSIS (PCA) uses the eigenvectors of the covariance matrix for a loss-free transformation of the original features into principal components (see e.g. [127]). The principal components are uncorrelated features, that are sorted according to the explained variance. This method is also often used for feature reduction, by omitting the lowest ranked features that do not explain a lot of variance. For 2D or 3D mapping, the top, most significant two or three principal components are used.

SAMMON'S NON-LINEAR MAPPING is an iterative multidimensional scaling method [128]. The algorithm maps the high-dimensional input space to a lower dimensional space, by iteratively reducing the error (or stress) based on

a user-defined (dis)-similarity measure for each compound employing gradient descent.

MULTIDIMENSIONAL SCALING USING MAJORIZATION (SMACOF) is an alternative multidimensional scaling approach [129], that computes the stress values with majorization to ensure a linear convergence rate.

SELF-ORGANIZING MAPS (SOMS) are a variant of Artificial Neural Networks (ANNs) [130] and were first applied to cheminformatics in 1994 [131]. SOMs try to retain the topological characteristics of the data, by mapping the high-dimensional input space to a lower dimensional set of neurons. In contrast to classical ANNs, this is done in an unsupervised way, that is without the use of a target variable. The array of neurons (usually a 2D grid) can be interpreted as the mapping result.

T-DISTRIBUTED STOCHASTIC NEIGHBOR EMBEDDING (T-SNE) is a variation of a stochastic neighbor embedding that uses conditional probabilities that represent similarities [132]. In particular, t-SNE utilizes a cost function that is easier to optimize (a Student-t distribution is used instead of a Gaussian to compute the similarity between two points). The effective number of neighbors that are used to compute the conditional probability distribution can be configured.

2.4.3 Visualization Tools for Small Molecule Datasets

Various visualization tools exist for inspecting chemical datasets. Before introducing a collection of tools in detail, we outline the visualization concepts that are employed by these methods.

As described in the previous section, dimensionality reduction helps in detecting correlations between feature and endpoint values. Many visualization tools for small molecule datasets include these dimensionality reduction techniques (ChemSpaceShuttle [133], MQN-Mapplet [134], Screening Assistant 2 [135], ViFrame [136], HiTSEE KNIME [137]).

Other visualization tools rely on clustering (Molecular Property eXplorer [138], Radial Clustergrams [139]). Clustering of compounds into subgroups according to their properties can provide useful information: clusters might resemble chemical categories having similar properties or sharing a common activity profile.

In general, tree-based or graph-based approaches convert the dataset into connected data structures (LASSO graph [140], SALI networks [141], SARANEA [142], Scaffold Hunter [143], Similarity–Potency Trees [144]). Hence, nodes in the trees (or

graphs) correspond to compounds and/or groups of compounds. The proximity of the nodes reflects the (dis-)similarity of the employed feature values. These tools can usually highlight the activity value of compounds and are therefore suitable for (Q)SAR information analysis as well.

(Q)SAR information in chemical datasets is usually hard to comprehend. The (Q)SAR assumption is that compounds with similar structure tend to have similar chemical and biological properties [145]. Consequently, small changes in structure often cause only small changes in activity. In this case, the so-called *activity landscape* [146] is considered to be smooth. The term activity landscape describes the distribution of endpoint values in the feature space. However, sometimes small changes in structure can cause big changes in activity. This is referred to as *activity cliff* [147, 148]. Therefore, an activity cliff can be defined by two compounds that have very similar feature values, but largely differing endpoint values. Activity cliffs are often visualized with the already mentioned approaches for visualization. Moreover, heat-maps (matrices of colored cells) are employed to highlight the corresponding pairs of compounds (Toxmatch 2 [149], [141]).

We review visualization approaches that directly aim at visualizing chemical datasets. Many of the following methods are available as free or proprietary software (see Table 2.2).

CHEMSPACESHUTTLE embeds multi-dimensional input into 3D with encoder networks and non-linear partial least squares [133]. Furthermore, it provides clustering with SOMs and can process large datasets. However, it is rather a general visualization tool, as it does not show any compound structures and chemical descriptors have to be pre-calculated with other software.

HITSEE is a visualization tool for the analysis of high-throughput screening experiments [137]. The user can select the top n compounds from the screen result, that will be embedded into 2D using structural fingerprints as similarity measure. The compounds are visually encoded as pie-charts that show activity and *logP*. The user can then explore the neighborhood of the selected hits by adding the most similar compounds that will be included into the embedding. It is implemented into the graphical workflow tool KNIME [152], but it is not freely available as it uses a proprietary library.

LASSO GRAPH (layered skeleton-scaffold organization) extends basic scaffold tree approaches [140]. Additionally to scaffolds, LASSO utilizes cyclic skeletons (CSKs) to build the graph structures. CSKs are derived by changing all heteroatoms to carbons and setting all bond orders to one. The activity of com-

Name	Ref	Availability	Operating-System	URL	Last update
ChemSpaceShuttle	[133]	free	linux-32bit	[1]	Jan 23, 2003
HiTSEE	[137]	proprietary	all	[2]	?
LASSO Graph	[140]	proprietary	?	?	?
Molecular Property eXplorer (MPX)	[138]	proprietary	?	?	?
The Molecule Cloud	[150]	open-source	all	on request	?
The MQN Mapplet	[134]	open-source	all	[3]	Aug 13, 2013
Radial Clustergrams	[139]	proprietary	?	?	?
SAR Matrices	[151]	proprietary	?	?	?
SARANEA	[142]	open-source	all	[4]	Sep 18, 2009
SALIViewer	[141]	open-source	all	[5]	Feb 11, 2012
Scaffold Hunter	[143]	open-source	all	[6]	Oct 7, 2013
Screening Assistant 2	[135]	open-source	windows	[7]	Aug 12, 2012
Similarity-Potency Trees	[144]	free	all	[8]	Aug 13, 2010
ViFrame	[136]	open-source	windows	[9]	Feb 16, 2013

	URL
1	`http://gecco.org.chemie.uni-frankfurt.de/ChemSpaceShuttle_light`
2	`http://hitsee.hs8.de`
3	`http://www.gdb.unibe.ch`
4	`www.lifescienceinformatics.uni-bonn.de`
5	`http://sali.rguha.net`
6	`http://scaffoldhunter.sourceforge.net`
7	`http://sa2.sourceforge.net`
8	`www.lifescienceinformatics.uni-bonn.de`
9	`http://siret.ms.mff.cuni.cz/projects`

Table 2.2: A list of visualization tools

pounds, which are summarized as the graph nodes, is indicated by color-coded pie charts.

MOLECULAR PROPERTY EXPLORER is a proprietary tool [138]. The software visualizes hierarchical clustering of datasets with tree maps. The tree map consists of nested rectangles, each rectangle corresponds to a sub-cluster of the tree. This method has the advantage that it shows the whole clustering tree at once. However, compounds, which are the leafs of the tree, are drawn in rectangles of largely different sizes, depending of the depth of the leaf. Moreover, the tool colors the rectangles according to compound property values, and additionally provides heat-maps that show feature values of multiple properties simultaneously in a compound-property-matrix.

THE MOLECULE CLOUD resembles word clouds, as the tool draws the most common substructures present in the dataset [150]. It depicts 2D images of the substituents. The size of the image refers to the frequency. Additionally, coloring can be used to highlight properties like activity. The tool is open-source and available on request.

THE MQN MAPPLET shows pre-calculated PCA embeddings of large databases with almost one billion molecules [134]. The PCA is based on 42 integer value descriptors called molecular quantum numbers (MQN) and produces 2D-maps of a fixed grid-size, hence each pixel is occupied by multiple compounds. The coloring can be changed interactively and corresponds to properties, like, e.g., number of rings or heavy atom count. The MQN Mapplet is available as open-source application.

RADIAL CLUSTERGRAMS visualize trees derived by hierarchical clustering [139]. The tree structure is mapped to concentric circles. The nodes of the tree correspond to sectors in the circle and the distance to the center reflects the depth within the tree. Color coding is used to highlight the average activity of each node. The user can zoom into interesting sections in the tree with a fish-eye lens.

SAR MATRICES is an approach to visualize structure activity relationship information in datasets using structural decomposition tables [151]. The table rows correspond to series of compounds having similar cores. The columns correspond to fragments, chemical groups that are present at individual substitution sites of each core compound. Matrix cells are colored according to the activity value of the corresponding composite fragment (if available in the

dataset). The authors present a method to create and rank the matrices automatically according to variable criteria.

SARANEA, an open-source program, creates network-like similarity graphs that can be used to explore (Q)SAR information [142]. Compounds are represented by nodes. Nodes are depicted larger if they are involved in the formation of activity cliffs and the color of a node is based on the activity. The similarity measure is based on structural fingerprints. The tool further supports neighborhood graphs, where compounds are drawn in circles around the inspected compound, the distance to the compound reflects the similarity.

THE STRUCTURE-ACTIVITY LANDSCAPE INDEX (SALI) can be used to identify activity cliffs [141]. The index is computed for pairs of compounds according to their chemical similarity (e.g. based on structural fingerprints), and on their activity value. The SALI values can be utilized to create heat-maps. Moreover, SALIViewer is a freely available program that creates graphs, where each node corresponds to a compound and each edge correspond to a compound pair with high SALI value.

SCAFFOLD HUNTER is another open-source tool [143]. A scaffold tree summarizes common substructures of the dataset compounds as nodes, and the compounds as leafs. The tree nodes can be colored according to feature values, and therefore allow to detect structure activity cliffs. Additionally, a plot view, spreadsheet view and dendogram view aid exploring the dataset. All views are connected and compounds that are selected in one view, will be highlighted in the other views as well. Scaffold Hunter requires a database to store the datasets and scaffold trees.

SCREENING ASSISTANT 2 is a open-source program that allows to import data in a database back-end and is aimed for large datasets with millions of compounds [135]. The database is accessible as chemical spreadsheet, and the software provides descriptor calculation, SMARTS similarity search, filtering, and some scaffold analysis, as well as basic PCA visualization.

SIMILARITY–POTENCY TREES (SPT) visualize SAR information present in a dataset as trees [144]. The tree connects two compound-nodes solely if they are structurally similar (based on fingerprints), not using the activity value. Instead the activity values are used to color the nodes accordingly, which helps to identify structural differences that cause discontinuous changes in activity levels. The SPT program is freely available.

VIFRAME is a visualization tool that supports 2D mapping of molecule datasets based on structural fingerprints [136]. The program is designed as pipeline-based framework, that allows developers to plugin their own algorithms. The provided embedding methods are PCA, SOMs and SMACCOF.

COMPARING SETS OF DESCRIPTORS USING SOMS [153]. This approach utilizes SOMs for a pairwise comparison of sets of descriptors with variable dimensions. The visualization can be inspected in addition to a calculated similarity score for a pair of descriptor sets. The dataset is embedded twice with SOMs to a 2D map, once with each feature set. A smooth color gradient is rearranged, according to where compounds have been moved from the first mapping result to the second mapping result: the more distorted the color map is, the less do the feature values correspond to each other. Moreover, regions with locally homogeneous coloring indicate subgroups of compounds where the sets correlate.

2.4.4 Visualization Approaches for Model Validation

The graphical analysis of machine learning models are often model dependent [154–156]. An example can be seen in an interactive visualization approach displaying decision trees using bars for each tree node [157]. Each bar contains colored instances, sorted according to the corresponding feature that is employed in this node. The coloring reflects the class distribution. Split points are indicated by lines, and can be modified or removed by the user. The system then rebuilds the tree according to the manual modifications.

Mineset is a data-mining tool [158] that allows to create various machine learning models and provides different visualization approaches. This includes 3D scatter plots, as well as model-dependent views for decision tree or naïve Bayes models. However, the software is currently not available (according to email correspondence, the distributing company was planning to release a re-engineered beta version in 2014).

Another approach for model independent visualization of classification results uses a 2D projection of the predicted dataset with SOMs [159]. Empty regions in the feature space are filled by sampling new instances. The maps are colored according to the class probability that is provided by the model for each prediction. The decision boundary (50% class probability) is indicated with a white line. Feature contours can be drawn over the map in order to interpret the space. Moreover, test instances can be overlayed, with their actual class colored, to show misclassified instances.

A model independent method is especially aimed for multi-class problems (classification with more than two disjoint classes) [160]. The visualization is exclusively based on the probability estimate provided by the classifier for each class value. The resulting plot displays a circle that is divided into radiants, each radiant accounts for one class. The more confident the classifier is with the prediction, the closer this instance is drawn to the edge of the circle in the corresponding radiant. However, this approach is limited, as it ignores the actual feature values of the instances themselves.

None of the available cheminformatics visualization tools focuses on visualizing (Q)SAR model validation results as such. The authors of ChemSpaceShuttle [133] discuss how their tool can be used to embed compounds with two different class values (drug/non-drug) into 3D space. Different embeddings based on different sets of feature values were investigated to decide which feature set is most suitable for separating the compounds according to their class values. However, this work did not include (Q)SAR modeling. Moreover, the software neither draws compound structures nor computes compound feature values.

3 A Large-Scale Empirical Evaluation of Cross-Validation and External Test Set Validation in (Q)SAR

Despite the above mentioned OECD guidelines that describe how to design and validate (Q)SAR models best practices for model validation are under discussion in the (Q)SAR community. Many (Q)SAR researchers [98, 102, 103, 110] consider validation with a single external test set as the "gold standard" to assess model performance and they question the reliability of cross-validation procedures. In contrast, best practices employed in statistics and machine learning [93, 96, 101] and some (Q)SAR researchers [6, 114] do recommend cross-validation procedures. To clarify this discrepancy empirically, we have designed and performed a large scale experimental comparison of plain[1] cross-validation to external test set validation. The idea is to apply the validation techniques to the same dataset, and consequently use the final validated models on a large amount of unseen compounds. This process is repeated numerous times with different datasets, algorithms, feature types and splitting techniques.

In the following, we specify design and implementation of the workflow that is used to compare the validation methods (Section 3.1) and report the results of our experiments (Section 3.2). Subsequently, we discuss whether cross-validation can be employed as an external validation method in Section 3.3. Our approach is summarized in Section 3.4.

3.1 Experimental Workflow Design

The workflow is illustrated in Figure 3.1. The *original dataset* is repeatedly split into a *working dataset*, and a large *reference dataset*. We will refer to this split as reference split, to avoid confusion with the external test set splits. The reference split is repeated 100 times. For each repetition, k-fold cross-validation and external test set validation are applied to the working dataset. We have chosen k=10 for the k-fold cross-validation as this is the most commonly used value. For external validation, we selected split sizes 10%, 30%, and 50%. Subsequent to validation, the final model from the respective validation methods is used to predict the compounds in the reference dataset. This allows us to make the following two comparisons:

1 As opposed to applying cross-validation for internal validation combined with external validation.

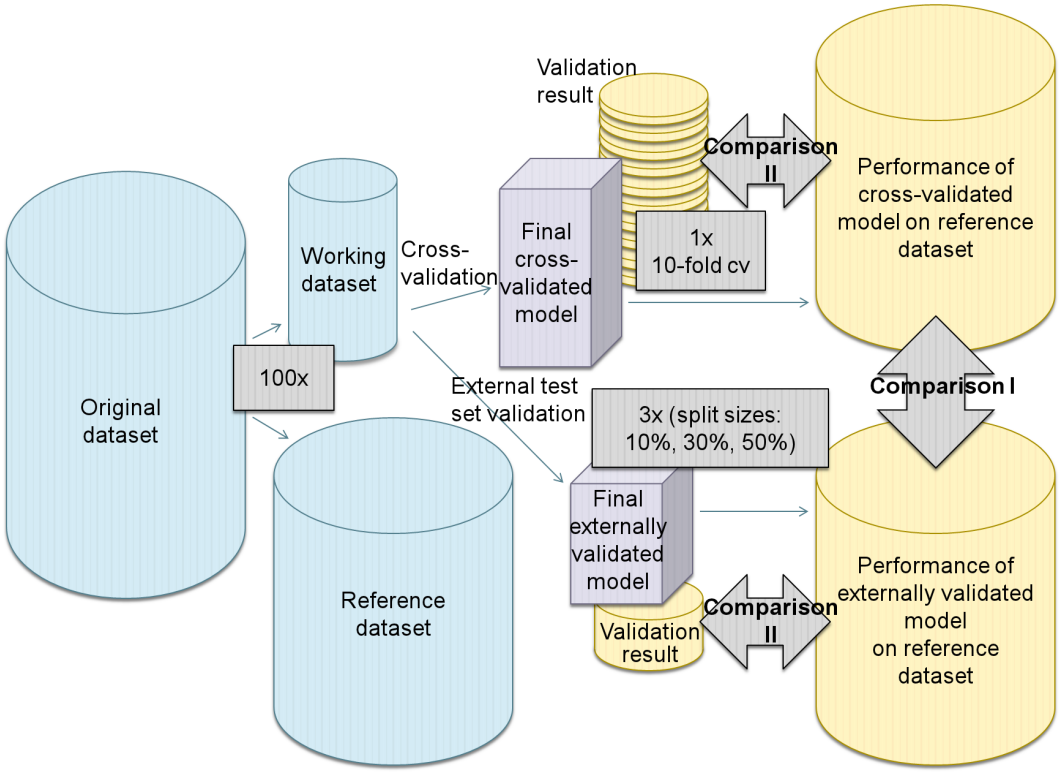


Figure 3.1: Experimental workflow. (The size of the icons indicates the amount of data, e.g., the final cross-validated model was built with the complete working dataset, the final externally validated model used less data.)

- *Comparison I* compares the actual performance of the final models from the different validation methods on the reference dataset. This will show the effect of using less data for model building as it is done by external test set validation.
- *Comparison II* evaluates the predictivity estimate from each validation method and the respective actual performance on the reference data. This will detect which validation method is more accurate and less variable.

The above described workflow resembles a hold-out validation. It is suitable to compare both validation methods with hold-out validation, because of the high number of repetitions and the large size of the input and reference datasets.

Please note that model building is done without optimization: we did not perform any feature selection nor parameter optimization. This step is dropped, because in the work presented here, we focus on performance evaluation instead of model selection. Therefore, we omit a possible nested loop of cross-validation within the cross-validation, and we omit internal validation within the external test set validation. This reduces the external test set validation to a single training-

Name	Description	Endpoint	Size(active)	Reference
Mutagenicity	Kazius-Bursi-Mutagenicity dataset	binary	4098 (2294)	[161]
KCNK9	Inhibitation of the two-pore domain potassium channel	binary	4914 (2094)	PubChem AID: 492992
KCC2	Identification of Novel Modulators of Cl- dependent Transport Process	binary	3382 (1815)	PubChem AID: 1714
Ames	Benchmark Data Set for In Silico Prediction of Ames mutagenicity	binary	6503 (3497)	[162]
MTP	Karthikeyan-Melting-Point dataset	numerical	4397	[163]
VP	EPI suite (MPBPWIN) test dataset: vapor pressure	numerical	2916	[164]
WS	EPI suite (WSKOWWIN) test dataset: water solubility	numerical	2293	[164]
Rat	Rat Acute Toxicity by Oral Exposure	numerical	10168	[165]

Table 3.1: Datasets used for the experiments. (The Ames and Rat dataset have been added to experiments at a later point of time, due to computational limitations we could run the experiments for those two datasets only 20 instead of 100 times.)

test split, which is sufficient as we only compare performance estimates of cross-validation and external test set validation. Please refer to the Section 3.3 whether cross-validation can be employed as an external validation method.

The workflow was executed 32 times, i.e. using 8 different datasets, with two algorithms and two different features types each. Table 3.1 provides a list and description of the employed datasets. Half of the datasets have a numerical endpoint, the others a binary nominal endpoint (e.g., active, inactive). Consequently, the corresponding (Q)SAR models are either regression or classification models.

For building (Q)SAR models, we employed the WEKA machine learning tool [166] (version 3.7.2). We selected two different, well-known and well-performing prediction models for each endpoint type: a support vector classifier (SMO) and random forests for classification, regression model trees (M5P) and support vector regression (SMOreg) for the numerical endpoints. All algorithms have been used with default settings.

We have created two different types of features for the compounds. The feature types were employed separately, each prediction model was applied twice on each dataset (once for each feature type). The first feature type is physico-chemical de-

Parameter	Values
Working dataset size	*500*, 100
External splitting method	*random*, stratified
Reference splitting method	*random*, outlier-sampling
Applicability domain	*disabled*, enabled

Table 3.2: Extended parameters to configure the workflow (default values are accentuated).

scriptors computed with the Chemistry Development Kit (CDK, version 1.7.4) [23]. We have computed all available descriptors, apart from the Ionization potential which took about 5 seconds per compound to be computed. After removing all features that failed to compute for at least one compound, and those that had equal values for all compounds, this method produced around 150 features for each dataset. The second employed feature type are structural fragments. These are binary features that either occur or do not occur in a compound. As structural fragments we have computed backbone refinement classes (BBRCs) with the tool fminer (as introduced in Secion 2.1.2.2). The fminer tool has a built-in filter mechanism to return only endpoint-correlated and non-redundant features. We use a relative minimum frequency of 5% and a significance level of 5% for the built-in filter (χ^2-test / Kolmogorov-Smirnov test). We further have added a maximum-cutoff of 1000 fragments, in case too many significant fragments have been found. As the numerical features do not depend on the endpoint (i.e., target variable), we have computed all features once for the entire dataset at the beginning of the workflow. The structural fragments are mined in a supervised (endpoint dependent) fashion, and are therefore created dynamically for each training set (to avoid information leakage).

3.1.1 Extended Workflow Parameters

Table 3.2 shows further parameters that were used to modify the experimental workflow.

We start the experiments by setting the *working dataset size* to 500 compounds. This is a common size when working with (Q)SAR models, and still leaves at least 3 times as many compounds in the reference dataset. We will then reduce this size to 100 to show the effect of a smaller dataset size on the selected validation technique.

The *external splitting technique* can be either random or stratified. Stratification ensures that descriptor and endpoint values are distributed evenly in the test and in the model building set. We have employed a technique described as stratified random sampling [113, 167]: first the data is separated into subgroups of similar compounds (clusters), and subsequently the compounds are sampled from each cluster.

The number of compounds sampled from the cluster corresponds to the particular cluster size. For clustering, we are using the dynamic tree cut method [168] that can be forced to compute a large enough number of clusters to achieve a good partitioning according to feature values. The input for the clustering algorithm is a distance matrix, indicating the pairwise distance of the dataset compounds. The distance matrix computation depends on the feature type: for the numeric features, we computed the Euclidean distance based on a principal component analysis (PCA) transformation of the feature values. For binary structural features the distance matrix is based on the Tanimoto similarity computed with the substructure fingerprints.

The *reference split* is done randomly or by using outlier splitting. Random splitting will yield on average a representative sample, as the reference split produces sets of 100 or 500 compounds (compared to the external split that produces much smaller sets). Therefore, validation and reference sets will have a similar feature range and distribution. To emphasize the influence of the selection bias on the validation method, we have implemented the sampling of an outlier distribution. This method ensures that the working dataset has a different distribution than the reference dataset. The outlier sampling method works as follows. At first 5% of the compounds are sampled randomly from the complete dataset. After computing the pairwise distance between all of the selected compounds the maximum outlier is selected (the compound with the highest average distance to all other compounds). This outlier compound is used as centroid of the outlier distribution. To sample the outlier distribution, we perform a probability weighted sampling from the whole dataset. Probabilities are computed according to the distance from each compound to the centroid compound: the closer a compound is to the outlier centroid compound, the higher is its probability of being selected[2]. The distance between two compounds has been computed as described above.

We have applied our models with *applicability domain* enabled and disabled (default). There are various different methods to compute the applicability domain [169]. We have chosen a model independent, distance-based approach: compounds are within the applicability domain of the model if their distance to the training dataset centroid is not too large. In more detail, a compound c is within the model applicability domain, iff $\mathtt{distance}(c, \mathtt{centroid}) \leqslant 2\times \mathtt{median}_{i\in\mathtt{train}}(\mathtt{distance}(c_i, \mathtt{centroid}))$. Centroid and distance computation depend on the feature type, as described above. For binary structural features, a consensus fingerprint is computed as centroid. The consensus fingerprint has each bit activated that is active in least 10% of the training compounds.

2 The sample probability decreases exponentially with growing distance.

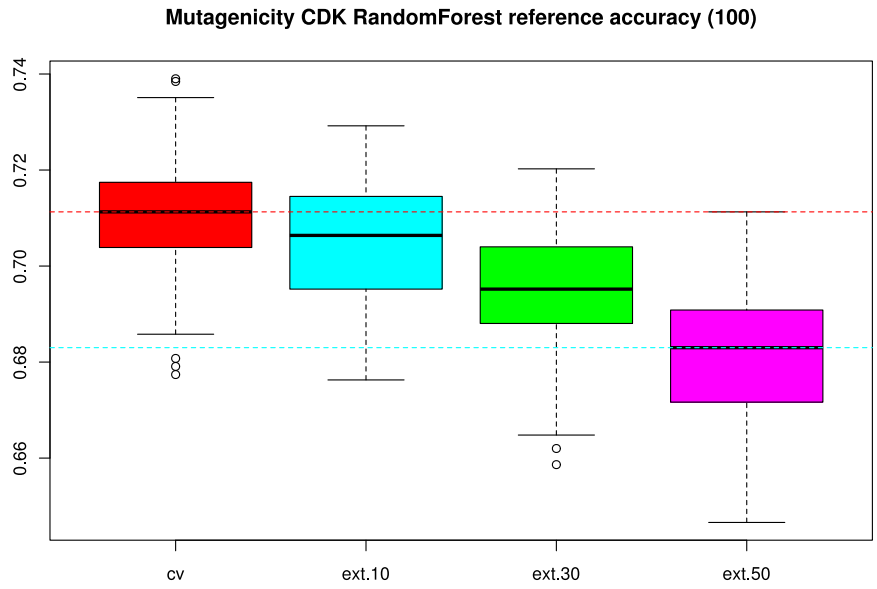


Figure 3.2: Comparison of the model performance on the reference dataset, using cross-validation (cv) and external test set validation (ext.10 - ext.50).

3.2 Experimental Results

We are using prediction accuracy and the concordance correlation coefficient to measure the performance of the respective classification or regression models. Accuracy is suitable for measuring the classification result as the datasets with binary endpoint values are well balanced[3]. For regression, we preferred the concordance correlation coefficient (CCC) to the conventionally more often used R^2-measure, as it has been shown to be a more stable statistic [170, 171]. (We have computed the results using R^2 as well, and could not find differences regarding the comparison of the studied validation methods.) Due to computational constraints, the experiments for 2 out of 8 datasets could only be repeated 20 instead of 100 times.

3.2.1 Performance on Reference Dataset

The performance of the reference dataset describes how performant the final model is when applied to unseen compounds (see *Comparison I* of the workflow, Figure 3.1). When using a 10-fold cross-validation, the final reported model is trained with 100% of the working dataset. External test set validation makes use of only

3 The least balanced dataset has 43% active compounds.

	cv	ext.10	ext.30	ext.50
cv		28(16)/4	32(30)/0	32(31)/0
ext.10	4/28(16)		31(28)/1	32(30)/0
ext.30	0/32(30)	1/31(28)		31(26)/1
ext.50	0/32(31)	0/32(30)	1/31(26)	

Table 3.3: Win-loss statistics for model performance (significant wins/losses are in brackets, measured with a paired t-test, significance level 5%).

	cv - ext.10	cv - ext.30	cv - ext.50
>5%	1	3	9
3-5%	0	3	5
1-3%	4	19	17
0-1%	23	7	1
0–1%	4	0	0
-1–3%	0	0	0
-3–5%	0	0	0
-5–100%	0	0	0

Table 3.4: Median performance loss for externally-validated model using a single test set, compared to cross-validation (for all 32 experiments each).

	cv	ext.10	ext.30	ext.50
cv		21(3)/11	29(11)/3	29(18)/3
ext.10	11/21(3)		29(9)/3	28(18)/4
ext.30	3/29(11)	3/29(9)		26(9)/6(1)
ext.50	3/29(18)	4/28(18)	6(1)/26(9)	

Table 3.5: Win-loss statistics for the variance of the model performance. A win corresponds to lower variance. Significance (shown in brackets) was calculated via F-test, significance level is 5%.

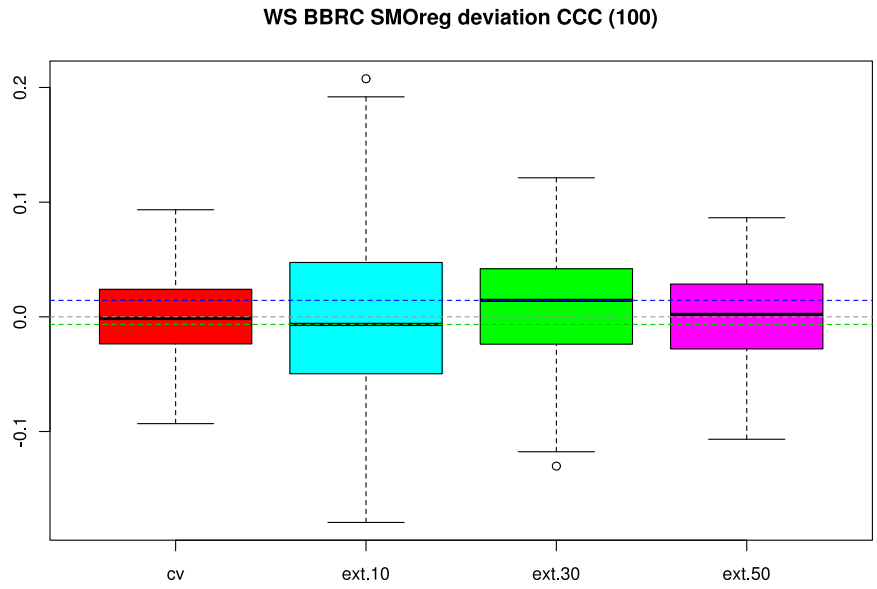


Figure 3.3: Example result for the deviation of the validation score from performance on reference data, for cross-validation and external test set validation (with difference split sizes).

90-50% of the data, as 10-50% of the data is split away for validating the model. We have used default settings as specified in the previous section.

Our experiments confirm, as expected, that the more data is used for model building, the better the model is. This is exemplified in Figure 3.2, that visualizes the accuracy of cross-validation and external test set validation, using three different split sizes. Table 3.3 contains win-loss statistics for all 32 experiments: the correlation between training dataset size and model performance holds for almost all experiments. Please note that the performance degradation for externally-validated models using a single test set, compared to cross-validated models could be shown to be significant in 16, 20, and 31 cases (for corresponding split sizes of 10, 30, and 50%). The magnitude of degradation depends on the training set size as well, as indicated in Table 3.4. An average performance loss compared to cross-validation of more than three percentage points of accuracy or CCC was estimated in 1, 6, and 14 cases (again corresponding to split sizes of 10, 30, and 50%). The same relation holds for the variance, as shown in Table 3.5: the more data is used for model training, the less variable the performance of the model is.

3.2.2 Deviation of Predictivity Estimate from Reference Dataset

The workflow allows to compare the predictivity estimates of the different validation techniques with the actual performance on unseen compounds (see *Comparison II* in Figure 3.1). To this end, we compute the distribution containing the pair-wise differences between the validation predictivity estimate and the reference dataset performance. An example result is shown in Figure 3.3 that displays the deviation for cross-validation and external test set validation with different split sizes: all four distributions have a median deviation around zero. Taking all 32 experiments into account, the median deviation is only higher than three percent for two or three experiments (for each validation method), as shown in Table 3.6. Furthermore, there is no visible trend for overestimation or underestimation for external test set validation. On the contrary, cross-validation underestimates the real model performance in 25 of 32 experiments. This pessimistic bias of 10-fold cross-validation is in agreement with the machine learning literature [97], and can be explained by the fact that it uses models that are trained with 90% of the data for validation, but the final cross-validated model uses 100% of the data. It is due to this bias that cross-validation has in many experiments a slightly higher median deviation compared to external test set validation (with 30 or 50%, see Table 3.7). However, the variance comparison is clearly in favor of the cross-validation approach (see Table 3.8). It should be noted that the variance could probably further be reduced by repeating the cross-validation. By design, the external test set validation can only be performed once.

The results of this section and the previous section show that external test set validation faces a trade-off regarding the test set size. Choosing a small test set reduces the model performance degradation, but gives a very variable external performance estimate. Choosing a large test set yields a less variable predictivity estimate, but at the expense of a less predictive model.

3.2.3 Additional Experimental Results

So far the workflow parameters have been used with default settings, as described in the methods section. The following sections show the influence of changing the configuration. For the remaining experiments, we focus on the numerical feature type. This will reduce the number of experiments to 16.[4]

4 Due to computational limitations, the reference split is only repeated 20 times instead of 100 times. Whenever the modified settings are compared to default settings, the exactly same 16 experiments with 20 repetitions only are used for the comparison.

	cv	ext.10	ext.30	ext.50
5-100%	0	0	0	0
3-5%	1(1)	3(2)	2(2)	1(1)
1-3%	1	5	4(2)	3(2)
0-1%	5	11	10	12
0–1%	20(4)	8	13(1)	14
-1–3%	4(4)	3(1)	3	2(1)
-3–5%	1	2	0	0
-5–100%	0	0	0	0

Table 3.6: Median deviation of predictivity estimate from performance on reference dataset. In brackets: number of experiments where the deviation is significantly higher/lower than zero, i.e. the validation method over-/underestimates (measured with t-test).

	cv	ext.10	ext.30	ext.50
cv		20(2)/12	10(2)/22(3)	10(3)/22(4)
ext.10	12/20(2)		7/25(1)	8/24(1)
ext.30	22(3)/10(2)	25(1)/7		16(1)/16(5)
ext.50	22(4)/10(3)	24(1)/8	16(5)/16(1)	

Table 3.7: Win-loss statistics for deviation from reference dataset. Win means the validation method has a lower median deviation. Significant differences are shown in brackets.

	cv	ext.10	ext.30	ext.50
cv		30(29)/2(2)	26(21)/6(3)	27(7)/5(5)
ext.10	2(2)/30(29)		2/30(27)	2/30(27)
ext.30	6(3)/26(21)	30(27)/2		3/29(12)
ext.50	5(5)/27(7)	30(27)/2	29(12)/3	

Table 3.8: Win-loss statistics to compare the variance of the deviation from the reference dataset. A win corresponds to a lower variance. Significant differences are indicated in brackets.

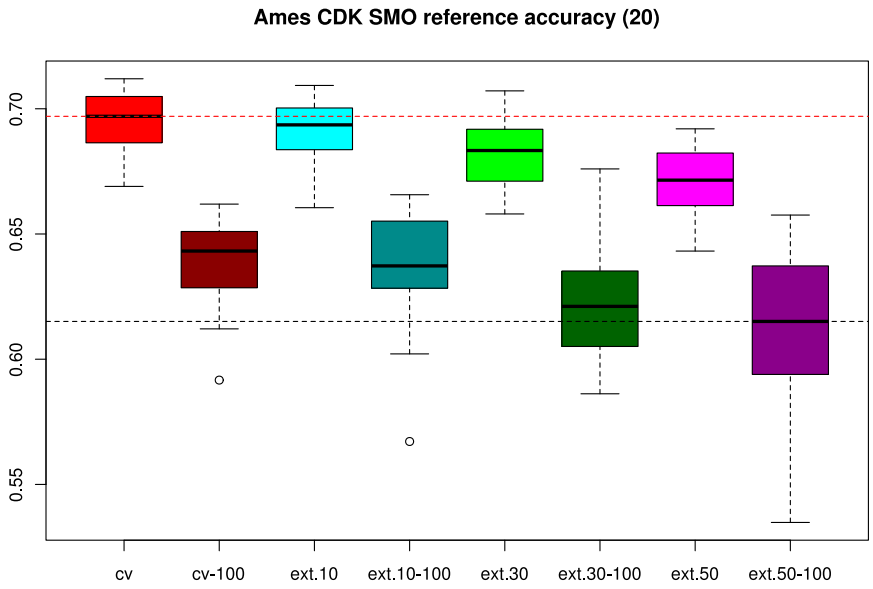


Figure 3.4: Model performance on reference dataset. Comparison of validation methods applied to 500 and 100 compounds (the latter is drawn in gray).

3.2.3.1 Reducing the Working Dataset Size from 500 to 100

Our results have previously shown that the model performance and variability depends on the dataset size. Consequently, the performance further drops and the variance increases in our results when using only 100 instead of 500 compounds for validation and training. Comparing the model performance of cross-validation and external test set validation yields the same results as previously analyzed. An example is shown in Figure 3.4.

Using less data for validation causes the deviation from the reference dataset to increase (see Table 3.9) and to be more variable. Furthermore, all validation techniques tend to overestimate the model predictivity. In contrary to the experiments with a dataset size of 500, cross-validation has a lower deviation compared to external test set validation. The results indicate that cross-validation is especially well-suited for small datasets.

3.2.3.2 Stratified Splitting for External Test Set Validation

Using stratified splitting instead of random splitting with a working dataset size of 500 shows only minor effects in our results. We presume that the sample sizes are already large enough to produce test datasets of very similar distributions as the training dataset. In contrast, applying stratified splitting when externally validating

	cv	ext.10	ext.30	ext.50
5-100%	1	9(3)	3(2)	4(3)
3-5%	2(1)	5	1	2
1-3%	2	0	6	5
0-1%	3	0	1	0
0–1%	3	0	1	2
-1–3%	4	1	4	3
-3–5%	1	1	0	0
-5–100%	0	0	0	0

Table 3.9: Median deviation of predictivity estimate from performance on reference dataset with a working dataset size of 100 (significantly higher/lower deviation as zero is shown in brackets).

	cv	ext.10-strat	ext.10	ext.30-strat	ext.30	ext.50-strat	ext.50
cv		16(5)/0	12(1)/4	15(10)/1	14(10)/2	16(13)/0	16(13)/0
ext.10-strat	0/16(5)		9/1/6(2)	14(6)/2	12(6)/4	15(11)/1	15(11)/1
ext.10	4/12(1)	6(2)/1/9		11(5)/5	14(9)/2	15(12)/1	15(11)/1(1)
ext.30-strat	1/15(10)	2/14(6)	5/11(5)		9(1)/7	15(6)/1	14(7)/2
ext.30	2/14(10)	4/12(6)	2/14(9)	7/9(1)		12(2)/4	13(10)/3
ext.50-strat	0/16(13)	1/15(11)	1/15(12)	1/15(6)	4/12(2)		13/3
ext.50	0/16(13)	1/15(11)	1(1)/15(11)	2/14(7)	3/13(10)	3/13	

Table 3.10: Win-loss statistics for model performance for random and stratified splitting with a working dataset of size of 100 (significant wins/losses in brackets).

	cv	ext.10-strat	ext.10	ext.30-strat	ext.30	ext.50-strat	ext.50
cv		12(2)/4	15(1)/1	11(1)/5	12(1)/4	6/10	10/6
ext.10-strat	4/12(2)		8(1)/8(1)	3/13(1)	5/11(2)	1/15(2)	5/11(2)
ext.10	1/15(1)	8(1)/8(1)		3(1)/13	2(1)/14	2/14(1)	2/14(1)
ext.30-strat	5/11(1)	13(1)/3	13/3(1)		10/6	4/12(1)	9/7
ext.30	4/12(1)	11(2)/5	14/2(1)	6/10		3/13	6/10
ext.50-strat	10/6	15(2)/1	14(1)/2	12(1)/4	13/3		12/4
ext.50	6/10	11(2)/5	14(1)/2	7/9	10/6	4/12	

Table 3.11: Win-loss statistics for deviation from reference dataset for random and stratified splitting with a working dataset of size of 100 (significant wins/losses in brackets).

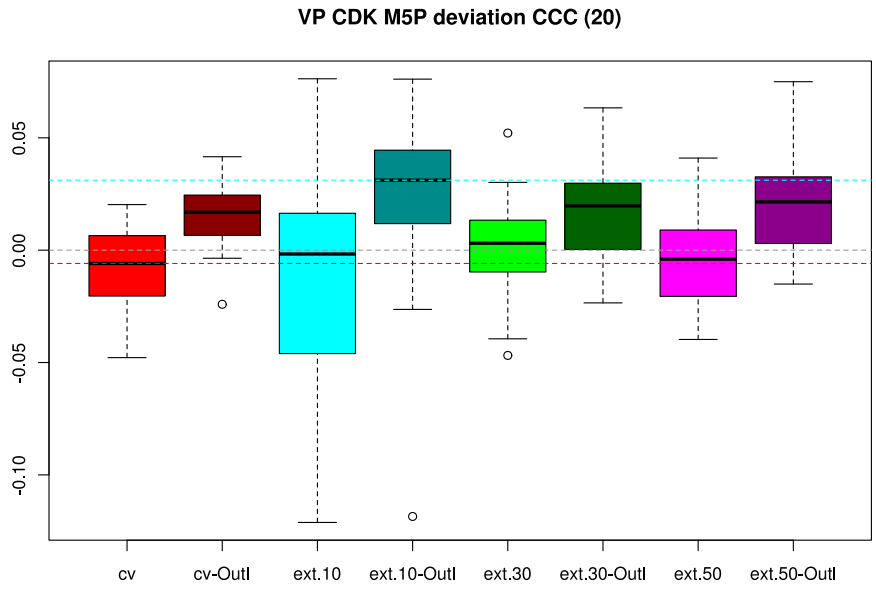


Figure 3.5: Deviation of model performance from actual performance on reference dataset. Comparison between random external test set splitting and outlier splitting (the latter is drawn in gray).

a dataset using 100 compounds, improves the validation result when splitting away 30% or 50% of the data: it produces better models and decreases the deviation from the reference datasets, as shown in Tables 3.10 and 3.11.[5]

At the same time the results show that despite improving external test set validation via stratified splitting, cross-validation still produces better performing models and has a lower deviation of the validation performance from the reference dataset. Only stratified splitting with 50% of the data as test set has a slightly lower deviation (considering the win-loss statistics), but yields a worse model in each of the experiments.

3.2.3.3 Learning and Validating with an Outlier Distribution

We added outlier splitting to the reference split, to analyze the performance of the different validation techniques when the built and validated models are applied to unseen compounds of a different distribution. As expected, the model performs less good compared to splitting the reference data away in a random, unbiased fashion. However, the results show that using the complete dataset to build the

5 There was no improvement for the 10% split that uses only 10 of 100 compounds as test set. We assume that the stratification technique we have implemented did not work for that few compounds.

	cv	ext.10	ext.30	ext.50
5-100%	4(3)	5(3)	4(4)	3(3)
3-5%	1(1)	4(3)	4(4)	3(2)
1-3%	6(4)	3(1)	4(3)	4(2)
0-1%	1	1	1	3
0–1%	4	3	2	2
-1–3%	0	0	1	1
-3–5%	0	0	0	0
-5–100%	0	0	0	0

Table 3.12: Median deviation of predictivity estimate from performance on reference dataset when using outlier splitting for the reference split (significantly higher/lower deviation as zero is shown in brackets).

	cv	cv-AD	ext.10	ext.10-AD	ext.30	ext.30-AD	ext.50	ext.50-AD
cv		5(2)/11(4)	12/4	10/6(1)	13(3)/3	7(2)/9(2)	9(2)/7	3(1)/13(4)
cv-AD	11(4)/5(2)		13(3)/3	13(1)/3	14(7)/2	12(2)/4	12(5)/4	10(1)/6
ext.10	4/12	3/13(3)		5/11(2)	7/9	4/12(3)	4/12	2/14(3)
ext.10-AD	6(1)/10	3/13(1)	11(2)/5		10(3)/6	6/10	6(3)/10	3/13
ext.30	3/13(3)	2/14(7)	9/7	6/10(3)		2(1)/14(5)	4/12(2)	4/12(6)
ext.30-AD	9(2)/7(2)	4/12(2)	12(3)/4	10/6	14(5)/2(1)		7(4)/9(3)	8/8(2)
ext.50	7/9(2)	4/12(5)	12/4	10/6(3)	12(2)/4	9(3)/7(4)		4/12(8)
ext.50-AD	13(4)/3(1)	6/10(1)	14(3)/2	13/3	12(6)/4	8(2)/8	12(8)/4	

Table 3.13: Win-loss statistics for the deviation from reference dataset, with AD enabled and disabled, for the outlier distribution split. Significant wins/losses (t-test) are in brackets.

model (e.g. apply cross-validation) still gives better results compared to using less training data.

As the unseen data is from a different distribution, all validation methods did overestimate the model performance on the reference data (see Figure 3.5 for an example). Table 3.12 shows that this overestimate is especially severe for external test set validation. The pessimistic bias of cross-validation diminishes this effect to some degree, and gives therefore the more accurate performance estimates. Nevertheless, one can conclude that applying models to unseen compounds that are different from the compounds that have been used for validating, all validation methods fail to give a good performance estimate.

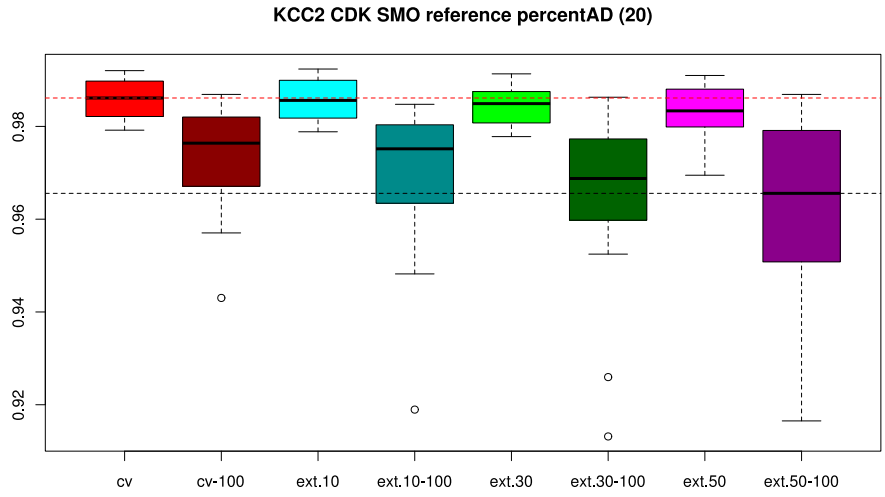


Figure 3.6: Ratio of compounds of the reference dataset inside the Applicability Domain of models built with 500 and 100 compounds (the latter is drawn in gray).

3.2.3.4 Employing Applicability-Domain

Applicability Domain (AD) methods ensure that a model only "interpolates, but does not extrapolate". Consequently, adding an AD approach to our workflow will probably improve the model performance as outliers are not predicted by the model. Secondly, it could decrease the deviation, in case outliers mainly occur in either the validation test set(s), or in the reference set.

The expected improvements could be measured in our results. When building models with 500 compounds, most of the test and reference data is inside the models AD (depending on the dataset about 95%). We therefore assume that the datasets we have used are rather homogeneous. The effect was larger when using a working dataset of only 100 compounds, as the AD of a prediction model that was built with less data is smaller (see Figure 3.6). Using the outlier distribution for learning provokes an increased effect with AD enabled as well. The model predictivity is higher, and the deviation form the reference dataset is lower (the latter is shown in Table 3.13). This shows the actual importance of AD in real-world applications: models are frequently applied to compounds that are not from the same distribution as the training and validation data.

Overall, AD improved the performance for cross-validation and external test set validation to an equal extent, and therefore did not affect the comparison between cross-validation and the external test set validation.

3.3 Discussion on Cross-Validation and External Validation

A more general definition of external and internal validation is that external validation is used to assess the final model performance, and internal validation is employed for model selection. The question arises whether cross-validation can be used for external validation at all. As it turns out, this depends on the precise definition on "external validation". Without aiming for a precise definition, one integral part is (a) that no information leakage is allowed: no test instance is allowed to be used during training (in which sense needs to be clarified below). The second integral part is that usually (b) the same method or type of model is validated that is actually used for making predictions on unseen data.

If external validation implies (i) that no instance from any test set is ever used for building the final model (see e.g. [98,103,116]), then no form of cross-validation (in which the complete dataset is repeatedly divided into disjoint training and test sets) can be regarded as external validation. If, however, we consider an external validation as valid, if, (ii) *in the training step(s) required for the performance estimation of the final model, none of the test instances is ever used*, then cross-validation could actually be used for external validation. Note that this latter definition excludes simple forms of information leakage, when instances from the test set are already seen during training. By contrast, it does not exclude cross-validation from external validation, as the above property must hold for each cross-validation training and test set.

A look into the literature suggests that the discussion of these topics is still ongoing: Recent papers [115, 172] recommended to use cross-validation for external validation instead of a single test set split. This external validation scheme requires internal validation to be applied on each training fold, for example with a nested loop of cross-validation. Eklund *et al.* [173] apply nested data partitioning loops: the inner loop is employed for model selection, the purpose of the outer loop is to assess the model performance. The authors note that it remains an open question how either a final model is constructed, or how a model can be chosen as final model. Hence, the approach to build the final model for the prediction of new instances (cf. (b) from above) is to some extent unclear in these studies. In contrast, Filzmoser *et al.* [174] apply a "repeated double cross-validation", to gain a reliable performance estimate for a PLS model using the optimal parameter. This parameter is eventually used to build a final model on the entire data, which therefore does not conform to (a) under the interpretation (i).

Whichever definition we want to adopt and whether or not we want to call cross-validation a method for external validation, our large-scale experimental results

have confirmed that cross-validation can be a performance estimation method of high accuracy and low variability in (Q)SAR studies.

3.4 Conclusions

Within the (Q)SAR community many researchers apply external validation using a single test set, mainly motivated by the common claim that it is the only way to reliably estimate model predictivity. Cross-validation in particular has the reputation of giving overoptimistic results compared to external test set validation.

We have designed a large scale experimental setup to compare both approaches, external test set validation to a plain cross-validation. In our experiments, the external test set validation often produced less performant models, as not all available information is used in model construction. Furthermore, our experiments have shown that the deviation of the predictivity estimate from the real performance on unseen compounds is less variable when using cross-validation. The results indicate that the external test set size should be chosen with care, as a small test dataset produces better models, while a large test dataset produces a less variable performance estimate.

The experiments further imply that cross-validation underestimates the real predictivity, and that it is in particular suited for small datasets. When applying external test set validation to small datasets, our experiments show improved validation results if the external test set is split in a stratified fashion. However, both validation methods fail to give good performance estimates if the distribution of the unseen compounds is different from the model building and validation data. In this scenario, using an Applicability Domain model is especially important.

We consider our work as once piece in the puzzle of how to validate (Q)SAR models correctly. There is certainly more than one way to assess the true predictivity of (Q)SARs. Given the experimental results reported in this chapter, one perhaps should not discard the cross-validation option from further consideration.

4 CheS-Mapper – Chemical Space Mapping and Visualization in 3D

This chapter describes how small molecule datasets can be visually explored in virtual 3D space with CheS-Mapper (Chemical Space Mapper) [8]. CheS-Mapper is a general, interactive, and open-source software that can be employed to inspect chemical datasets of small molecules. It maps the compounds into virtual 3D space and was designed to enable scientific researchers to investigate compounds and their features.

Compared to existing methods that we have reviewed in Section 2.4.3, CheS-Mapper is a unique combination of clustering, dimensionality reduction, and 3D viewer. The distinguishing feature of the tool is that each compound is represented by its (3D) structure instead of substituting it by a dot or a node. Moreover, the tool is generic, as it is up to the user to select the features and the similarity measure that are employed within the mapping process. Hence, the computed clusters as well as the 3D positions are based on a user-defined chemical similarity. In contrast to some existing open-source tools that are limited to a distinct operating system, depend upon the installation of an additional database or require a specific input format, CheS-Mapper is platform independent, requires no installation, and accepts a wide range of chemical formats.

The following Section 4.1 describes the application workflow, mapping process and the capabilities of the 3D viewer. The subsequent Section 4.2 outlines the implementation of the application. Finally, CheS-Mapper is applied to real-world datasets in Section 4.3.

4.1 Methods

The CheS-Mapper program is a graphical application that can be used to visualize chemical datasets of small molecules (compounds). The application is divided into two main parts, namely *Chemical Space Mapping* and *Visualization*. The overall workflow can be seen in Figure 4.1.

In the first part, the Chemical Space Mapping, the molecules in the dataset are pre-processed. The compounds are grouped together into clusters and embedded into 3D space. The user has to select the features of the compound that are used as input for the clustering and the embedding algorithms. The features employed for this can either be features already precomputed in the original dataset or can be

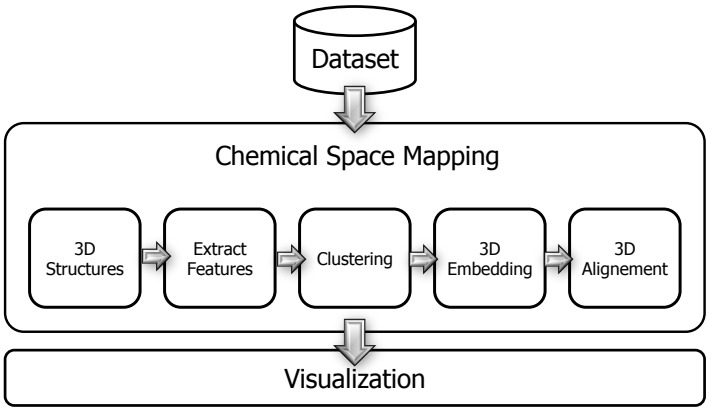


Figure 4.1: The CheS-Mapper application workflow is divided into two main parts: Chemical Space Mapping (can be configured with a wizard) and 3D Visualization.

computed by the application. A range of chemical descriptors or structural features is available, as well as various clustering and embedding algorithms. Finally, the compounds of each cluster can be aligned according to common substructures. The preprocessing is described in more detail in the following section.

The second part of the application, the visualization, is then used to explore the dataset in a virtual three dimensional space. Hence, the application is based on a molecular 3D viewer, that provides basic yet intuitive capacities like rotating and zooming. Special highlighting functions allow properties of the dataset to be visually highlighted. This includes highlighting of cluster assignments, compound features, endpoint values, and structural fragments. Furthermore, the compounds of each cluster can be superimposed to provide a better overview of the whole dataset, and to point out structural (dis-)similarities. Compounds and clusters of compounds can be deleted and exported in standard file formats for further analysis. The visualization is described in more detail in Section 4.1.2.

4.1.1 Configuring Chemical Space Mapping with the CheS-Mapper Wizard

As the mapping process offers a wide variety of options, like which clustering algorithm to use, or which 3D embedding technique to employ, we have designed a dedicated wizard to ease the use of the application. Each step is well-documented

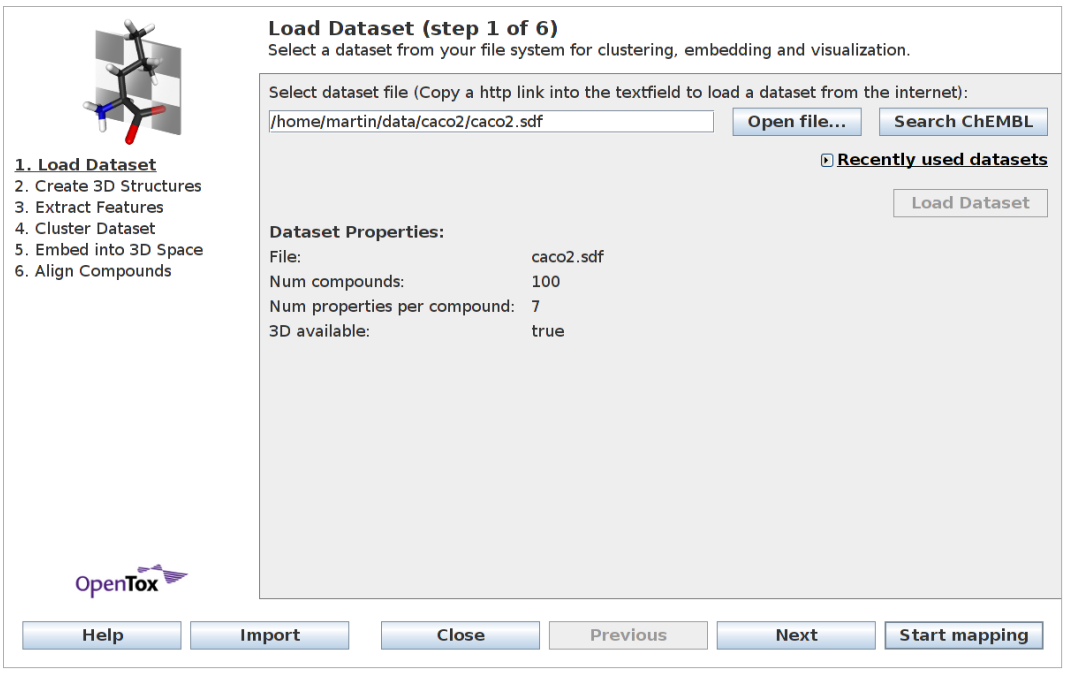


Figure 4.2: Wizard Step 1 - Load dataset: A wide range of chemical file formats is supported. Users can load datasets from the local file system, as well as directly from the internet.

and provides reasonable default settings to support the novice user. Expert users will appreciate that most algorithms are highly configurable. The current wizard configuration can be stored and exchanged, as documented in Section 5.2.1.5. Overall, there are six steps in the wizard, each step is described in one of the following sections.

4.1.1.1 Load Dataset

The dataset is selected in the first wizard step (see Figure 4.2). CheS-Mapper uses the Chemistry Development Kit (CDK) to read the dataset, which assures that a wide range of chemical formats is supported[1]. Additionally, CheS-Mapper accepts files in comma-separated values format (CSV, including a SMILES or InCHi column) that can be easily created with spreadsheat applications (like e.g. Microsoft Excel). The wizard is able to load datasets from the local file system, as well as from the Internet. When the dataset is entirely loaded, the wizard displays the number of compounds, the number of features for each compound, and a flag that indicates whether 3D structure information is already available in the dataset or has yet to be calculated (see the next section).

1 Details on supported dataset formats can be found on the project web-page `http://ches-mapper.org`.

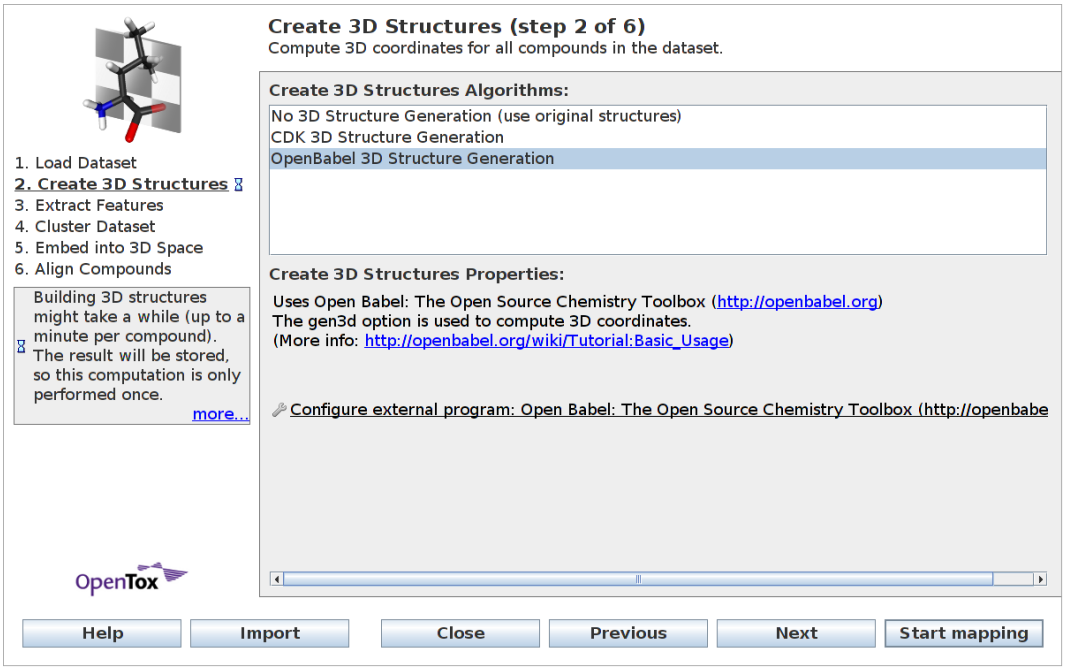


Figure 4.3: Wizard Step 2 - Create 3D Structures: In cases, where the 3D structures of compounds is not already available, users can calculate these 3D structures with the chemical libraries CDK or Open Babel.

Additionally, datasets allocated at an OpenTox [175] dataset service can be used. The OpenTox project provides a API definition for a predictive toxicology framework, a publicly available dataset service is for example the AMBIT web service [24].

Moreover, we have integrated ChEMBL Web Services[2]. This allows to directly search the ChEMBL database using SMILES similarity or search by substructure. SMILES similarity returns compounds that are similar to a provided canonical SMILES string exceeding a user-defined similarity cutoff score (e.g. 75%). Substructure search returns compounds that contain the provided canonical SMILES as substructure. The query result is used by CheS-Mapper as input dataset for the remaining workflow.

4.1.1.2 Create 3D Structures

In this wizard step, three-dimensional structure can be calculated for the compounds in case it is not already present in the original dataset. The wizard window for the structure generation can be seen in Figure 4.3, where the 3D builders of the chemical libraries CDK and Open Babel [22] are exposed. The CDK structure

2 The ChEMBL Web Services are documented online: `https://www.ebi.ac.uk/chembl/ws`

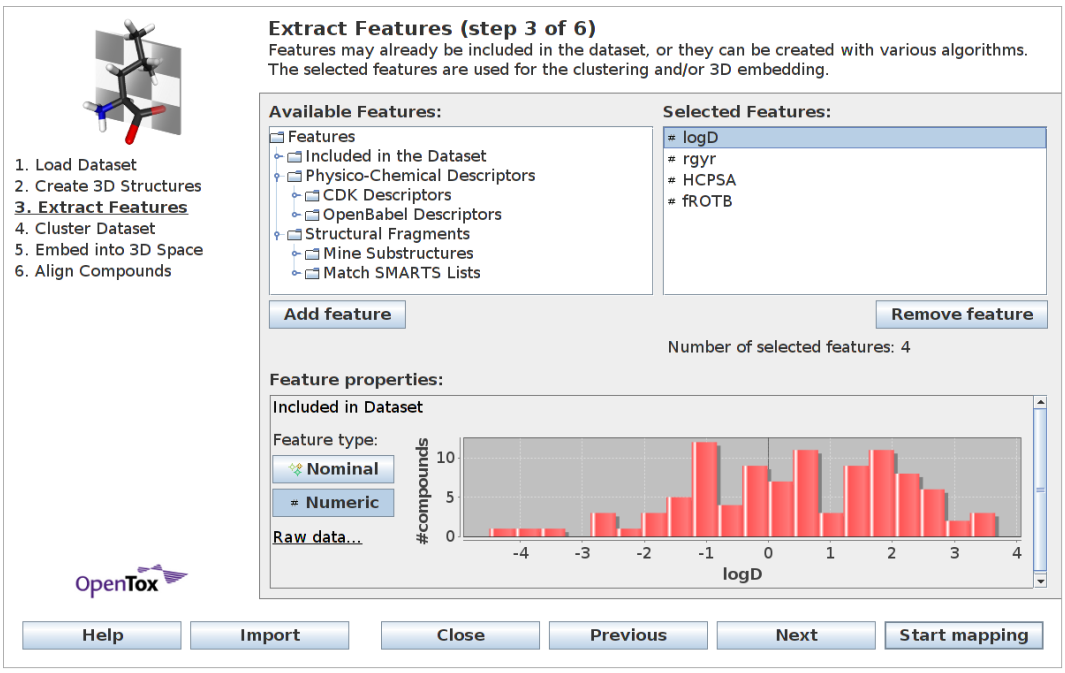


Figure 4.4: Wizard Step 3 - Extract features: Users can select features that are precomputed in the dataset or can be computed by CheS-Mapper. Compounds with similar feature values are likely to be clustered together and are embedded closer to each other in 3D space.

builder allows to choose from two different forcefields, MM2 [176] and MMFF94 [28]. However, at the current state the CDK builder (v1.4.18) tends to produce unreliable results, and the Open Babel builder should be preferred if available. The Open Babel 3D builder is using the MMFF94 forcefield. Building 3D structures is a time consuming process, and can sometimes take up to a few minutes per compound (see Section 4.4 for runtime experiments). However, this has to be done only once for each dataset, the results are cached for each single compound (e.g. by SMILES or MDL molfile) and the structures are available for subsequent runs of the program. In case the 3D structure information is already provided in the original dataset, one can select *No 3D structure generation*. In case no 3D structures are calculated, 2D flat structures are later shown in the viewer.

4.1.1.3 Extract Features

In step three of the wizard, the user can select features (see Figure 4.4) that are employed in the subsequent steps for clustering and 3D embedding. Thus, compounds with similar feature values are likely to be clustered together into the same cluster. Likewise, these similar compounds are embedded closer to each other in 3D space, while compounds that have mainly different feature values will have 3D positions far from each other.

The CheS-Mapper application distinguishes between numerical and nominal features. Nominal features separate compounds into distinct categories. For example, a nominal feature representing an activity could have values *active*, *moderately-active*, *inactive*. Numeric features have continuous floating point numbers as values, like e.g. *logP* or molecular weight. The software tries to guess the correct feature type when reading in the features present in the dataset. The feature type can be changed manually, if the feature value domain includes distinct numeric values and both feature types are possible.

Within this wizard step, it is also possible to pre-compute and plot the feature values (see the chart in Figure 4.4) in order to aid in the decision which features to select. Three different types of features are available:

INCLUDED IN DATASET Precomputed or experimentally measured properties that are available in the original dataset. Most (Q)SAR datasets have a biological or toxic endpoint that is stored in the dataset. Often, the datasets also contain features that have been computed by some external software.

PHYSICO-CHEMICAL DESCRIPTORS The Chemistry Development Kit and the Open Babel library provide a range of descriptor calculators that produce numerical features. This includes relatively simple features like the molecular weight, or the number of rotatable bonds as well as sophisticated chemical descriptors like the van der Waals volume.

STRUCTURAL FRAGMENTS Structural fragments are encoded as SMARTS strings. In the CheS-Mapper application, a structural fragment is represented as a binary nominal feature with value *match* or *no-match*: the feature has value *match* if the compound contains the fragment (i.e. it matches the SMARTS string), the value is *no-match* if the fragment is not contained in the compound. There are two different ways of computing structural fragments in CheS-Mapper:

- The first approach mines fragments dynamically. This method searches the compounds in the dataset for substructures that occur according to a user-defined minimum frequency threshold. Currently, the only implemented method is the Open Babel fingerprint *FP2*, that enumerates all linear fragments of size less or equal to *7* atoms.
- The second approach matches pre-defined lists of SMARTS fragments with the dataset compounds. Currently, CheS-Mapper has 22 SMARTS lists integrated. This includes two built-in lists from CDK (Functional Groups and Klekota-Roth Biological Activity). Moreover, fragment lists

have been extracted from Open Babel fingerprints (lists of functional groups from fingerprints FP3 and FP4, and the MACCS keys list), and from the ToxTree application [177]. Additional SMARTS lists present results from recent publications (e.g., alerts that model drug induced phospholipidosis [178]). A user-defined SMARTS file can be added by clicking the *Add SMARTS file* button.

The structural fragment computation can be configured by the user with various parameters: the minimum frequency defines a threshold for the number of compounds that must be matched by the fragment. A flag decides whether fragments should be skipped that occur in each compound. CDK or Open Babel can be selected as SMARTS matching software, whereas Open Babel is much faster and should be preferred.

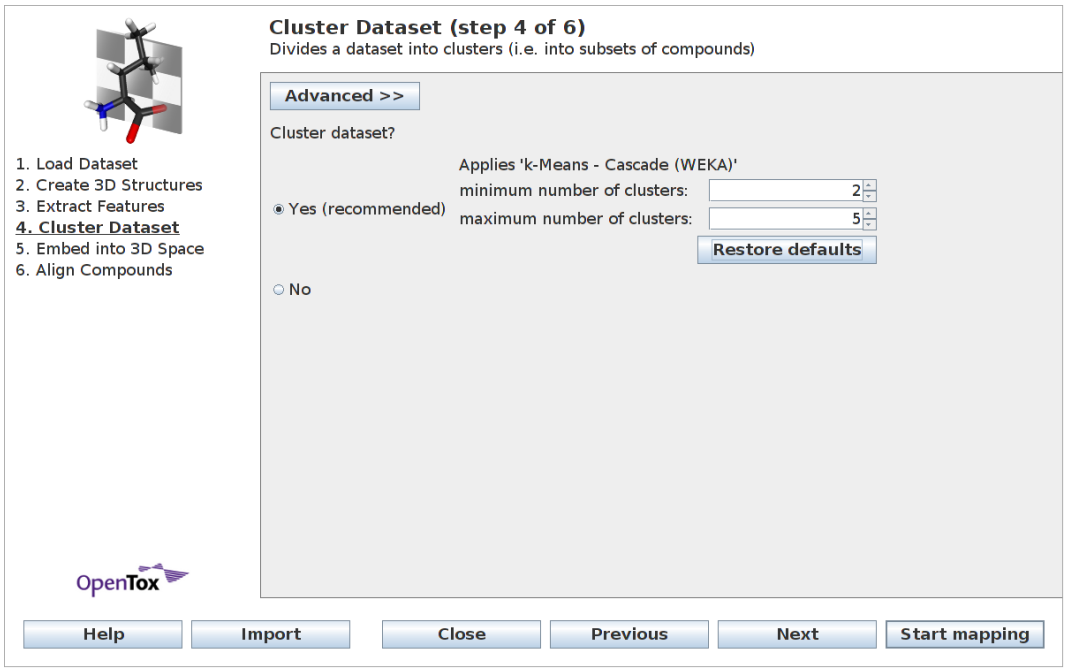
4.1.1.4 Cluster Dataset

The cluster settings are configured in wizard step 4. Clustering divides the dataset into subgroups. In general, compounds having similar feature values are grouped together in one cluster. Only features that have been selected in the previous step are used as input to the clustering algorithm. Clustering provides several benefits for the visualization: the user is given indication if subgroups exist within the dataset. Moreover, the dataset is easier accessible, as large datasets can hardly be shown on the screen at once. Furthermore, computing the maximum common subgraph inside a cluster can give further insights towards the structural similarity of the compounds.

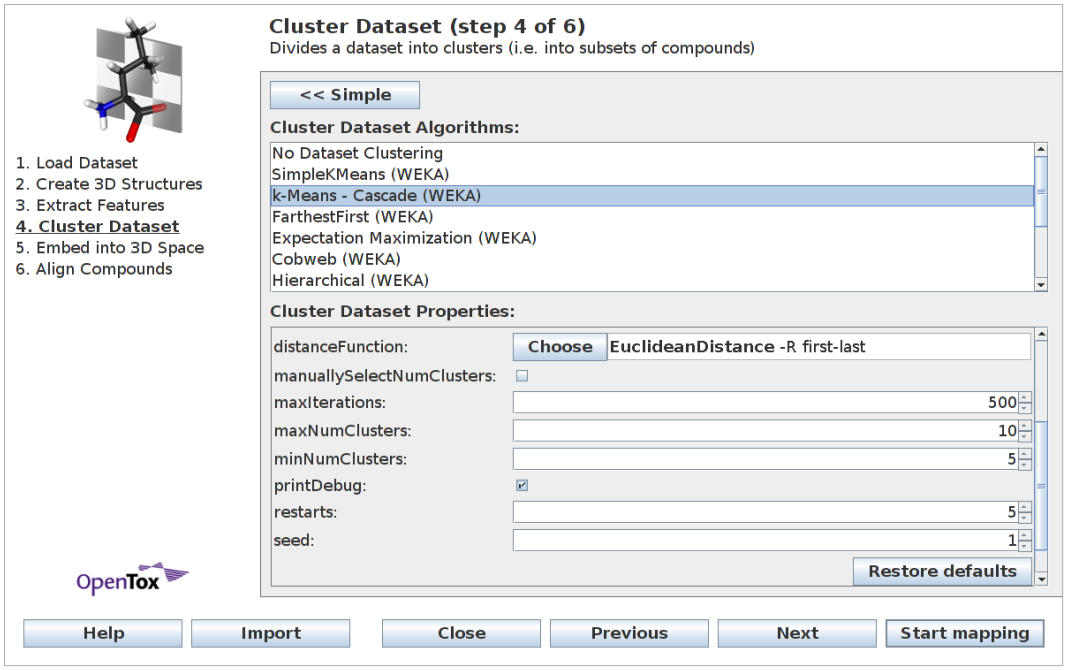
Figure 4.5a shows the *simple* view of wizard step 4. The user can set lower and upper bounds on the possible number of clusters. The Cascade k-Means algorithm is used for clustering, as described below. Alternatively, there is the choice to *not* to cluster the dataset. This is the only viable option if no features have been selected in the previous step.

The *advanced* view of wizard step 4 provides a range of algorithms to choose from (see Figure 4.5b). Cluster algorithms from the statistics library R [78], and the data-mining library WEKA [77] can be employed within CheS-Mapper. The cluster methods by R rely on a local installation of the R system on the users' computer, while the WEKA routines are built into the CheS-Mapper. Guidance on which algorithm to select can be found on the project homepage `http://ches-mapper.org`. The following cluster algorithms are available:

k-MEANS is a basic clustering technique assigning compounds to k randomly initialized centroids using some distance function. The compounds are as-



(a) With the simple view, users can set the lower and upper bound on the number of clusters.



(b) Within the advanced view, users can choose (and configure) a cluster algorithm from the statistics library R or the data-mining library WEKA.

Figure 4.5: Wizard Step 4 - Cluster Dataset

signed to each centroid, then the centroid is re-computed, being the center of all compounds belonging to this cluster. Subsequently, the compounds are re-assigned to the nearest centroid. This is repeated iteratively, until the algorithm converges. This method is available in two different implementations (R and WEKA). The major difference between both implementations is that the one by R includes the concept of random restarts. In both versions, the number of clusters (k) has to be given by the user. This is not ideal, as the user would have to find the right number of clusters manually.

CASCADE k-MEANS uses a range of values for k, performs random restarts of the k-Means algorithm, and automatically selects the best value. To this end, the quality of each cluster result is evaluated with the Calinski-Harabasz criterion [179]. This method is available as R implementation [180] and as built-in implementation that employs WEKA's k-Means method. The latter method is the default clustering algorithm in CheS-Mapper.

HIERARCHICAL CLUSTERING is a well established clustering approach that starts by considering each compound as a single cluster and subsequently merges two clusters at a time. A distance matrix with all pairwise distances between clusters is computed, to identify the pair which is closest in distance space, that will be merged. Various different set distance schemes to compute the distance between clusters are available. In the CheS-Mapper application, hierarchical clustering is provided in two implementations, R and WEKA. Again, this method has the drawback that the number of clusters has to be set to a fixed number by the user. Therefore, a dynamic version that automatically detects the number of clusters, implemented in R, is available [181].

EXPECTATION MAXIMIZATION models the data as mixture of Gaussians, i.e. each cluster is represented by one Gaussian distribution. This is a more general approach of the k-Means clustering that can model clusters of different spatial expansions (the Gaussians can have different standard deviations on covariances) in contrary to k-Means (where each compound is assigned to the closest centroid). The WEKA implementation of EM Clustering has a built-in functionality to auto-detect the number of clusters by using cross-validation: Starting at 1, it iteratively increases the number of clusters, using the log-likelihood of the test-fold compounds as quality measure. If the log-likelihood decreases, the previous number of clusters is used.

COBWEB is a hierarchical conceptual clustering algorithm, implemented in WEKA. It splits, merges, or inserts nodes in a hierarchical tree according to the category utility of a node [182].

FARTHEST FIRST is somewhat similar to k-Means, with the difference that the centroids are chosen as follows: It starts with a random data point, and chooses the point farthest from it. Subsequently, the next point that is farthest away from the already chosen points is selected until k points are obtained [183].

Moreover, the option 'Manual Cluster Assignment' can be selected to import a pre-computed cluster assignment that is already stored as a feature in the dataset file.

4.1.1.5 Embed into 3D Space

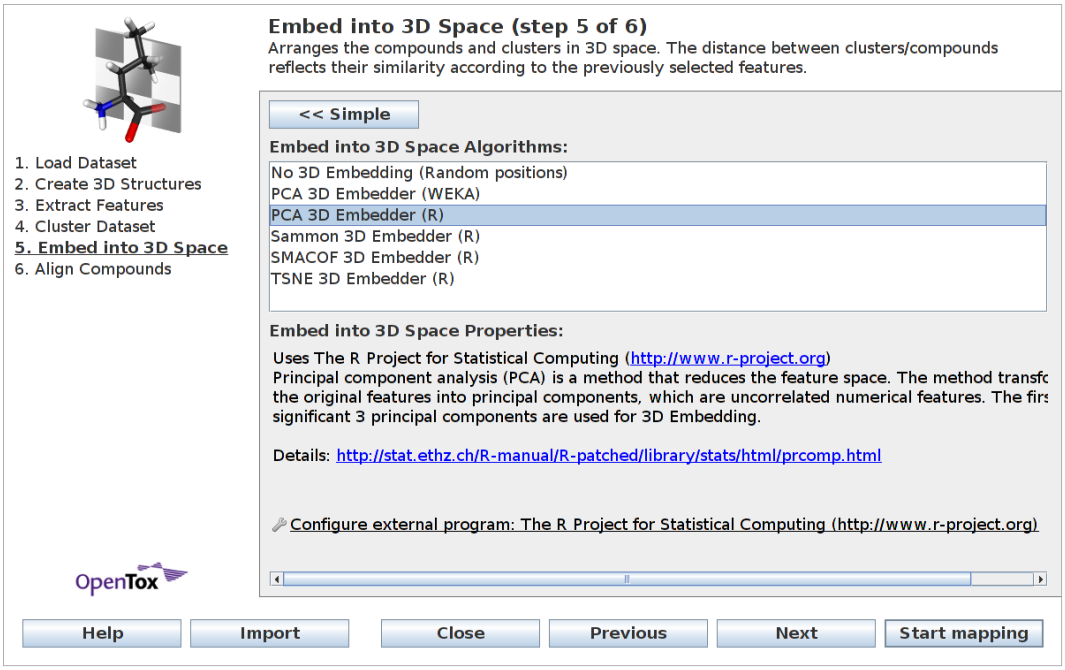
Wizard step 5 handles the 3D embedding of compounds. The embedding algorithm uses the feature values selected in step 3 as input. Accordingly, each compound is assigned a position in 3D space, such that the spatial proximity of two compounds reflects their similarity based on the features. Hence, the n-dimensional input space (n corresponding to the number of features selected in wizard step 3) is converted to 3 dimensions. We have discussed various techniques for dimensionality reduction in section 2.4.2.

Figure 4.6a shows the *simple* view for this wizard step: the user has to decide if the compounds should be embedded. In this case a principal component analysis is applied to calculate 3D coordinates. When deciding not to embed the compounds, the compounds will be arranged at random positions in 3D space. This is the only feasible method if there are no compound features available. Runtime experiments and guidance on which algorithm to select can be found at `http://ches-mapper.org`. The following embedding methods are available in the *advanced* wizard view (see Figure 4.6b):

PRINCIPAL COMPONENT ANALYSIS (PCA) is a method that reduces the feature space. Within CheS-Mapper, two different implementations are provided: WEKA and R. The PCA method is computationally not so expensive when compared to other embedding techniques and possesses therefore faster runtime than the methods below.

SAMMON'S NON-LINEAR MAPPING is an iterative multidimensional scaling method [128]. The algorithm is only available as R implementation that typically converges in about 50 iterations. Sammon embedding is the only technique in CheS-Mapper that allows to configure the (dis-)similarity measure. The default measure is the Euclidean distance, it could however be suitable to select another of the 48 currently available (dis-)similarity measures. For

(a) With the simple view, users can decide whether to embed a dataset according to its features values, or not.



(b) Within the advanced view, users can choose and configure an embedding algorithm.

Figure 4.6: Wizard Step 5 - Embed into 3D Space

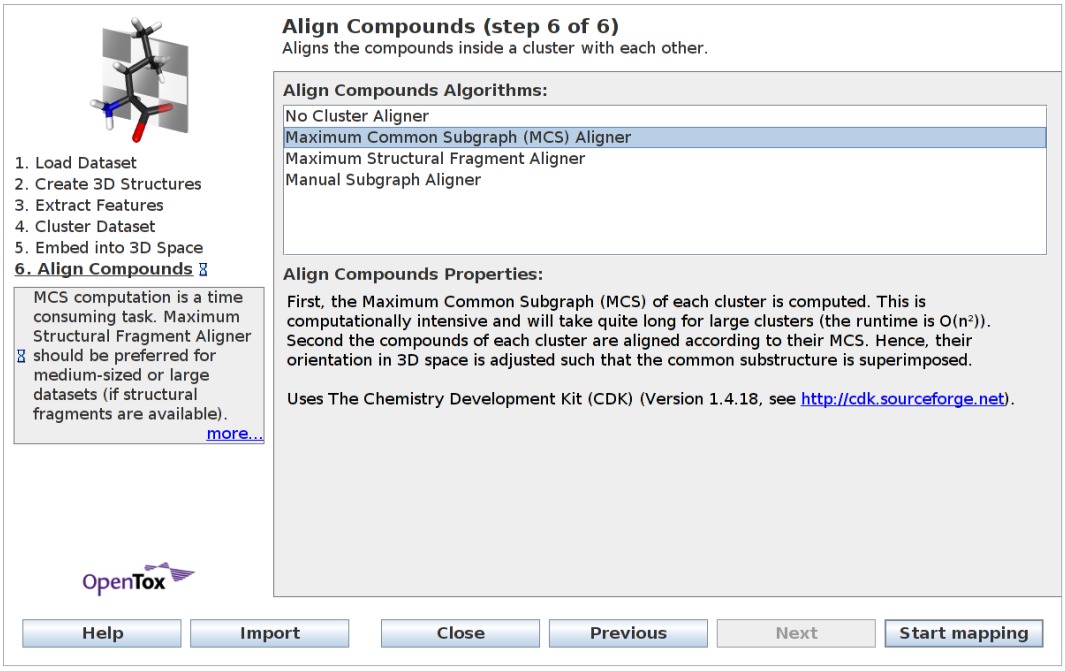


Figure 4.7: Wizard Step 6 - Align Compounds: Users can enable 3D alignment of compounds, which is appropriate when the dataset contains structurally similar compounds.

example, Tanimoto similarity might be a reasonable choice if only structural features are selected, as it ignores joint absences of fragments.

SMACOF employs multidimensional scaling using majorization using R [129]. For this method, CheS-Mapper converts the feature values to a distance matrix using the Euclidean distance. To reduce the runtime of this method, the user can set a maximum-number-of-iterations parameter.

T-DISTRIBUTED STOCHASTIC NEIGHBOR EMBEDDING (T-SNE) is also only available within R [132]. The parameter perplexity defines the effective number of neighbors that are used to compute the conditional probability distribution. Again, the maximum number of iterations can be reduced by the user in order to decrease the runtime.

In order to measure how well the computed 3D positions reflect the compound features values, CheS-Mapper can compute an embedding quality measure (see Section 5.2.1.1).

4.1.1.6 Align Compounds

The final wizard step allows to configure the alignment of the compounds within a cluster according to a common substructure (see Figure 4.7). Hence, a structural

fragment has to be computed that matches all compounds of this cluster. There are three methods to derive this structural fragment:

MAXIMUM COMMON SUBGRAPH (MCS) ALIGNER computes the maximum common subgraph of each cluster. The CheS-Mapper program uses the CDK library for the MCS computation: CDK provides a method to find all common substructures of a molecule pair. A list of all MCS candidates is created while this method is applied subsequently to all compounds within a cluster. In cases where this procedure does not produce an MCS, no alignment is performed.

MAXIMUM STRUCTURAL FRAGMENT ALIGNER requires structural features to be selected in the *Extract Feature* wizard step. It uses the largest structural feature that matches all compounds of each cluster for alignment.

MANUAL SUBGRAPH ALIGNER is an expert method that allows the user to specify the SMARTS fragment manually. This method differs from the two above, as the same fragment is used to align all clusters.

The compounds of each cluster are superimposed according to the common substructure and oriented in 3D space such that "they face the same direction". This is done pairwise for two compounds, each compound is seperately superimposed with the first compound of the cluster. The compounds are rotated until the two matching regions in both compounds have the smallest possible root mean square deviation (RMSD). The superimposition can be perfomed in CheS-Mapper using one of the chemical libraries CDK and Open Babel. CDK has the Kabsch algorithm [184] implemented, Open Babel provides the *obfit* command. The alignment method can be used to compare the structure of structurally similar compounds that are assigned to the same cluster (see example in Sections 4.3.2).

4.1.2 CheS-Mapper Viewer

The viewer window of the CheS-Mapper application is shown to the user when the mapping process is completed. It is based on the 3D library Jmol, and displays all compounds in virtual 3D space. The mouse can be used to zoom, rotate and translate the 3D viewer. Detailed documentation on how to control the viewer can be found on the project homepage.

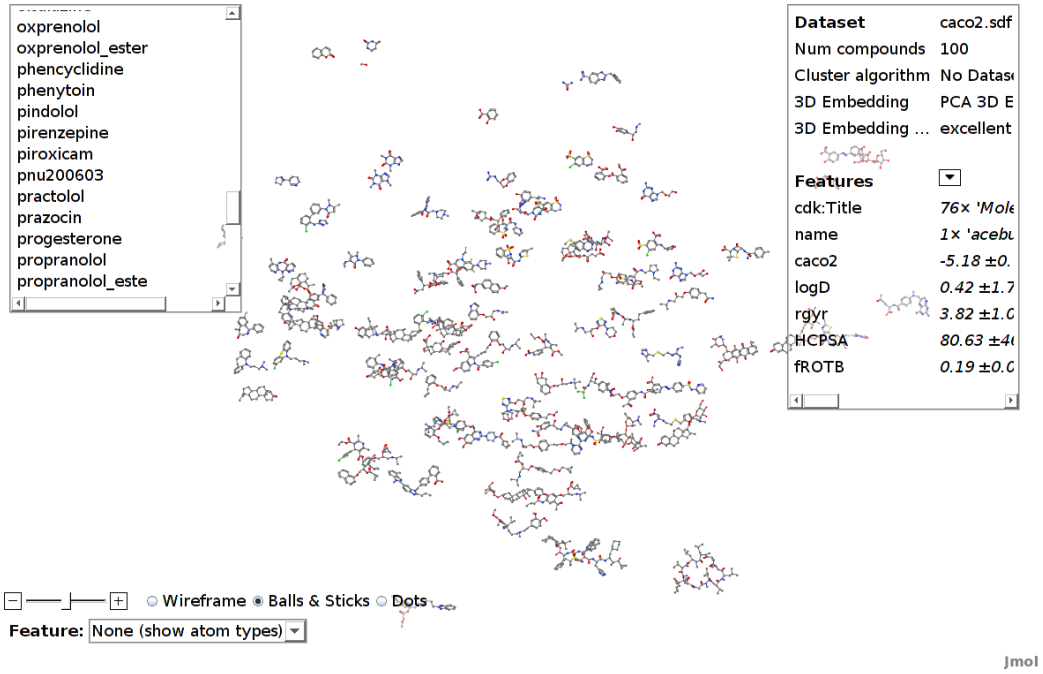


Figure 4.8: The CheS-Mapper viewer showing the Caco-2 permeability dataset. The compound list (on the left-hand side) can be used to select compounds. General dataset information and mean feature values are provided on the right-hand side. The control panel is located on the bottom left-hand side.

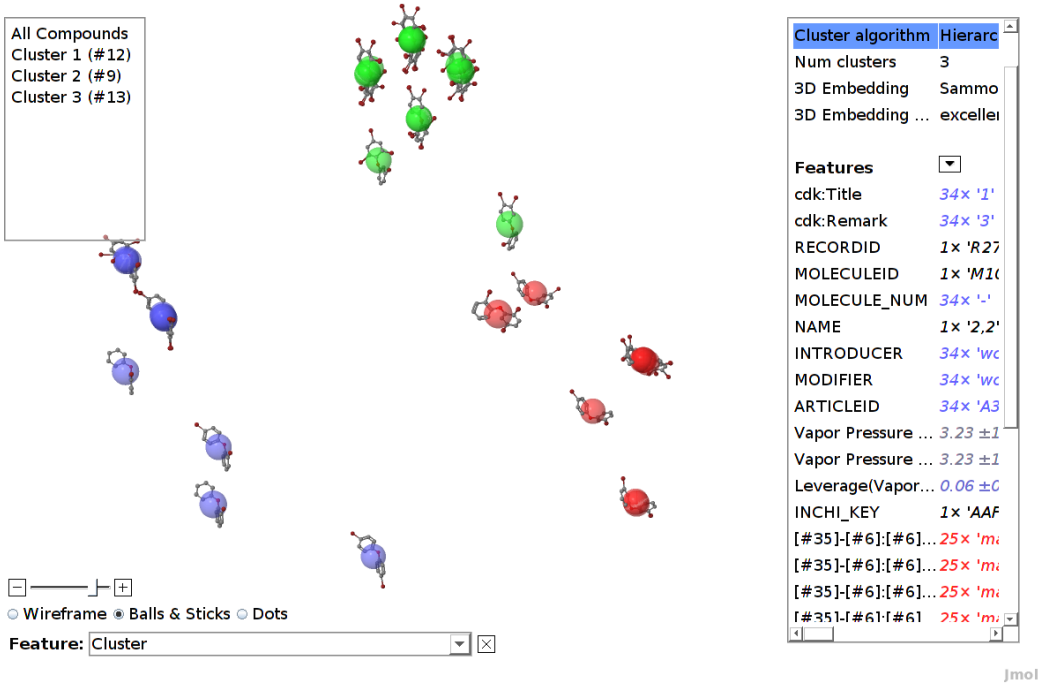


Figure 4.9: A dataset including polybrominated diphenyl ethers (PBDE) is clustered into 3 clusters. Structural features are employed for clustering and embedding.

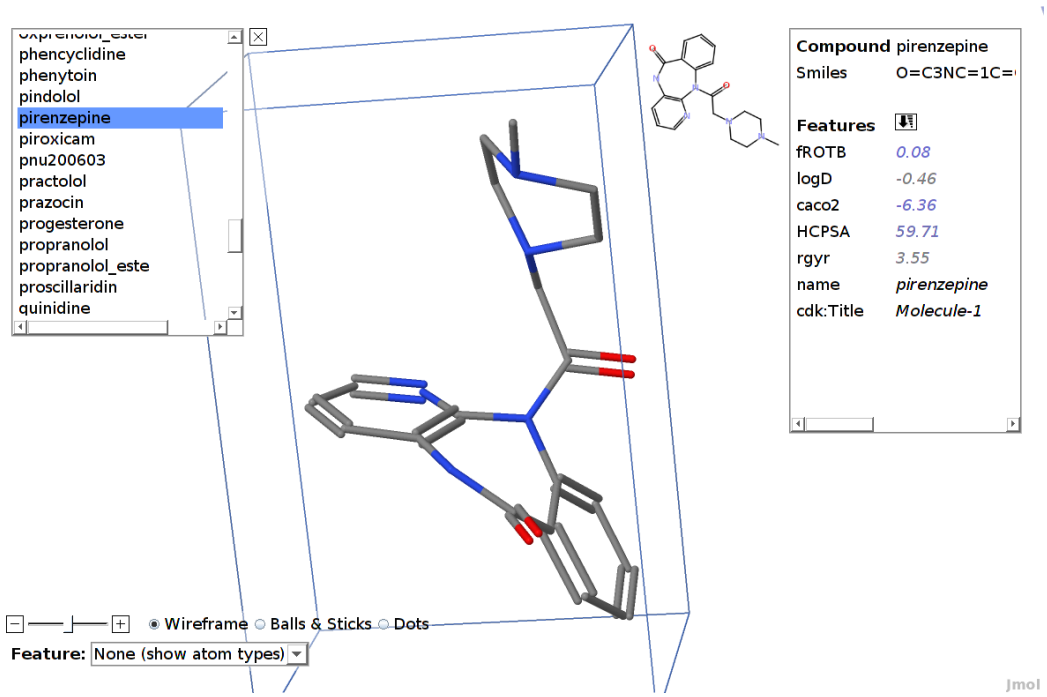


Figure 4.10: Zooming in on compound *pirenzepine*. The compound is depicted using the wireframe setting. Compound features are listed on the right-hand side. Feature values for *fROTB*, *caco2*, and *HCPSA* are relatively low and therefore colored in blue. The *fROTB* value of pirenzepine differs the most from the values in the entire dataset, therefore this feature is ranked at the top.

4.1.2.1 View Organization

CheS-Mapper shows the compounds of the embedded dataset in the center of the 3D viewer (see Figure 4.8). The 3D positions of compounds have been calculated by the selected embedding algorithm. Hence, compounds that are similar based on the selected feature values, will be located close to each other in 3D space.

Lists to select clusters and compounds are located at the top left side of the viewer (see a screenshot with clustering enabled in Figure 4.9). The cluster list can be utilized to select a cluster, or alternatively, the user can move the mouse over one of the cluster compounds. When clicking on a cluster, the view will zoom into this cluster, hiding the remaining dataset compounds. Subsequently, a single compound can be inspected by using the compound list or by hovering the mouse over the compound. When clicking on a compound, the view automatically zooms in on the structure (see Figure 4.10).

The information panel on the right-hand side of the screen shows the properties of the currently selected cluster or compound, or of the entire dataset. In particular, all features in the dataset and the (mean) feature value of the currently selected ele-

ment are presented. If a compound is selected, the provided information includes a 2D image, the compound SIMLES, and feature values (see right side of Figure 4.10). Please note that the feature values of clusters or compounds are colored: relatively high numeric feature values are drawn red, low values are colored in blue.

4.1.2.2 Highlight Clusters and Features

By default, compounds are drawn in standard CPK coloring[3]. When features are selected for highlighting, the compounds are colored according to their feature value. For numerical values, this is done with a color gradient, a continuous range of blue, white and red color. Accordingly, compounds with low feature values are colored blue, while compounds with high feature values are colored red (see Figure 4.11). Highlighting of nominal features assigns a distinct color to each value. Additionally, the numeric or nominal feature value of each compound can be labeled explicitly: the feature value is written next to each compound or cluster. Instead of changing the color of the whole compound, feature values can also be highlighted using translucent spheres that are overlayed over each compound (see e.g. Figure 4.9). This preserves the standard atom coloring of compounds. If clustering is enabled, the compounds can be highlighted according to their cluster assignment (see Figure 4.9). The user can switch manually between the different highlight modes using the drop down menu on the bottom left of the screen.

When a specific feature is selected, a chart showing the histogram of the feature values is displayed in the bottom right corner. When a cluster and/or a compound is selected, their feature values are indicated in the chart (see Figures 4.12 and 4.14).

As described above, a feature may represent a structural fragment encoded as a SMARTS string. When such a structural feature is selected, the SMARTS structure is matched and highlighted in the corresponding compounds (see Figure 4.12). Additionally, if cluster alignment according to a common fragment is enabled, this fragment can be highlighted as well (see Figure 4.13)).

4.1.2.3 Change Compound Depiction

Several depiction modes can be selected with the corresponding buttons at the bottom left of the viewer. *Wireframe* is the default option, that draws only the bonds of each compound (see Figure 4.10). *Balls & Sticks* additionally depicts atoms with spheres (Figure 4.13). The *Dots* option hides the compound structure and draws single spheres instead. This option might be preferable if the user wants to highlight

3 CPK is the the standard color convention for chemical elements, designed by and named after the chemists Corey, Pauling, and Koltun.

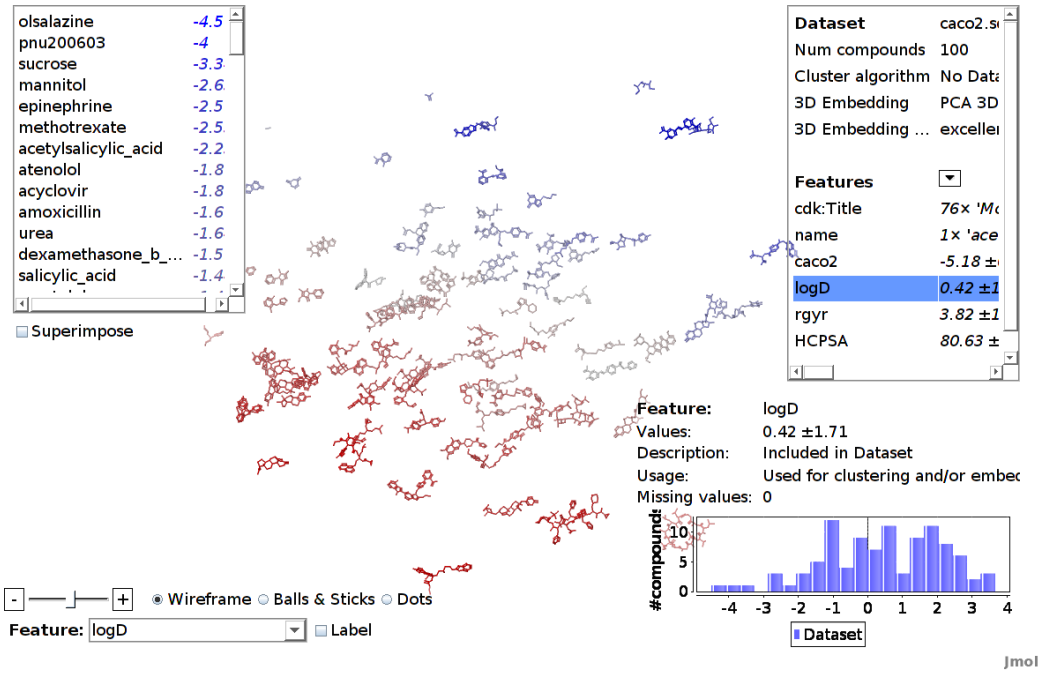


Figure 4.11: Highlighting *logD* feature values in the Caco-2 permeability dataset. The compound color has changed according to the feature value. Compounds with similar *logD* values are located close to each other in 3D space, as this feature was used for 3D embedding. The compound list at the left-hand side shows the *logD* value for each compound and the list is sorted according to the *logD* value. A histogram depicting the feature value distribution in the dataset is on the bottom right-hand side.

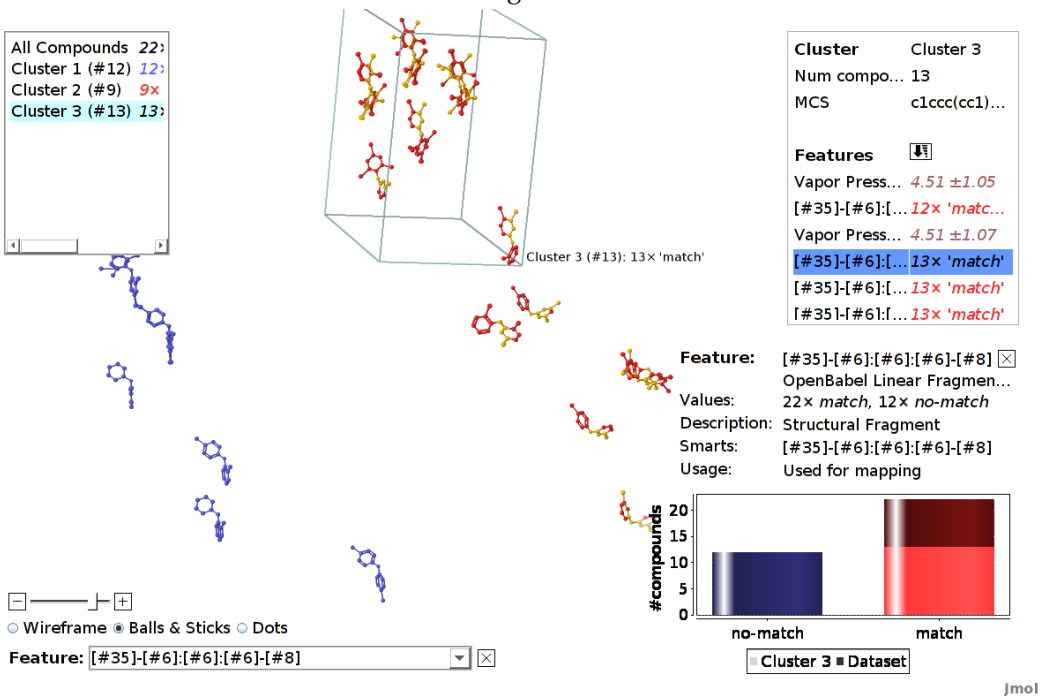


Figure 4.12: The compounds of the PBDE dataset have been highlighted according to the structural fragment ”Br-c:c:c-O”. The matching atoms in each compounds are drawn in orange.

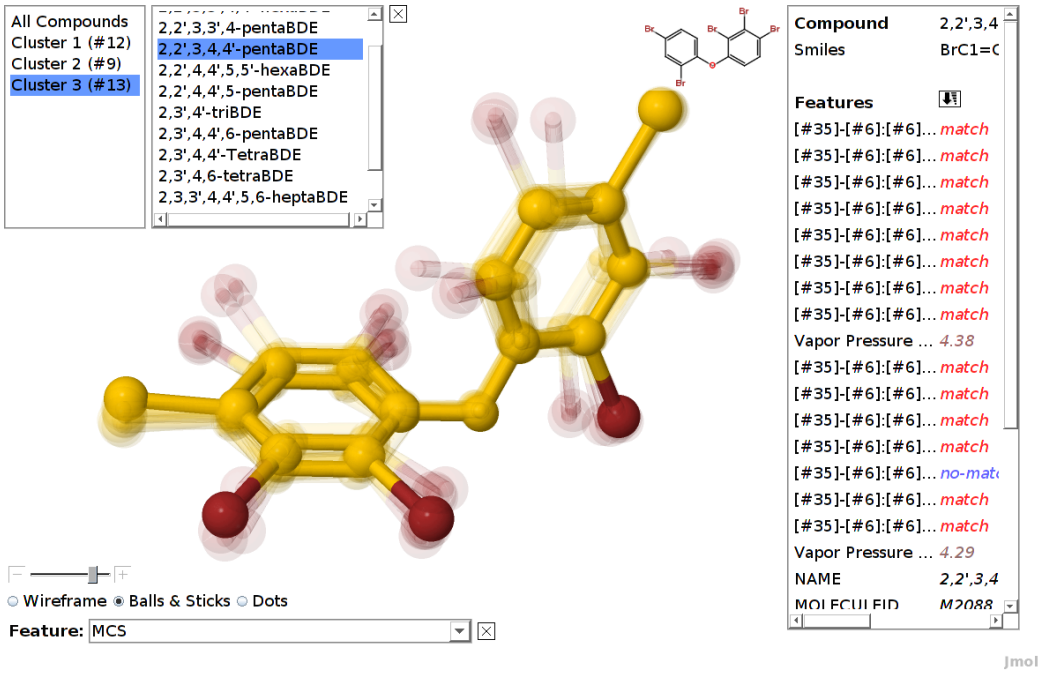


Figure 4.13: The compounds of a cluster of the PBDE dataset are superimposed according to their maximum common subgraph.

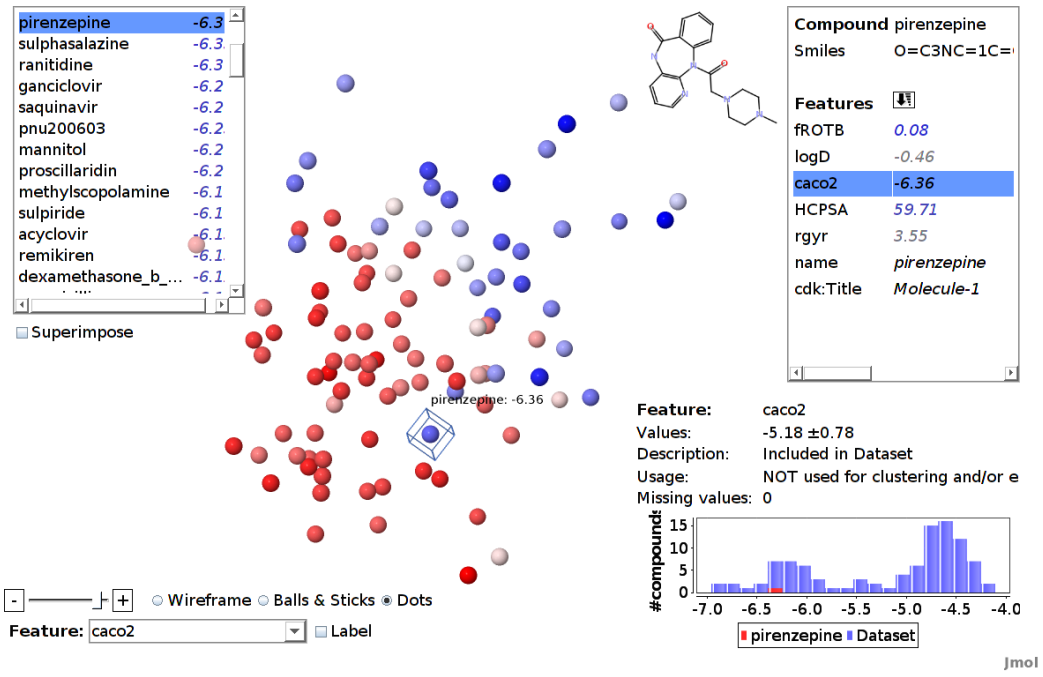


Figure 4.14: Highlighting the endpoint of the Caco-2 permeability dataset. Selecting the endpoint feature shows the activity space (or landscape). The endpoint was not employed for 3D embedding. The depiction setting is set to the depiction option *Dots*. The compound pirenzepine is selected (indicated with a blue box), the compound information including a 2D image of the compound is shown on the right-hand side. The compound forms an activity cliff, as its endpoint value differs from its neighbor compounds.

the feature value distribution in the data space for large datasets (as exemplified in Figure 4.14).

4.1.2.4 Adjust the Size of Compounds

In order to get a more detailed and closer view on compounds or clusters, the user can use the zooming function. This will show a smaller section of the whole dataset in larger scale.

Additionally, there is the possibility to adjust the size of the compounds in 3D space, which affects the distance between compounds to each other. This can be controlled with the slider or the "+" and "-" buttons on the bottom left of the screen. To decrease the compound size can be advantages, as sometimes compounds are located very close to each other and may overlap in the field of view.

4.1.2.5 Superimpose Compounds

Superimposition moves the compounds of each cluster to the cluster center. Accordingly, they will overlap each other. This method may provide a better overview when a large clustered dataset is visualized. Furthermore, it emphasizes structural similarities of the compounds inside the cluster. This is especially useful, if the compounds of the clusters are aligned according to a common substructure (see Figure 4.13).

4.1.2.6 Export or Remove Clusters and Compounds

Compounds or whole clusters can be (temporally) removed from the view, not changing the original dataset. To actually modify the data, the compounds can be (partially) exported as SD-file or CSV file, including cluster assignments and computed features. If available, the SDF export will contain the compound 3D structures. The exported data can than be used for further processing.

4.1.2.7 Export Images

The presented figures in this chapter are screen-shots of the application to illustrate the viewers functionality[4]. Additionally, CheS-Mapper has a dedicated export function to create high-resolution images of the compounds, omitting the controllers and lists.

4 Therefore, the default display options of the CheS-Mapper viewer have been customized for printing: the default background is changed from black to white, and the font size is increased.

4.2 Implementation

The CheS-Mapper software is implemented in Java. It is provided as a Java Web Start application, which can directly be started from a web browser. Additionally, the program can also be downloaded as stand-alone version. CheS-Mapper is an open-source project hosted at GitHub (`http://github.com/mguetlein/ches-mapper`). The code architecture allows developers to easily integrate novel algorithms, e.g. for clustering, 3D structure calculation, etc. A range of Java libraries is integrated into the project: the 3D viewer for molecules Jmol[5], the Chemistry Development Kit (CDK [23]), and the data mining workbench WEKA [77]. Extended functions are provided in CheS-Mapper in case the free software tools Open Babel [22] and R [78] are installed on the local computer. Open Babel is a C++ library for cheminformatics that can be used for additional 3D computing, matching of SMARTS (Smiles Arbitrary Target Specification) fragments, and structural fragment mining. The statistical computing tool R is exploited by CheS-Mapper for clustering and embedding.

4.3 Use Cases – Applying CheS-Mapper to Real World Datasets

In the following, CheS-Mapper is applied to two real-world datasets for demonstrating its functionalities. Additionally, we examine published statements about the datasets.

4.3.1 Mapping a Dataset using Integrated Features

We use the CheS-Mapper application to visualize and verify work on the correlation of Caco-2 permeation with simple molecular properties [185]. The authors provide 100 structural diverse compounds, with five numeric features that are stored in the dataset. One of the features is the actual endpoint Caco-2 permeability ($\log P_{app}$). The remaining four molecular descriptors are: experimental distribution coefficient (*logD*), high charged polar surface area (*HCPSA*), radius of gyration (*rgyr*), and fraction of rotatable bonds (*fROTB*). In their work, the authors describe that these four features are valuable descriptors for building a QSAR model to predict Caco-2 permeation.

We employ the CheS-Mapper wizard to embed the compounds in 3D space according to those four properties using principal components analysis (see Fig-

5 `http://www.jmol.org`

ure 4.8). Note that the actual endpoint (*caco2*) was not used. A copy of the dataset and a detailed tutorial can be found at `http://ches-mapper.org`.

Figure 4.11 shows a screenshot of the CheS-Mapper viewer, with *LogD* values highlighted: the compounds are colored according to their feature value of *LogD*, compounds with low values are colored in blue, compounds with high values are colored in red. As this feature was used for embedding, compounds with similar values are close to each other. The same holds if we select the remaining three features. In contrary, the actual endpoint was not used as input for the embedding algorithm. Still, compounds that are close to each other tend to have a similar *caco2* value (see Figure 4.14). This supports the authors findings, the endpoint is indeed correlated to the feature values presented in the dataset.

Additionally, we easily detected one compound with the Viewer that violates the correlation between feature values and endpoint: Figure 4.10 shows the compound *pirenzepine*. It has a relatively low endpoint value of -6.35, and is therefore drawn in blue. This compound attracts our attention, as it is close to compounds with high endpoint values, i.e. it is located next to many red compounds. By employing CheS-Mapper, we effortlessly gained insights into a dataset and a critical compound: *pirenzepine* is the training compound with the highest prediction error using the QSAR model in the cited article.

A more detailed analysis of this dataset including embedding quality calculation, investigation of activity cliffs, and visual validation of (Q)SAR modeling can be found in Section 5.3.1.

4.3.2 Structural Clustering using Open Babel Fingerprints

We use the CheS-Mapper program to apply structural fragment mining to a small dataset with structural similar compounds. The dataset consists of 34 polybrominated diphenyl ethers (PBDEs) with experimentally measured endpoint (*Vapor pressure*) [186]. It was used to build and validate (Q)SPR models with physico-chemical descriptors (computed with the Dragon software). The authors determined the feature *T(O...Br)* to be the most significant descriptor with respect to the endpoint. This feature describes the sum of the topological distance between oxygen and bromine. This can be visually verified with CheS-Mapper.

We select linear fragments (up to a size of 7 atoms) as features in the wizard. We skip uninformative fragments that occur in each compound of the dataset, which yields 15 distinctive fragments, all containing bromine. Note that the actual target endpoint is not selected for clustering and embedding. Again, a copy of the dataset and a detailed tutorial can be found on the project homepage.

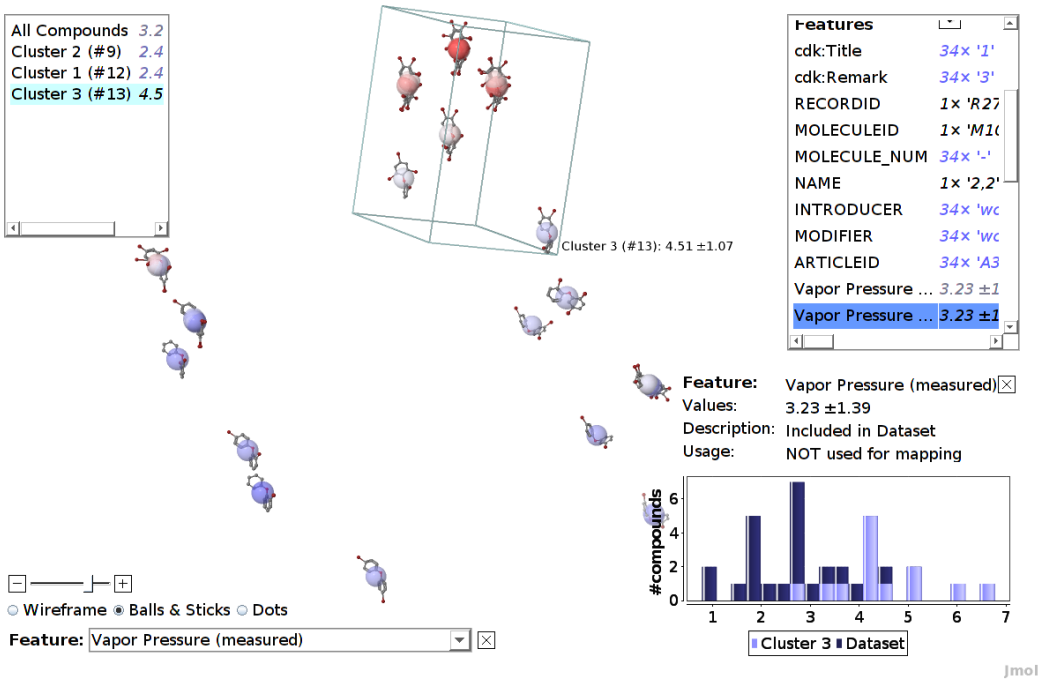


Figure 4.15: The compounds of the PBDE dataset are highlighted according to their endpoint value. The selected cluster contains the compounds with the highest endpoint values.

The viewer shows that there are three clusters of about equal size (see Figure 4.9). We highlight the endpoint in Figure 4.15 to show that the endpoint can in fact be modeled with these 15 structural features: features that are located next to each other have a similar endpoint values. Furthermore, one of the clusters contains all compounds with high endpoint values.

We now determine the feature properties of the three clusters. We select the structural feature *[#35]-[#6]:[#6]:[#6]-[#8]* that corresponds to a linear sequence of 5 atoms: bromine, three aromatic carbons, and oxygen (the SMARTS is equal to *Br-c:c:c-O*). Figure 4.12 shows that the CheS-Mapper viewer colors the compounds according to occurrence, as well as it highlights where the smarts matches in each compound. Selecting feature *Br-c:c-O* shows that the compounds in cluster number three (the one with the highest endpoint values) exclusively match both fragments. This supports the findings [186] that the endpoint value depends indeed on the distance between the bromine atom and the ether group.

CheS-Mapper can be a valuable tool for structurally very similar compounds, when employing MCS alignment together with the superimpose mechanism. Figure 4.13 shows the compounds of cluster number three superimposed onto each other. The compounds are aligned according to their common substructure, enabling the user to identify structural similarities or dissimilarities.

Dataset	Compounds	Description	Reference
PBDE	34	Polybrominated diphenyl ethers (endpoint 'Vapor pressure')	[186]
Caco2	100	Diverse compounds tested for Caco-2 permeability	[185]
COX2	467	Cyclooxygenase-2 (COX-2) inhibitors	[187]
CPDBAS	1508	Carcinogenic Potency Database - All Species	[188]

Table 4.1: List of datasets employed for empirical evaluation.

Dataset	Size	CDK	OpenBabel
PBDE	34	3s	1m, 22s
Caco2	100	2m, 12s	10m, 21s
COX2	467	51s	38m, 19s
CPDB	1508	14m, 3s	n/a

CDK	CDK 3D Structure Generation
OpenBabel	OpenBabel 3D Structure Generation

Table 4.2: Runtime of building 3D structures.

4.4 Empirical Evaluation of CheS-Mapper Functionalities

The use cases presented in this work exhibit the application of CheS-Mapper to a range of real-world datasets of various sizes: the largest of theses datasets contains 613 compounds (see Section 5.3.4). We have successfully tested CheS-Mapper with datasets including over 10.000 compounds (not shown in this thesis). Therefore, CheS-Mapper has dedicated functionalities to facilitate the analysis of large datasets (like, e.g., drawing only dots instead of compound structures (see Figure 5.10), and applying clustering to inspect only subsets of a large dataset at a time). However, pre-processing datasets can require huge computational effort, depending on the dataset size, the number and type of selected features, and the selected algorithms. To this end, we have evaluated the runtime[6] of the pre-processing steps in CheS-Mapper on four datasets with up to 1508 compounds (see Table 4.1). Please note, that CheS-Mapper caches results of pre-processing steps: each calculation has to be performed only once.

The results presented in this Section can provide some guidance when applying CheS-Mapper to large datasets. Additionally to the algorithm runtimes, we measure the quality of the 3D embedding.

6 The experiments have been performed on an Intel(R) Core(TM) i5-4200U CPU with 1.60GHz and 8G main memory. CheS-Mapper is currently not optimized to make use of multiple cores.

Dataset	PC OB	PC	MC OB	MC	Funct	Bio	FP2 OB	MoSS
PBDE	2s	6s	<1s	1s	1s	7s	<1s	<1s
Caco2	3s	31s	<1s	4s	3s	23s	<1s	<1s
COX2	7s	2m, 16s	1s	13s	8s	1m, 48s	1s	42s
CPDBAS	16s	6m, 4s	1s	23s	25s	4m, 41s	3s	n/a

PC OB	14 PC Descriptors created with Open Babel
PC	271 PC Descriptors created with CDK
MC OB	166 MACCS structural fragments matched with Open Babel
MC	166 MACCS structural fragments matched with CDK
Funct	307 Function groups matched with CDK
Bio	4860 Klekota-Roth Biological Activity fragments matched with CDK
FP2 OB	Linear fragments (FP2) mined with Open Babel (min freq: 10)
MoSS	MoSS - Molecular Substructure Miner (min freq: 10)

Table 4.3: Calculating differet sets of molecular descriptors.

As described in Section 2.1.1.2, building 3D structures is a time consuming optimization task. Calculating the three dimensional conformation of 100 compounds with Open Babel took about 10 minutes (see Table 4.2). The integrated CDK 3D builder is faster but does not produce reliable results.

In Table 4.3, we show the runtime of descriptor calculation algorithms on the investigated datasets. Computing 271 PC descriptors with CDK took more than 6 minutes on the largest dataset[7]. In general, the Open Babel library is faster than the integrated CDK library, especially when matching SMARTS fragments. Moreover, mining linear sub-structures with Open Babel (FP2) is very fast, even on large datasets. The graph miner MoSS [189] is slower, and requires a reasonable minimum threshold to be set[8]. However, it yields more complex graph sub-structures.

The runtimes of cluster algorithms presented in Table 4.4 have been measured using 271 molecular descriptors as input. The starred algorithm is the default cluster option (see Section 4.1.1.4). The two slowest clustering methods (Cascading k-Means with R and Expectation Maximization) automatically determine the best cluster size. These methods can be accelerated by adjusting their default parameters. As noted above, we do not evaluate the quality of the clustering result. In the future, we consider extending CheS-Mapper by calculating and providing cluster performance measures (e.g., the Calinski-Harabasz criterion [179]).

7 We have omitted a single descriptor (ionization potential) that takes about five seconds for a compound

8 The MoSS library failed on CPDB because it could not read a compound structure within the dataset.

Dataset	CkM*	kM	CkM	FF	EM	CW	H	kM R	CkM R	H R	DC R
PBDE	<1s	<1s	<1s	<1s	<1s	<1s	<1s	<1s	2s	<1s	<1s
Caco2	<1s	<1s	1s	<1s	8s	<1s	<1s	<1s	3s	<1s	<1s
COX2	4s	<1s	11s	<1s	36s	<1s	1s	2s	21s	<1s	1s
CPDBAS	13s	1s	49s	<1s	n/a	2s	26s	17s	7m, 39s	3s	4s

CkM*	k-Means - Cascade (WEKA) – Default Clusterer
kM	SimpleKMeans (WEKA)
CkM	k-Means - Cascade (WEKA)
FF	FarthestFirst (WEKA)
EM	Expectation Maximization (WEKA)
CW	Cobweb (WEKA)
H	Hierarchical (WEKA)
kM R	k-Means (R)
CkM R	k-Means - Cascade (R)
H R	Hierarchical (R)
DC R	Hierarchical - Dynamic Tree Cut (R)

Table 4.4: Runtime of clustering algorithms with 271 PC descriptors.

Dataset	Features	PCA*	Q	PCA R	Q	SM R	Q
PBDE	271	<1s	0.99	<1s	0.99	<1s	0.97
Caco2	271	<1s	0.94	<1s	0.94	<1s	0.94
COX2	271	1s	0.91	1s	0.91	2s	0.83
CPDBAS	271	2s	0.92	8s	0.92	23s	0.73

PCA*	PCA 3D Embedder (WEKA) – Default 3D-Embedder
PCA R	PCA 3D Embedder (R)
SM R	Sammon 3D Embedder (R)

Table 4.5: Duration of calculating 3D embedding and embedding quality (denoted with "Q") using 217 PC descriptors.

Dataset	Features	PCA*	Q	PCA R	Q	SM R	Q	SM-T R	Q
PBDE	15	<1s	0.97	<1s	0.97	<1s	0.98	<1s	0.99
Caco2	224	<1s	0.73	<1s	0.73	<1s	0.65	<1s	0.77
COX2	585	17s	0.94	2s	0.94	5s	0.94	7s	0.95
CPDBAS	1304	8m, 34s	0.76	47s	0.76	2m, 53s	0.69	3m, 16s	0.86

PCA*	PCA 3D Embedder (WEKA) – Default 3D-Embedder
PCA R	PCA 3D Embedder (R)
SM R	Sammon 3D Embedder (R)
SM-T R	Sammon 3D Embedder (R) Tanimoto

Table 4.6: Duration of calculating 3D embedding and embedding quality (denoted with "Q") using structural fragments (Open Babel FP2).

Dataset	Cluster	MCS	Max Fragment
PBDE	4	2s	2s
Caco2	2	7s	1s
COX2	3	21s	2s
CPDBAS	3	8s	6s

MCS	Maximum Common Subgraph (MCS) Aligner
Max Fragment	Maximum Structural Fragment Aligner

Table 4.7: Runtime of 3D alignment algorithms (including MCS computation)

We have measured the runtime of embedding the dataset into 3D space and calculating the embedding quality. This experiment has been executed with PC descriptors (Table 4.5) as well as structural fragments (Table 4.6). The computation of the embedding quality (see Section 5.2.1.1) suffers from quadratic runtime with respect to the number of compounds and feature values. Therefore, we have added an option to CheS-Mapper to disable embedding quality computation when working with large datasets. PCA shows in general good runtime and embedding quality when using numeric descriptors[9]. Sammon embedding with Tanimoto distance can increase the embedding quality in case structural fragments are selected.

Table 4.7 contains the runtimes of 3D aligning the compounds in each cluster. Clustering was performed with the default algorithm based on structural fragments (computed with Open Babel FP2). Using structural fragments to compute clusters for 3D alignment is recommended as it tends to produce structurally sim-

9 The runtime discrepancy for structural fragments between PCA performed with WEKA and PCA performed with the R library is due to a an in-efficient handling of nominal binary features within the WEKA PCA.

ilar clusters (structurally dis-similar compounds are not well-suited for 3D alignment). The runtime of computing the maximum common sub-graph (MCS) depends on cluster sizes and compound structures (e.g., the COX2 dataset contains many structurally similar compounds and therefore yields large common fragments). Maximum fragment alignment requires less time as it uses the largest common structural fragment that has been calculated during feature computation.

4.5 Conclusions

We presented the CheS-Mapper application, a tool to visualize and explore chemical datasets. In a preprocessing step, the dataset is mapped into a virtual three-dimensional space. A key part of the preprocessing is the choice of features, which is done by the user. Features can either be provided in the dataset, or the application can calculate physico-chemical descriptors as well as structural fragments. The selected features are then used for clustering and 3D embedding. Hence, compounds that have similar feature values are likely to be clustered together, and are close to each other in 3D space. Subsequently, a 3D viewer shows the embedded dataset, and enables the user to explore the dataset and its properties. Numerical features, as well as structural features can be highlighted within the viewer.

This makes CheS-Mapper a visualization tool that could be used to explore structure-activity relationship (SAR) information in datasets, as well as to present chemical compounds and compound features to others. It is freely available and an open-source project.

5 CheS-Mapper 2.0 for Visual Validation of (Q)SAR models

This chapter presents our work on visual validation of (Q)SAR models [9], as an approach for the graphical inspection of (Q)SAR model validation results. We will motivate the method in the following Section 5.1. The approach applies the 3D viewer CheS-Mapper, that has been extended to facilitate the analysis of (Q)SAR information and allows the visual validation of (Q)SAR models. The new functionalities of CheS-Mapper 2.0 and our visual validation approach is described in Section 5.2. Subsequently, visual validation is applied to real-world datasets in Section 5.3. Finally, Section 5.4 provides a summary and discussion.

5.1 Motivation

Visualization of (Q)SAR information in chemical datasets is a very active field of research in cheminformatics (see Section 2.4.3). Many approaches are being developed that help to understand existing correlations between the structure of chemical compounds, their physico-chemical properties, and biological or toxic effects. These correlations are employed by (Q)SAR models to predict the activity of unseen compounds. The predictive performance of (Q)SAR models can then be evaluated with numerous statistical validation techniques. There is, however, to the best of the author's knowledge, no visualization method yet that incorporates (Q)SAR predictions and validation results. One reason for this might be that most (Q)SAR models are the results of applying statistical machine learning approaches to chemical datasets and the resulting models are sometimes opaque and it is commonly not an easy task to extract the reasoning behind a prediction. Some models induced by machine learning approaches, however, are relatively easy to understand, like decision trees, rule learners or nearest neighbor models. The predictions of these models are easy to comprehend, as long as the number of features that are employed for predictions is not too large (e.g. the size of the decision tree is reasonably small). Therefore, several model-dependent visualization tools exist [154, 155, 157, 158]. In contrast, many other models can rather be seen as black boxes, like artificial neural networks or support vector machines. In addition, these complex models are often more predictive than intuitive and simpler models.

In this work, we propose a model-independent visual analysis of validation results employing the 3D viewer CheS-Mapper [8]. The presented approach does not

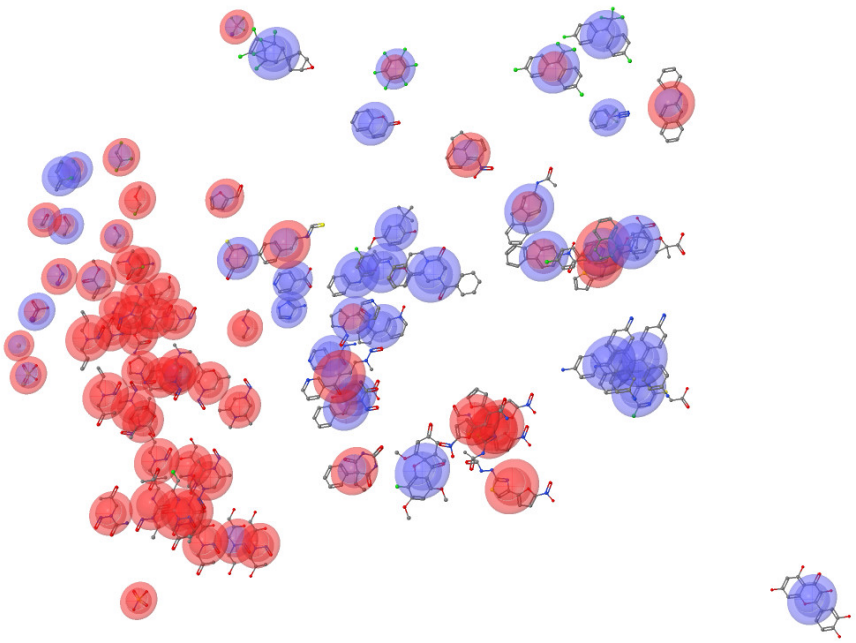


Figure 5.1: Comparing actual and predicted activity values with CheS-Mapper. The CPDB hamster dataset is embedded into 3D space, based on 14 physico-chemical (PC) descriptors that have been computed with CheS-Mapper using Open Babel. The compounds are predicted with a 3-Nearest Neighbor approach employing the same PC features. The inner sphere corresponds to the actual endpoint activity, with class values active (red) and inactive (blue). The outer, flattened spheroid depicts the prediction.

investigate how predictions are made by each model, but rather allows the comparison of actual and predicted activity values in the feature space (see Figure 5.1). We call this approach *visual validation*. It should be regarded as complementary to the standard statistical validation. Visually examining (Q)SAR model validation results can aid in understanding the model itself as well as the modeled data, and can furthermore yield the following benefits:

DATA CURATION: It is important to inspect (groups of) misclassified compounds (in case of classification), or compounds with high prediction error (regression). Investigating possible reasons for the erroneous predictions might aid in detecting errors in the training data, like mis-measured endpoint values. The researcher might as well discover that the misclassifications are outliers or that more training data is required.

MODEL IMPROVEMENT: Another possible reason for bad model performance may be improper feature choice, e.g. the available features can not be used to distinguish between some active and inactive compounds. Moreover, the selected model might be too specific (overfitting) or too general (underfitting).

Additionally, visual validation can show the effect of different model parameters.

MECHANISTIC INTERPRETATION: It is also possible to extract knowledge from groups of compounds that are correctly classified. Compounds with similar feature values and endpoint values might have similar modes of action. Consequently, visual validation can support the researcher in deriving a mechanistic interpretation. Mechanistic interpretation and proper model validation are requirements of the OECD guidelines for valid (Q)SAR models [190]. To this end, visual validation could also help to improve the acceptance of (Q)SAR models by regulatory authorities as alternative testing methods.

5.2 Visual Validation of (Q)SAR Models

The following Section 5.2.1 introduces new functionality of CheS-Mapper 2.0 for (Q)SAR information analysis and visual validation. Next, we describe how the tool can be used to visually validate (Q)SAR models in Section 5.2.2. In the subsequent Section 5.3, we present actual use cases and how the new features of CheS-Mapper 2.0 were employed to achieve the goals of the use cases. An overview of the new features of CheS-Mapper 2.0, the use cases, and the connection among them is shown in Table 5.1.

5.2.1 New Features for Visual Validation

New functionalities have been added to CheS-Mapper 2.0 for (Q)SAR information analysis and visual validation. A novelty is that the compound list (at the left-hand side of the viewer in Figures 5.2 and 5.3) is completed with feature values of the currently selected feature for each compound and sorted according to this value. This extension (referred to as (a) in Table 5.1) facilitates the identification of compounds with the highest or lowest features values. Moreover, we added the option to highlight two features at once by adding a second, flattened spheroid (Figure 5.1). This novel functionality (see Table 5.1 (b)) can be used to directly compare the values of two features.

5.2.1.1 Measuring Embedding Quality and Embedding Stress

It is not always possible to compress the feature space without loss of information. This is especially the case if many diverse and/or uncorrelated features are

New feature

(a) Sorting of compound/cluster list according to selected feature
(b) Highlighting two features simultaneously
(c) Computing embedding quality and distances
(d) Determination of common properties of compounds/clusters
(e) Compute mean SALI values to detect activity cliffs
(f) KNIME integration

Dataset	**Use case**	(a)	(b)	(c)	(d)	(e)	(f)
Caco-2	Inspect (Q)SAR information using integrated features	✓		✓		✓	
	Compare (Q)SAR models and validation methods	✓	✓		✓		✓
Cox-2	Inspect (Q)SAR information by mining structural fragments	✓		✓	✓	✓	
	Inspect validation results with respect to activity cliffs	✓			✓	✓	
CPDB Hamster	Compare modeling with different feature sets	✓	✓	✓	✓	✓	
EPA FHM	Inspect applicability domain algorithms	✓		✓	✓		

Table 5.1: Overview of new features and their application to a variety of use cases. The features are described in more detail in Section 5.2.1. We illustrate the application of the new features in Section 5.3.

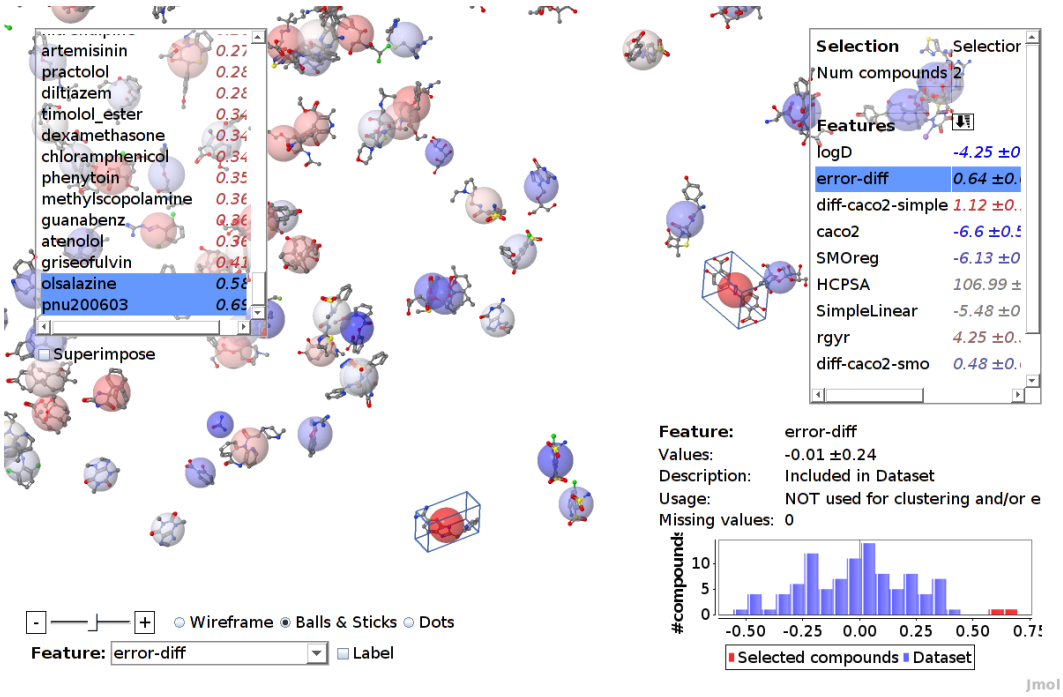


Figure 5.2: Inspecting the prediction error difference of two (Q)SAR approaches. The prediction error difference ($\text{error}_{\text{simple-linear}} - \text{error}_{\text{support-vector}}$) of two regression approaches for the Caco-2 dataset is selected. Simple linear regression performed especially worse for the two selected compounds (olsalazine and pnu200603). Both compounds have a very low *logD* value (*logD* is the top feature in the feature list on the right-hand side).

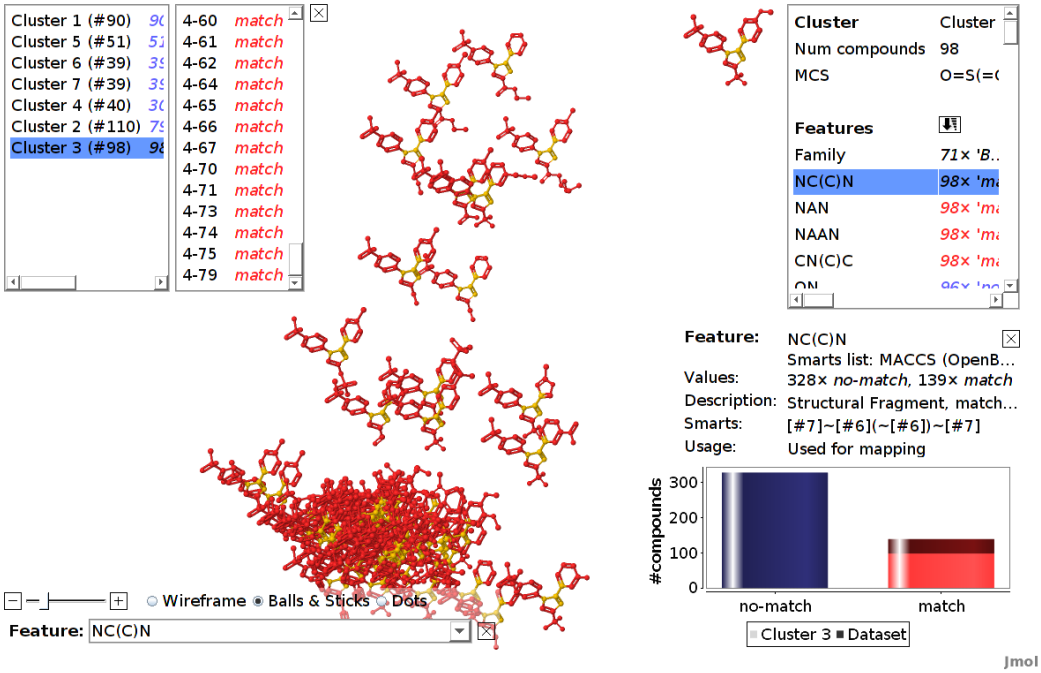


Figure 5.3: Cluster 3 of the COX-2 dataset is selected. Only compounds of the selected cluster 3 are visible. At the top left-hand side, cluster and compound list can be used to select another cluster or a compound within the active cluster. The feature list of cluster 3 at the top right-hand side is sorted according to specificity. The structural feature *NC(C)N* is very specific, as it matches mostly compounds within this cluster (as indicated by the bar chart). The fragment is highlighted in each compound structure with orange color.

selected by the user. CheS-Mapper 2.0 computes a global embedding quality measure that describes how well the feature values are reflected by the 3D positions of the compounds (see Table 5.1 (c)). A standard stress function [125] has the disadvantage that it cannot be used to compare the embedding of different datasets: it is commonly defined as the sum of squares between the pairwise distances in the high-dimensional representation (feature values) and the low-dimensional representation (3D positions). Instead, CheS-Mapper computes the Pearson's product-moment correlation coefficient between the distance pairs. The distances based on the feature values are computed using the (dis-)similarity measure of the selected embedding algorithm. The 3D distance values are computed using the Euclidean distance, resembling the human user's perception of distances between compounds in 3D space. The embedding quality ranges from 1 (perfect correlation) to 0 (no correlation) to -1 (negative correlation). A warning is given to the user if the correlation is below 0.6, corresponding to moderate or weak embedding quality [191].

In some use cases, the overall embedding is good, apart from some outlier compounds that might have largely differing feature values. Therefore, CheS-Mapper provides the embedding stress for each compound. We define embedding stress as 1− Pearson's correlation coefficient between the distance pairs of the corresponding compound to all other compounds in the dataset. Accordingly, compounds with a value close to 0 have low embedding stress, whereas a value close to 1 corresponds to high stress.

The global embedding quality is presented to the user at the top right-hand side of the viewer (see Figure 4.8). The embedding stress can be highlighted with the drop down menu coloring compounds with low embedding stress in blue, while compounds with high embedding stress are colored in red.

CheS-Mapper can also compute and highlight the distance from all compounds in the dataset to a particular compound, based on the features and distance measure that were used for the 3D embedding. When a good 3D embedding is feasible, this distance mirrors the proximity between compounds in 3D space. However, if the 3D embedding is poor, or a particular compound has a high embedding stress, this function allows to determine the nearest neighbors for a particular compound.

5.2.1.2 Determination of Common Properties of Compounds

When exploring a clustered dataset with CheS-Mapper, a common task is to identify the reasoning of why compounds are assigned to the same cluster. Similarly, the user might want to determine why two particular compounds are located close to each other in 3D space. In both cases, the user is looking for common properties

of groups of compounds that separate these instances from the remainder of the dataset.

This kind of information is dynamically provided by CheS-Mapper 2.0 (see Table 5.1 (d)), even for large datasets with numerous features: the feature list (on the right-hand side of the viewer) is sorted depending on the currently selected cluster or the currently selected compound/s. In more detail, the list is sorted in descending order according to the specificity of the feature values of the selected elements. Hence, the *most important* features that distinguish the selected compounds from the remaining dataset can be found at the top of the list. Examples are given in Figure 4.10 and Figure 5.3. The specificity of features is computed by comparing the feature values of the selected elements to the feature values of the entire dataset. To this end, statistical tests are applied and the features are sorted in ascending order according to the p-value: low p-values indicate that the tested distributions differ from each other and high p-values indicate similar distributions. A χ^2-test is exercised for nominal feature values [192]. For numeric features, we employ one-way analysis of variance (ANOVA) [192]. When comparing the numeric feature value of a single compound to the overall feature values distribution, the ANOVA test is not applicable. We do therefore apply the χ^2-test on binned numerical data to compute the p-value for numeric features of single compounds[1].

5.2.1.3 Analysis of Activity Space and Activity Cliffs

CheS-Mapper can be employed for various purposes, including the analysis of datasets without endpoint activity information. However, the typical use cases are the analysis of (Q)SAR information and of the activity landscape of a small molecule dataset. Commonly, activity landscapes depict activity values in two feature dimensions. As CheS-Mapper provides an additional third dimension (3D space), the term activity space is more appropriate. Inspecting the activity space of a dataset with CheS-Mapper requires that the endpoint values are stored in the dataset, but not employed for 3D embedding. Highlighting the endpoint feature in the CheS-Mapper viewer presents the activity space, as shown in Figure 4.14. The user can detect activity cliffs by locating compounds that stand out in the color coding, when compared to neighboring compounds in nearby 3D space. This indicates that these compounds have differing endpoint values, yet similar feature values.

1 The Apache Commons Mathematics Library is employed for statistical testing (`http://commons.apache.org/math`). To test the specificity of numeric features of single compounds, equal-width binning is applied with initially 20 bins. Hence, the numeric data is divided into categories using 20 intervals of equal width. If intervals without any compounds exist, the number of intervals is decreased by one and the binning method is reapplied. To produce a compact data representation, this process is iteratively repeated until no empty bins exist.

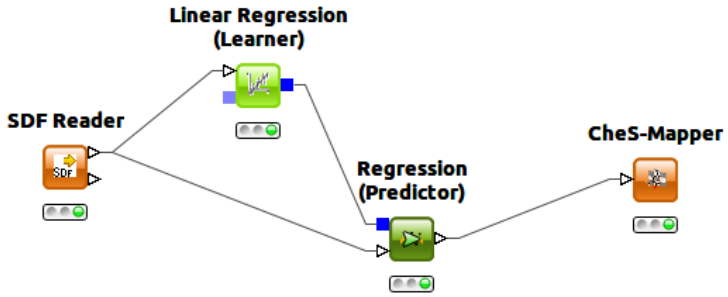


Figure 5.4: A simple KNIME workflow including the CheS-Mapper visualization node. CheS-Mapper is employed to visually inspect the modeled activity of linear regression, based on properties that are stored in a SD-File.

Additionally, CheS-Mapper provides a new functionality to automatically reveal activity cliffs by computing the Structure-Activity Landscape Index (SALI) (see Table 5.1 (*e*)). The SALI value is computed of pairs of compounds and a high SALI value indicates that a pair resembles an activity cliff[2] [141]. We transform the SALI value matrix to a feature (with a single value for each compound) by calculating the mean SALI values for each compound. Additionally, CheS-Mapper provides the standard deviation and maximum pairwise SALI values. Hence, compounds forming activity cliffs can be determined and inspected. Activity cliffs can further be studied by investigating common properties of a particular compound and its neighboring compounds. For instance, the user might detect that the features that have been selected for embedding cause an activity cliff, as some active and inactive compounds cannot be distinguished [193].

5.2.1.4 CheS-Mapper Extension for KNIME

We have included CheS-Mapper into KNIME (Konstanz Information Miner), a graphical framework for data analysis [152] (see Table 5.1 (f)). The framework has various extensions for cheminformatics and machine learning, and can therefore be employed for (Q)SAR modeling. The CheS-Mapper node for KNIME is a pure visualization node that envisions data which has been arbitrarily processed within KNIME (see `http://tech.knime.org/book/ches-mapper-node-for-knime-trusted-extension`). A simple example for using CheS-Mapper within KNIME is shown in Figure 5.4. In this case, CheS-Mapper is applied for analyzing prediction results of a regression model (follow the link above to find a detailed description of this workflow).

2 The Structure-Activity Landscape Index (SALI) is high for compound pairs c_1 and c_2 that have dissimilar activity values A_1 and A_2 but are structurally similar ($sim(c_1, c_2)$ is close to 1):
$SALI_{c_1,c_2} = (A_1 - A_2)/(1 - sim(c_1, c_2))$

5.2.1.5 Store Configuration for Chemical Space Mapping

A novel functionality in CheS-Mapper is that the current mapping settings, which are configured in the wizard, can be stored to a file and shared with others. This is especially helpful when exploring the same dataset in a team, as it preserves different useful configurations like e.g. the selected features, clustering, and 3D embedding settings. The configuration includes the random seed for randomized approaches (like e.g., k-means clustering), to ensure that the mapping settings always produce the same mapping result.

5.2.2 Visually Validating (Q)SAR Models in CheS-Mapper 2.0

Visual validation describes the graphical inspection of (Q)SAR model validation. Initially, a (Q)SAR model is built and validated on a compound dataset. The dataset is then visualized with CheS-Mapper, using the same features for embedding that were used to validate the (Q)SAR model. Additionally, the endpoint values and the prediction results of the model are employed within the visualization. Consequently, CheS-Mapper allows the inspection of actual and predicted activity in the feature space. The visual validation approach can be re-iterated to take possible insights from the visualization steps into account for model re-building. However, re-iteration should be handled with care to prevent model overfitting or chance correlation (as discussed below).

In general, the proposed method can be performed with an arbitrary validation scheme, like e.g. test set validation or repeated k-fold cross-validation. The predicted endpoint can be qualitative or quantitative (i.e. classification and regression models can both be analyzed).

5.2.2.1 Mapping the Feature Space of the (Q)SAR Model

When performing visual validation with CheS-Mapper the dataset is mapped into 3D space based on the same features as employed by the (Q)SAR modeling. Hence, when exploring the embedded dataset with CheS-Mapper, the user is provided an intuitive view on the feature space that is employed as input for the (Q)SAR approach. Depending on the model, it is likely that similar compounds are predicted with similar activity values. For example, when performing classification, compounds will be assigned the same class value if they are on the same side of the decision boundary.

5.2.2.2 Comparing Actual and Predicted Activity

The main idea of visual validation is that the user is able to simultaneously compare the experimental and predicted endpoint values in the feature space. These values can be highlighted in CheS-Mapper, by coloring the compounds according to their actual or predicted activity value. Moreover, the application can highlight both activity values at the same time (see Figure 5.1). Hence, the user is able to identify compounds that have been misclassified by the (Q)SAR model and investigate possible reasons. A model might under-fit the target concept which results in an overly smooth predicted activity space, or if classification is applied, in large areas of compounds with the same predicted class. In contrast, the (Q)SAR model might be too complex and overfit the data. Furthermore, the model could fail in predicting compounds that form activity cliffs. As described above, activity cliffs can be detected and investigated with CheS-Mapper. A possible solution for misclassified compounds that form activity cliffs might be to add additional features that aid in distinguishing active and inactive compounds.

Directly highlighting the prediction error (instead of individually selecting predicted and actual endpoint values) allows the selection of groups of misclassified compounds with CheS-Mapper. Therefore, the user can detect common properties of these compounds and investigate possible weaknesses of a model. When performing classification, the probability (or confidence) of a classifier for each prediction is a useful extension for the visualization of decision boundaries. These are areas in feature space where the compound predictions change from one class to another (e.g. from active to inactive). As some (Q)SAR models do not provide probability estimates, repeated validation can overcome this limitation, as described in the next section.

5.2.2.3 Visually Validating Repetitive Validation Approaches

Arbitrary (Q)SAR modeling software can be employed for visual validation. Modeling and validation results have to be stored in the CheS-Mapper input dataset file (e.g. SD-File or CSV-File), in addition to the model input features and actual endpoint values. In more detail, the following validation results should be available for each predicted compound:

- Predicted endpoint value (class or numeric regression value)
- The prediction error (difference between actual and predicted endpoint value, optionally the squared-error for regression)
- A probability or confidence measure (if available)

- The applicability domain value (inside/outside or continuous, if available)

As mentioned above, our presented visual validation approach can be employed using any arbitrary validation technique. Depending on the selected validation approach, compounds can even be predicted multiple times. When applying a single training test set split, test set compounds will be predicted only once. A k-fold cross-validation yields a prediction for every compound in the dataset. When using a repetitive sampling scheme, like bootstrapping or a n-times repeated k-fold cross-validation, compounds are predicted multiple times by (Q)SAR models trained on different subsets of the data. As mentioned previously (Section 2.3), a repetitive validation approach should be preferred for small datasets to avoid overfitting (caused by e.g., "parameter fiddling"). In particular, visual validation should not be used to maximize the prediction accuracy on a single test set, as this would most likely not improve the predictivity of the (Q)SAR model. For visual validation, each repetition (or run of the validation approach) could be inspected separately, but it is more reliable to inspect the aggregated result. Consequently, multiple predictions for each compound should be combined as follows:

- For numeric predictions, the mean predicted value is preferable. When performing binary classification, the prediction can be transformed to continuous values between 0 and 1 (this is e.g. the ratio how often a class was predicted as active). For classification with multiple classes, the majority class prediction or an *inconclusive value* could be used.
- The prediction error can be averaged with standard techniques, like accuracy for classification and root-mean-squared-error for regression.
- The mean of the probability or confidence is adequate.
- The applicability domain value should be transformed to a continuous 0-1 value, corresponding to the ratio of how often the compound was inside the applicability domain.

5.2.2.4 Limitations

As already stated before, the exact reasoning behind predictions is hard to comprehend for humans in many (Q)SAR modeling approaches. A prediction algorithm whose predictions can be easily understood with CheS-Mapper is a k-Nearest Neighbor algorithm, as illustrated in Figure 5.1. Nevertheless, even if predictions are not comprehensible to researchers, inspecting and comparing actual and predicted activity values can provide valuable information (as discussed above).

Another limitation of our approach is that it relies on a good mapping of the data into 3D space. Often, CheS-Mapper can achieve high embedding quality as it employs a third dimension (compared to standard 2D mapping approaches) and provides various embedding algorithms with configurable distance measures. However, dimensionality reduction without loss of information is not always feasible, especially on large and diverse datasets and when applying non-redundant and uncorrelated descriptors (which is preferable for (Q)SAR modeling). In these cases, CheS-Mapper yields an oversimplified and compressed view of the feature space and the spatial distance does not resemble the descriptor-based similarity for all compounds. As described above, CheS-Mapper allows detecting poorly embedded compounds by computing and highlighting embedding stress. Moreover, the descriptor-based distance to a dedicated compound can be calculated to detect neighboring compounds. An additional functionality that helps to overcome this limitation is the computation of activity cliffs, which is not dependent on the embedding.

5.3 Use Cases

We employ visual validation with CheS-Mapper to analyze real-world datasets that include experimentally derived activity endpoints. Table 5.1 provides an overview of the subsequent use cases. We show how researchers can employ the new functionalities to investigate the correlation between feature values and activity values. Furthermore, we explore (Q)SAR model prediction and validation results with CheS-Mapper, and inspect different applicability domain approaches that exclude different compounds from a dataset.

5.3.1 Comparing (Q)SAR Models for Caco-2 Permeation

We applied CheS-Mapper to visualize and verify work on the correlation of Caco-2 permeation with simple molecular properties in Section 4.3.1. We will now demonstrate CheS-Mapper 2.0 functionalities, and subsequently compare two (Q)SAR modeling approaches applied to this data.

As noted before, when highlighting *logD*, we observe that compounds with similar *logD* values are close to each other, as this feature was used for embedding (see Figure 4.11). In fact, the dataset is almost perfectly embedded into 3D space using principal components analysis ($\mathtt{Pearson} : 0.99$). This is due to the fact that the number of dimensions has to be reduced only by one: from 4 molecular descriptors to a 3 dimensional space. Additionally, inter-correlation of feature values simplifies the

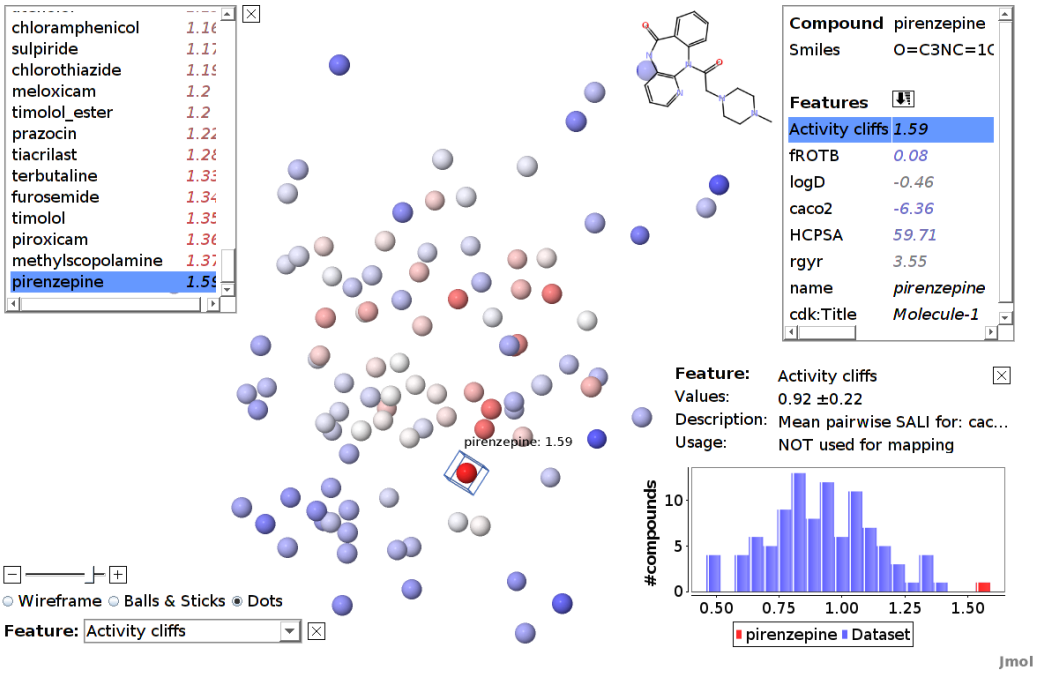


Figure 5.5: Highlighting activity cliffs within the Caco-2 permeability dataset. The mean SALI values are computed and highlighted. The compound pirenzepine is selected. It is the compound with the highest mean SALI value, as indicated in the histogram (at the bottom right-hand side) and in the compound list on the left-hand side (pirenzepine is at the bottom of the sorted compound list). Alternatively, the feature with the maximum SALI value or the standard deviation can be selected: pirenzepine has the highest maximum SALI value in the dataset (4.01) and the second highest standard deviation (0.8).

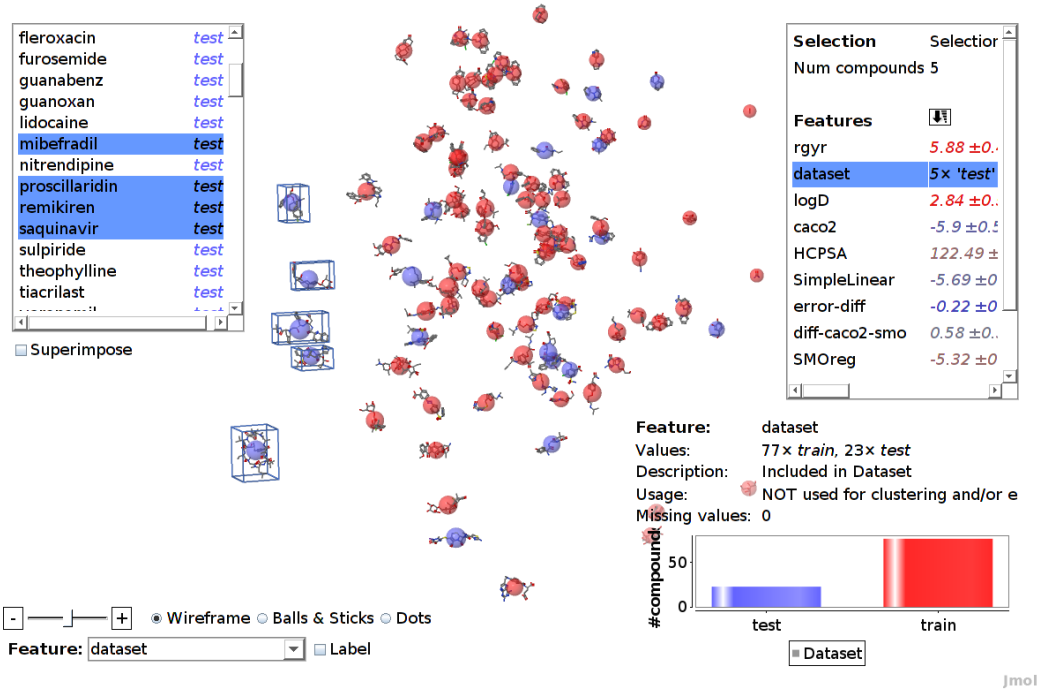


Figure 5.6: Visualizing the selected test set compounds of the Caco-2 dataset. The screenshot shows the distribution of training and test compounds in the feature space. The five selected test compounds share in particular high values for *radius of gyration* (*rgyr*) and *logD*. As a result, both features are the top of the feature list on the right-hand side.

dimensionality reduction (e.g. most compounds with high feature values of *high charged polar surface area (HCPSA)* have a low *logD* value). Even though the endpoint was not used for embedding, compounds that are close to each other tend to have a similar endpoint value as exemplified in Figure 4.14, i.e. the activity space (or landscape) is smooth. In our previous assessment, we were also able to visually detect the compound *pirenzepine* (selected in Figure 4.14; the viewer has zoomed in on this compound in Figure 4.10). It attracted our attention in CheS-Mapper, as it has a relatively low endpoint value (and is therefore colored in blue), but is spatially close to compounds with high endpoint values (colored in red). Hence, it is part of an activity cliff as its endpoint value differs from compounds with similar feature values. A new function of CheS-Mapper is to automatically locate activity cliffs. Therefore, compounds can be sorted and highlighted according to their pairwise SALI values. For this dataset, *pirenzepine* stands out as the compound with the highest mean and second highest standard deviation (see Figure 5.5).

We use two different (Q)SAR approaches to model Caco-2 permeability. Instead of adopting the training test split that was used in the original article [185], we apply a leave-one-out cross-validation procedure to compare support vector regression and simple linear regression. The visual validation workflow is implemented with the CheS-Mapper extension for KNIME [152] and is described in Section A.1 of the appendix. The visual validation with CheS-Mapper shows, as expected, that *pirenzepine* has the highest prediction error in simple linear regression and the second highest prediction error in support vector regression. According to statistical validation with KNIME, the R^2 value of support vector regression is 0.54, simple linear regression attains a value of 0.51. We investigate the reason for the predictivity difference with CheS-Mapper. We highlight the prediction error difference for each compound to determine which compounds are predicted more accurately by which approach (see Figure 5.2). The distribution of the prediction error difference is depicted as histogram in the figure: it indicates that the overall less accurate result of simple linear regression is mainly due to the prediction of two compounds. Using CheS-Mapper, we can easily determine common properties of these two compounds (*pnu200603* and *olsalazine*): the two compounds have the lowest *logD* feature values in the dataset. Consulting the original publication confirms the assumption that the *logD* value causes the high prediction error in linear regression. *logD* is treated differently from the other three input features, as the function to predict the endpoint value includes a cutoff for high and low *logD* values[3]. This

3 The formula to predict the caco2 endpoint (from [185]):
$\mathrm{logP_{eff}} = -4.358 + 0.317 \times \mathrm{min}(\mathrm{max}(-1.8, \mathrm{logD}), 2.0) - 0.00558 \times \mathrm{HCPSA} - 0.179 \times \mathrm{rgyr} + 1.074 \times \mathrm{f_{rotb}}$

cannot be modeled by simple linear regression, and causes the inferior predictivity compared to support vector regression.

Finally, this use case demonstrates shortcomings of external test set validation compared to cross-validation. We have shown in Chapter 3 that not using the complete data for building the final (Q)SAR model, which is used to predict unseen compounds, will yield a less predictive model. This is especially the case if all compounds of an entire region of the dataset are removed from the model building set and split away into the test set. There is no information given on how the test set split was performed in the original article, however, the test set includes 5 neighboring compounds (see Figure 5.6). Accordingly, each of the 5 compounds is predicted with a higher error by a support vector model built on the training data (mean error 0.86) compared to the leave-one-out approach (mean error 0.58). This indicates that the final model of external test set validation has a lower predictivity for similar unseen compounds and/or has a smaller applicability domain.

5.3.2 Structural Clustering of COX-2 Data

We apply CheS-Mapper to structurally cluster and embed a dataset using MACCS keys [50], and investigate if these structural fragments are suitable to model the inhibitory potential of the dataset compounds. The dataset contains 467 COX-2 inhibitors [187, 194], that have been tested for the selective inhibition of the human enzyme Cyclooxygenase-2 (COX-2). The experimentally derived activity of each compound is stored in the dataset as IC_{50} value (half maximal inhibitory concentration). The inhibitors are structurally very similar, as they have to fit the active site of the COX-2 enzyme. We apply visual validation using MACCS keys, as a recent attempt to model the dataset with these features failed [48]. As proposed in previous work [187, 194], we transform the numeric endpoint to a binary nominal endpoint. Equal-frequency discretization yields 234 active compounds with $IC_{50} \leqslant 0.12\mu Mol$. A random forest classifier based on the structural fragments achieves 0.75 accuracy, validated with a 10-times repeated 10-fold cross-validation. To compute the structural fragments, CheS-Mapper matches the 166 SMARTS fragments of the MACCS list with the dataset compounds. This generates 97 nominal features with a minimum frequency of 10.

For visual validation, we employ the features as input for hierarchical clustering using the dynamic tree cut method that is included in CheS-Mapper to automatically compute the number of clusters [181]. Sammon's non-linear mapping is used for 3D-embedding [128]. We employ the Tanimoto similarity measure for the clustering and embedding techniques. Finally, we enable 3D alignment according

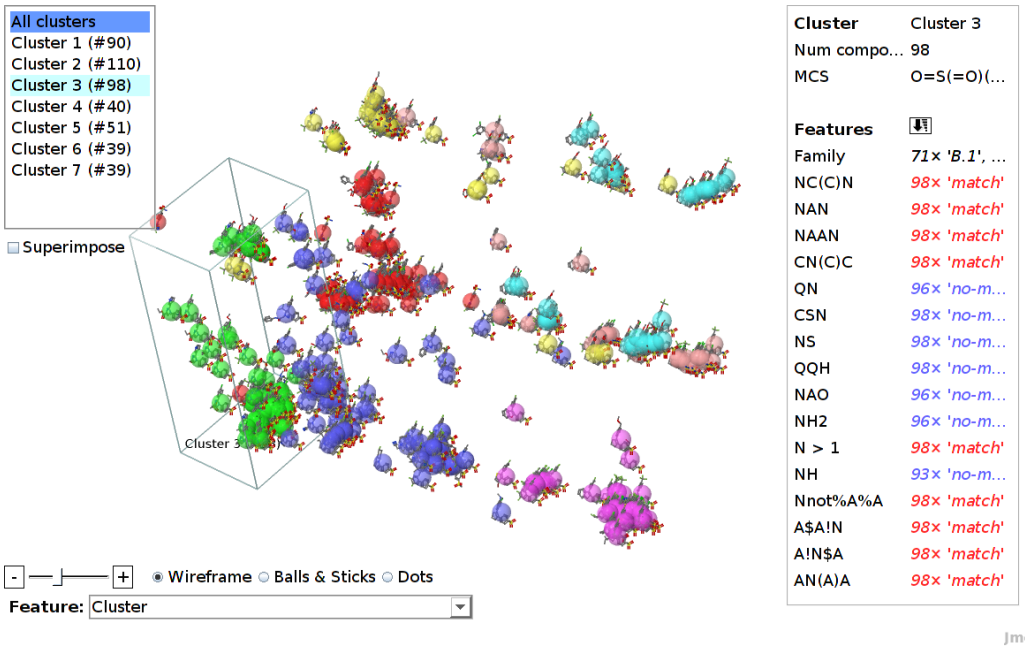


Figure 5.7: The COX-2 dataset is clustered into 7 clusters. The compounds are highlighted according to their cluster assignment. Cluster 3 is selected (as indicated by the box), and the a summary of feature values is shown on the right-hand side. Spheres are employed for highlighting (instead of changing the color of the structure).

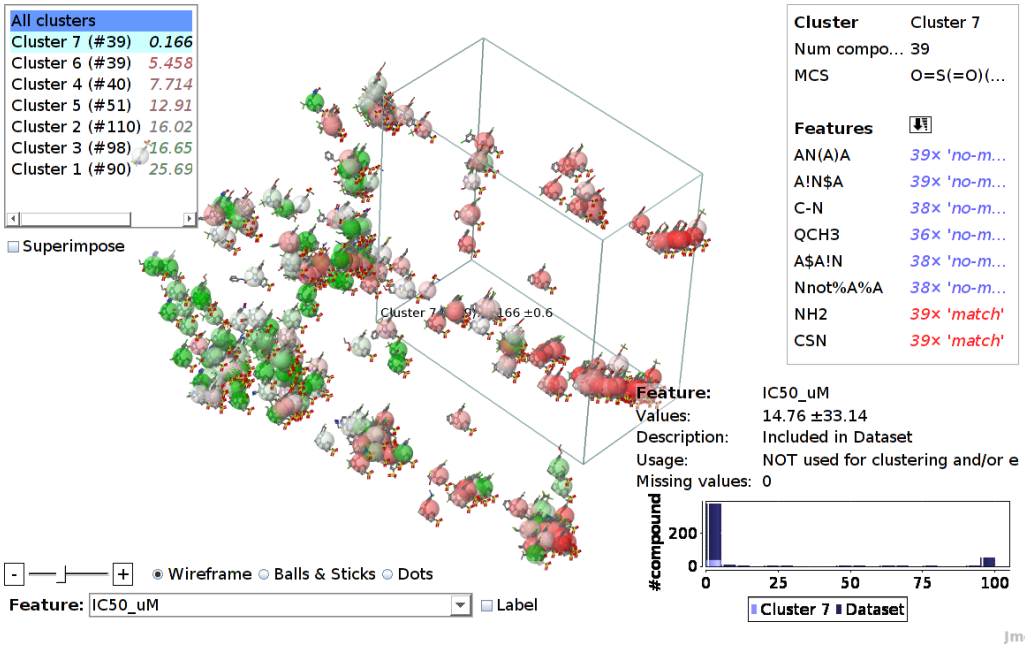


Figure 5.8: Highlighting IC_{50} values within the COX-2 dataset. The endpoint value of the COX-2 dataset is selected, showing the activity space (or landscape). A new function of CheS-Mapper has been used to modify the highlighting colors, using red for active compounds with low feature values and applying a log-transformation. Compounds with high feature values are located predominantly on the right-hand side. The selected cluster 7 includes many active compounds.

to the maximum common subgraph (MCS). The mapping result is shown in Figure 5.7 and divides the dataset into 7 clusters of different sizes (ranging from 39 to 110 compounds). Moreover, CheS-Mapper gives a warning to the user that 467 compounds have been mapped to only 333 distinct positions in 3D space due to identical feature values of numerous compounds. Hence, several of the structurally similar compounds cannot be distinguished with the 97 fragments matched by the SMARTS list (as discussed in detail below). Even though the dimensionality reduction cannot be achieved without loss of information, the distance in the 3D-space resembles the Tanimoto distance well for most compound pairs ($\mathtt{Pearson} : 0.89$). Highlighting the target endpoint (see Figure 5.8) shows that clustering and embedding apparently separate active and inactive compounds. Compounds with low IC_{50} values are mostly on the right-hand side (drawn in red), and compounds with high values (green) mostly on the left-hand side. Similarly, the endpoint values of the clusters do largely differ from each other: as an example, 34 of 39 compounds in cluster 7 are categorized as active. Accordingly, cluster 7 has a much lower mean IC_{50} value compared to other clusters (see cluster list on the left-hand side of Figure 5.8).

When investigating the clustering result, the user is usually interested in the most specific features that *define* a cluster. As the features are sorted according to specificity, CheS-Mapper makes this information easily accessible. As an example, the most specific structural feature for cluster 3, that comprises 98 predominantly inactive compounds, is the SMARTS fragment *NC(C)N*. It matches each compound of this cluster. In contrast, most of compounds in the dataset (328 of 467) do not contain this fragment. This can be seen in (the chart of) Figure 5.3, where the view has zoomed in on cluster 3, and the corresponding feature was selected. Furthermore, we applied 3D alignment using the maximum common subgraph. As this dataset consists of structurally very similar compounds, large common fragments have been found. Cluster 3 shares the fragment *O=S(=O)(c1ccc(cc1)n2ccnc2(cc))C*, that is highlighted orange in Figure 5.9. The superimposition simplifies the structural comparison of clusters within the dataset.

When computing activity cliffs for this dataset, CheS-Mapper reveals that 95 compounds share equal feature values with another compound in the dataset that has the opposite nominal endpoint value. Consequently, many of these compounds are misclassified by the (Q)SAR approach. For instance, these compounds account for the majority of compounds that are misclassified in every single repetition of the cross-validation (34 of 51 compounds). We conclude that the fragments based on the MACCS keys do provide valuable (Q)SAR information, but cannot distinguish numerous active and inactive compounds. This probably caused the bad modeling

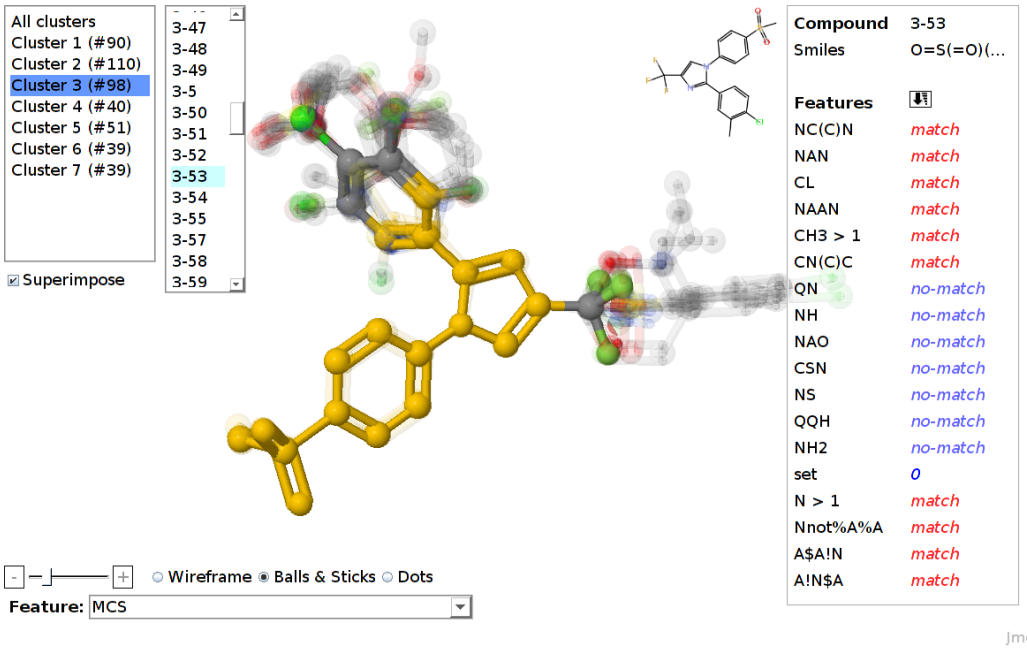


Figure 5.9: Superimposition of compounds that are aligned in 3D space. Cluster 3 of the COX-2 dataset has been 3D aligned according to the *maximum common subgraph* (MCS). In this screen-shot, the compounds are superimposed to compare the compound structures. The MCS feature is selected and therefore highlighted in orange. The depiction setting for compounds is *Balls & Sticks*.

performance in the work cited above [48]. Including additional fragments could aid to improve the (Q)SAR model.

5.3.3 Investigating Input Features for Carcinogenicity Models

We visually validate the effect of exchanging the descriptors used by a (Q)SAR algorithm. To this end, we select a subset of the Carcinogenic Potency Database (CPDB) [188] for various species. The database contains 86 compounds that have an activity value for hamster carcinogenicity assigned (active or inactive). We compute two different sets of features for these compounds with CheS-Mapper: 308 physicochemical (PC) descriptors using CDK and Open Babel, and 287 structural fragments. The structural features have been calculated by matching the compounds with three predefined SMARTS lists included in Open Babel. We choose the random forest implementation from the WEKA workbench as classification algorithm and compare two different approaches: *(Q)SAR-1* is built using only the physicochemical descriptors, while *(Q)SAR-2* exploits a combination of both feature sets. We apply a 5-times repeated 10-fold cross-validation to validate both variants.

(Q)SAR-2 achieved a classification accuracy of 75%, and significantly outperformed *(Q)SAR-1* that had a classification accuracy of only 67%. Apparently, using both feature types allows to build a more predictive model.

As CheS-Mapper's 3D embedding is based on the features, we start the program twice to (simultaneously) compare two different embeddings. When highlighting the actual endpoint value, we note that the compounds are roughly separated according to their class value. The separation (and thus the decision boundary) is less distinctive when using only PC features (Figure 5.10) compared to adding structural fragments (Figure 5.11). This indicates that it is easier for *(Q)SAR-2* to predict the endpoint, than for the *(Q)SAR-1* approach. Comparing the misclassifications of both approaches, we detect two compounds that have always been correctly classified by *(Q)SAR-2*, but not by *(Q)SAR-1*.

The inactive compound *Isonicotinic acid* (DSSTox-RID 20757) is selected in Figure 5.11 (marked with a label and drawn as 2D picture at the top right-hand side). In the embedding based on both feature types it is located in entirely inactive space. It is correctly classified by *(Q)SAR-2* in 5 of 5 repetitions of the cross-validation. In contrast, this compound was misclassified as active 2 out of 5 times by *(Q)SAR-1*. As previously described, the feature list at the top right-hand side is sorted according to specificity. Hence, *carboxylic acid* is the structural feature that distinguishes this compound the most from the remaining dataset compounds. With the help of CheS-Mapper, we detect that this compound is one of 6 compounds in the dataset that have at least one carboxyl group. All 6 carboxylic acids are inactive for this endpoint, which indicates why this compound is classified correctly when taking structural fragments into account. Moreover, we can select the nearest neighbors of the compound and inspect what they have in common. The 5 nearest neighbors are inactive, and moreover, the compound *Isonicotinic acid* and its 4 nearest neighbors are all vinylogous esters. The corresponding SMARTS fragment for vinylogous esters is matched by 12 compounds in the dataset, 10 of them are being inactive.

The mixture *1,2-Dimethylhydrazine, 2HCl* (DSSTox-RID 20517, selected in Figure 5.10) has the endpoint value active. It is always correctly classified by *(Q)SAR-2*, but misclassified 3 out of 5 times without structural fragments as features. Again, we gain insights when inspecting the most meaningful features for this compound separately, and for the compound including its neighbors. In fact, the compound, and its 3 nearest neighbors, belong to a group of 6 compounds that contain the fragment hydrazine (two connected aliphatic Nitrogen atoms). 5 of these 6 compounds have an active endpoint value in this dataset.

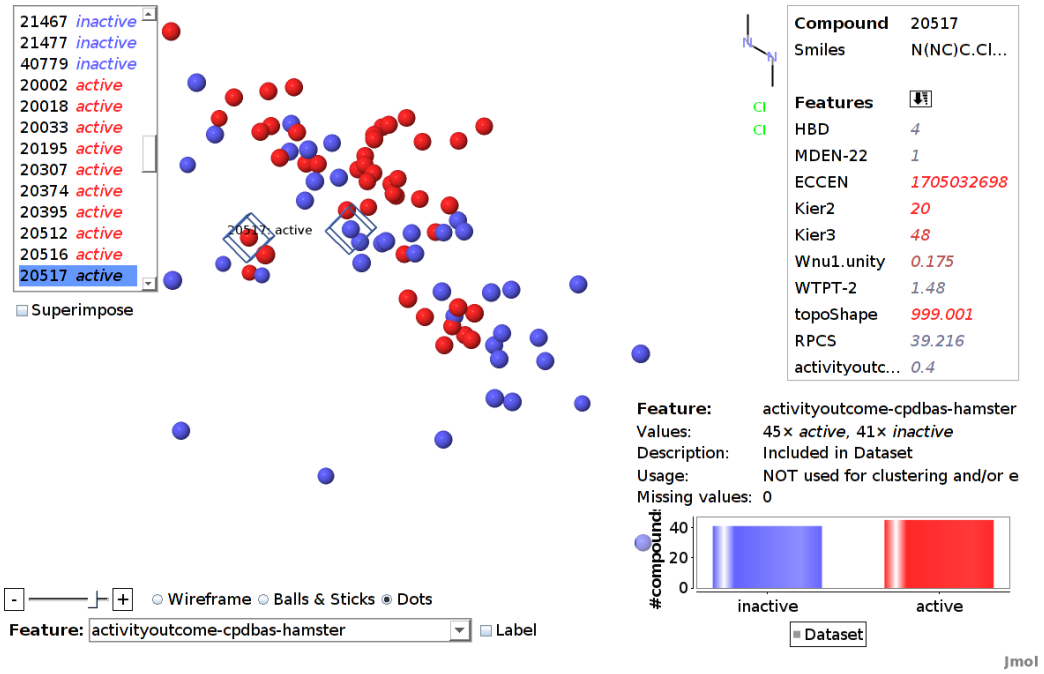


Figure 5.10: Applying PC descriptors to embed the CPDB hamster dataset. Compound 20517 (*active*) is selected, compound 20757 (*inactive*) is also marked with bounding boxes. The actual activity values are highlighted. (Dots are drawn instead of compound structures to illustrate the distribution of class values and the selection of compounds.)

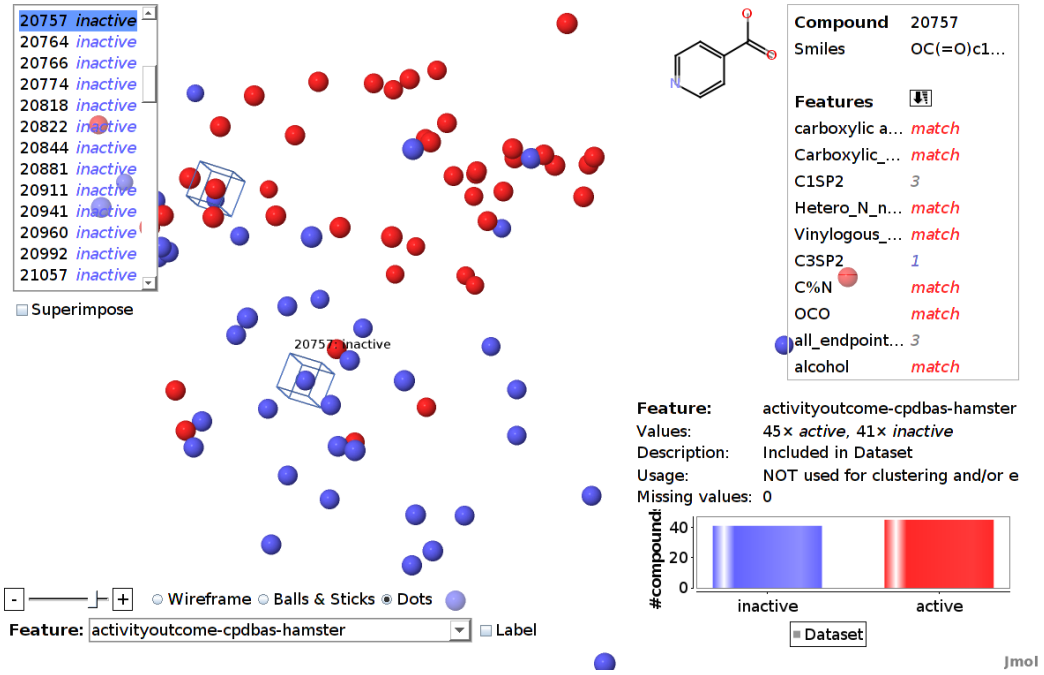


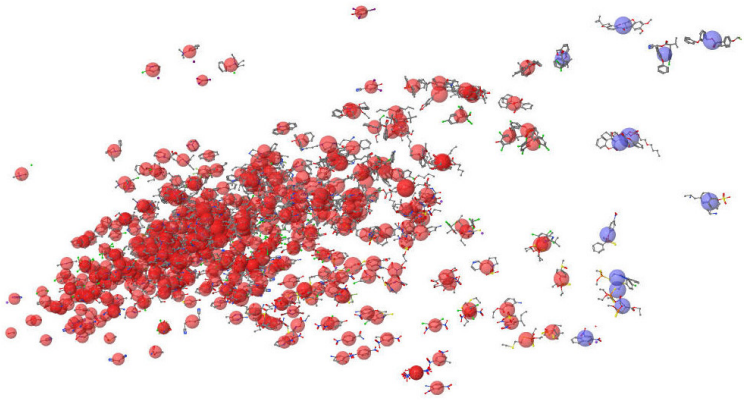
Figure 5.11: Applying PC and structural features to embed the CPDB hamster dataset. Compound 20757 (*inactive*) is selected, compound 20517 (*active*) is also marked with bounding boxes. The actual activity values are highlighted. (Dots are drawn instead of compound structures to illustrate the distribution of class values and the selection of compounds.)

5.3.4 Analyzing Applicability Domains for Fish Toxicity Prediction

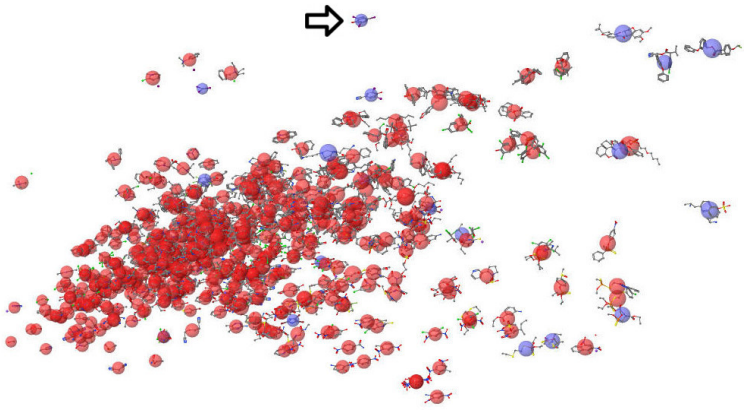
As final visual validation use case, we examine different applicability domain (AD) approaches. The use of ADs is a necessity due to the vast size of chemical space and to assure that a (Q)SAR model only *interpolates*, but does not *extrapolate*. We have introduced various AD methods in Section 2.2.4 above. AD models can be regarded as prediction algorithms, statistical models that predict whether a compound is inside or outside of the model domain. Similar to statistical (Q)SAR models, single predictions may be hard to reproduce.

As example, we select a fish toxicity dataset (Fathead Minnow Acute Toxicity) [1], published by the US EPA. The endpoint is highly correlated to physico-chemical (PC) descriptors. We have used five physico-chemical descriptors as a basis for the AD computation: molecular weight, number of bonds, octanol/water partition coefficient (*logP*), topological polar surface area (*TPSA*), and molar refractivity. Figure 5.12 shows three different AD methods applied to this dataset. The compound embedding is the same for all methods, and was performed with Sammon's mapping using default settings. The embedding quality is excellent ($Pearson : 1$), i.e. the distance between two compounds in 3D-space perfectly resembles the Euclidean distance between compound feature values. CheS-Mapper reveals that the PC feature values are correlated in this dataset, especially the values of molecular weight, number of bonds and molar refractivity (compounds on the left-hand side of the figures have low values, compounds on the right-hand side have high feature values).

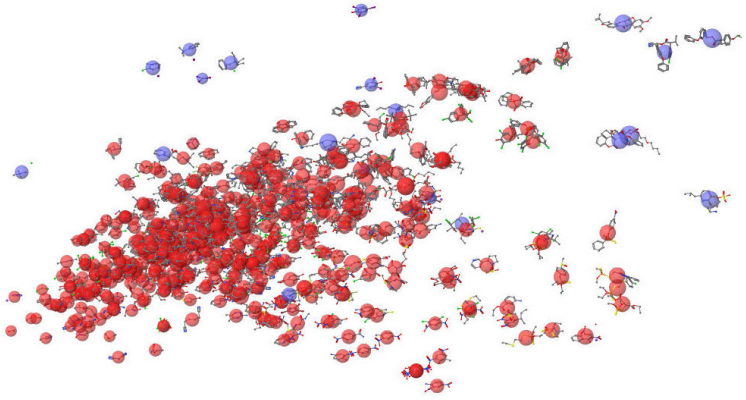
Without going into detail regarding the functionality of the AD methods, we describe some characteristics and (dis-)advantages of the AD approaches that can be investigated using CheS-Mapper. The distance-based approach using the Euclidean distance to the centroid is shown in Figure 5.12a: if compounds differ too much from the *centroid compound* (a virtual compound with mean feature values), they are excluded from the AD. As for most AD methods, the user has to set the threshold manually (we have selected 3 times the mean distance to the centroid). One disadvantage of this approach is that outliers with extreme values for a single feature are often not excluded when features are correlated. This is circumvented by the leverage approach [195]. This is a centroid distance based approach as well, but neglects inter-correlation of feature values (by computing the distance using the diagonal elements of the hat matrix). As a result, the marked compound at the top center of the embedding (*2,4,6-Triiodophenol*) is excluded from the AD with the leverage approach (see Figure 5.12b), but not with the Euclidean distance centroid approach. It is the second heaviest compound in the dataset and therefore an outlier, but it has



(a) Centroid distance-based method using the Euclidean distance



(b) Leverage method, excluded compound *2,4,6-Triiodophenol* is marked



(c) k-NN distance-based method using the Euclidean distance

Figure 5.12: Three applicability domain (AD) approaches applied to the Fathead Minnow Acute Toxicity, based on five physico-chemical descriptors. Compounds that are inside the AD are highlighted in red, compounds that are outside the AD are colored blue.

moderate number of bonds and moderate molar refractivity. Both centroid distance based approaches have the disadvantage that in diverse datasets not only individual separate outliers are removed, but also groups of outlying, similar compounds (see bottom right-hand area in the figures). The k-nearest neighbor (k-NN) distance based AD approach (Figure 5.12c) overcomes this disadvantage: compounds are only excluded from the AD if the distance to k nearest neighbors is too high (we set k to 3, and the mean distance has to be $\leqslant 3$ times the mean k-NN distance).

Which of the three approaches is more suitable for this dataset depends on the applied (Q)SAR model. Therefore, CheS-Mapper helps to understand AD methods and allows inspecting compounds that are excluded from the dataset.

5.4 Conclusions

In this chapter, we presented how visual validation can be performed with CheS-Mapper 2.0, an improved and updated version of our 3D viewer for small molecule datasets. In particular, CheS-Mapper now allows to analyze activity cliffs, to detect common properties of subgroups of compounds within the dataset, and to calculate the 3D embedding quality.

In our work, visual validation is understood as the graphical analysis of (Q)SAR model validation results. Therefore, the predicted dataset is embedded into 3D space, based on the same features that have been employed for (Q)SAR modeling. The highlighting functionality of CheS-Mapper allows to compare the predictions to the actual activity values within the feature space. The user can in particular inspect how the model predicted compounds that form activity cliffs. Visual validation can aid the (Q)SAR model developer to select appropriate features, to detect possible inconsistencies within the data, and to investigate strengths and weaknesses of the employed (Q)SAR approach. Re-iterating (Q)SAR modeling, statistical validation and visualization can improve the model predictivity and supports the researcher in mechanistically interpreting model predictions, which is an important requirement for the acceptance of (Q)SAR models as alternative testing method.

6 Conclusions

In this thesis, we presented work on visualization and validation of (Q)SAR information and (Q)SAR models. The final chapter summarizes the results, outlines how the dedicated parts of this thesis complement one another, and discusses future work.

6.1 Summary

We presented our large-scale experimental study to compare cross-validation and external test set validation in Chapter 3. The goal of comparing the two validation methods was to determine which method yields better models and which method provides better predictivity estimates for the validated (Q)SAR modeling approach. Hence, we compared the validation result to the actual performance on a large reference dataset. The results indicate that external test set validation produces weaker models due to sacrificing model building data for creating an external test set. Moreover, the predictivity estimate of cross-validation is less variable compared to external test set validation.

In the subsequent Chapter 4, we presented CheS-Mapper, a 3D viewer for small molecule datasets. CheS-Mapper is a freely available, open-source application that can be applied to investigate datasets with chemical compounds in virtual 3D space. It can load pre-computed features as well as compute physico-chemical or structural features. Subsequently, the dataset compounds are embedded into 3D space, such that compounds with similar feature values are located close to each other. Moreover, CheS-Mapper supports clustering, 3D alignment of compounds, and has various dedicated highlighting functions to show and analyze compound feature values.

Finally, we have described how the enhanced version CheS-Mapper 2.0 can be employed for visual validation in Chapter 5. In our work, visual validation denotes the graphical inspection of (Q)SAR modeling results by a human expert. CheS-Mapper can be applied to analyze the (Q)SAR information provided in the dataset by selecting physico-chemical or structural features for 3D-embedding, and subsequently highlight the actual endpoint activity of compounds within 3D space. The activity value of a particular compound can then be compared to the prediction of the (Q)SAR modeling approach. To support this analysis, we have extended the functionalities of CheS-Mapper, e.g. to compute activity cliffs.

6.2 Application to (Q)SAR Modeling

Statistical validation and graphical visualization are rather contrary approaches. Nevertheless, the different parts of this thesis are closely connected, as they are essential and complementary building blocks within the model building process.

Inspecting and cleaning the data (data curation) is frequently recommended as the first step within (Q)SAR modeling. This includes the inspection of compound structures and examination of endpoint values for potential measurement errors. If the selected features depend on the 3D structure of a compound, the computed 3D structure should be examined, as well as the calculated descriptor values. CheS-Mapper is well-suited for this task, as it can not only be used for inspecting the data, but also for feature computation and exporting.

Statistical validation is essential to select the best algorithm and feature set for (Q)SAR modeling. As outlined, it is still under discussion how to validate (Q)SAR models correctly. Proper validation helps to create predictive models with an accurately estimated predictivity estimate. Our results indicate to prefer cross-validation to an external test set validation. With regard to visually analyzing the model validation results, cross-validation has also the advantage of predicting each compound in the dataset and thus provides more data for the visual analysis (compared to external test set validation).

Visual validation supplements the statistical validation process by providing further insights. It aids detecting strengths and weaknesses of the modeling approach, by inspecting common feature values of groups of correctly classified or mis-classified compounds. Moreover, it indicates why some models perform better than others and allows inspecting the prediction of compounds that form activity cliffs. Insights from visual validation can be taken into account for a possible reiteration of the modeling process and thus can potentially improve the modeling result.

6.3 Future Work

In the future, we consider adding model building functionalities to CheS-Mapper, in order to build and visually validate (Q)SAR models directly within the software. Currently, CheS-Mapper does not support (Q)SAR modeling. For the here presented visual validation approach, external modeling software has to be applied (e.g. with KNIME, as presented in Section 5.3.1). Predicting biological or chemical activities based on the selected features within the software is relatively easy to implement, as the WEKA machine learning library is already included into the

CheS-Mapper software. The main challenge of this extension is to integrate the visual validation workflow into the graphical user interface (GUI). The modeling and validation functionality should be clear yet at the same time comprehensive, and protect the user from pitfalls like over-fitting (through e.g. "parameter fiddling") or information leakage (see Section 2.3.3). Moreover, the derived model should be applicable to unseen compounds and include the computation of applicability domains.

Moreover, we plan to integrate unfolded Circular Fingerprints as structural fragments (like e.g., ECFPs, see Section 2.1.2.2). These features haven been shown to be well suited for (Q)SAR modeling, and could therefore yield good 3D embeddings of compounds with respect to their activity endpoint information. The main challenges for this integration will be to intelligently reduce the number of fragments and to highlight single fragments. The latter is demanding due to the fact that ECFP fragments are presented by hash-codes where atom mappings for compounds have to be calculated explicitly.

Additionally, a possible use case can be found in the field of biotransformation prediction. CheS-Mapper could be employed to refine biotransformation rules by mining enzyme functional data. These rules encode compound transformations based on a substructure that acts as reaction center [196]. However, many biontransformation rules are not very selective and lead to a combinatorial explosion when predicting transformation pathways [197]. The maximum common subgraph (MCS) computation within CheS-Mapper (see Section 4.1.1.6) could be extended to include the rule reaction center (for groups of compounds that share the same Enzyme Commission (EC) number). CheS-Mapper is very well suited to inspect the calculated fragments with respect to their biochemical interpretability. In case no common fragments are found, the compound set can be split into homogeneous subgroups.

BIBLIOGRAPHY

[1] C L Russom, S P Bradbury, S J Broderius, D E Hammermeister, and R A Drummond. Predicting modes of toxic action from chemical structure: Acute toxicity in the fathead minnow (Pimephales promelas). *Environmental Toxicology and Chemistry*, 16(5):948–967, 1997.

[2] C Helma. *Predictive Toxicology*. CRC Press, March 2005.

[3] A Cherkasov, E N Muratov, D Fourches, A Varnek, I I Baskin, M Cronin, J Dearden, P Gramatica, Y C Martin, R Todeschini, V Consonni, V E Kuz'min, R Cramer, R Benigni, C Yang, J Rathman, L Terfloth, J Gasteiger, A Richard, and A Tropsha. QSAR Modeling: Where Have You Been? Where Are You Going To? *Journal of Medicinal Chemistry*, 2013.

[4] T M Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.

[5] T I Netzeva, A Worth, T Aldenberg, R Benigni, M T D Cronin, P Gramatica, J S Jaworska, S Kahn, G Klopman, C A Marchant, G Myatt, N Nikolova-Jeliazkova, G Y Patlewicz, R Perkins, D Roberts, T Schultz, D W Stanton, J J M van de Sandt, W Tong, G Veith, and C Yang. Current status of methods for defining the applicability domain of (quantitative) structure-activity relationships. The report and recommendations of ECVAM Workshop 52. *Alternatives to laboratory animals: ATLA*, 33(2):155–173, April 2005.

[6] D M Hawkins. The problem of overfitting. *J. Chem. Inf. Comput. Sci.*, 44(1):1–12, 2004.

[7] M Gütlein, C Helma, A Karwath, and S Kramer. A Large-Scale Empirical Evaluation of Cross-Validation and External Test Set Validation in (Q)SAR. *Molecular Informatics*, 32(5-6):516–528, 2013.

[8] M Gütlein, A Karwath, and S Kramer. CheS-Mapper - Chemical Space Mapping and Visualization in 3d. *Journal of Cheminformatics*, 4(1):7, March 2012.

[9] M Gütlein, A Karwath, and S Kramer. CheS-Mapper 2.0 for visual validation of (Q)SAR models. *Journal of Cheminformatics*, 6(1):41, September 2014.

[10] J Gasteiger. The Scope of Chemoinformatics. In Johann Gasteiger, editor, *Handbook of Chemoinformatics*, pages 3–5. Wiley-VCH Verlag GmbH, 2003.

[11] G W King, P C Cross, and G B Thomas. The Asymmetric Rotor III. Punched-Card Methods of Constructing Band Spectra. *The Journal of Chemical Physics*, 14(1):35–42, 1946.

[12] F K Brown. Chapter 35. Chemoinformatics: What is it and How does it Impact Drug Discovery. *Annual Reports in Medicinal Chemistry - ANNU REP MED CHEM*, 33:375–384, 1998.

[13] W L Chen. Chemoinformatics: Past, Present, and Future†. *Journal of Chemical Information and Modeling*, 46(6):2230–2255, November 2006.

[14] T Engel. Basic Overview of Chemoinformatics†. *Journal of Chemical Information and Modeling*, 46(6):2267–2277, November 2006.

[15] N Brown. Chemoinformatics—an Introduction for Computer Scientists. *ACM Comput. Surv.*, 41(2):8:1–8:38, February 2009.

[16] P Willett. From chemical documentation to chemoinformatics: 50 years of chemical information science. *Journal of Information Science*, 34(4):477–499, August 2008.

[17] A C Brown. XLIV.—On the Theory of Isomeric Compounds. *Earth and Environmental Science Transactions of the Royal Society of Edinburgh*, 23(03):707–719, 1864.

[18] P J Hansen and P C Jurs. Chemical applications of graph theory. Part I. Fundamentals and topological indices. *Journal of Chemical Education*, 65(7):574, July 1988.

[19] P G Dittmar, J Mockus, and K M Couvreur. An Algorithmic Computer Graphics Program for Generating Chemical Structure Diagrams. *Journal of Chemical Information and Modeling*, 17(3):186–192, August 1977.

[20] A M Clark, P Labute, and M Santavy. 2d Structure Depiction. *Journal of Chemical Information and Modeling*, 46(3):1107–1123, 2006.

[21] A A Gakh, M N Burnett, S V Trepalin, and A V Yarkov. Modular Chemical Descriptor Language (MCDL): Stereochemical modules. *Journal of Cheminformatics*, 3(1):5, January 2011.

[22] N O'Boyle, M Banck, C James, C Morley, T Vandermeersch, and G Hutchison. Open babel: An open chemical toolbox. *Journal of Cheminformatics*, 3(1):33, 2011.

[23] C Steinbeck, C Hoppe, S Kuhn, M Floris, R Guha, and E L Willighagen. Recent developments of the Chemistry Development Kit (CDK) - an open-source java library for chemo- and bioinformatics. *Current Pharmaceutical Design*, 12(17):2111–2120, 2006.

[24] N Jeliazkova and V Jeliazkov. AMBIT RESTful web services: an implementation of the OpenTox application programming interface. *Journal of Cheminformatics*, 3(1):18, 2011.

[25] D J Diller and K M Merz. Can we separate active from inactive conformations? *Journal of computer-aided molecular design*, 16(2):105–112, February 2002.

[26] J Sadowski and J Gasteiger. From atoms and bonds to three-dimensional atomic coordinates: automatic model builders. *Chemical Reviews*, 93(7):2567–2581, November 1993.

[27] D K Agrafiotis, A C Gibbs, F Zhu, S Izrailev, and E Martin. Conformational Sampling of Bioactive Molecules: A Comparative Study. *Journal of Chemical Information and Modeling*, 47(3):1067–1086, 2007.

[28] T A Halgren. Merck molecular force field. i. basis, form, scope, parameterization, and performance of MMFF94. *Journal of Computational Chemistry*, 17(5-6):490–519, 1996.

[29] M A Miteva, F Guyon, and P Tufféry. Frog2: Efficient 3d conformation ensemble generator for small compounds. *Nucleic acids research*, 38(Web Server issue):W622–627, July 2010.

[30] J S Puranen, M J Vainio, and M S Johnson. Accurate conformation-dependent molecular electrostatic potentials for high-throughput in silico drug discovery. *Journal of computational chemistry*, 31(8):1722–1732, June 2010.

[31] J Sadowski, J Gasteiger, and G Klebe. Comparison of Automatic Three-Dimensional Model Builders Using 639 X-ray Structures. *Journal of Chemical Information and Modeling*, 34(4):1000–1008, July 1994.

[32] A Moll, A Hildebrandt, H Lenhof, and O Kohlbacher. BALLView: a tool for research and education in molecular modeling. *Bioinformatics*, 22(3):365–366, February 2006.

[33] W Humphrey, A Dalke, and K Schulten. VMD: Visual molecular dynamics. *Journal of Molecular Graphics*, 14(1):33–38, February 1996.

[34] CAS Registry System. *Journal of Chemical Information and Computer Sciences*, 18(1):58–58, February 1978.

[35] R G Freeland, S A Funk, L J O'Korn, and G A Wilson. The Chemical Abstracts Service Chemical Registry System. II. Augmented Connectivity Molecular Formula. *Journal of Chemical Information and Computer Sciences*, 19(2):94–98, 1979.

[36] W J Wiswesser. How the WLN began in 1949 and how it might be in 1999. *Journal of Chemical Information and Computer Sciences*, 22(2):88–93, 1982.

[37] S Ash, M A Cline, R W Homer, T Hurst, and G B Smith. SYBYL Line Notation (SLN): A Versatile Language for Chemical Structure Representation†. *Journal of Chemical Information and Computer Sciences*, 37(1):71–79, January 1997.

[38] D Weininger. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, February 1988.

[39] S Heller, A McNaught, S Stein, D Tchekhovskoi, and I Pletnev. InChI - the worldwide chemical structure identifier standard. *Journal of Cheminformatics*, 5(1):7, January 2013.

[40] I Pletnev, A Erin, A McNaught, K Blinov, D Tchekhovskoi, and S Heller. InChIKey collision resistance: an experimental testing. *Journal of Cheminformatics*, 4(1):39, December 2012.

[41] D J Gluck. A Chemical Structure Storage and Search System Developed at Du Pont. *Journal of Chemical Documentation*, 5(1):43–51, February 1965.

[42] H L Morgan. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *Journal of Chemical Documentation*, 5(2):107–113, 1965.

[43] F C Bernstein, T F Koetzle, G J B Williams, E F Meyer Jr., M D Brice, J R Rodgers, O Kennard, T Shimanouchi, and M Tasumi. The protein data bank: A computer-based archival file for macromolecular structures. *Archives of Biochemistry and Biophysics*, 185(2):584–591, January 1978.

[44] G V Gkoutos, P Murray-Rust, H S Rzepa, and M Wright. Chemical Markup, XML, and the World-Wide Web. 3. Toward a Signed Semantic Chemical Web of Trust. *Journal of Chemical Information and Computer Sciences*, 41(5):1124–1130, September 2001.

[45] C Hansch and J M Clayton. Lipophilic character and biological activity of drugs II: The parabolic case. *Journal of Pharmaceutical Sciences*, 62(1):1–21, January 1973.

[46] R Todeschini and V Consonni. *Molecular Descriptors for Chemoinformatics*. John Wiley & Sons, October 2009.

[47] K Schomburg, H Ehrlich, K Stierand, and M Rarey. From Structure Diagrams to Visual Chemical Patterns. *Journal of Chemical Information and Modeling*, 50(9):1529–1535, September 2010.

[48] K Myint, L Wang, Q Tong, and X Xie. Molecular Fingerprint-based Artificial Neural Networks QSAR for Ligand Biological Activity Predictions. *Molecular pharmaceutics*, 9(10):2912–2923, October 2012.

[49] F Costa and K De Grave. Fast Neighborhood Subgraph Pairwise Distance Kernel. *Proceedings of the 26th International Conference on Machine Learning*, pages 255–262, 2010.

[50] J L Durant, B A Leland, D R Henry, and J G Nourse. Reoptimization of MDL Keys for Use in Drug Discovery. *Journal of Chemical Information and Computer Sciences*, 42(6):1273–1280, November 2002.

[51] J Klekota and F P Roth. Chemical substructures that enrich for biological activity. *Bioinformatics*, 24(21):2518–2525, November 2008.

[52] L Dehaspe, H Toivonen, and R D King. Finding Frequent Substructures in Chemical Compounds. pages 30–36. AAAI Press, 1998.

[53] X Yan and J Han. gSpan: graph-based substructure pattern mining. In *2002 IEEE International Conference on Data Mining, 2002. ICDM 2003. Proceedings*, pages 721–724, 2002.

[54] S Nijssen and J Kok. Frequent graph mining and its application to molecular databases. In *2004 IEEE International Conference on Systems, Man and Cybernetics*, volume 5, pages 4571–4577 vol.5, October 2004.

[55] A Maunz, C Helma, and S Kramer. Large-scale Graph Mining Using Backbone Refinement Classes. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 617–626, New York, NY, USA, 2009. ACM.

[56] D Rogers and M Hahn. Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010.

[57] H M Berman, J Westbrook, Z Feng, G Gilliland, T N Bhat, H Weissig, I N Shindyalov, and P E Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, January 2000.

[58] J Gasteiger, M Reitz, and O Sacher. Multidimensional Exploration into Biochemical Pathways. In *ABSTRACTS OF PAPERS OF THE AMERICAN CHEMICAL SOCIETY*, volume 221, pages U459–U459. AMER CHEMICAL SOC 1155 16TH ST, NW, WASHINGTON, DC 20036 USA, 2001.

[59] S Heller. The NIST/EPA/MSDC Mass Spectra Database, PC Version 3.0. *Journal of Chemical Information and Computer Sciences*, 31(2):352–354, 1991.

[60] Y Wang, J Xiao, T O Suzek, J Zhang, J Wang, and S H Bryant. PubChem: a public information system for analyzing bioactivities of small molecules. *Nucleic Acids Research*, 37(suppl 2):W623–W633, July 2009.

[61] E E Bolton, Y Wang, P A Thiessen, and S H Bryant. Chapter 12 PubChem: Integrated Platform of Small Molecules and Biological Activities. In Ralph A. Wheeler and David C. Spellmeyer, editor, *Annual Reports in Computational Chemistry*, volume Volume 4, pages 217–241. Elsevier, 2008.

[62] A Gaulton, L J Bellis, A P Bento, J Chambers, M Davies, A Hersey, Y Light, S McGlinchey, D Michalovich, B Al-Lazikani, and J P Overington. ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Research*, 40(D1):D1100–D1107, September 2011.

[63] E L Willighagen, A Waagmeester, O Spjuth, P Ansell, A J Williams, V Tkachenko, J Hastings, B Chen, and D J Wild. The ChEMBL database as linked open data. *Journal of Cheminformatics*, 5(1):23, May 2013.

[64] S Heller. The Beilstein Online Database. In *The Beilstein Online Database*, number 436 in ACS Symposium Series, pages 1–9. American Chemical Society, August 1990.

[65] B Bienfait and P Ertl. JSME: a free molecule editor in JavaScript. *Journal of Cheminformatics*, 5(1):24, May 2013.

[66] J M Barnard. Substructure searching methods: Old and new. *Journal of Chemical Information and Computer Sciences*, 33(4):532–538, July 1993.

[67] L C Ray and R A Kirsch. Finding Chemical Records by Digital Computers. *Science*, 126(3278):814–819, October 1957.

[68] C Hansch, P P Maloney, T Fujita, and R M Muir. Correlation of Biological Activity of Phenoxyacetic Acids with Hammett Substituent Constants and Partition Coefficients. *Nature*, 194(4824):178–180, April 1962.

[69] A Leo, P Y C Jow, C Silipo, and C Hansch. Calculation of hydrophobic constant (log P) from .pi. and f constants. *Journal of Medicinal Chemistry*, 18(9):865–868, September 1975.

[70] G Klopman. Artificial intelligence approach to structure-activity studies. Computer automated structure evaluation of biological activity of organic molecules. *Journal of the American Chemical Society*, 106(24):7315–7321, November 1984.

[71] R D Cramer, D E Patterson, and J D Bunce. Comparative molecular field analysis (CoMFA). 1. Effect of shape on binding of steroids to carrier proteins. *Journal of the American Chemical Society*, 110(18):5959–5967, August 1988.

[72] I H Witten and E Frank. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition*. Morgan Kaufmann, July 2005.

[73] G H John and P Langley. Estimating Continuous Distributions in Bayesian Classifiers. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI'95, pages 338–345, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[74] V Vapnik. *The Nature of Statistical Learning Theory*. Springer, 2000.

[75] J R Quinlan. Learning With Continuous Classes. In *Proceedings of the 5th Australian joint Conference on Artificial Intelligence*, pages 343–348. World Scientific, 1992.

[76] Y Wang and I H Witten. Inducing Model Trees for Continuous Classes. In *In Proc. of the 9th European Conf. on Machine Learning Poster Papers*, pages 128–137, 1997.

[77] M Hall, E Frank, G Holmes, B Pfahringer, P Reutemann, and I H Witten. The WEKA data mining software: an update. *SIGKDD Explor. Newsl.*, 11:10–18, November 2009.

[78] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0.

[79] J Demšar, B Zupan, G Leban, and T Curk. Orange: From Experimental Machine Learning to Interactive Data Mining. In Jean-François Boulicaut, Floriana Esposito, Fosca Giannotti, and Dino Pedreschi, editors, *Knowledge Discovery in Databases: PKDD 2004*, number 3202 in Lecture Notes in Computer Science, pages 537–539. Springer Berlin Heidelberg, January 2004.

[80] L Breiman. Random Forests. *Machine Learning*, 45(1):5–32, October 2001.

[81] J E Ridings, M D Barratt, R Cary, C G Earnshaw, C E Eggington, M K Ellis, P N Judson, J J Langowski, C A Marchant, M P Payne, W P Watson, and T D Yih. Computer prediction of possible toxic action from chemical structure: an update on the DEREK system. *Toxicology*, 106(1–3):267–279, January 1996.

[82] Y Woo and D Y Lai. OncoLogic: a mechanism-based expert system for predicting the carcinogenic potential of chemicals. In *Predictive Toxicology*, pages 385–413. CRC Press, 2005.

[83] G Patlewicz, D W Roberts, A Aptula, K Blackburn, and B Hubesch. Workshop: use of "read-across" for chemical safety assessment under REACH. *Regulatory toxicology and pharmacology: RTP*, 65(2):226–228, March 2013.

[84] Organisation for Economic Co-operation and Development (OECD). *GUIDANCE DOCUMENT FOR USING THE OECD (Q)SAR APPLICATION TOOLBOX TO DEVELOP CHEMICAL CATEGORIES ACCORDING TO THE OECD GUIDANCE ON GROUPING OF CHEMICALS*. OECD Environment, Health and Safety Publications, Paris, France, ENV/JM/MONO(2009)5, 2009.

[85] J C Dearden, M T D Cronin, and K L E Kaiser. How not to develop a quantitative structure-activity or structure-property relationship (QSAR/QSPR). *SAR and QSAR in environmental research*, 20(3-4):241–266, 2009.

[86] J Jaworska and N Nikolova-Jeliazkova. QSAR applicabilty domain estimation by projection of the training set descriptor space: a review. *Alternatives to laboratory animals : ATLA*, 33(5):445–59, 2005.

[87] L Eriksson, J Jaworska, A P Worth, M T D Cronin, R M McDowell, and P Gramatica. Methods for reliability and uncertainty assessment and for applicability evaluations of classification- and regression-based QSARs. *Environmental health perspectives*, 111(10):1361–1375, August 2003.

[88] B W Silverman. *Density Estimation for Statistics and Data Analysis*. CRC Press, April 1986.

[89] A Maunz, M Gütlein, M Rautenberg, D Vorgrimmler, D Gebele, and C Helma. lazar: a modular predictive toxicology framework. *Predictive Toxicology*, 4:38, 2013.

[90] Council of the European Union European Parliament. *Regulation (EC) No 1907/2006 of the European Parliament and of the Council of 18 December 2006*

concerning the Registration, Evaluation, Authorisation and Restriction of Chemicals (REACH), establishing a European Chemicals Agency. Brussels, Belgium, 2003/0256/COD, 2006.

[91] Organisation for Economic Co-operation and Development (OECD). *GUIDANCE DOCUMENT ON THE VALIDATION OF (QUANTITATIVE) STRUCTURE-ACTIVITY RELATIONSHIP [(Q)SAR] MODELS*. Paris, France, ENV/JM/MONO(2007)2, 2007.

[92] European Chemicals Agency (ECHA). *Evaluation under REACH Progress Report 2013*. European Chemicals Agency (ECHA), Helsinki, Finland, 2014.

[93] T M Mitchell. Machine learning and data mining. *Commun. ACM*, 42(11):30–36, 1999.

[94] S C Larson. The shrinkage of the coefficient of multiple correlation. *Journal of Educational Psychology*, 22(1):45–55, 1931.

[95] T G Dietterich. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation*, 10:1895–1923, 1998.

[96] T Hastie, R Tibshirani, and J H Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer, New York, NY, USA, 2009.

[97] S Arlot and A Celisse. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40–79, 2010.

[98] P Gramatica. Principles of QSAR models validation: internal and external. *QSAR Comb. Sci.*, 26(5):694–701, 2007.

[99] J J Heckman. Sample selection bias as a specification error. *Econometrica*, 47(1):153–161, 1979.

[100] B Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Twenty-first international conference on Machine learning*, ICML '04, pages 114–122, New York, NY, USA, 2004. ACM.

[101] R Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *International joint conference on artificial intelligence*, volume 14, pages 1137–1145, 1995.

[102] A G Saliner, T I Netzeva, and A P Worth. Prediction of estrogenicity: validation of a classification model. *SAR QSAR Environ. Res*, 17(2):195–223, 2006.

[103] A P Worth, A Bassan, A Gallegos, T I Netzeva, G Patlewicz, M Pavan, I Tsakovska, and M Vracko. *The Characterisation of (Quantitative) Structure-Activity Relationships: Preliminary Guidance*. European Chemicals Bureau, Ispra (VA) Italy, EUR 21866 EN, 2005.

[104] P A Lachenbruch and M R Mickey. Estimation of error rates in discriminant analysis. *Technometrics*, 10(1):1–11, 1968.

[105] D M Allen. The relationship between variable selection and data agumentation and a method for prediction. *Technometrics*, 16(1):125–127, 1974.

[106] L. Breiman, J. Friedman, C.J. Stone, and R.A. Olshen. *Classification and regression trees*. Chapman & Hall/CRC, New York, NY, USA, 1984.

[107] J R Quinlan. Learning with continuous classes. In *5th Australian joint Conference on Artificial Intelligence*, pages 343–348. Singapore, World Scientific, 1992.

[108] M Kearns. A bound on the error of cross validation using the approximation and estimation rates, with consequences for the training-test split. *Neural Comput.*, 9(5):183–189, 1996.

[109] M Kearns, Y Mansour, A Y Ng, and D Ron. An experimental and theoretical comparison of model selection methods. *Machine Learning*, 27(1):7–50, 1997.

[110] A Golbraikh and A Tropsha. Beware of q2! *J. Mol. Graphics Modell.*, 20(4):269 – 276, 2002.

[111] P Gramatica, E Giani, and E Papa. Statistical external validation and consensus modeling: A qspr case study for koc prediction. *J. Mol. Graphics Modell.*, 25(6):755 – 766, 2007.

[112] I Kahn, D Fara, M Karelson, U Maran, and P L Andersson. Qspr treatment of the soil sorption coefficients of organic pollutants. *J. Chem. Inf. Model.*, 45(1):94–105, 2005.

[113] J T Leonard and K Roy. On selection of training and test sets for the development of predictive qsar models. *QSAR Comb. Sci.*, 25(3):235–251, 2006.

[114] D M Hawkins, S C Basak, and D Mills. Assessing model fit by cross-validation. *J. Chem. Inf. Comput. Sci.*, 43(2):579–586, 2003.

[115] A Tropsha. Best practices for qsar model development, validation, and exploitation. *Molecular Informatics*, 29(6-7):476–488, 2010.

[116] K Baumann and N Stiefl. Validation tools for variable subset regression. *J. Comput.-Aided Mol. Des.*, 18(7):549–562, 2004.

[117] P Smialowski, D Frishman, and S Kramer. Pitfalls of supervised feature selection. *Bioinformatics*, 26(3):440–443, 2010.

[118] S Varma and R Simon. Bias in error estimation when using cross-validation for model selection. *BMC Bioinf.*, 7(1):91, 2006.

[119] G C Cawley and N L C Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *J. Mach. Learn. Res.*, 99:2079–2107, 2010.

[120] H Kubinyi, Fred A, and T Mietzner. Three-dimensional quantitative similarity−activity relationships (3d qsiar) from seal similarity matrices. *J. Med. Chem.*, 41(14):2553–2564, 1998.

[121] A O Aptula, N Jeliazkova, T W Schultz, and M T D Cronin. The better predictive model: High q2 for the training set or low root mean square error of prediction for the test set? *QSAR Comb. Sci.*, 24(3):385–396, 2005.

[122] D K Agrafiotis, S Alex, H Dai, A Derkinderen, M Farnum, P Gates, S Izrailev, E P Jaeger, P Konstant, A Leung, V S Lobanov, P Marichal, D Martin, D N Rassokhin, M Shemanarev, A Skalkin, J Stong, T Tabruyn, M Vermeiren, J Wan, X Y Xu, and X Yao. Advanced biological and chemical discovery (ABCD): Centralizing discovery knowledge in an inherently decentralized world. *Journal of Chemical Information and Modeling*, 47(6):1999–2014, 2007. PMID: 17973472.

[123] O Spjuth, J Alvarsson, A Berg, M Eklund, S Kuhn, C Mäsak, G Torrance, J Wagener, E Willighagen, C Steinbeck, and J Wikberg. Bioclipse 2: A scriptable integration platform for the life sciences. *BMC Bioinformatics*, 10(10), 2009.

[124] K R Lawson and J Lawson. LICSS - a chemical spreadsheet in microsoft excel. *Journal of Cheminformatics*, 4(1):3, February 2012.

[125] L J P van der Maaten, E O Postma, and H J van den Herik. *Dimensionality Reduction: A Comparative Review*. Citeseer, 2008.

[126] M Reutlinger and G Schneider. Nonlinear dimensionality reduction and mapping of compound libraries for drug discovery. *Journal of Molecular Graphics and Modelling*, 34:108–117, April 2012.

[127] Y A Ivanenkov, N P Savchuk, S Ekins, and K V Balakin. Computational mapping tools for drug discovery. *Drug Discovery Today*, 14(15–16):767–775, August 2009.

[128] J W Sammon. A nonlinear mapping for data structure analysis. *Computers, IEEE Transactions on*, C-18(5):401 – 409, may 1969.

[129] J De Leeuw and P Mair. Multidimensional scaling using majorization: Smacof in R. *Journal of Statistical Software*, 2006.

[130] T Kohonen. Self-Organization and Associative Memory. *Self-Organization and Associative Memory, 100 figs. XV, 312 pages.. Springer-Verlag Berlin Heidelberg New York. Also Springer Series in Information Sciences, volume 8*, -1, 1988.

[131] J Gasteiger, X Li, and A Uschold. The beauty of molecular surfaces as revealed by self-organizing neural networks. *Journal of Molecular Graphics*, 12(2):90–97, June 1994.

[132] L van der Maaten and G Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research 9*, 2008.

[133] A Givehchi, A Dietrich, P Wrede, and G Schneider. ChemSpaceShuttle: A tool for data mining in drug discovery by classification, projection, and 3d visualization. *QSAR & Combinatorial Science*, 22(5):549–559, 2003.

[134] M Awale, R van Deursen, and J Reymond. MQN-Mapplet: Visualization of Chemical Space with Interactive Maps of DrugBank, ChEMBL, PubChem, GDB-11, and GDB-13. *Journal of Chemical Information and Modeling*, 53(2):509–518, February 2013.

[135] V L Guilloux, A Arrault, L Colliandre, S Bourg, P Vayer, and L Morin-Allory. Mining collections of compounds with Screening Assistant 2. *Journal of Cheminformatics*, 4(1):20, August 2012.

[136] P Skoda and D Hoksza. Chemical space visualization using ViFrame. In *2013 IEEE/ACIS 12th International Conference on Computer and Information Science (ICIS)*, pages 541–546, 2013.

[137] H Strobelt, E Bertini, J Braun, O Deussen, U Groth, T U Mayer, and D Merhof. HiTSEE KNIME: a visualization tool for hit selection and analysis in high-throughput screening experiments for the KNIME platform. *BMC Bioinformatics*, 13(Suppl 8):S4, May 2012.

[138] C Kibbey and A Calvet. Molecular Property eXplorer: A Novel Approach to Visualizing SAR Using Tree-Maps and Heatmaps. *Journal of Chemical Information and Modeling*, 45(2):523–532, 2005.

[139] D K Agrafiotis, D Bandyopadhyay, and M Farnum. Radial Clustergrams: Visualizing the Aggregate Properties of Hierarchical Clusters. *Journal of Chemical Information and Modeling*, 47(1):69–75, January 2007.

[140] D Gupta-Ostermann, Y Hu, and J Bajorath. Introducing the LASSO Graph for Compound Data Set Representation and Structure-Activity Relationship Analysis. *Journal of Medicinal Chemistry*, 55(11):5546–5553, June 2012.

[141] R Guha and J H Van Drie. Structure–activity landscape index: Identifying and quantifying activity cliffs. *Journal of Chemical Information and Modeling*, 48(3):646–658, 2008. PMID: 18303878.

[142] E Lounkine, M Wawer, A M Wassermann, and J Bajorath. SARANEA: A Freely Available Program To Mine Structure-Activity and Structure-Selectivity Relationship Information in Compound Data Sets. *Journal of Chemical Information and Modeling*, 50(1):68–78, January 2010.

[143] K Klein, O Koch, N Kriege, P Mutzel, and T Schäfer. Visual Analysis of Biological Activity Data with Scaffold Hunter. *Molecular Informatics*, 32(11-12):964–975, 2013.

[144] M Wawer and J Bajorath. Similarity-potency trees: A method to search for SAR information in compound data sets and derive SAR rules. *Journal of Chemical Information and Modeling*, 50(8):1395–1409, 2010.

[145] M A Johnson, G M Maggiora, and A C S Meeting. *Concepts and applications of molecular similarity*. Wiley, 1990.

[146] G M Maggiora. On outliers and activity cliffs–why QSAR often disappoints. *Journal of chemical information and modeling*, 46(4):1535, August 2006.

[147] M Lajiness. Evaluation of the Performance of Dissimilarity Selection Methodology. In *QSAR, rational approaches to the design of bioactive compounds*, pages 201–204. Distributors for the US and Canada, Elsevier Science, 1991.

[148] V Shanmugasundaram and G Maggiora. Characterizing property and activity landscapes using an information-theoretic approach. *222nd American Chemical Society National Meeting*, 2001.

[149] N Jeliazkova and V Jeliazkov. Chemical landscape analysis with the OpenTox framework. *Current topics in medicinal chemistry*, 12(18):1987–2001, 2012.

[150] P Ertl and B Rohde. The Molecule Cloud - compact visualization of large collections of molecules. *Journal of Cheminformatics*, 4(1):12, July 2012.

[151] A M Wassermann, P Haebel, N Weskamp, and J Bajorath. SAR Matrices: Automated Extraction of Information-Rich SAR Tables from Large Compound Data Sets. *Journal of Chemical Information and Modeling*, 52(7):1769–1776, July 2012.

[152] M R Berthold, N Cebron, F Dill, T R Gabriel, T Kötter, T Meinl, P Ohl, C Sieb, K Thiel, and B Wiswedel. KNIME: The Konstanz Information Miner. In Christine Preisach, Professor Dr Hans Burkhardt, Professor Dr Lars Schmidt-Thieme, and Professor Dr Reinhold Decker, editors, *Data Analysis, Machine Learning and Applications*, Studies in Classification, Data Analysis, and Knowledge Organization, pages 319–326. Springer Berlin Heidelberg, Berlin Heidelberg, January 2008.

[153] S Bremm, T von Landesberger, J Bernard, and T Schreck. Assisted Descriptor Selection Based on Visual Comparative Data Analysis. *Computer Graphics Forum*, 30(3):891–900, 2011.

[154] H Hofmann, A P J M Siebes, and A F X Wilhelm. Visualizing Association Rules with Interactive Mosaic Plots. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '00, pages 227–235, New York, NY, USA, 2000. ACM.

[155] J Han and N Cercone. RuleViz: A Model for Visualizing Knowledge Discovery Process. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '00, pages 244–253, New York, NY, USA, 2000. ACM.

[156] D Cook, D Caragea, and V Honavar. Visualization for classification problems, with examples using support vector machines. In *in: Proceedings of the COMPSTAT 2004, 16th Symposium of IASC*, 2004.

[157] M Ankerst, M Ester, and H Kriegel. Towards an effective cooperation of the user and the computer for classification. In *Proc. 6th Intl. Conf. on Knowledge Discovery and Data Mining*, KDD '00, pages 179–188, New York, NY, USA, 2000. ACM Press.

[158] C Brunk, J Kelly, and R Kohavi. *MineSet: An Integrated System for Data Mining*. -, 1997.

[159] P Rheingans and M desJardins. Visualizing High-Dimensional Predictive Model Quality. In *In Proceedings of IEEE Visualization 2000*, pages 493–496, 2000.

[160] C Seifert and E Lex. A Novel Visualization Approach for Data-Mining-Related Classification. In *Information Visualisation, 2009 13th International Conference*, pages 490–495, July 2009.

[161] J Kazius, R McGuire, and R Bursi. Derivation and validation of toxicophores for mutagenicity prediction. *J. Med. Chem.*, 48(1):312–320, 2005.

[162] K Hansen, S Mika, T Schroeter, A Sutter, A ter Laak, T Steger-Hartmann, N Heinrich, and K-R Müller. Benchmark data set for in silico prediction of ames mutagenicity. *J. Chem. Inf. Model.*, 49(9):2077–2081, 2009. PMID: 19702240.

[163] M Karthikeyan, R C Glen, and A Bender. General melting point prediction based on a diverse compound data set and artificial neural networks. *J. Chem. Inf. Model.*, 45(3):581–590, 2005.

[164] US EPA. *EPI Suite Data - ISISBase & SDF for the Estimation Programs Interface Suite*. United States Environmental Protection Agency, Washington, DC, USA, 2012.

[165] H Zhu, T M Martin, L Ye, A Sedykh, D M Young, and A Tropsha. Quantitative structure−activity relationship modeling of rat acute toxicity by oral exposure. *Chem. Res. Toxicol.*, 22(12):1913–1921, 2009. PMID: 19845371.

[166] M Hall, E Frank, G Holmes, B Pfahringer, P Reutemann, and I H Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, 2009.

[167] W G Cochran. *Sampling Techniques*. John Wiley & Sons, Inc., New York, NY, 1977.

[168] P Langfelder, B Zhang, and S Horvath. Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r. *Bioinformatics*, 24(5):719–720, 2008.

[169] T I Netzeva, A P Worth, T Aldenberg, R Benigni, M T D Cronin, P Gramatica, J S Jaworska, S Kahn, G Klopman, C A Marchant, et al. Current status of methods for defining the applicability domain of (quantitative) structure-activity relationships. *ATLA, Altern. Lab. Anim.*, 33(2):1–19, 2005.

[170] L I-K Lin. A concordance correlation coefficient to evaluate reproducibility. *Biometrics*, 45(1):pp. 255–268, 1989.

[171] N Chirico and P Gramatica. Real external predictivity of qsar models. part 2. new intercomparable thresholds for different validation criteria and the need for scatter plot inspection. *J. Chem. Inf. Model.*, 52(8):2044–2058, 2012.

[172] R Hajjo, V Setola, Bryan L Roth, and A Tropsha. Chemocentric informatics approach to drug discovery: Identification and experimental validation of selective estrogen receptor modulators as ligands of 5-hydroxytryptamine-6 receptors and as potential cognition enhancers. *J. Med. Chem.*, 55(12):5704–5719, 2012.

[173] M Eklund, O Spjuth, and J E Wikberg. The c1c2: A framework for simultaneous model selection and assessment. *BMC Bioinf.*, 9(1):360, 2008.

[174] P Filzmoser, B Liebmann, and K Varmuza. Repeated double cross validation. *J. Chemom.*, 23(4):160–171, 2009.

[175] B Hardy, N Douglas, C Helma, M Rautenberg, N Jeliazkova, V Jeliazkov, I Nikolova, R Benigni, O Tcheremenskaia, S Kramer, T Girschick, F Buchwald, J Wicker, A Karwath, M Gütlein, A Maunz, H Sarimveis, G Melagraki, A Afantitis, P Sopasakis, D Gallagher, V Poroikov, D Filimonov, A Zakharov, A Lagunin, T Gloriozova, S Novikov, N Skvortsova, D Druzhilovsky, S Chawla, I Ghosh, S Ray, H Patel, and S Escher. Collaborative development of predictive toxicology applications. *Journal of Cheminformatics*, 2(1):7, August 2010.

[176] N L Allinger. Conformational analysis. 130. MM2. A hydrocarbon force field utilizing V1 and V2 torsional terms. *Journal of the American Chemical Society*, 99(25):8127–8134, 1977.

[177] G Patlewicz, N Jeliazkova, R J Safford, A P Worth, and B Aleksiev. An evaluation of the implementation of the cramer classification scheme in the toxtree software. *SAR and QSAR in Environmental Research*, 19(5-6):495–524, 2008.

[178] K R Przybylak and M T D Cronin. In silico studies of the relationship between chemical structure and drug induced phospholipidosis. *Molecular Informatics*, 30(5):415–429, 2011.

[179] T Caliński and J Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27, 1974.

[180] J Oksanen, R Kindt, P Legendre, B O'Hara, and M H H Stevens. *The vegan Package*, 2007. http://r-forge.r-project.org/projects/vegan.

[181] P Langfelder, B Zhang, and S Horvath. Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r. *Bioinformatics*, 24(5):719–720, 2008.

[182] D H Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2:139–172, 1987. 10.1007/BF00114265.

[183] S Dasgupta and P M Long. Performance guarantees for hierarchical clustering. *Journal of Computer and System Sciences*, 70(4):555 – 569, 2005. Special Issue on COLT 2002.

[184] W Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5):922–923, September 1976.

[185] T. J. Hou, W. Zhang, K. Xia, X. B. Qiao, and X. J. Xu. Adme evaluation in drug discovery. 5. correlation of caco-2 permeation with simple molecular properties. *Journal of Chemical Information and Computer Sciences*, 44(5):1585–1600, 2004.

[186] E Papa, S Kovarich, and P Gramatica. Development, validation and inspection of the applicability domain of qspr models for physicochemical properties of polybrominated diphenyl ethers. *QSAR & Combinatorial Science*, 28(8):790–796, 2009.

[187] G W Kauffman and P C Jurs. QSAR and k-Nearest Neighbor Classification Analysis of Selective Cyclooxygenase-2 Inhibitors Using Topologically-Based Numerical Descriptors. *Journal of Chemical Information and Computer Sciences*, 41(6):1553–1560, November 2001.

[188] L S Gold, N B Manley, T H Slone, and L Rohrbach. Supplement to the Carcinogenic Potency Database (CPDB): results of animal bioassays published in the general literature in 1993 to 1994 and by the National Toxicology Program in 1995 to 1996. *Environmental Health Perspectives*, 107(Suppl 4):527–600, August 1999.

[189] C Borgelt, T Meinl, and M Berthold. MoSS: A Program for Molecular Substructure Mining. In *Proceedings of the 1st International Workshop on Open Source Data Mining: Frequent Pattern Mining Implementations*, OSDM '05, pages 6–15, New York, NY, USA, 2005. ACM.

[190] EC. *Annexes to the Final Report on Principles for Establishing the Status of Development and Validation of (Quantitative) Structure-Activity Relationships [(Q)SARs]*. Paris, France, (ENV/JM/TG(2004)27/ANN), 2004.

[191] J D Evans. *Straightforward Statistics for the Behavioral Sciences*. Duxbury Press, Pacific Grove, January 1996.

[192] D Howell. *Statistical Methods for Psychology*. Cengage Learning, Boston, US, January 2012.

[193] J L Medina-Franco. Activity Cliffs: Facts or Artifacts? *Chemical Biology & Drug Design*, 81(5):553–556, 2013.

[194] J J Sutherland, L A O'Brien, and D F Weaver. Spline-Fitting with a Genetic Algorithm: A Method for Developing Classification Structure-Activity Relationships. *Journal of Chemical Information and Computer Sciences*, 43(6):1906–1915, November 2003.

[195] J Neter, M Kutner, W Wasserman, and C Nachtsheim. *Applied Linear Statistical Models*. McGraw-Hill/Irwin, Chicago, 4 edition edition, February 1996.

[196] B K Hou, L P Wackett, and L B M Ellis. Microbial Pathway Prediction: A Functional Group Approach. *Journal of Chemical Information and Computer Sciences*, 43(3):1051–1057, May 2003.

[197] K Fenner, J Gao, S Kramer, L Ellis, and L Wackett. Data-driven extraction of relative reasoning rules to limit combinatorial explosion in biodegradation pathway prediction. *Bioinformatics*, 24(18):2079–2085, September 2008.

A Supplementary Material

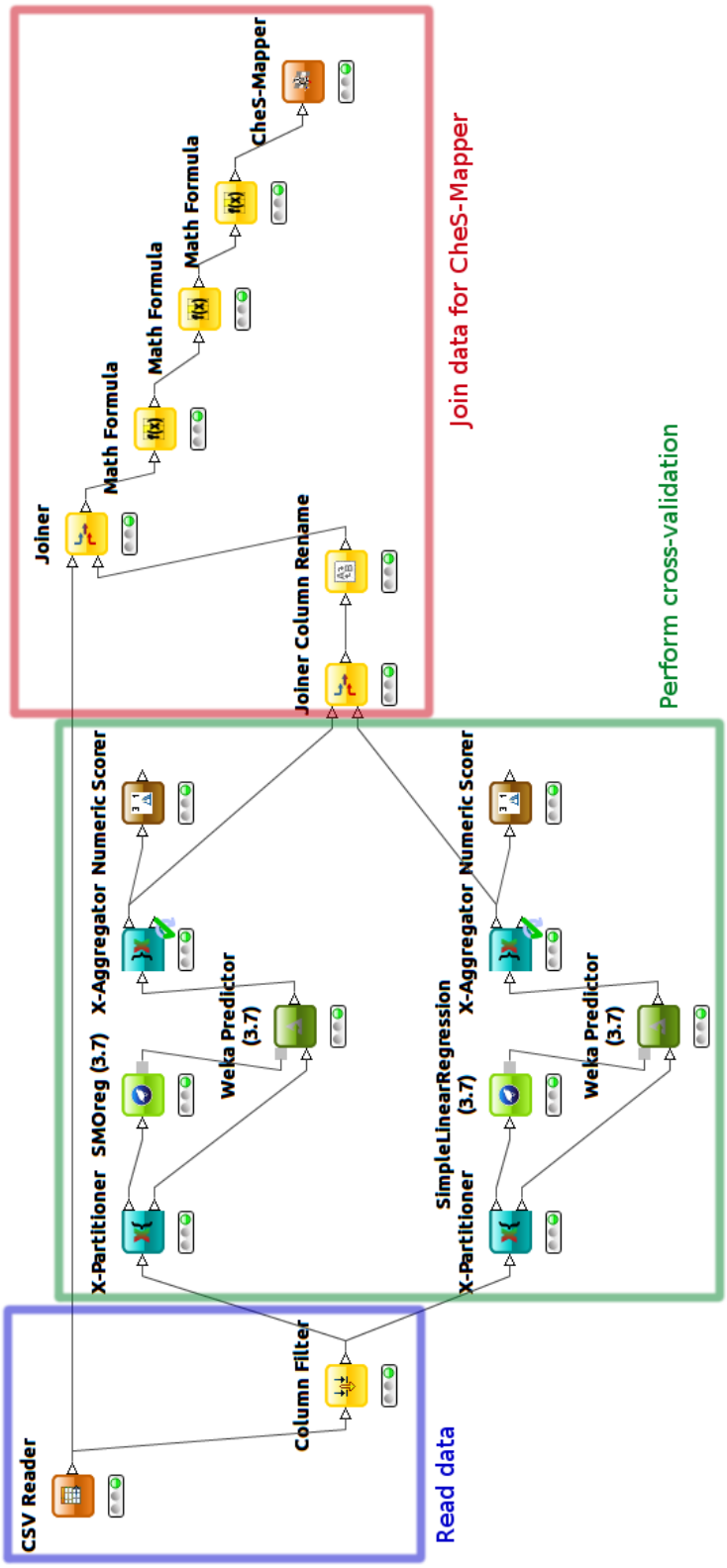
A.1 A KNIME Workflow for Visually Validating LOO-CV

This section describes how KNIME is used to visually validate regression approaches for Caco-2 permeation (see Results section). We apply two different (Q)SAR approaches to model the numeric endpoint. Instead of adopting the training test split that was used in the original article, we apply a leave-one-out cross-validation procedure to compare support vector regression and simple linear regression. The visual validation workflow is implemented with the CheS-Mapper extension for KNIME. The corresponding workflow is shown in the figure below and can be distinguished into 3 main steps:

READ DATASET The data is loaded into the system and data columns that are not used for modeling are filtered.

PERFORM CROSS-VALIDATION Two identical cross-validations are performed. The compared learning schemes are support vector machines (*SMOreg*) and linear regression, both from KNIME's WEKA extension, using default settings. The node *Numeric Scorer* computes the statistical predictivity of both models.

JOIN DATA FOR CHES-MAPPER Before transferring the data into the visualization node, the modeling results are joined with each other and with the previously removed columns. Also, the prediction errors per compound, as well as the difference between both errors are computed (using the *Math Formula* nodes).



Employing KNIME for cross-validation and visual validation of (Q)SARs. This KNIME workflow employs CheS-Mapper to visually validate two regression approaches (support vector regression and simple linear regression) that are validated with leave-one-out cross-validation.

CV has been removed from the online version.

CV has been removed from the online version.