UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE BIOLOGIA ANIMAL

A Clustering Analysis of the Chemical Metric Space

Tiago Filipe dos Santos Pacheco

Mestrado em Bioinformática e Biologia Computacional
Especialização em Bioinformática

Dissertação orientada por:
Prof. Doutor André Osório e Cruz de Azerêdo Falcão

2018

Resumo

O tempo médio de vida da espécie humana tem vindo a aumentar significativamente, sendo a indústria farmacêutica responsável por parte desse sucesso. O tempo médio de produção de um fármaco situa-se entre os 10 e os 15 anos e o seu custo tem vindo a crescer anualmente. A quiminformática permite a redução destas adversidades, recorrendo a ferramentas informáticas com a capacidade de prever propriedades químicas e biológicas. Uma abordagem utilizada para esta previsão é a dos modelos Relação Estrutura-Atividade Quantitativa, que se baseia na relação entre a semelhança de estrutura de fármacos e o conhecimento das suas atividades. Na verdade, alguns modelos utilizados atualmente utilizam algoritmos de elevada complexidade, incapazes de fazer previsões para grandes quantidades de dados. Neste contexto, na elaboração do presente trabalho, foi desenvolvido um algoritmo de agrupamento que permitisse definir farmacologicamente o espaço molecular. A performance deste algoritmo foi avaliada para um conjunto de dados considerável, provenientes da base de dados ZINC, de modo a verificar diversos aspetos importantes, como por exemplo, se este seria capaz de produzir resultados que permitissem definir o espaço molecular. Com base nos resultados produzidos pelo algoritmo, foram definidos farmacologicamente os agrupamentos gerados, de acordo com regras lógicas, recorrendo a uma base de dados de atividades, nomeadamente o ChEMBL 23. Este processo permitiu a criação de uma base de dados, posteriormente utilizada na construção de uma interface gráfica de busca. Desta forma, para um composto desconhecido, será possível verificar a que agrupamento este se encontra mais próximo, extrapolando a informação de alvos a ele ligado para o novo fármaco.

Palavras Chave: Algoritmo Brotherhood, Interface de busca, Modelo Quantitativo de Relação Estrutura-Atividade, Processo de Agrupamento, Quimioinformática

Abstract

The average life expectancy of the human species has been growing significantly and the pharmaceutical industry is a part of this success. The average time of production of a drug is between 10 and 15 years and the cost of it has been growing annually. Cheminformatics allows the reduction of these adversities, using computer tools capable of predicting chemical and biological properties. An approach used is the Quantitative Structure Activity Relationship models. These, make use of the relationship between the similarity of drug's structure and the knowledge of their activities. In fact, some models currently used, make use of highly complex algorithms, unable to make predictions for large amounts of data. Thus, this work had the purpose to develop a clustering algorithm that allowed to define pharmacologically the molecular space. The algorithm performance was evaluated for a considerable data set, from the ZINC database, in order to verify several important aspects, such as, the ability to produce results that allowed to define the molecular space. Based on the results produced by the algorithm, the clusters generated, according to logical rules, were pharmacologically defined using a database of activities, namely ChEMBL 23. This process allowed the creation of a database, later used in the construction of a search graphical user interface. So, for an unknown compound, it will be possible to verify which is the closest cluster, extrapolating the target information attached to it, to the new drug.

Keywords: Brotherhood Algorithm, Cheminformatics, Clustering Process, Quantitative Structure-Activity Relationship model, Search User Interface

Resumo Alargado

O tempo médio de vida da espécie humana tem vindo a aumentar significativamente nas últimas décadas, sendo que a indústria farmacêutica tem contribuido em grande parte para esse sucesso. Apesar do infindável número de possíveis compostos, desde 1827 até 2013, apenas 1453 foram registados na Food and Drug Administration. O tempo médio para a produção de um fármaco situa-se entre os 10 e os 15 anos e o seu custo médio tem vindo a crescer quase exponencialmente.

A quiminformática permite reduzir o impacto destas adversidades, uma vez que, com recurso a ferramentas e tecnologias informáticas, permite a previsão de propriedades químicas e biológicas. Uma das abordagens mais comum para a previsão in silico é a dos modelos Relação Estrutura-Atividade Quantitativa, que se baseia na correlação entre a semelhança de estrutura entre fármacos e o conhecimento das suas atividades. Deste modo, é possível prever que dois fármacos com uma estrutura semelhante possuam atividades semelhantes.

Assim, um possível algoritmo que poderia obter bons resultados, seria um que apresentasse a capacidade de, para cada molécula, a comparar com todas as moléculas para as quais já se conhece informação acerca das suas atividades, sendo que, no caso de uma semelhança superior a um valor definido, extrapolár-se-ia a informação de atividades para a molécula a comparar. A verdade é que, apesar de este ser um hipotético método com a capacidade de obter bons resultados, não é prático. Quando não possuímos qualquer informação acerca de centenas de milhões de moléculas e temos apenas informação conhecida acerca de um milhão de moléculas, por exemplo, a complexidade associada para uma previsão deste género não é computacionalmente tratável. Imaginemos que temos uma molécula desconhecida e queremos comparar a sua estrutura com a de um milhão de moléculas já estudadas. Isto custar-nos-ia um milhão de comparações "in silico". Se tivermos um milhão de moléculas desconhecidas e o objetivo for comparar a sua estrutura com outro um milhão de moléculas conhecidas, para este caso, seria necessário realizar 1,000,000,000,000 de comparações.

Assim, foi necessário neste trabalho encontrar uma solução com a capacidade de lidar com esta quantidade de dados e ainda assim, obter bons resultados. Neste contexto, foi desenvolvido um novo algoritmo de agrupamento de dados, de base heurística, de modo a definir farmacologicamente as diferentes regiões do espaço molecular. De seguida, foi construída uma base de dados com a capacidade de armazenar esta informação, a qual foi utilizada na construção de uma interface de busca, cujo intuito é o de, para novas moléculas, fazer uma previsão de possíveis alvos.

O algoritmo "Brotherhood" é então um algoritmo de agrupamento de base heurística desenvolvido com o intuito de lidar com conjuntos de dados de grande dimensão. Este requer 3 parâmetros de entrada: um ficheiro, com uma lista de moléculas (uma por linha) com o formato (Identificador da Molécula, Identificador SMILES); um valor limite entre 0.0 e 1.0, que é utilizado no sentido de definir se uma molécula tem ou não uma determinada relação com o agrupamento e, finalmente, um valor limite entre 0.0 e 1.0, que é utilizado no sentido de definir se uma molécula pertence ou não a um agrupamento filho. Para cada uma das moléculas presentes, esta pode: pertencer a um agrupamento se a sua estrutura molecular apresentar semelhança superior ao primeiro valor limite com todas as moléculas desse agrupamento; pertencer a um agrupamento filho, se possuir semelhança estrutural superior ao primeiro limite, com pelo menos uma molécula do agrupamento, e semelhança estrutural superior ao segundo limite com todas as moléculas do agrupamento filho; criar um novo agrupamento filho, caso tenha semelhança estrutural superior ao primeiro limite, com pelo menos uma das moléculas do agrupamento, mas não preencher os requisitos para se juntar a um agrupamento filho já existente; por último, criar um novo agrupamento, caso nenhuma das condições anteriores ocorra. A semelhança estrutural é calculada traduzindo os canonical SMILES em descriptores 2D, como os Extended Conectivity Fingerprint(ECFP) 4 e 6, e posteriormente comparados segundo o coeficiente de Tanimoto, descrito na literatura como o mais utilizado e o que obtém melhores resultados para este tipo de modelos. Por fim, o algoritmo retorna dois ficheiros: o primeiro, com a organização de toda a estrutura de agrupamento realizada, os valores limite utilizados, o número de agrupamentos gerados, o número de agrupamentos filho gerados e ainda o tempo, em segundos, necessário à realização de todo o processo; um segundo ficheiro, com o identificador da molécula e o identificador SMILES da primeira molécula de cada agrupamento gerado, tantos quanto o número de agrupamentos gerados.

De modo a avaliar a performance do algoritmo, foram realizadas três análises distintas, recorrendo sempre a conjuntos de dados provenientes de uma base de dados designada por ZINC. Na primeira análise, o objetivo era avaliar o tempo necessário de execução, variando apenas os dois parâmetros de entrada, valores de limite. Na segunda análise, foi avaliada a relação entre a ordem e as moléculas pertencentes ao conjunto de dados com o número de agrupamentos gerados e o tempo necessário à execução. Por último, na terceira análise, foi efetuada uma avaliação que permitisse determinar a partir de que quantidade de conjunto de dados seria possível gerar uma quantidade de agrupamentos com a capacidade de representar o espaço molecular.

Em relação à primeira análise foram aplicados quatro conjuntos de valor limite (0.5-0.3, 0.3-0.5, 0.3-0.3 e 0.2-0.2) a doze conjuntos de dados com dimensões compreendidas entre 1,000 e 5,000,000. Assim, foi possível verificar que o aumento do primeiro valor limite (0.5-0.3), ao gerar demasiados agrupamentos, mesmo em conjuntos de dados reduzidos, tornava o tempo de execução do algoritmo demasiado elevado. Com a utilização do conjunto (0.3-0.5) verificava-se a mesma situação, sendo que o tempo elevado de execução não resultava de um aumento do número de agrupamentos, mas sim do aumento dos agrupamentos filho. Reduzindo significativamente os dois limites para 0.2-0.2 foi possível reduzir o tempo de execução, contudo, o facto de gerar um número bastante reduzido de agrupamentos e agrupamentos filho fez com que estes fossem maiores, o que levou a um tempo de execução superior quando comparado com o tempo de execução utilizando um conjunto de limites de 0.3-0.3.

Na segunda análise, foram utilizados três conjuntos para conjuntos de dados desde 1,000 a 100,000 sendo que, cada conjunto foi baralhado cinco vezes. Desta forma, foi possível não só avaliar a influência do processo de agrupamento das moléculas pertencentes a cada conjunto mas também a ordem do mesmo. Foi assim possível verificar que apesar de todas as variações anteriormente mencionadas, o tempo de execução e os agrupamentos e agrupamentos filho gerados não variavam significativamente.

Por último, na terceira análise, foram utilizados dois conjuntos de valor limite (0.2-0.2 e 0.3-0.3) aplicados a doze conjuntos de dados com quantidades entre 1,000 e 5,000,000. Para cada um, foi calculada e avaliada a proporção de agrupamentos e agrupamentos filho gerados face ao número de moléculas utilizado para os gerar. Deste modo, foi possível traçar dois gráficos que demonstram que o aparecimento de novos agrupamentos vai diminuindo com o aumento da quantidade de dados, o que permite concluir que o espaço molecular vai sendo progressivamente definido até estabilizar. Por fim, foi realizado o processo de agrupamento com dois milhões de moléculas e com cinco milhões de moléculas. De seguida, para cada um desses processos foi verificado se para um novo conjunto de dois milhões de moléculas estes iriam pertencer a uma já definida região do espaço (agrupamento) ou gerariam um novo agrupamento. Foi possível verificar que, para o 1º agrupamento com 2 milhões, apenas 4822 (0.24%) moléculas de um novo conjunto de 2 milhões não pertenceriam a qualquer agrupamento já definido. Com o 2º agrupamento, com 5 milhões de moléculas, apenas 1531 (0.07%) moléculas de um novo conjunto de 2 milhões não pertenceriam a uma região já existente. Assim, desta forma, foi reforçada a ideia de que o algoritmo "Brotherhood" apresentaria a capacidade de definir mais de 99% do espaço molecular de um conjunto de dados significativamente grande.

Após esta definição, tornou-se essencial atribuir informação a cada grupo molecular, sendo que, foi definido cada agrupamento farmacologicamente, com base na informação presente na base de dados ChEMBL, versão 23. Assim, foram utilizados os resultados, de dois conjuntos de agrupamentos, provenientes de um processo de agrupamento com cinco milhões de moléculas e os conjuntos de parâmetros de valores limite de 0.2-0.2 e 0.3-0.3. Na realidade, como representação de cada agrupamento, foi utilizado o primeiro elemento, designado como o centróide do agrupamento.

A base de dados ChEMBL possui diversa informação relativa à atividade entre compostos e alvos. Contudo, nem toda a informação presente é necessária nem se encontra imediatamente disponível para ser utilizada no contexto deste projeto. Desta forma, foi realizada uma extracção e manipulação da informação, de acordo com algumas regras lógicas definidas, de modo a que fosse possível, para o máximo de atividades composto-alvo, classificá-las segundo três categorias: Activa, Inactiva e Desconhecida. Assim, para cada composto cuja informação se encontrava disponível, foi realizada a sua ligação a todos os centróides próximos, nomeadamente todos aqueles cuja semelhança estrutural era superior a 0.2 (nos centroides provenientes do processo de agrupamento com conjunto 0.2-0.2) e superior a 0.3 (nos centroides provenientes do processo de agrupamento com conjunto 0.3-0.3). Como resultado desta ação, foi construída uma base de dados em que cada composto-atividade-alvo se encontrava, paralelamente, ligado a dois conjuntos distintos de centróides.

Por último, foi construída uma interface gráfica de busca, cujo objetivo é o de, para um composto desconhecido, verificar a que centróide este se encontra mais próximo, extrapolando a informação de alvos a ele ligado, para a nova e desconhecida molécula.

Com o término da construção da interface, é possível afirmar que os principais objetivos da tese foram alcançados com sucesso,existindo agora uma nova alternativa, de modo a prever possíveis alvos para novos compostos.

Face ao que foi desenvolvido neste projeto, é proposto para um futuro trabalho, a validação da interface, recorrendo a novas moléculas cujos alvos sejam conhecidos e não se encontrem presentes na base de dados. Desta forma, poderá ser interessante uma atualização contínua à base de dados de suporte à interface, efetuando uma análise mais exploratória aos dados nela contida.

Acknowledgements

First of all, I want to thank my grandfather, who passed away while I was performing this work. It's thanks to him that I am the person I am and it was also for him that I promised to conclude this project successfully. Thank you, Grandfather, I know you would be very proud.

Secondly, I want to thank my advisor, Prof. Dr André Falcão, for the dedication, patience and wisdom transposed over hours, often weekly. Even in the busiest times, he always found time to help me.

I also want to thank my girlfriend, Inês, who never let me give up and always had a friendly word in the more difficult moments.

To my grandmother, mother, sister and the rest of the family who have given me the necessary support, not only during the Master's, but in life, that allowed me to achieve my goals successfully.

To my friends, Madalena Pavão and Sofia Pires, who lived almost daily with me. Not only were they able to help me in implementing this project but they were also able to maintain a healthy spirit throughout the process.

Last but not least, I want to thank to FCiências.ID – Associação para a Investigação e o Desenvolvimento de Ciências (projeto MIMED projeto PTDC/EEI-ESS/4923/2014) for funding my master thesis, without it my research wouldn't be possible.

Contents

List of Figures

List of Tables

List of Abbreviations

- AC50 - Activity Concentration 50%
- CSS - Cascading Style Sheets
- DBSCAN - Density-Based Spatial Clustering of Applications with Noise
- EC50 -Efficacy Concentration 50%
- ECFP - Extended-Connectivity Fingerprints
- ED50 - Effective Dose 50%
- FDA - Food and Drug Administration
- GI50 - Growth Inhibition 50%
- HTML - HyperText Markup Language
- IC50 - Inhibitory concentration 50%
- IUPAC - International Union of Pure and Applied Chemistry
- InChI - International Chemical Identifier
- KDD - Knowledge Discovery from Data
- Ki - Inhibitory constant
- MIC - Minimum Inhibitory Concentration
- MTV - Model-Template-View
- MySQL - My Structured Query Language
- PHP - Hypertext Preprocessor
- QSAR - Quantitative Structure-Activity Relationship
- RDKit - Rational Discovery Kit
- SMARTS - SMiles ARbitrary Target Specification
- SMILES - Simplified Molecular Input Line Entry System
- ZINC - Zinc Is Not Commercial

Chapter 1

Introduction

1.1 Motivation

According to the United Nations, between 1900 and 2000, the human population has grown from 2 to 7 billions, and the projections show that it can reach 11 billions by the end of the 21th century. The advances of life sciences, such as medicine, chemistry, biology and informatics took a major role in this exponential growth of population.(United, 2017) In fact, it's the relation of all those fields that contributed to new drug development techniques that are directly related to the increase of average life expectancy. As stated by Michael, Food and Drug Administration (FDA) approved a total of 1453 compounds between 1827 and 2013, being more than 800 in the last 35 years. (Kinch *et al.*, 2014; Pharmaceutical Research and Manufacturers of America, 2016) However, this is a small number, since there are an infinity of possible compounds. Recent numbers show that the average time to go through all the process of drug development is 10-15 years and the average cost is \$2.6 billion, since less than 12 % enter clinical trials and even less are approved.(Pharmaceutical Research and Manufacturers of America, 2016) The cost of of drug development is, at the moment, higher than ever, and has been increasing year after year.(figure 1.1)

The process of drug development, can be divided into four main steps (figure 1.2): Discovery and development; Pre-clinical research, Clinical research and Drug Review. (FDA, 2018)

Discovery and development - At this stage, researchers discover new drugs (10,000-15,000) through new insights about a disease process, applying many molecular tests to the compounds or using new technologies to find possible beneficial effects

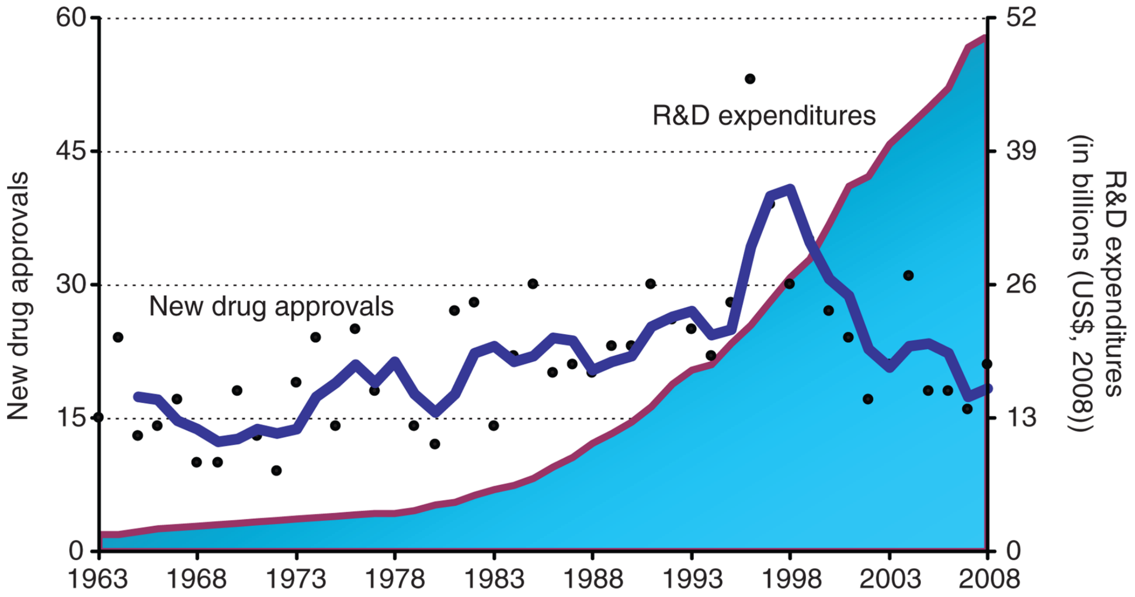


Figure 1.1: New drug approvals (dots), represented on the left vertical axis, and pharmaceutical R&D expenditures (shaded area), represented on the right vertical axis, in the United States from 1963 to 2008. R&D expenditures are presented in terms of constant 2008 dollar value. The trend line is a 3-year moving average. (Kaitin, 2010)

against a large number of diseases.There are even treatments that show unanticipated effects, that is, the researchers are looking for a molecule that would act in a specific target, and it ends to be tested to a completely different one. After discovering promising compounds, they conduct experiments to gather information like best dosage, mechanisms of action and side effects. At the end of this stage, only an average of 250 compounds goes to the next phase.

Pre-clinical research - Before testing a drug on humans, it is necessary to know that the compound doesn't have the potential to cause serious harm (toxicity). In order to understand that, there are several in vivo and in vitro experiments to perform. On average, only 5 of the 250 compounds are approved to clinical trials.

Clinical research - After pre-clinical research shows that compounds aren't toxic to humans, it's time for the clinical trials. Those have to be planned according to specific rules and protocols. Three different phases are defined: phase I (20-100 volunteers), phase II (100-500 volunteers) and phase III (1000-5000 volunteers). Usually, only one of the five compounds goes through all three phases and achieves the market approval. It's possible that all 5 fail, and pharmaceuticals have to return to drug and discovery stage.

Drug Review - When a compound has proved to be safe for humans and effective for it's intended use, the pharmaceutical company can file the application to market the drug. If the drug passes all the rigorous controls, it can then be sold in the market.

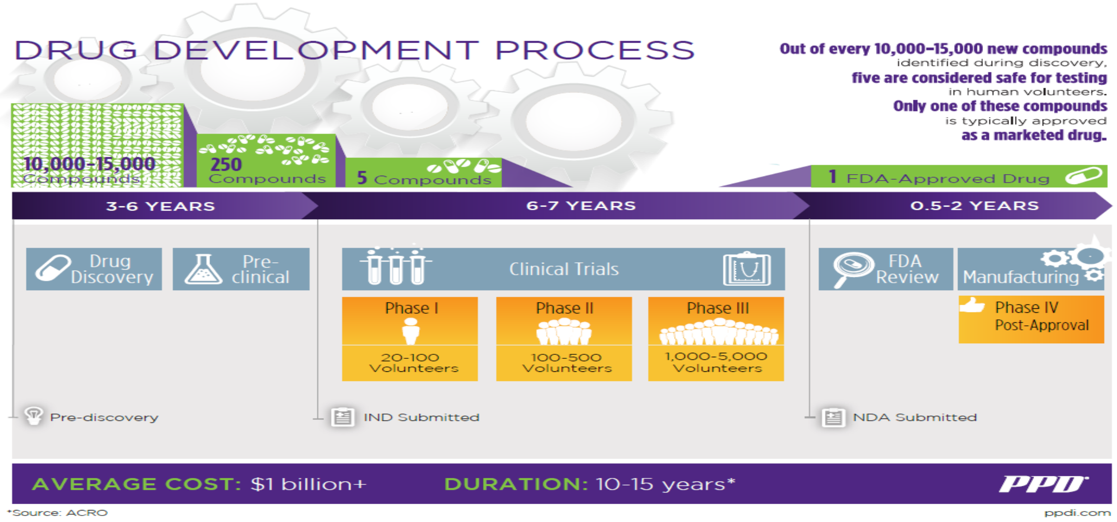


Figure 1.2: Drug Development Process (Pharmaceutical, 2018)

.

As seen before, the drug development process has an huge cost and its infeasible to know all the biological and chemical properties of all of the 10,000 compounds at start. Thereby, it's necessary to have tools and technologies that predict with precision those properties. Cheminformatics is the area responsible to predict those properties. In 1998, Dr. Brown gave his definition of cheminformatics as: "Cheminformatics is the mixing of information resources to transform data into information and information into knowledge for the intended purpose of making better decisions faster in the area of drug lead identification and organization."(Chen, 2006) Nowadays, cheminformatics extends the drug development process.

1.2 Objectives

One of the most common approaches for in silico activities prediction is the Quantitative Structure-Activity Relationship (QSAR) models.(Nantasenamat *et al.*, 2010) The QSAR models are designed to correlate structure similarities of drugs with activities knowledge.

It means that, whenever we have activity information about a compound, we can predict for a new compound similar activities if both share structure properties.(Vilar & Costanzi, 2012) However, the use of these models are not that simple. An algorithm that could predict with good results could be: for our molecule, m, the algorithm would compare it's structure to all the molecule structures with known information, using a threshold value to split the similar from dissimilar molecules, using the similar ones to predict results for our m molecule. An algorithm like this, however, would take an exponential amount of time to solve. Let's imagine this scenario: if we have 1 molecule and we would like to compare it's structure with 1 million of compounds that we already have information, it would "cost us" 1 million of in silico comparisons. Imagine that we have 1 million of unknown compounds and we want to predict information about them. Now, our machine would have to make 1,000,000,000,000 comparisons. This problem is impracticable in real time, so it means that it is necessary to rearrange a solution for our prediction models. The aim of this project is to implement a system of hierarchical grouping of molecules using a new clustering based algorithm, in order to define pharmacological regions inside the molecular space. Subsequently, it was made a database and a web application that makes use of algorithm results to predict targets for all type of unknown compounds.

1.3 Schedule of work

In the table 1.1 is possible to see all the activities, realized during this research, carried out during the presented time line.

Activities	**Year**													
	2017			**2018**										
	Month													
	10	**11**	**12**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
Familiarization with the modeling techniques of chemical similarity	X	X												
Familiarization with the test data and definition of subsets	X	X												
Testing of clustering strategies		X	X											
Implementation of the chosen clustering strategy			X	X	X									
Characterization of the various clusters pharmacologically, according to the literature information and annotated databases					X	X	X							
Design and construction of the database							X	X						
Implementation of the search user interface								X	X	X	X	X		
Interface test and validation												X	X	
Bibliographic research	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Report writing											X	X	X	X

Table 1.1: Schedule of work

1.4 Overview

This document is organized and divided into 7 chapters.

- (Previously mentioned) **Chapter 1 - Introduction**.
- **Chapter 2 - Background**. Along the background it will be explained important concepts and definitions regarding the project theme.
- **Chapter 3 - Methods and Data**. In this chapter, it's described clustering methods and the data used.
- **Chapter 4 - Clustering Analysis**. Here, all the results obtained through the application of the developed Clustering method to all the datasets, in all lines of action are presented, analyzed and discussed.
- **Chapter 5 - Defining Clusters Pharmacologically**. Within this chapter, relevant results are used in order to define clusters pharmacologically by creating a database.
- **Chapter 6 - Search User Interface**. Using the database created, a search user interface is created and described in more detail.
- **Chapter 7 - Conclusions**. Finally, this chapter makes a final conclusion about all the project.

Chapter 2

Background

2.1 Molecular Representations

2.1.1 Identifiers

Since the 60s that computers have been used to store and manipulate chemical structures.(Warr, 2011) Some of the applications have already been addressed in this document, like similarity searching and processes in drug discovery. The need of a machine-readable representation of chemical structure is/was a need to complete those tasks successfully.(Warr, 2011)

Line notations are than linear representations of chemical structures of a molecule as a line of a text. Some of this characterizations may have some advantages, such as: being human-readable and human-writable; easily entered into a software and canonical representations (unique representation of a molecule).(Boyle, 2012) Two of the most widely used nowadays, are Simplified Molecular Input Line Entry System (SMILES) and IUPAC International Chemical Identifier (InChI and InChIKey).

2.1.1.1 Simplified Molecular Input Line Entry System (SMILES)

SMILES is a chemical notation language developed in the end of 1980 at Pomono College and later implemented by Daylight Chemical Information Systems. The algorithm responsible for generating the SMILES notation have specific and simple rules that allow the final result to be easy to understand by humans.(Weininger, 1988)

Rules:

1) Atoms. Atoms are represented by their periodic table symbol inside of square brackets.

The brackets aren't needed if elements are part of the "organic subset" (B, C, N, O, P, S, F, Cl, Br and I). Whenever represented without brackets, the elements must have the following premise: the number of attached hydrogens conforms to the lowest normal valence consistent with explicit bonds;

2) Bonds. Single, double, triple and aromatic bonds are represented by the symbols -, =, #, and :, respectively. Single and aromatic bonds may be, and usually are, omitted. E.g.: CC, C=C, C#N

3) Branches. Branches are represented by the inclusion of the atom in parentheses and can be nested or stacked. E.g.: CCN(CC)CC

4) Cyclic Structures. Cyclic structures are represented by breaking one single (or aromatic) bond in each ring. The ring-opening and ring-closure bonds are followed by a digit. E.g.: C1CCCCC1

5) Disconnected Structures. Disconnected structures are represented by a dot ('.') separating them. E.g.: [Na+].[O-]c1ccccc1

6) Aromaticity. Aromatic structures are represented by writing the atoms in the aromatic ring in lower case letters. E.g.: c1ccccc1C(=O)O

There is no perfection in anything and the SMILES approach is no exception. One of the drawbacks of this format is the fact that each molecule representation isn't canonical. However, there are many algorithms that make use of SMILES and turn it into a canonical form.(Boyle, 2012)

2.1.1.2 InChi and InChiKey

The IUPAC International Chemical Identifier (InChI) is a machine-readable string of symbols that unequivocally represent in a computer a compound.(Heller *et al.*, 2013) One of the greatest advantages is the fact that it is an open source and non-proprietary system. (Heller *et al.*, 2015; Warr, 2015) The InChI system makes use of his layered format in order to represent the compound information, where each layer contains a specific type of information.(Heller *et al.*, 2013, 2015; Warr, 2015)

The layers are characterized as:

1. Formula

2. Connectivity (no formal bond orders)

   (a) Disconnected metals

   (b) Connected metals

3. Isotopes

4. Stereochemistry

   (a) Double bond

   (b) Tetrahedral

5. Tautomers (on or off)

Each layer in the InChI string is separated by the slash (/), followed by a lower-case letter (except the first layer).

A structure with 100+ atoms gives a very long string, which is an identified problem when using a search engine such as Google or Yahoo. (Heller *et al.*, 2013) The InChIKey was the answer for that problem. InChIKey is a shorted hash-based InChI derivative, with 27-characters and based on a SHA-256 cryptographic hash function.(Heller *et al.*, 2013; Warr, 2015) A small possibility of finding two structures with the same InChIKey is possible due to hash code collisions, however, since 2007, only two of these cases have been reported.(Warr, 2015)

2.1.2 Descriptors

In the last topic, it was described how to identify a molecule computationally. However, in order to compare between structures, it is necessary to have a comparable definition for the molecule structures. Descriptors are terms that characterize specific information about an active compound.(Khan, 2016; Roy *et al.*, 2015) The information encoded by descriptors generally depends on the kind of molecular representation and the defined algorithm for its calculations. There are several types of characterizations that describe the compounds in a different way, used in QSAR models, such as Geometrical, Thermodynamic, Electronic, Constitutional and Topological descriptors.(Khan, 2016) (Figure 2.1)

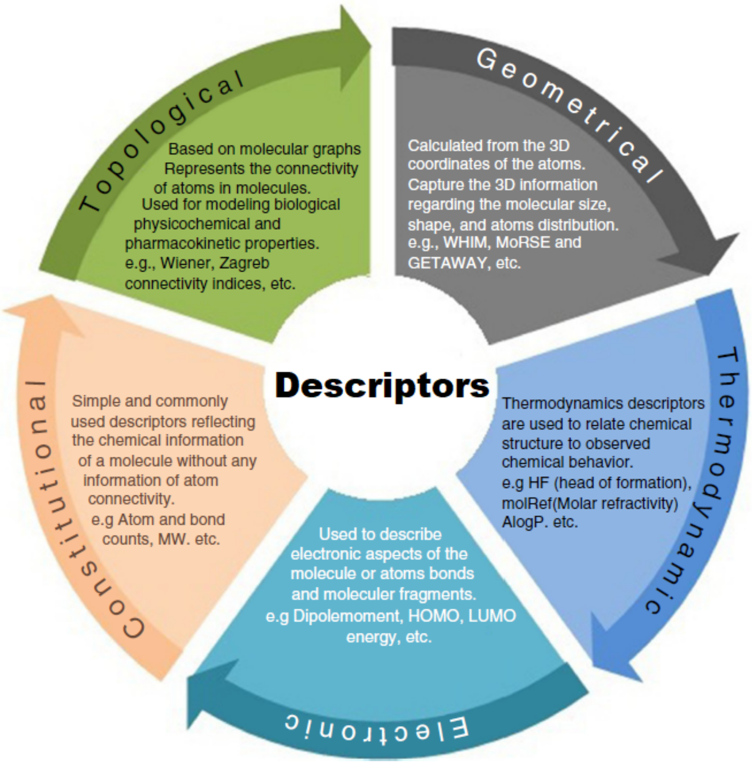

Descriptors
Topological
Based on molecular graphs Represents the connectivity of atoms in molecules. Used for modeling biological physicochemical and pharmacokinetic properties. e.g., Wiener, Zagreb connectivity indices, etc.
Geometrical
Calculated from the 3D coordinates of the atoms. Capture the 3D information regarding the molecular size, shape, and atoms distribution. e.g., WHIM, MoRSE and GETAWAY, etc.
Thermodynamic
Thermodynamics descriptors are used to relate chemical structure to observed chemical behavior. e.g HF (head of formation), molRef(Molar refractivity) AlogP. etc.
Electronic
Used to describe electronic aspects of the molecule or atoms bonds and moleculer fragments. e.g Dipolemoment, HOMO, LUMO energy, etc.
Constitutional
Simple and commonly used descriptors reflecting the chemical information of a molecule without any information of atom connectivity. e.g Atom and bond counts, MW. etc.

Figure 2.1: Representation of Molecular Descriptors Used in Quantitative Structure–Activity Relation (QSAR) Modeling.(Khan, 2016)

2.1.2.1 Extended-Connectivity Fingerprints (ECFPs)

Extended-connectivity fingerprints (ECFPs) are a novel class of topological fingerprint, formulated in graph theoretic approach, for molecule characterization explicitly designed to capture molecular features relevant to molecular activity. (Rogers & Hahn, 2010; Roy, 2004) ECFPs are suited to tasks related to predicting and gaining insight into drug activity and in methods such as similarity searching, clustering and virtual screening. (Hu *et al.*, 2009; Rogers & Hahn, 2010) Like other fingerprints, ECFPs are encoded as a binary bit vector string. The presence of a specific substructure is represented as the bit 1 and the absence as 0. (Gortari *et al.*, 2017) (Figure 2.2)

In fact, they have a more complex generation process since they use the relative position of each atom. ECFP generation process has three sequential stages: (Rogers & Hahn, 2010)

1. An initial assignment stage, where each atom has an integer assigned to it.
2. An iterative updating stage, where each atom integer is updated to reflect the integers assigned to each other atoms.

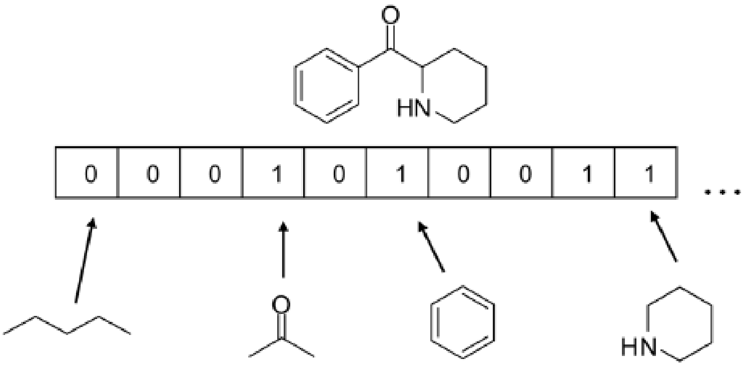


Figure 2.2:
Representation of a chemical structure as a binary vector (Gortari *et al.*, 2017)

3. Finally, a duplicate identifier removal stage, where multiple occurrences of the same feature are reduced to a single representative feature.

There are different types of ECFP fingerprint according to different diameters, such as ECFP_0, ECFP_2, ECFP_4 and ECFP_6. The difference between all of them is the diameter of circular atoms neighbors considered for each atom. Sometimes, it's possible that different approaches generate the same fingerprint, for example, if the molecule is too small and the same diameter covers all the bonds.(Rogers & Hahn, 2010) The most common used are ECFP_4 and ECFP_6 since they generally have the best performance.(Skinnider *et al.*, 2017)

2.1.3 Similarity Measures

Similarity measures or distance metrics are a need to compare fingerprints in order to quantify the similarity between two chemical structures.(Skinnider *et al.*, 2017) Chemical similarity measure can be described has three components: (Chen & Reynolds, 2002; Todeschini *et al.*, 2012)

- Structural representation, used to characterize the structures to be compared;
- Weighting schemes, to assign different importances to each features/substructures;
- Similarity coefficient, that provides the mathematical function for calculating a similarity value based on (possible weighted) values of structural descriptors.

Before present similarity measures, it is necessary to define what similarity between two compounds means. Thus, molecules as the ones we've seen can be described as binary vectors. Let's see an example of two molecules as binary vectors, x and y, each with p substructures with values being 0 or 1. Since each feature can be 0 or 1, and we have 2 vectors, we can have a maximum of 4 combinations. Those four combinations can be seen in the contingency table 2.1. (Todeschini *et al.*, 2012)

	y = 1	**y = 0**	
x = 1	a	b	a + b
x = 0	c	d	c + d
	a + c	b + d	p

Table 2.1: Frequency Table of the Four Possible Combinations for Two Binary Vectors

The contingency table can be read as:

- **a** (x=1 and y=1) is the number of features which x and y share
- **b** (x=1 and y=0) is the number of features which x has and y lacks
- **c** (x=0 and y=1) is the number of features which x lacks and y has
- **d** (x=0 and y=0) is the number of features which x and y both lacks
- **a** + **b** is the number of presence of substructures in x
- **a** + **c** is the number of presence of substructures in y
- **a** + **d** represents the similarity between the x and y vectors
- **b** + **c** represents the dissimilarity between the x and y vectors
- **p** is the total number of variables, (a+b+c+d), which is the length of each binary vector

There are several similarity measures that make use of binary fingerprints similarity and dissimilarity terms. Some of the most commonly applied are present on table 2.2.(Holliday *et al.*, 2002; Todeschini *et al.*, 2012)

No.	Name	Formula
1.	Jaccard/Tanimoto	$\frac{a}{a+b+c}$
2.	Dice	$\frac{2a}{2a+b+c}$
3.	Russell/Rao	$\frac{a}{p}$
4.	Sokal/Sneath	$\frac{a}{a+2b+2c}$
5.	Kulczynski	$\frac{a}{b+c}$
6.	Simple Matching	$\frac{a+d}{p}$
7.	Hamann	$\frac{a+d-b-c}{p}$
8.	Rogers/Tanimoto	$\frac{a+d}{b+c+p}$
9.	Baroni-Urbani/Buser	$\frac{\sqrt{ad}+a}{\sqrt{ad}+a+b+c}$
10.	Ochiai/Cosine	$\frac{a}{\sqrt{(a+b)(a+c)}}$
11.	Forbes	$\frac{pa}{(a+b)(a+c)}$
12.	Fossum	$\frac{n(a-\frac{1}{2})^2}{(a+b)(a+c)}$
13.	Simpson	$\frac{a}{min(a+b,a+c)}$
14.	Pearson	$\frac{ad-bc}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$
15.	Yule	$\frac{ad-bc}{ad+bc}$

Table 2.2: Most Used Similarity Measures

According to several studies, Tanimoto is the most appropriate similarity measure in the search of similarity patterns. (Bajusz *et al.*, 2015; Chen & Reynolds, 2002; Todeschini *et al.*, 2012)

2.1.4 Quantitative Structure-Activity Relationship (QSAR)

As seen in the last topic, it is possible to compare two distinct structures and retain a value representative of the similarity. However, this alone doesn't give us any information. Quantitative Structure-Activity Relationship (QSAR) models were first described by Corwin Hansch in the 60s and, nowadays, make use of the principle that similar structures may share the same biological activities and physiochemical properties.(Chen & Reynolds, 2002; Cherkasov *et al.*, 2015) Statistical and machine learning models such as Clustering are between the most common approaches to automatize predictions for large databases instead of ultrahigh-throughput screening of large databases.(Kausar & Falcao, 2018; Polishchuk, 2017)

2.1.5 Drug Activity Measures

The existence of prior laboratory investigation about compound-target biological activities is the reason that makes possible to use QSAR models to predict activities for unknown molecules, since these make use of known information to make those predictions. When defining targets/receptors to a drug by laboratory experiments, it would be desirable that a drug would act only on the receptor or biological site of interest, at all concentrations, and wouldn't interact with others at any achievable concentration. Unfortunately, no drug have this ideal property.(Neubig *et al.*, 2003) To quantify the action of each drug, for all type of experiments, at different concentrations, for different targets, the use of experimental measures is mandatory. The table 2.3 shows some of the most common experimental measures of drug action and their descriptions.

2.2 Cheminformatics Databases

Chemical information (such as the properties of a drug, the relationship between different compounds or the drug-target relationship) increases exponentially everyday. Safe storage, the possibility of manipulation with different tools and access everywhere are some of the essential requirements these days. Thus, there are several chemical databases whose stored information may be different according to the purpose for which the project intends to respond. Some of the most widely used databases worldwide are, for example: DrugBank, PubChem, ChEMBL and ZINC.

- **DrugBank**.(Wishart *et al.*, 2018) "DrugBank is a comprehensive, freely available web resource containing detailed drug, drug target, drug action and drug interaction information about FDA-approved drugs as well as experimental drugs going through the FDA approval process". Contains 2,358 drugs approved by FDA and others, 4,501 compounds from experimental drugs in phases I/II/III and more then 365,000 drug-drug interactions.

Measure	Description
Ki	Inhibitory constant. Concentration needed of inhibitor to reduce an activity between ligand-receptor. (Mohan *et al.*, 2013; Waley, 1982)
IC50	Inhibitory concentration 50%. Concentration needed of inhibitor to reduce an activity by 50% between ligand-receptor. (Mohan *et al.*, 2013; Neubig *et al.*, 2003)
MIC	Minimum Inhibitory Concentration (MIC). Lowest concentration of an anti-microbial that will inhibit the visible growth of a microorganism after overnight incubation. (Andrews, 2001)
Inhibition	Concentration needed of inhibitor to reduce an activity between ligand-receptor. (Waley, 1982)
Potency	Concentration/amount needed to produce an effect with a determined magnitude. (Neubig *et al.*, 2003)
Activity	Concentration needed to produce an activity.(Shockley, 2016)
EC50	Efficacy Concentration 50%. Concentration needed to produce 50% of the maximal possible effect.(Mohan *et al.*, 2013; Neubig *et al.*, 2003)
GI50	Growth Inhibition 50%. Concentration of drug needed to inhibit the growth by 50%.(Marx *et al.*, 2003)
ED50	Effective Dose 50%. Dose needed to produce 50% of the maximal response to that drug.(Mohan *et al.*, 2013; Neubig *et al.*, 2003)
AC50	Activity Concentration 50%. Concentration needed to produce 50% of maximal activity.(Shockley, 2016)

Table 2.3: Experimental Measures of Drug Action and their Descriptions

- **PubChem.**(Kim *et al.*, 2016) PubChem is a public repository for information on chemical substances and their biological activities. Launched in 2004, has rapidly grown to a key chemical information resource that serves scientific communities in many areas such as cheminformatics, chemical biology, medicinal chemistry and drug discovery. In 2015, PubChem had more than 157 million provided chemical substances descriptions, 60 million unique chemical structures and 1 million biological assay descriptions. The database data is provided by more than 350 contributors, such as universities, government agencies, pharmaceutical companies, chemical vendors, publishers and some other chemical biology resources. The data exchange between other chemical databases is very common.

- **ChEMBL.**(Gaulton *et al.*, 2014, 2017) "ChEMBL is an open large-scale bioactivity database containing information largely manually extracted from the medicinal chemistry literature. Information regarding the compounds tested (including their structures), the biological or physicochemical assays performed on these and the targets of these assays are recorded in a structured form, allowing users to address a broad range of drug discovery questions." In 2017, the database contained information extracted from more than 65,000 publications, 1.6 million distinct compounds, 14 million activity values from 1.2 million assays. These assays are mapped to approximately 11,000 targets, including 9,052 proteins (which 4,255 are human). Data can be used in different applications, like identification of suitable chemical tools for a target and large scale data mining, such as the construction of predictive models for targets.

- **ZINC.**(Sterling & Irwin, 2015) ZINC (ZINC Is Not Commercial) is a public access database and a tool set, developed to enable ready access to compounds for virtual screening, ligand discovery, benchmarking and force field development. Initially developed as an exclusive compounds database, it has been updated more recently to ZINC15 version, that is designed to bring together biology and cheminformatics, with a tool that makes it easier to use for non experts, remaining full programmable for cheminformaticians and computational biologists.

Chapter 3

Methods and Data

3.1 Data Mining

Data Mining is a popularly used term as a synonym of Knowledge Discovery from Data (KDD). This process can be described in the following 7 steps: **1. Data Cleaning** (remove of inconsistent data); **2. Data Integration** (combination of multiple data sources); **3. Data Selection** (retrieved relevant data from database); **4. Data Transformation** (transformation of data to appropriate mining form); **5. Data Mining** (application of methods to extract data patterns); **6. Pattern Evaluation** (identification of interesting patterns representing knowledge); **7. Knowledge Presentation** (visualization and knowledge representation to users).(Han *et al.*, 2012)

3.1.1 Cluster Analysis

Cluster analysis consists in the process of partitioning a dataset into subdatasets. Each subset is defined as a cluster, where objects in a cluster are similar to one another, yet dissimilar to objects in other clusters. The set of clusters generated from a cluster analysis is commonly referred as clustering. Thus, different clustering methods may generate different clustering for the same dataset. The discovery of previously unknown groups in the data is one of the most useful resources of clustering.(Han *et al.*, 2012)

3.2 Clustering Algorithms

There are several clustering algorithms, classified into Hierarchical, if in each iteration a parent-child relationship is being established between clusters or nonhierarchical, if the results produced are a single partition of the dataset. (Lipkowitz & Boyd, 2002)

3.2.1 Hierarchical Algorithms

In the Hierarchical algorithms, there are divisive and agglomerative branches. Agglomerative algorithms have a bottom-up strategy, starting with each object forming its own cluster and iteratively merging them into large clusters until a final merge into a single cluster. Divisive make use of a top-down strategy, starting by a single cluster with all objects, splitting them into smaller clusters.(Han *et al.*, 2012; Lipkowitz & Boyd, 2002)

3.2.1.1 Hierarchical Agglomerative Algorithms

The most commonly hierarchical clustering algorithms methods are implemented using what is called the stored-matrix algorithm, since the starting point of all the algorithms is a matrix of all pairwise proximities between all the objects to be clustered. After that, the algorithm scans the matrix to find the most similar pair of clusters and merge them into a new cluster. The matrix is updated and it's scanned over and over until just one cluster remains. For a N dataset the stored-matrix algorithm requires O(n^2) time and space for creation and O(n^3) for clustering.(Lipkowitz & Boyd, 2002)

3.2.1.2 Hierarchical Divisive Algorithms

One of the most used Divisive algorithms is Divisive Analysis (DIANA). All the n objects start by belonging to an initial cluster. The cluster is split according to a principle. The clustering process ends only when the principle can no longer divide more. In this algorithm, when the n is large it is computationally prohibitive to examine all possibilities.(Han *et al.*, 2012)

3.2.2 Nonhierarchical Algorithms

Nonhierarchical algorithms make use of different techniques to build clusters. For example, a single-pass method (used in Leader Algorithm) where the partition is achieved through a single pass through the dataset; a relocation method, where objects are moved from one cluster to another to improve the initial estimation of clusters (used in K-means); and even those who make use of density-based methods (used in Density-Based Spatial Clustering of Applications with Noise - DBSCAN), regard the distribution of descriptors across the dataset as generating patterns of high and low density that, when identified, can be used to separate the compounds into clusters. (Han *et al.*, 2012; Lipkowitz & Boyd, 2002)

3.2.2.1 Leader Algorithm

The Leader algorithm starts by setting the number of clusters to zero. Then, uses the first object in the data set to start the first cluster. To all the next objects, it calculates the similarity between them and, iteratively, all the first elements (Leader) of each cluster. If its similarity exceeds some threshold, the object belongs to a cluster; otherwise it tries the next cluster or generates a new one. This method is simple to implement and fast, however it is order dependent.(Lipkowitz & Boyd, 2002)

3.2.2.2 K-means

K-means is a centroid-based partiotining technique. It is necessary to give at start, a dataset with n objects and a k number of clusters to partitionate the dataset. At start, it chooses arbitrarily k objects from the dataset as the initial cluster centroids, then for each of the remaining objects it is made the assignment to the most similar centroid. The mean value of each cluster is calculated, becoming the mean object the new centroid. This process of assignment/new calculated centroid is repeated until no changes. However, this process has two main disadvantages since it is necessary, at start, to mention the number of centroids desired and it is too time consuming in large datasets, with a complexity of $\mathrm{O(n^{dk+1}logn)}$, being d dimensions. (Han *et al.*, 2012)

3.2.2.3 Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

DBSCAN is an algorithm known for finding clusters with arbitrary shape as the "S" shape/oval clusters and can detect noise/outliers in the data. The main strategy makes use of dense regions in the data space, separated by sparse regions. The algorithm requires a dataset; a user-specified-parameter e>0 used to specify the radius of a neighborhood considered for every object; and MinPts>0 that allows a object to be considered a core object of the cluster if it has at least MinPts objects at e radius. So, an object can be a core member of a cluster if it has at least MinPts at e radius of distance. A border member of cluster if it's at e radius distance of a core object or noise if none of the previously premises happen. (Han *et al.*, 2012; Lipkowitz & Boyd, 2002)

The complexity of the algorithm is O(nlogn) if a spatial index is used, and O(n^2) otherwise. This algorithm however is too sensitive to the setting of parameters.

3.2.3 Brotherhood Algorithm

To define different pharmacological regions in the chemical space it is necessary to have large datasets with a big variety of information. The previously mentioned algorithms and even variants of them described in the literature are some of the most used algorithms in clustering cheminformatics data. However, most of them are unable to treat large datasets and are too sensitive to user-specified parameters.(Ahmad & Dang, 2015)

Brotherhood algorithm is an heuristic clustering algorithm, based on Leader algorithm with a single-pass method, designed with the purpose of handling large datasets. By using two related layers of clusters (Clusters and Son-Clusters) it allows to reduce the number of clusters without creating large partitions that would compromise the clustering in large datasets. Another difference comparing to Leader is the fact that for a molecule to belong to a specific cluster it is not only necessary to have a threshold higher than the specified with the first molecule but with all the molecules of that cluster.

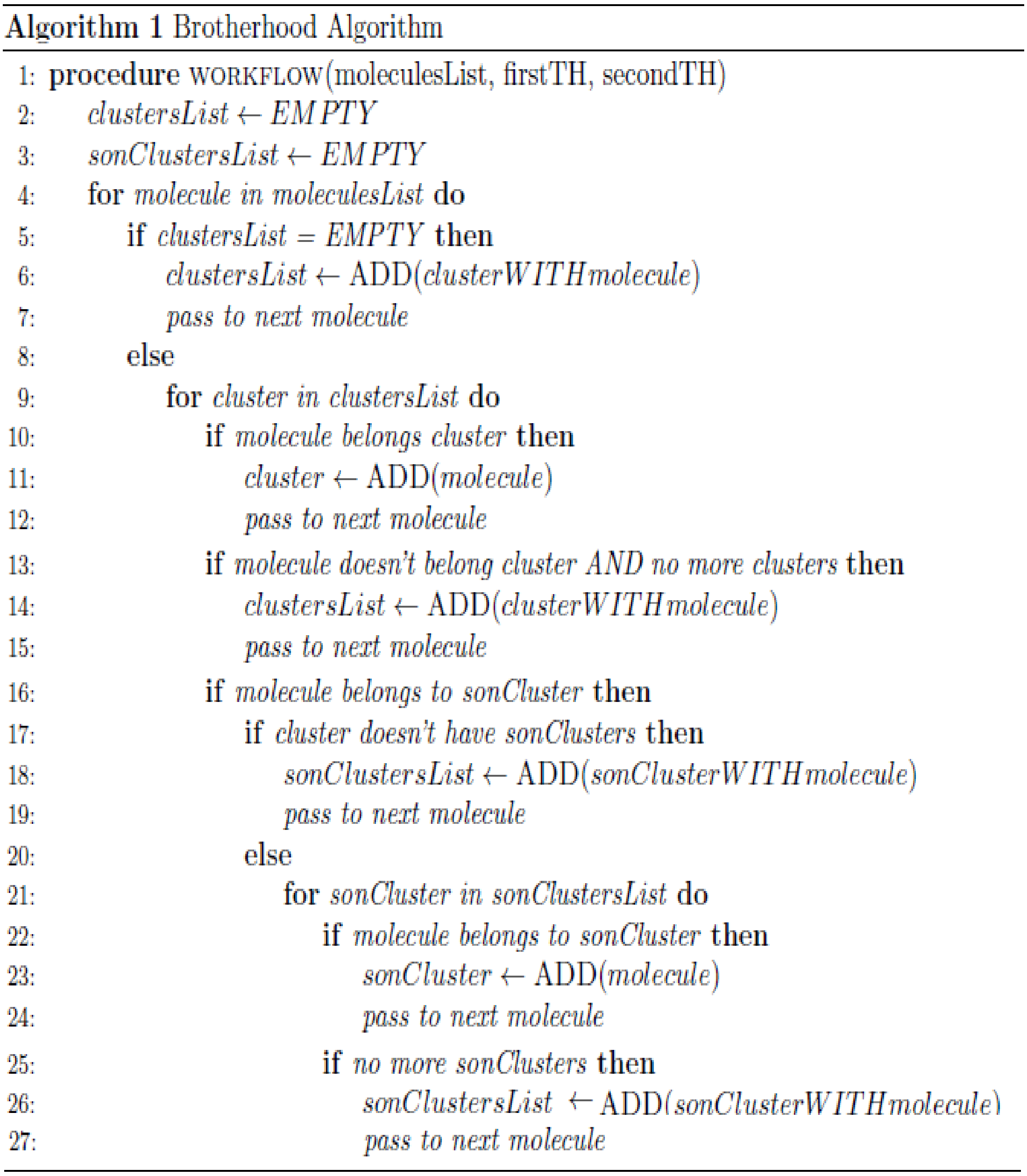
The algorithm requires three entry parameters: a **dataset moleculesList**, as a list of molecules (one per line) with the format (molecule ID, SMILES identifier); a **first threshold**, between 0.0 and 1.0, as a cut-off value for a molecule belong or not to a specific cluster, and finally a **second threshold**, between 0.0 and 1.0, as a cut-off value for a molecule belong or not to a specific Son-Cluster. The following premises are equally valid:

- **For a molecule to belong to a cluster**. Necessary that, the result of Tanimoto[1] similarity measure between that molecule and all those molecules belonging to the cluster, be always greater than the first threshold.

- **For a molecule to belong to a son-cluster**. Necessary that the result of Tanimoto similarity measure between that molecule and at least one of the molecules belonging to a cluster(father) be greater than the first threshold and Tanimoto similarity measure between that molecule and all those belonging to the son-cluster, be always greater than the second threshold.

[1] According to literature, most used in QSAR studies and with the best results

The workflow is simplified in the pseudo-code of the algorithm:



Algorithm 1 Brotherhood Algorithm

```
procedure WORKFLOW(moleculesList, firstTH, secondTH)
    clustersList ← EMPTY
    sonClustersList ← EMPTY
    for molecule in moleculesList do
        if clustersList = EMPTY then
            clustersList ← ADD(clusterWITHmolecule)
            pass to next molecule
        else
            for cluster in clustersList do
                if molecule belongs cluster then
                    cluster ← ADD(molecule)
                    pass to next molecule
                if molecule doesn't belong cluster AND no more clusters then
                    clustersList ← ADD(clusterWITHmolecule)
                    pass to next molecule
                if molecule belongs to sonCluster then
                    if cluster doesn't have sonClusters then
                        sonClustersList ← ADD(sonClusterWITHmolecule)
                        pass to next molecule
                    else
                        for sonCluster in sonClustersList do
                            if molecule belongs to sonCluster then
                                sonCluster ← ADD(molecule)
                                pass to next molecule
                            if no more sonClusters then
                                sonClustersList ← ADD(sonClusterWITHmolecule)
                                pass to next molecule
```

Figure 3.1: Pseudocode of Brotherhood algorithm

.

After executing the algorithm, with the three parameters required, the expected results are two .txt files.

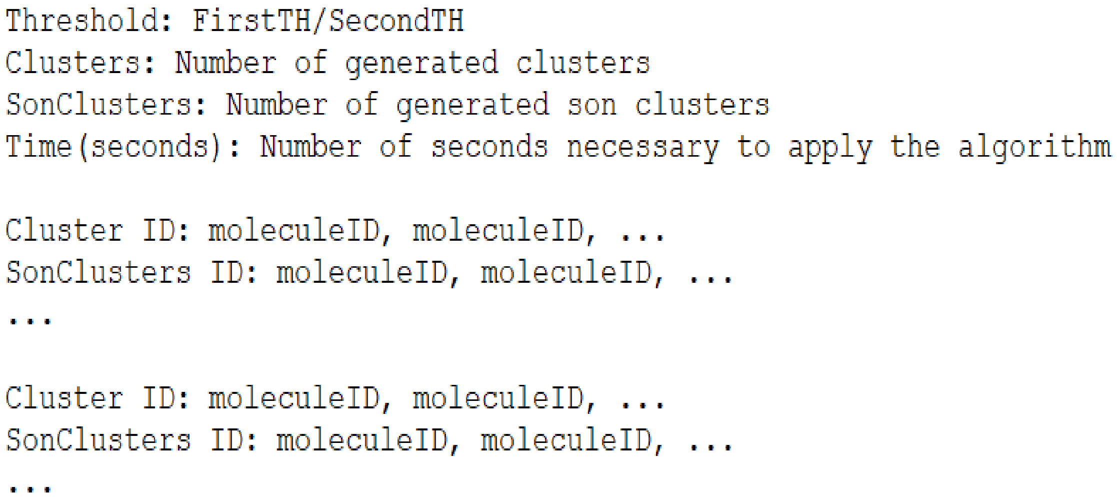
- First file with the name Dataset_FirstTH_SecondTH_Output.txt. E.g (myList_0.5_0.5_Output.txt) Following organization:

- **Thresholds** parameters given to execute the algorithm;
- **Number of clusters** generated;
- **Number of son-clusters** generated;
- **Time (in seconds)** of execution;
- **Representation of clusters and son-clusters** generated

Format of file similar to Figure 3.2

- Second File with the name Dataset_Centroids.txt. Following organization:
  - **First molecule of each generated cluster** (as many as the generated clusters), with molecular identifier and canonical SMILES.

Format of file similar to Figure 3.3

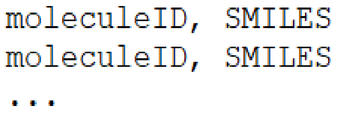


```
Threshold: FirstTH/SecondTH
Clusters: Number of generated clusters
SonClusters: Number of generated son clusters
Time(seconds): Number of seconds necessary to apply the algorithm

Cluster ID: moleculeID, moleculeID, ...
SonClusters ID: moleculeID, moleculeID, ...
...

Cluster ID: moleculeID, moleculeID, ...
SonClusters ID: moleculeID, moleculeID, ...
...
```

Figure 3.2: First File - A .txt file with the similar aspect representing the clustering results
.



```
moleculeID, SMILES
moleculeID, SMILES
...
```

Figure 3.3: Second File - A .txt file with the similar aspect representing centroids (first molecules) of each cluster
.

The implementation of the algorithm was made using Python(release 3.6.3), being the molecules SMILES processed and compared using RDKit library.

3.3 Chemical Information Processing

There are several chemical tools/libraries that could be used to implement the task described above. Open Babel (Pybel) and RDKit are two of the most used chemical toolkits, both are free to use and have an open source code. However, according to google trends, RDKit is more searched than Open Babel. (Figure 3.4)(googleTrends, 2018) This usually translates into greater support among users in solving complex problems.

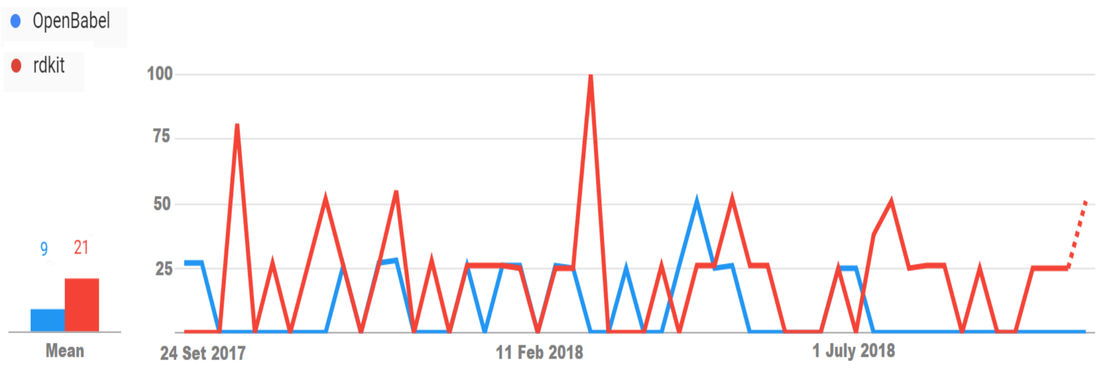


Figure 3.4: Google Trends - The numbers represent the search interest relative to the highest point in the graph of a given region(in this case global) in a given period. A value of 100 represents the peak popularity of a term. A value of 50 means that the term had half the popularity. A score of 0 means that there was not enough data on the term.
.

3.3.1 OpenBabel/Pybel

OpenBabel is a C++ toolkit that allows the reading and writing of molecular file formats (more than 80 supported) as well as molecular data processing. This toolkit supports SMiles ARbitrary Target Specification (SMARTS) structure searching and molecular fingerprints (daylight and structural-key based). Pybel is the python module that provides access to the OpenBabel toolkit.(Boyle, 2012)

3.3.2 RDKit

RDKit is an open source toolkit for cheminformatics with core data structures and algorithms developed in C++ with bindings for Python, Java and C#. Originally developed at Rational Discovery, is currently being used and developed within the Novartis Institutes for BioMedical Research. (Landrum, 2018; Tosco *et al.*, 2014)

Unlike Pybel, RDKit allows to turn SMILES into 2D descriptors like ECFP_4 and ECFP_6 and compare those descriptors using multiple similarity measures such as Tanimoto, Dice, Cosine, Sokal, Russel, among others.(Landrum, 2018)

3.4 Data

By using the Brotherhood algorithm it is expected a fast algorithm, less sensitive and more manageable to entry parameters and the ability of partitioning the chemical space through the use of large datasets. In the next chapter,divided in three phases, it was used different sets of the same database, ZINC Database, more specifically the Standard All Purchasable:

- In the first phase, clustering process was made to test the amount of time needed to run the algorithm using small and large datasets, including an analysis of clusters and son-clusters generated. For that, 12 randomly selected datasets from 1,000 to 5,000,000 molecules were used.

- With the second phase, the purpose was to evaluate if there was a relationship between the amount and order of molecules per dataset with the generated clusters, son-clusters and time of execution. The clustering process was made in 21 randomly selected datasets, each randomly ordered 5 times, in amounts from 1,000 to 100,000 molecules, totalizing 105 runs.

- In the third and last phase, the goal was to evaluate if the clustering process in large amounts of molecules were able to partition most of the chemical space. The clustering was made for 2 different datasets with 2,000,000 and 5,000,000 molecules. Then, for 2,000,000 new molecules, it was verified whether they would be part of any of the previously generated clusters. In this way it would be possible to verify if the molecular space was adequately divided.

Chapter 4

Clustering Analysis

This chapter shows the results obtained through the application of the developed algorithm to all the datasets through all the three lines of action. For each phase, results will be discussed.

The clustering process was executed in a machine with a Intel Core Processor (Broadwell) with a base frequency of 2.2Ghz 4Mb cache and 20 cores with 32GB of RAM running a Debian GNU/Linux 8 (jessie).

4.1 Phase I

The algorithm was tested 4 times with the 12 described datasets, changing only the two entry threshold parameters. The thresholds (first threshold - second threshold) used were: 0.2-0.2; 0.3-0.3; 0.3-0.5 and 0.5-0.3. Since the purpose of the algorithm is to generate a treatable and small number of clusters, the thresholds used couldn't be too high, otherwise, there would be a risk of having a huge number of clusters and the algorithm complexity problem would remain. In the table 4.1 is represented the time (in seconds) needed to run the algorithm for each dataset, for each set of thresholds and the number of clusters and Son-Clusters generated.

For the thresholds of 0.5-0.3, the table does not display all the data. This happens because for relatively small datasets, the time required to apply clustering is already too time consuming. For the 200,000 molecules dataset, for example, comparing the time required with the remaining sets of parameters presented it's possible to see that it takes more than 20x and even 40x. The justification for this is due to the fact that the first threshold (0.5) may be too high. In this case, for a molecule to belong to a cluster, it must have structural similarity greater than 50% with all of the molecules in this cluster

0.2-0.2			
Mol	**Clust**	**Son-Clust**	**Time**
1,000	266	281	1
2,000	377	561	2
5,000	650	1,356	5
10,000	900	2,633	11
20,000	1,282	4,943	26
50,000	1,848	9,980	87
100,000	2,442	16,537	186
200,000	3,166	27,315	488
500,000	4,415	50,118	1,602
1,000,000	5,520	76,582	3,886
2,000,000	6,819	115,048	9,659
5,000,000	8,673	183,207	39,331

0.3-0.3			
Mol	**Clust**	**Son-Clust**	**Time**
1,000	465	198	1
2,000	723	504	2
5,000	1,293	1,524	7
10,000	1,917	3,196	17
20,000	2,893	6,308	47
50,000	4,433	14,345	145
100,000	6,057	25,338	337
200,000	8,067	43,839	765
500,000	11,420	85,527	2,197
1,000,000	14,613	136,118	4,807
2,000,000	18,314	212,542	10,621
5,000,000	23,820	352,539	30,129

0.3-0.5			
Mol	**Clust**	**Son-Clust**	**Time**
1,000	465	288	1
2,000	723	778	2
5,000	1,293	2,620	7
10,000	1,917	6,004	18
20,000	2,893	12,936	50
50,000	4,433	33,979	167
100,000	6,057	66,576	427
200,000	8,067	126,626	1,049
500,000	11,420	279,624	3,447
1,000,000	14,613	485,319	9,310
2,000,000	18,314	816,916	22,979
5,000,000	23,820	1,492,022	78,631

0.5-0.3			
Mol	**Clust**	**Son-Clust**	**Time**
1,000	954	1	2
2,000	1,868	3	6
5,000	4,620	28	39
10,000	8,702	166	150
20,000	16,100	606	551
50,000	33,493	2,982	2,672
100,000	55,657	8,079	8,371
200,000	87,595	20,078	24,550
500,000	-	-	-
1,000,000	-	-	-
2,000,000	-	-	-
5,000,000	-	-	-

Table 4.1: Time (in seconds) necessary to run the algorithm, clusters and Son-Clusters generated for each datasets with each set of thresholds parameters.

or, to be in a son-cluster, it must have structural similarity greater than 50% with at least one of them. If this doesn't happen often, the growth of number of clusters will be fast. Thus, in the course of the clustering process, more and more comparisons are necessary as more and more clusters are created, which makes the time needed to run the algorithm increase exponentially.

Analyzing the results obtained for set 0.3-0.5, it is possible to verify that, compared with the other two (0.2-0.2, 0.3-0.3), the time differences are not that big when it is applied the process in small datasets. However, as the dataset increases, the differences start to to be noticed. In the 3 biggest datasets, for example, the time required is already more than double. The justification for this turns out to be similar to the previously set 0.5-0.3. The number of clusters doesn't grow so quickly but the number of son-clusters does.

Comparing the execution time between sets 0.2-0.2 and 0.3-0.3, it is possible to verify a curious fact. Clustering time is always smaller in set 0.2-0.2, except in the largest dataset (5,000,000), where time is approximately 1.25x bigger. This happens because being thresholds lower, the number of clusters generated are actually smaller. However, the size of each of them increases significantly compared to the clusters generated with 0.3-0.3 threshold. The fact that clusters are larger could mean that the average value distribution of similarity approaches more than 0.2 threshold than 0.3. There are fewer clusters and these are larger, consequently, it may be necessary to compare a molecule to a large part of the molecules present in the cluster, until it finds an appropriate place for that molecule to belong. In the figure 4.1, it is possible to observe the charts of time for each set, according to each dataset, in a more graphical and simpler way.

With the results seen previously, it is possible to verify that it is through the balance between thresholds that is possible to achieve a reduced number of clusters with a good time performance of the algorithm. We observed four combinations of thresholds with the final conclusions:

- By increasing the first threshold (0.5-0.3), many clusters are generated which, even for small datasets, make the time for execution too consuming.

- By increasing the second threshold (0.3-0.5), many Son-Clusters are generated which make the clustering process slower.

- By decreasing the two parameters (0.2-0.2), less clusters are generated (either clusters and Son-Clusters) which make each cluster larger and consequently required more time to conclude the clustering process.

- Through intermediate threshold values (0.3-0.3) it is possible to have a balanced amount of cluster and acceptable sizes that end up reducing the execution time.

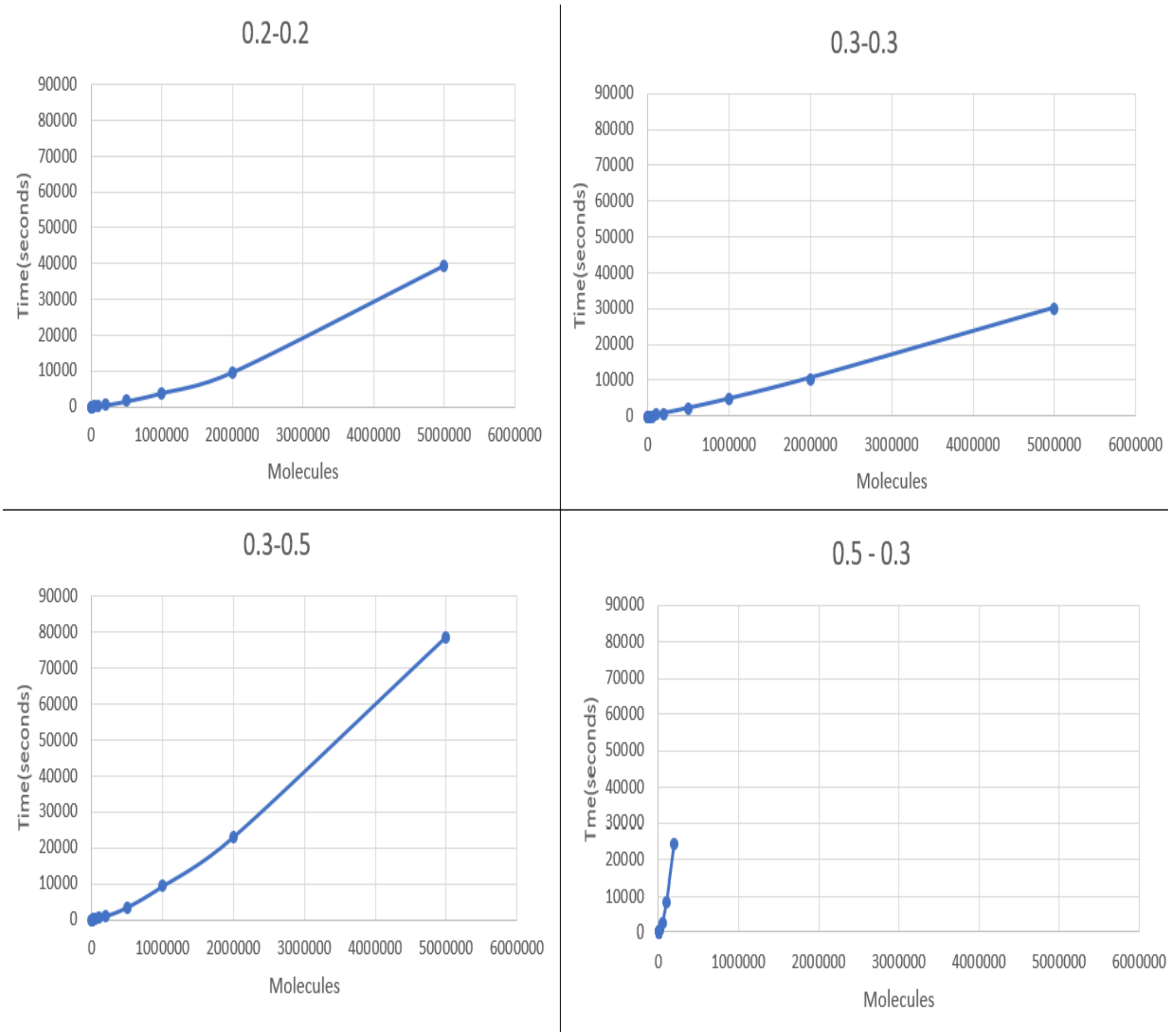


Figure 4.1: Charts of Time for each set (of thresholds) for each dataset.

.

4.2 Phase II

In phase II, the algorithm was tested 105 times with the same set of thresholds (0.3-0.3). Three different sets for each of the seven amounts of molecules from 1,000 to 100,000 molecules with five different randomly orders. In the appendix A, it is presented in tables all the data obtained.

To analyze the data, it was made a chart for each of the properties: Clusters, Son-Clusters and Time. In fact, boxplots were plotted in each chart, for each of the amounts of the datasets. (figures 4.2,4.3, 4.4)

It is possible to observe that only 2 outliers were detected in the chart of time, regarding the values "525" and "511". Since the server where the clustering process was made is shared, and only those 2 values were registered as outliers it may represent a moment where the server got overloaded.

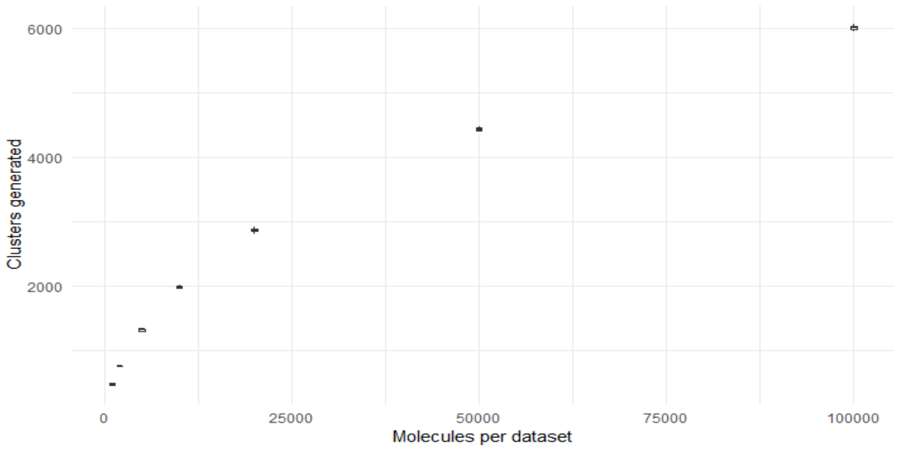


Figure 4.2: Chart with boxplots - Clusters generated for each amount of the dataset

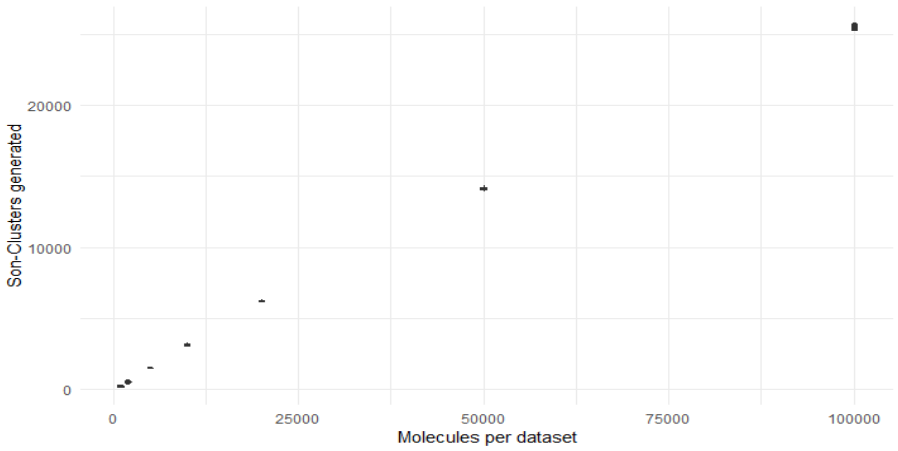


Figure 4.3: Chart with boxplots - Son-Clusters generated for each amount of the dataset

By making for each chart, the log x and log y and plotting a line crossing the mean of each boxplot, the following charts were obtained figures 4.5, 4.6, 4.7.

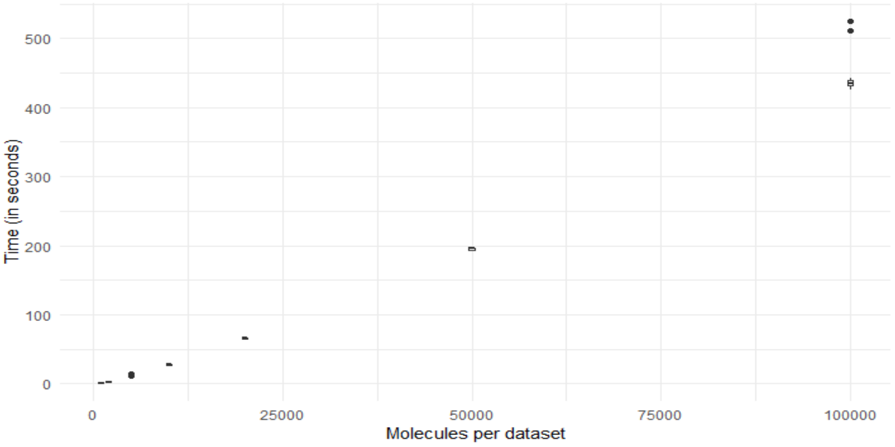


Figure 4.4: Chart with boxplots - Time necessary to run for each amount of the dataset

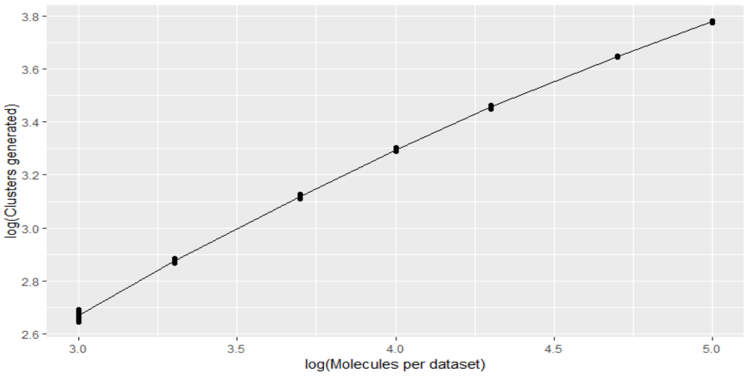


Figure 4.5: Chart with boxplots - log(Clusters) generated for each amount of the log(dataset)

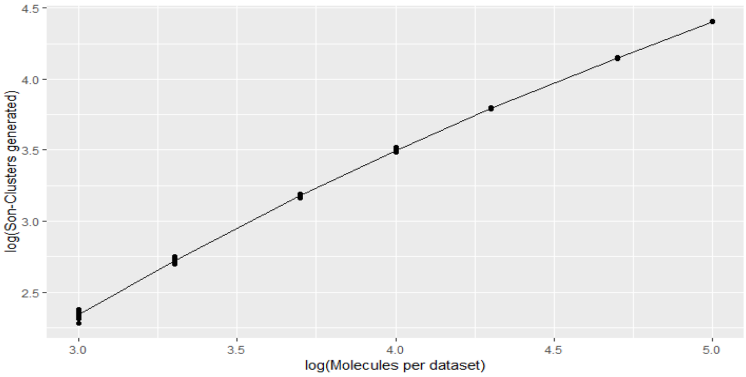


Figure 4.6: Chart with boxplots - log(Son-Clusters) generated for each amount of the log(dataset)

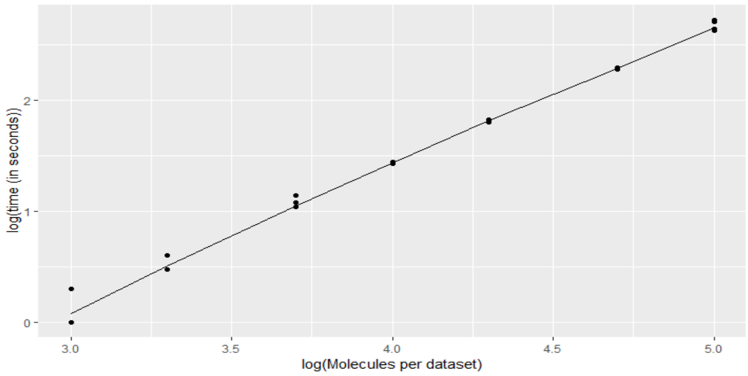


Figure 4.7: Chart with boxplots - log(Time) necessary to run for each amount of the log(dataset)

It is possible to see that in the 3 cases, by doing the log $nrMolecules$ and log y, being y Clusters, Son-clusters or time, it results in a linear relationship. Whenever two quantities plotted in logarithmic axes show a linear relationship, it indicates that the two quantities have a power law distribution. So, it is possible to say that independently of the order and the constitution of the datasets, clusters, son-clusters and time have a linear relationship between the dataset that generated them.

4.3 Phase III

This phase has the following line action: first, the clustering process was applied for 2 datasets (2 million and 5 million molecules). Then, with a new dataset with 2 million molecules, it was made the verification if they would belong to an existing Cluster or Son-Cluster or if they would create a new Cluster or Son Cluster. Before starting this process, it was evaluated the relationship between clusters/Son-Clusters and the number of molecules that originated them. With this evaluation, it is possible to observe whether the speed of generated clusters increases or decreases as the dataset increases. The results presented on table 4.2 and 4.3 make use of the results obtained in the phase I, when applied the algorithm to the 12 datasets with the thresholds as 0.2-0.2 and 0.3-0.3. The charts in figure 4.8 and 4.9 represent the data in tables 4.2 and 4.3, respectively.

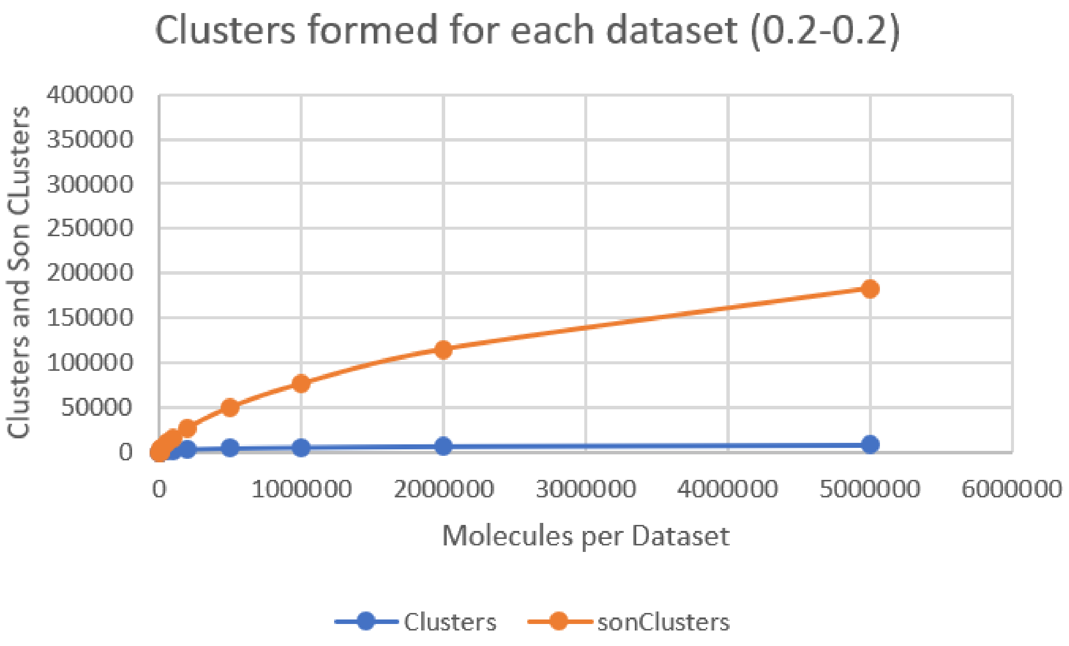


Figure 4.8: RelationShip between the number of clusters generated and dataset that originated them when using parameters 0.2-0.2 as entry.
.

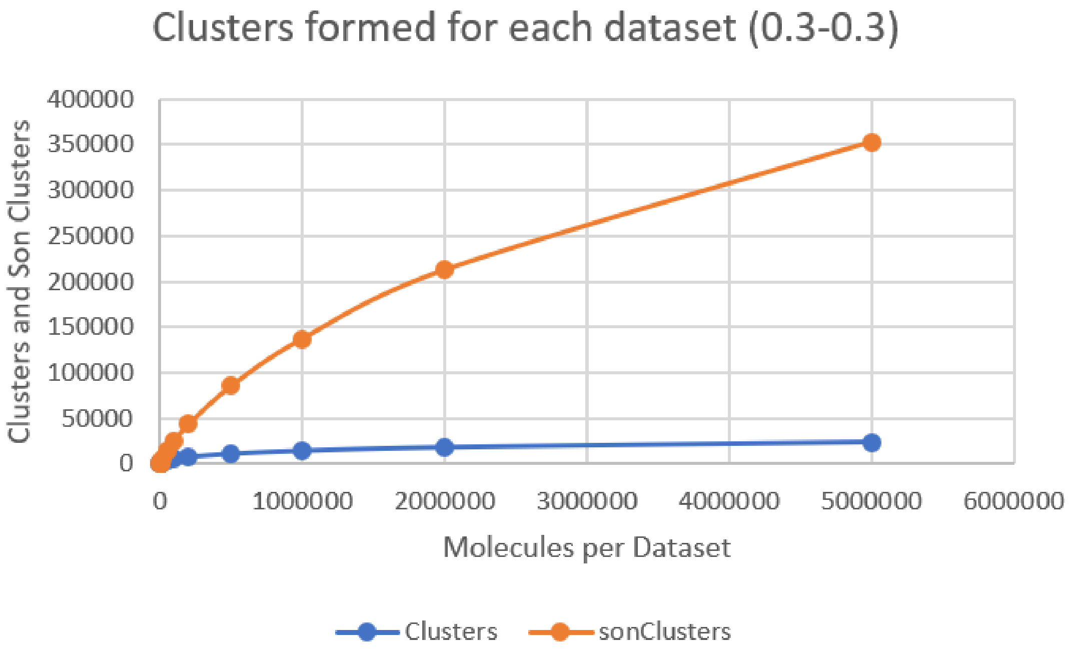


Figure 4.9: RelationShip between the number of clusters generated and dataset that originated them when using parameters 0.3-0.3 as entry
.

0.2-0.2				
Molecules	**Clusters**	**son-Clusters**	**C/Mol**	**SC/Mol**
1,000	266	281	0.2660	0.2810
2,000	377	561	0.1885	0.2805
5,000	650	1,356	0.1300	0.2712
10,000	900	2,633	0.0900	0.2633
20,000	1,282	4,943	0.0641	0.2472
50,000	1,848	9,980	0.0370	0.1996
100,000	2,442	16,537	0.0244	0.1654
200,000	3,166	27,315	0.0158	0.1366
500,000	4,415	50,118	0.0088	0.1002
1,000,000	5,520	76,582	0.0055	0.0766
2,000,000	6,819	115,048	0.0034	0.0575
5,000,000	8,673	183,207	0.0017	0.0366

Table 4.2: Relationship between Clusters and Son-Clusters with the dataset using parameters 0.2-0.2.

By analyzing both tables, it is possible to verify that the relationship between the number of clusters/Son-Clusters generated and the number of molecules that generated them is decreasing as dataset increases, which means that for every iteration less and less clusters are being generated. Observing figures 4.8 and 4.9 it is possible to see that the curve of generated clusters is decreasing and showing a sign of stabilization.

After this, as mentioned, it was executed the algorithm process for 2 million randomly selected molecules with thresholds 0.3-0.3, with the results presented in table 4.4.

Then, for other set of randomly selected 2 million molecules the assignment for each molecule was made, so they had to fill one of the following categories: Inside Cluster, Inside Son-Cluster, Similarity with Cluster but no Son-Cluster (would generate new Son-Cluster), No Assignment (would generate new cluster). (Table 4.5)

Observing the table 4.5, the most important information we can retain is that in an universe of 2 million molecules, 4,822 molecules wouldn't be linked in anyway to any of the clusters. In other words, only 0.24% of the molecules wouldn't be assigned to a cluster in anyway.

The previously procedure was also applied for a clustering of 5 million molecules and an assignment with the same set of 2 million molecules, being the results presented in tables 4.6 and 4.7.

In this case, with a previously clustering of 5 million molecules, only 1,531 of the 2 million molecules wouldn't be linked to a cluster, which represents 0.07% of them.

0.3-0.3				
Molecules	**Clusters**	**son-Clusters**	**C/Mol**	**SC/Mol**
1,000	465	198	0.4650	0.1980
2,000	723	504	0.3615	0.2520
5,000	1,293	1,524	0.2586	0.3048
10,000	1,917	3,196	0.1917	0.3196
20,000	2,893	6,308	0.1447	0.3154
50,000	4,433	14,345	0.0887	0.2869
100,000	6,057	25,338	0.0606	0.2534
200,000	8,067	43,839	0.0403	0.2192
500,000	11,420	85,527	0.0228	0.1711
1,000,000	14,613	137,270	0.0146	0.1373
2,000,000	18,314	212,542	0.0092	0.1063
5,000,000	23,820	352,539	0.0048	0.0705

Table 4.3: Relationship between Clusters and Son-Clusters with the dataset using parameters 0.3-0.3.

Thresholds	0.3-0.3
Clusters	18,310
Son-Clusters	210,715
Time(seconds)	10,575

Table 4.4: Results of clustering for 2 million randomly selected molecules with thresholds 0.3-0.3

Inside Cluster	117878
Inside Son-Cluster	1,763,283
Similarity with cluster but no Son-Cluster	114,728
No Assignment	4,822

Table 4.5: Assignment of 2 million randomly selected molecules with thresholds 0.3-0.3

Thresholds	0.3-0.3
Clusters	23,918
Son-Clusters	350,175
Time(seconds)	29,993

Table 4.6: Results of clustering for 5 million randomly selected molecules with thresholds 0.3-0.3

If the pharmacological definition for each cluster had already been made, it would be possible to predict some information for each of the molecules with this assignment process. Thus, in the first case, it wouldn't be possible to predict for 4,822 molecules

Inside Cluster	113,494
Inside Son-Cluster	1,835,030
Similarity with cluster but no Son-Cluster	50,656
No Assignment	1,531

Table 4.7: Assignment of 2 million randomly selected molecules with thresholds 0.3-0.3

(0.24% of the 2 million) and in the second case it wouldn't be possible to make any prediction for 1,531 molecules (0.07% of the 2 million).

The cluster method used is based on an heuristic process, so it is necessary to ignore some information, sacrificing optimum results, in order to make the decision faster and sometimes even possible. In the last case, we would then be able to predict for 99.93% of the molecules.

Chapter 5

Defining Clusters Pharmacologically

The main purpose of this chapter is to make use of the results obtained with the algorithm, in order to define each of the clusters with pharmacological information from ChEMBL_23 database. Two lists of clusters were defined: 8,673 clusters (generated through 5 million ZINC dataset with 0.2-0.2 thresholds) and 23,820 clusters (generated through 5 million ZINC dataset with 0.3-0.3 thresholds). The first element of each cluster was used to represent it. So, in fact, there is a list of 8,673 and 23,820 molecules, defined as centroids (they are not the center of the cluster but are representative of it).

First, ChEMBL_23 data was processed and filtered in order to keep only the relevant information relative to activity. E.g., if a compound is active or inactive to a target. Then, each of those compounds were linked to a cluster/centroid. Finally, a database was created with all the information in order to be used in the user interface.

5.1 ChEMBL_23 Data Processing

The ChEMBL_23 is a database that contains 72 tables with different kind of information relative to bioactivities. However, not all information is relevant for the goal of this task. The information needed was retrieved from 7 tables and not all columns were required. In the figure 5.1 it is presented the tables and columns used.

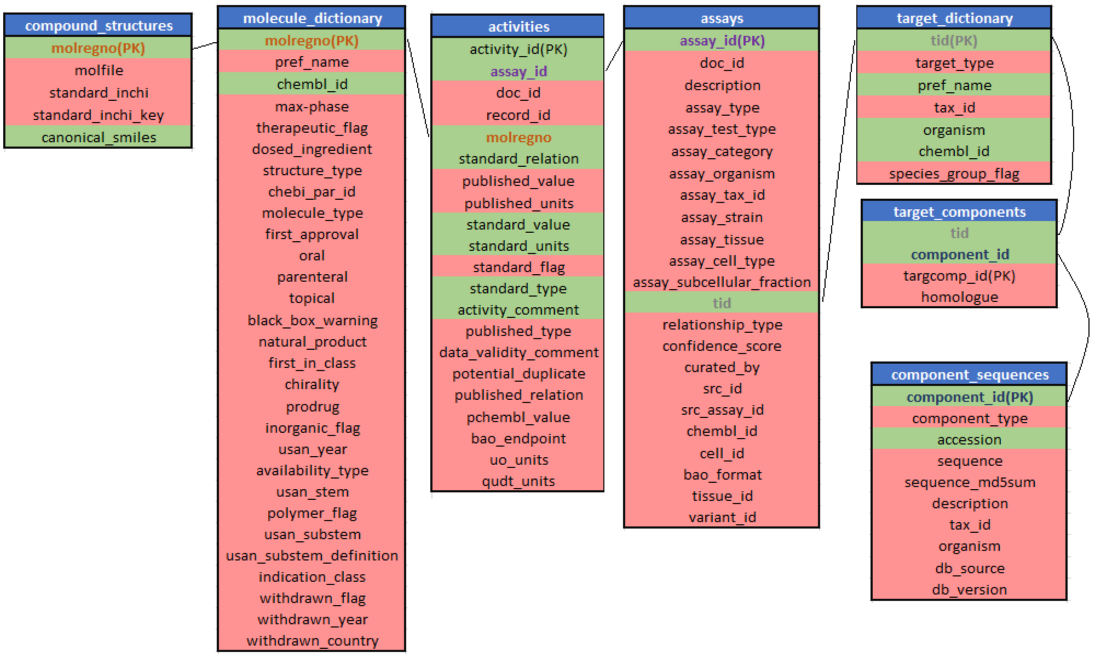


Figure 5.1: Seven retrieved tables from ChEMBL_23. Green background represents selected columns and red background represents unselected. The lines between columns represent the columns that inter ligate all table information.

The tables and columns above mentioned have the following description: (number of entries for each table is mentioned in brackets)

- compound_structures (1,818,302)

  Table storing various structure representations (e.g., Molfile, InChI) for each compound

  - molregno: Internal Primary Key for the compound structure and foreign key to molecule_dictionary table
  - canonical_smiles: Canonical smiles, generated using pipeline pilot

- molecule_dictionary(1,742,024)

  Non redundant list of compounds/biotherapeutics with associated identifiers

  - molregno: Internal Primary Key for the molecule
  - chembl_id: ChEMBL identifier for this compound (for use on web interface etc)

- activities(14,675,320)

  Activity 'values' or 'end points' that are the results of an assay recorded in a scientific document. Each activity is described by a row.

  – activity_id: Unique ID for the activity row
  – assay_id: Foreign key to the assays table (containing the assay description)
  – molregno: Foreign key to compounds table
  – standard_relation: Symbol constraining the activity value (e.g. $>$, $<$, $=$)
  – standard_value: Same as PUBLISHED_VALUE but transformed to common units: e.g. mM concentrations converted to nM.
  – standard_units: Selected 'Standard' units for data type: e.g. concentrations are in nM.
  – standard_type: Standardised version of the published_activity_type (e.g. IC50 rather than Ic-50/Ic50/ic50/ic-50)
  – activity_comment: Describes non-numeric activities i.e. 'Slighty active', 'Not determined'

- assays(1,238,241)

  Table storing a list of the assays that are reported in each document. Similar assays from different publications will appear as distinct assays in this table.

  – assay_id: Unique ID for the assay
  – tid: Target identifier to which this assay has been mapped. Foreign key to target_dictionary. From ChEMBL_15 onwards, an assay will have only a single target assigned.

- target_dictionary(11,538)

  Target Dictionary containing all curated targets for ChEMBL. Includes both protein targets and non-protein targets (e.g., organisms, tissues, cell lines)

  – tid: Unique ID for the target
  – pref_name: Preferred target name: manually curated
  – organism: Source organism of molecular target or tissue, or the target organism if compound activity is reported in an organism rather than a protein or tissue

  - chembl_id: ChEMBL identifier for this target (for use on web interface etc)

- target_components(9,512)

  Links molecular target from the target_dictionary to the components they consist of (in the component_sequences table). For a protein complex or protein family target, for example, there will be multiple protein components in the component_sequences table.

  - tid: Foreign key to the target_dictionary, indicating the target to which the components belong.
  - component_id: Foreign key to the component_sequences table, indicating which components belong to the target.

- component_sequences (7,758)

  Table storing the sequences for components of molecular targets (e.g., protein sequences), along with other details taken from sequence databases (e.g., names, accessions). Single protein targets will have a single protein component in this table, whereas protein complexes/protein families will have multiple protein components.

  - component_id: Primary key. Unique identifier for the component.
  - accession: Accession for the sequence in the source database from which it was taken (e.g., UniProt accession for proteins).

Using the information retrieved from the 7 tables, the purpose was to reorganize and generate 3 simpler tables with the following information: Compounds_table (information about compounds), Activities_table (activity level, for a given compound to a specific target), Targets_table (information about targets).(figure 5.2)

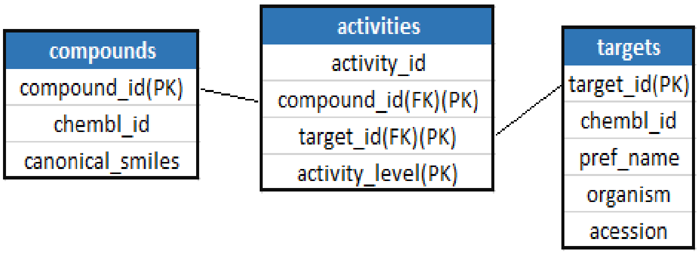
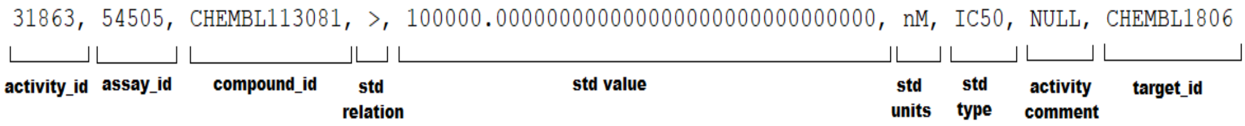


Figure 5.2: ChEMBL information reorganized into simpler tables.

Most of the information of those 3 tables is easily accessed and well organized in the ChEMBL_23 tables, however, the information for activity level is not that accurate.

From activities table of ChEMBL_23, it is possible to obtain that information through the observation and evaluation of the following columns: standard_relation, standard_value, standard_units, standard_type and activity_comment. (Figure 5.3)

In order to do the transformation of all that data into "Active", "Inactive" and "Unknown" fields, it was necessary to apply some rules, because, for more than 14 million entries, it is necessary that the process is automatized.



```
31863, 54505, CHEMBL113081, >, 100000.000000000000000000000000000000, nM, IC50, NULL, CHEMBL1806
```

Figure 5.3: Activity entry example.

To be able to generate those rules, it is necessary to analyze the data present in those columns and find logic patterns. There are more than 2000 different standard_units and near 6000 different standard_type of assays. Despite having so many different standard_types, the 10 most common represent almost 85% of the near 14 million activity entries, and from those 2000 standard_units, most of them are either represented by percentage, concentration or quantity units.

So, these are the logic rules to turn activities into activity_level (Active/Inactive/Unknown):

1. Through the analysis of activity_comment it's possible to describe the activity as "Active" or "Inactive". List of exchange comments to Active/Inactive present in Appendix B.

   E.g. "slight Inhibition" turned to "Active". "Ineffective" turned to "Inactive"

2. If there is no comment to conclude about activity and standard unit is %:

   - The combination between std_relation and std_value is: $>$ 0, the activity is described as "Active".
   - The combination between std_relation and std_value is: $<=$ 0, the activity is described as "Inactive".

     E.g. An inhibition assay have $>$ 30 %. It means that a compound inhibits more than 30% when applied to a specific target. Whenever a value is negative, it is considered as an enhancer instead of an inhibitor, so it's described as inactive.

3. If there is no comment to conclude about activity and standard unit is **not** %:
   - When std_relation is <; =; <=; ~; ; « the activity is considered Active.
   - When std_relation is >; >=; » the activity is considered Inactive.

     E.g. If std_type is IC50, std_relation <, std_units is nM and std_value is 50, it's considered Active. It is evaluated that to achieve 50% inhibition of the target it is necessary a concentration less (<) than 50nM. In other words, 50nM is able to inhibit already more than 50%. However, if every fields maintains the same, but std_relation is >, it is considered Inactive. Since 50 nM isn't able to achieve 50% of inhibition, it is necessary, at least, a concentration higher (>) than 50 nM. Once not an infinity range of concentrations were tested, it's not possible to know if at any concentration the inhibition would occur.
4. If the previously 3 rules couldn't be applied, the activity between compound and target is considered Unknown.

By applying the previously rules it was possible to characterize each activity between compound and target into "Inactive", "Active" and "Unknown".

It is important to mention two important cases. Sometimes, there are different assays regarding the same compound-target activity with contraries information. So, for a compound-target activity it is possible to have "Active" and "Inactive" activity_level (whenever this happens, the information is still used). In the same way, it is possible to have multiple activity_level for compound-target activity saying in all cases, the same activity_level. For example, having 3 different assays saying that a compound-target activity_level is "Active" (whenever this happens, only 1 entry is used).

To define each cluster using activities information, it is necessary to link each compound to the correspondent cluster.

5.2 Link between compounds-activities-targets and clusters

In order to link compounds-activities-targets information to each of the clusters, it was used the similarity between the compounds and centroids. For each compound, it was registered the most similar centroid and all those higher than 0.2 for the list of 8,673, and higher than 0.3 for the list of 23,820. The reason for not keeping only the most similar is because it was verified that in some cases, the difference between the most similar and the second one was to close. One of the cases, for example, had 0.2247 similarity to a centroid and 0.2222 similarity with another. Both cases may contain precious information that couldn't be discarded.

5.3 Database Construction

Through the previously extraction and manipulation of the data from ChEMBL_23 and the linkage between compounds-activities-targets information to the centroids of the clusters, it was constructed a database.

The database was implemented in MySQL. MySQL is the most popular open source SQL database management system which is developed, distributed and supported by Oracle Corporation.(MySQL, 2018) The development process was made through linux server and phpMyAdmin, which is a free software tool written in PHP, intended to handle the administration of MySQL from a web user interface.(phpMyAdmin, 2018)

The developed Chemical Database contains information about centroids generated through the clustering with entry parameters 0.2-0.2 and 0.3-0.3 (zincID, SMILES and the positions of 1's in an ECFP_6 vector), compounds (chemblID), activities(with information about the activity level of a compound to a target) and targets (chemblID, preferable name, organism and accession number if exists). Each compound can be linked to the closest centroid and all those closer than 0.2 for centroids generated with 0.2-0.2 and all those closer than 0.3 for centroids generated with 0.3-0.3. A centroid can have multiple compounds associated. Regarding activities, each compound can be related to multiple targets and each target can be related to multiple compounds.

In the figure 5.4 it is presented the database scheme.

Each of the tables have the following description and attributes description:

- centroids8673 - Table with 8,673 centroids from clusters generated through 5 million ZINC sample with Brotherhood algorithm with entry parameters as 0.2-0.2.(8,673 entries)
  - centroidID - A natural key, from 1 to 8,673
  - zincID - The ZINC ID of the correspondent molecule
  - SMILES - Molecule representation

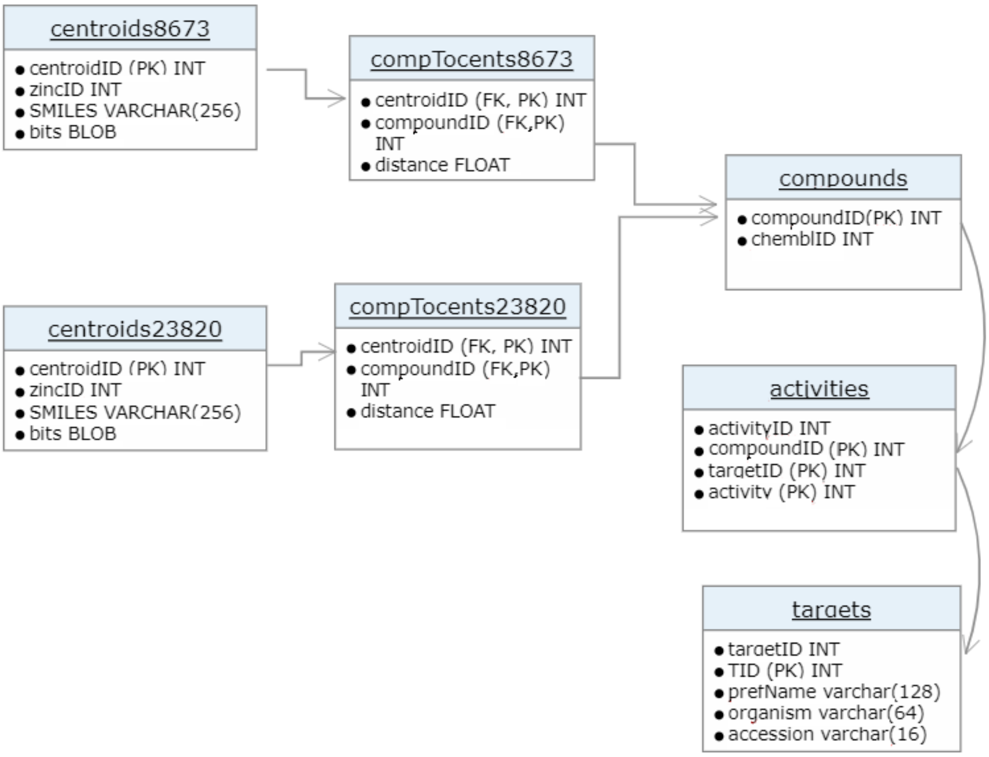


Figure 5.4: Database Scheme

  - bits - SMILES is turned into a ECFP vector. bits represent the position of the 1's in the correspondent vector.

- centroids23820 - Table with 23,820 centroids from clusters generated through 5 million ZINC sample with Brotherhood algorithm with entry parameters as 0.3-0.3.(23,820 entries)

  - centroidID - A natural key, from 1 to 23,820
  - zincID - The ZINC ID of the correspondent molecule
  - SMILES - SMILES is turned into a ECFP bits vector
  - bits - bits represent the position of the 1's in the correspondent vector.

- compTocents8673 - Table linking ChEMBL compounds to the closest centroid and all other centroids with similarity higher than 0.2.(8,312,439 entries)

  - centroidID - Foreign key to the centroids8673 table
  - compoundID - Foreign key to the compounds table

  - distance - Similarity measure value, between 0 and 1, of a compound to the centroid.

- compTocents23820 - Table linking ChEMBL compounds to the closest centroid and all other centroids with similarity higher than 0.3.(1,823,079 entries)
  - centroidID - Foreign key to the centroids23820 table
  - compoundID - Foreign key to the compounds table
  - distance - Similarity measure value, between 0 and 1, of a compound to the centroid.
- compounds - Table with ChEMBL compounds.(1,727,581 entries)
  - compoundID - A Natural key from 1 to 1,727,581
  - chemblID - The ChEMBL ID corresponding to the compound.
- activities - Table that registers the activity level from a compound to a target.(9,957,429 entries)
  - activityID - The ChEMBl activity ID that originated the entry
  - compoundID - Foreign key to the compounds table
  - targetID - Foreign key to the targets table
  - activity - Activity level of the activity. 0:Inactive; 1:Active; 2:Unknown.
- targets - Table with the ChEMBL targets.(10,827 entries)
  - targetID - A Natural key from 1 to 10827
  - TID - The ChEMBL ID corresponding to the target
  - prefName - The preferable name of the target (according to ChEMBL)
  - organism - The organism of the corresponding target
  - accession - The accession number, if exists.

The previously database was used to the construction of a search user interface.

Chapter 6

Search User Interface

With the information stored in the database (only the branch of 8,673 centroids was used, however it's possible to switch in seconds of coding) created in the last chapter, a search user interface was developed with the purpose of predict targets for untested compounds, among other predictions.

The developed system can be seen in two stages:

- Back end (data access stage): The back end is constituted by the database (developed in mySQL v5.5.59) and the back end engine (developed with Python 3.7.1 using Django Framework v2.1.3).

- Front end (presentation stage): On the other hand, front end is constituted by the interface presentation and funcionalities (developed with HTML5, CSS3, Bootstrap v4 and Javascript).

6.1 System Architecture

In order to facilitate future updates to the system, it is necessary to be properly organized, otherwise a simple change could imply changes throughout the system. So, the Model-Template-View (MTV) architecture pattern was the one used.(Django, 2018)

6.1.1 Model-Template-View

The Model-Template-View is a software design pattern, similar to the widely known Model-View-Controller. However, since the controller is the framework itself, in django it's known as MTV. It's a collection of three important components, as the name implies: Models, Templates and Views.

- Model: Provides an abstraction layer (the "models") for structuring and manipulating the data of your Web application.
- Template: The template layer provides a designer-friendly syntax for rendering the information to be presented to the user.
- View: The concept of "views" to encapsulate the logic responsible for processing a user's request and for returning the response.

6.2 Interface

The interface of this system is divided into four sections that can be accessed in the navigation bar: Home, Description, Tool and Contacts. (figure 6.1)

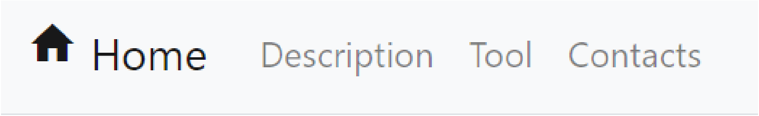


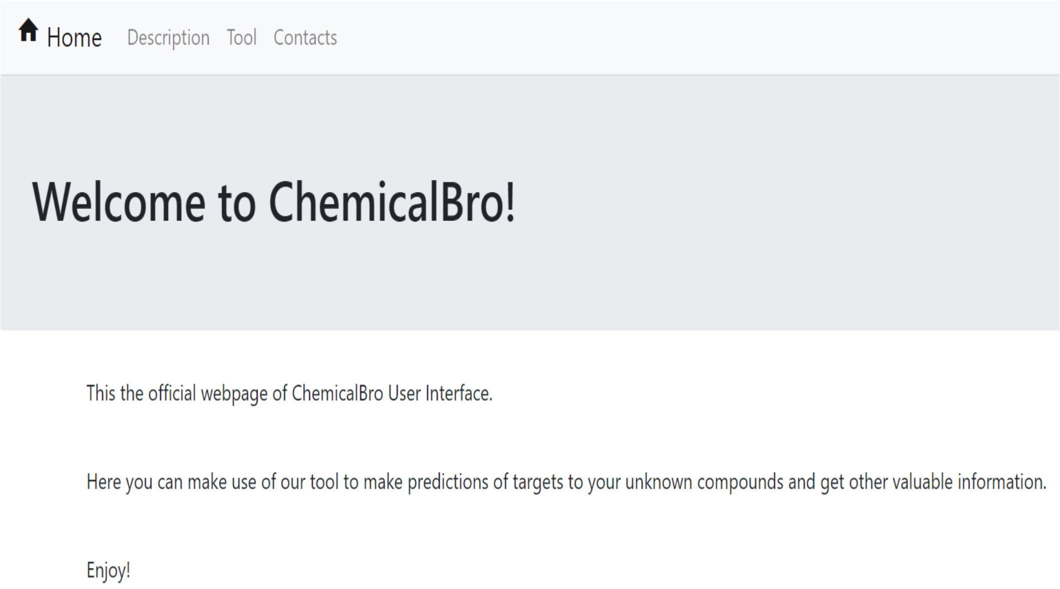
Figure 6.1: Navigation Bar of ChemicalBro Search Interface

6.2.1 Home

The home section is a page where the user is received and contains a brief description of the purpose of the ChemicalBro interface. (figure 6.2)

6.2.2 Description

In the description section there is a more detailed explanation about the options that can be chosen and which are the entry parameters and the expected output. (figure 6.3)



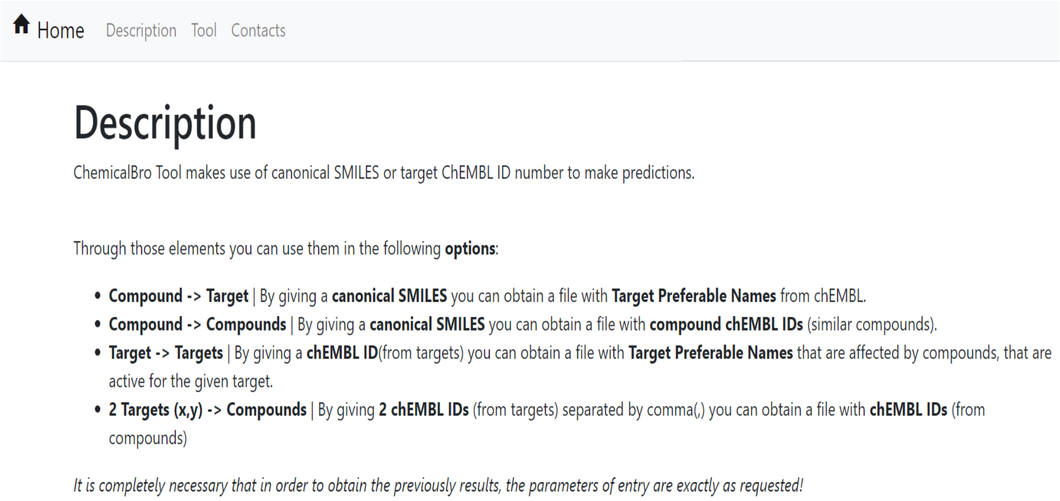
Welcome to ChemicalBro!

This the official webpage of ChemicalBro User Interface.

Here you can make use of our tool to make predictions of targets to your unknown compounds and get other valuable information.

Enjoy!

Figure 6.2: Home page



Description

ChemicalBro Tool makes use of canonical SMILES or target ChEMBL ID number to make predictions.

Through those elements you can use them in the following **options**:

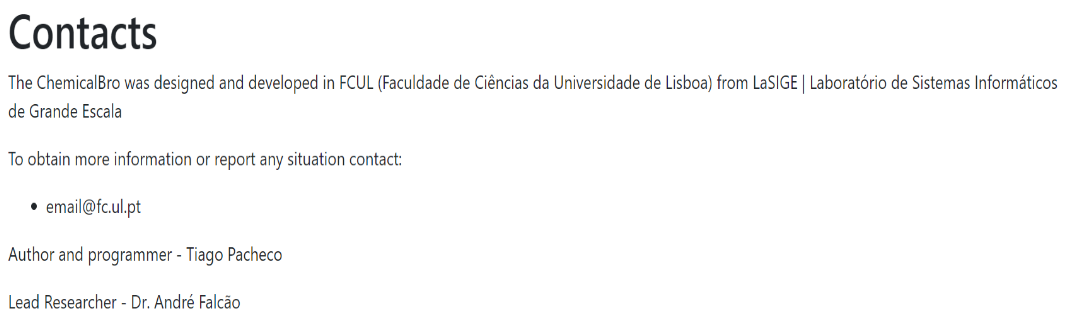
- **Compound -> Target** | By giving a **canonical SMILES** you can obtain a file with **Target Preferable Names** from chEMBL.
- **Compound -> Compounds** | By giving a **canonical SMILES** you can obtain a file with **compound chEMBL IDs** (similar compounds).
- **Target -> Targets** | By giving a **chEMBL ID**(from targets) you can obtain a file with **Target Preferable Names** that are affected by compounds, that are active for the given target.
- **2 Targets (x,y) -> Compounds** | By giving **2 chEMBL IDs** (from targets) separated by comma(,) you can obtain a file with **chEMBL IDs** (from compounds)

It is completely necessary that in order to obtain the previously results, the parameters of entry are exactly as requested!

Figure 6.3: Description page

6.2.3 Contacts

Here, there are more details about who developed the system and how it is possible to contact to obtain more information or report any problem. (figure 6.4)



Contacts

The ChemicalBro was designed and developed in FCUL (Faculdade de Ciências da Universidade de Lisboa) from LaSIGE | Laboratório de Sistemas Informáticos de Grande Escala

To obtain more information or report any situation contact:

- email@fc.ul.pt

Author and programmer - Tiago Pacheco

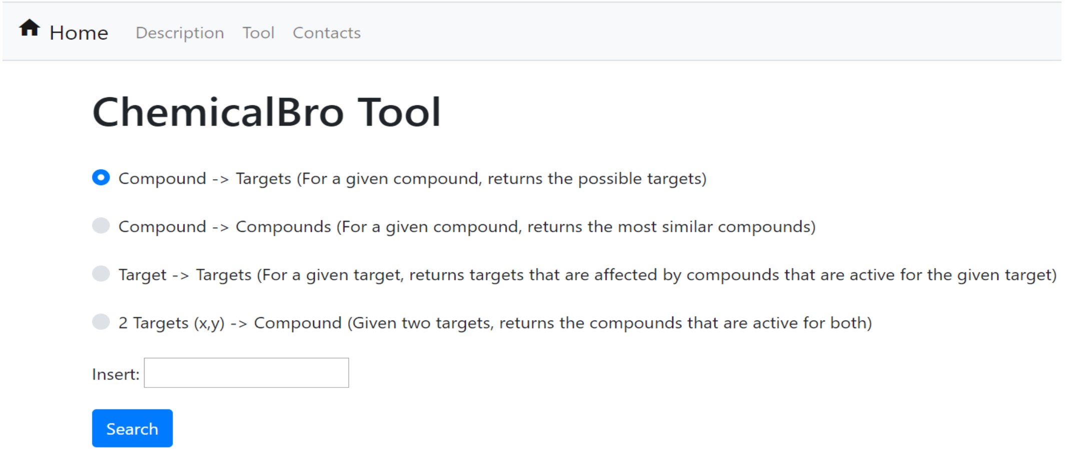
Lead Researcher - Dr. André Falcão

Figure 6.4: Contacts page

6.2.4 Tool

In the Tool page, there are four options that can be chosen and a brief description for each one. An insert box and a search button are present. (figure 6.5) When the option is chosen, the insert box is correctly filled and the search button is clicked, the user browser automatically downloads a .csv file with the results.

It is important to mention that the first and second options require a canonical SMILES (E.g."Cc1c(cnc(n1)N)C(=O)C") and third and fourth options require ChEMBL ID's (E.g."340" and "340,370" respectively).



ChemicalBro Tool

● Compound -> Targets (For a given compound, returns the possible targets)

○ Compound -> Compounds (For a given compound, returns the most similar compounds)

○ Target -> Targets (For a given target, returns targets that are affected by compounds that are active for the given target)

○ 2 Targets (x,y) -> Compound (Given two targets, returns the compounds that are active for both)

Insert:

Figure 6.5: Tool page

There are 4 types of .csv result files:

- First option generates: CompToTargResults.csv
- Second option generates: CompToCompResults.csv
- Third option generates: TargToTargResults.csv

- Fourth option generates: 2TargToCompResults.csv

6.2.4.1 Results examples per Option

Aspirin and Paracetamol are two compounds known for participating in the irreversible inhibition of cyclooxygenase implicated in the prostaglandin synthesis, in the inflammation process.(Infarmed, 2008, 2011)

First Option Example
In the first option, by giving the canonical SMILES from Aspirin (CC(=O)OC1=CC=CC=C1C(=O)O) it is expected to obtain targets related to prostaglandin synthesis and cyclooxygenase. By doing the search, it's obtained 1,002 possible targets, being 10 related to Cyclooxygenase and prostaglandin. (table 6.1)

#	ChEMBLID	Preferable Name
64	CHEMBL5658	Prostaglandin E synthase
65	CHEMBL1293255	15-hydroxyprostaglandin dehydrogenase [NAD+]
23	CHEMBL2096674	Cyclooxygenase
24	CHEMBL230	Cyclooxygenase-2
25	CHEMBL221	Cyclooxygenase-1
33	CHEMBL2949	Cyclooxygenase-1
35	CHEMBL2094253	Cyclooxygenase
42	CHEMBL4102	Cyclooxygenase-2
61	CHEMBL2860	Cyclooxygenase-1
768	CHEMBL4321	Cyclooxygenase-2

Table 6.1: Option 1 Results for Aspirin

Second Option Example

In the second option, by giving the canonical SMILES from Aspirin (CC(=O)OC1=CC=CC=C1C(=O)O) it is expected to obtain, for example, a compound like Paracetamol (acetaminophen) (CC(=O)NC1=CC=C(C=C1)O - CHEMBL112) since both have an high structure similarity and because of that are related to the irreversible inhibition of cyclooxygenase in the inflammation process. By making that search, Acetaminophen appears as a similar compound to Aspirin.

Third Option Example

In the third option, it requires a target and returns other targets that are affected by a compound active for the given target. Aspirin, as mentioned, is present in the inhibition of cyclooxygenases however, it is also known for an effect anti platelet aggregations. So, in this option, by giving a target such as cyclooxygenase, it is expected to get targets related to platelets aggregations. The expected results appear with targets such as: CHEMBL2007, Platelet-derived growth factor receptor alpha; CHEMBL2095189, Platelet-derived growth factor receptor; CHEMBL1913, Platelet-derived growth factor receptor beta; CHEMBL250, Platelet activating factor receptor

Fourth Option Example

In the last option, by giving 2 targets, it is expected to obtain compounds active to both. With the case study presented, by giving two targets such as 5658,2094253 (CHEMBL5658 - Prostaglandin E synthase and CHEMBL2094253 - Cyclooxygenase) it is expected to get compounds such as the mentioned Aspirin and Paracetamol. As expected, both compounds appear in the results file, including other anti inflammatory compounds such as CHEMBL521-Ibuprofen and CHEMBL563-Flurbiprofen.

Case Study - A new molecule with unknown information on ChEMBL

In December 2018, in Journal of Medicinal Chemistry, it was published a new study with the following title: "Discovery and Characterization of the Potent and Highly Selective (Piperidin-4-yl)pyrido[3,2-d]pyrimidine based in vitro Probe BAY-885 for the Kinase ERK5".(Nguyen *et al.*, 2018) In this study, it's presented that probe BAY-885 inhibits a ERK kinase known for having an important play role in various cellular processes, such as proliferation, differentiation, apoptosis and cell survival. ERK is also known as a therapeutic target for several cancers.

The mentioned compound have the chemical formula "C25H28F3N7O2" and the following canonical SMILES "O=C(NC1=CC=C(CN2CCN(CC)CC2)C(C(F)(F)F)=C1) NC3=CC=C(OC4=NC(N)=NC=C4)C=C3". Also, there are no studies of activity presented on ChEMBL, being this a "ghost" compound. So, this molecule could be used in the ChemicalBro interface to find possible targets. By choosing the option 1 - Compound to Targets, it is expected to find ERK related targets as possible results since the recent study shows that the BAY-885 inhibits ERK targets. The results of the .csv file are 868 possible targets, being 3 of them relevant to the case study:

- #225, CHEMBL4040, MAP kinase ERK2
- #346, CHEMBL3385, MAP kinase ERK1
- #741, CHEMBL1907606, Mitogen-activated protein kinase;ERK1/ERK2

With the the results mentioned above, it's possible to observe promising results for the ChemicalBro interface. In this specific case study, for an unstudied molecule, it suggests targets that are related to those presented in the recent laboratory study.

Chapter 7

Conclusions

Cheminformatics has been a crucial approach in the process of discovering new drugs by the pharmaceutical industries, and the premise that similar drugs have similar activities has proved to be quite valid. Despite this, the increasing amount of data has greatly hampered the application of forecasting methods.

One of the main objectives of this work was the creation and development of a clustering algorithm, based on heuristics, capable of becoming an auxiliary tool in predicting new therapeutic targets for unknown compounds.

Through the analysis and evaluation of algorithm performance, it was possible to draw some important conclusions. In the first place, it was possible to verify, for several data sets, how input thresholds influences not only the run time but also the number of clusters generated. This factor becomes very important since the number of clusters generated must be as small as possible but should allow to define with quality the molecular space. Then, it was possible to observe that the order of the data, in a given data set, does not drastically influence the definition of the molecular space. Finally, an assessment was made of how well defined the molecular space was. Through this, it was possible to verify that the number of clusters generated, with increasing data, was becoming smaller and smaller, with molecular space almost entirely defined. To confirm this analysis, for a new set of 2 million data it was possible to verify that only 0.07% of the data wouldn't have place in the molecular space already defined, which can be seen as a rather small dimension.

In fact, the results obtained by the algorithm can be seen as a division of molecular space and not a definition of it. To make this definition, data from an activity database, named ChEMBL v23, was used. Through a logical procedure, this process was performed, giving rise to a database, capable of reflecting the definition of the clusters.

Using this database, it was possible to create a graphical search interface. This allows, for a new unknown drug (by giving the canonical SMILES), to predict possible therapeutic targets and even to find similar new compounds. By providing a ChEMBL target ID, it is possible to get other targets that are affected by active compounds to the given target. By providing two ChEMBL ID's targets, it is possible to obtain compounds that are active for both.

The fulfill of all the steps mentioned above is a further step in the direction of predicting, with quality, new biological and biochemical properties. However, the work shouldn't end here.

In the line of action of this work, it would be interesting to develop new features in the user interface and with new releases of ChEMBL it is also possible to populate the database with more info. Also, it would be interesting to evaluate and compare the predictions obtained using the branch of 8,673 centroids vs 23,820 centroids, in order to see which are more accurate.

References

AHMAD, P.H. & DANG, S. (2015). Performance Evaluation of Clustering Algorithm Using Different Datasets. *International Journal of Advance Research in Computer Science and Management Studies*, **3**, 167–173. 20

ANDREWS, J.M. (2001). JAC Determination of minimum inhibitory concentrations. *Journal of Antimicrobial Chemotheraphy*, **48**, 5–16. 15

BACHMANN, K.A. & LEWIS, J.D. (2005). Predicting inhibitory drug-drug interactions and evaluating drug interaction reports using inhibition constants. *Annals of Pharmacotherapy*, **39**, 1064–1072.

BAJUSZ, D., RÁCZ, A. & HÉBERGER, K. (2015). Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations ? *Journal of Cheminformatics*, 1–13. 13

BOYLE, N.M.O. (2012). Towards a Universal SMILES representation - A standard method to generate canonical SMILES based on the InChI. 1–14. 7, 8, 23

BOYLE, N.M.O., MORLEY, C. & HUTCHISON, G.R. (2008). Pybel : a Python wrapper for the OpenBabel cheminformatics toolkit. *Chemistry Central Journal*, **7**, 1–7.

CHEN, W.L. (2006). Chemoinformatics: Past, Present, and Future. *Journal of Chemical Information and Modeling*, **46**, 2230–2255. 3

CHEN, X. & REYNOLDS, C.H. (2002). Performance of Similarity Measures in 2D Fragment-Based Similarity Searching : Comparison of Structural Descriptors and Similarity Coefficients. *J. Chem. Inf. Comput*, 1407–1414. 11, 13, 14

CHERKASOV, A., MURATOV, E.N., FOURCHES, D., VARNEK, A., IGOR, I., CRONIN, M., DEARDEN, J., GRAMATICA, P., MARTIN, Y.C., CONSONNI, V., KUZ, V.E. &

Cramer, R. (2015). QSAR Modeling: Where have you been? Where are you going to? **57**, 4977–5010. 14

Django, D.S.F. (2018). Django documentation. *(2018-11-22)*, https://docs.djangoproject.com/en/2.1/. 47

FDA (2018). The drug development process. *(2018-08-02)*, https://www.fda.gov/ForPatients/Approvals/Drugs/ucm405382.htm. 1

Gaulton, A., Hersey, A., Bellis, L.J., Chambers, J., Davies, M., Kru, F.A., Light, Y., Mak, L., Mcglinchey, S., Nowotka, M., Papadatos, G., Santos, R. & Overington, J.P. (2014). The ChEMBL bioactivity database : an update. *Nucleic Acids Research*, **42**, 1083–1090. 16

Gaulton, A., Hersey, A., Patr, A., Chambers, J., Mendez, D., Mutowo, P., Atkinson, F., Bellis, L.J., Cibri, E., Davies, M., Dedman, N., Karlsson, A., Magari, P., Overington, J.P., Papadatos, G. & Smit, I. (2017). The ChEMBL database in 2017. *Nucleic Acids Research*, **45**, 945–954. 16

googleTrends (2018). Openbabel vs rdkit. *(2018-09-29)*, https://trends.google.com/trends/explore?cat=31q=open%20babel,rdkit. 23

Gortari, E.F.D., Jacas, C.R.G., Mayorga, K.M. & Franco, J.L.M. (2017). Database fingerprint (DFP): an approach to represent molecular databases. *Journal of Cheminformatics*, 1–9. 10, 11

Han, J., Kamber, M. & Pei, J. (2012). *Data Mining Concepts and Techniques*. Elsevier Inc., Waltham, third edit edn. 17, 18, 19

Heller, S., McNaught, A., Stein, S., Tchekhovskoi, D. & Pletnev, I. (2013). InChI - The worldwide chemical structure identifier standard. *Journal of Cheminformatics*, **5**, 1. 8, 9

Heller, S.R., McNaught, A., Pletnev, I., Stein, S. & Tchekhovskoi, D. (2015). *InChI, the IUPAC International Chemical Identifier*, vol. 7. Journal of Cheminformatics. 8

Holliday, J.D., Hu, C.y. & Willett, P. (2002). Grouping of Coefficients for the Calculation of Inter-Molecular Similarity and Dissimilarity using 2D Fragment Bit-Strings. *Combinatorial Chemistry & High Throughput Screening*, **5**, 155–166. 13

Hu, Y., Lounkine, E. & Bajorath, J. (2009). Improving the Search Performance of Extended Connectivity Fingerprints through Activity-Oriented Feature Filtering and Application of a Bit-Density- Dependent Similarity Function. 10

Infarmed (2008). Resumo das caracterÍsticas do medicamento - aspirina. *(2018-12-10)*, http://app7.infarmed.pt/infomed/download$_{f}icheiro.php?med_{i}d = 640tipo_{d}oc = rcm.$51

Infarmed (2011). Resumo das caracterÍsticas do medicamento - paracetamol. *(2018-12-10)*, http://app7.infarmed.pt/infomed/download$_{f}icheiro.php?med_{i}d = 50148tipo_{d}oc = rcm.$51

Kaitin, K. (2010). Deconstructing the Drug Development Process: The New Face of Innovation. *Clin Pharmacol Ther.*, **87**, 356–361. 2

Kausar, S. & Falcao, A.O. (2018). An automated framework for QSAR model building. *Journal of Cheminformatics*, 1–23. 14

Khan, A.U. (2016). Descriptors and their selection methods in QSAR analysis : paradigm for drug design. **21**, 1291–1302. 9, 10

Kim, S., Thiessen, P.A., Bolton, E.E., Chen, J., Fu, G., Gindulyte, A., Han, L., He, J., He, S., Shoemaker, B.A., Wang, J., Yu, B., Zhang, J. & Bryant, S.H. (2016). PubChem Substance and Compound databases. *Nucleic Acids Research*, **44**, 1202–1213. 15

Kinch, M.S., Haynesworth, A., Kinch, S.L. & Hoyer, D. (2014). An overview of FDA-approved new molecular entities: 1827-2013. *Drug Discovery Today*, **19**, 1033–1039. 1

Landrum, G. (2018). RDKit Documentation. 24

Lipkowitz, K.B. & Boyd, D.B. (2002). *Reviews in Computational Chemistry*, vol. 18. John Wiley and Sons Ltd, Hoboken, United States, 18th edn. 18, 19

Marx, K.A., O'Neil, P., Hoffman, P. & Ujwal, M.L. (2003). Data Mining the NCI Cancer Cell Line Compound GI50 Values: Identifying Quinone Subtypes Effective Against Melanoma and Leukemia Cell Classes. *Journal of Chemical Information and Computer Sciences*, **43**, 1652–1667. 15

MOHAN, C., LONG, K.D. & MUTNEJA, M. (2013). An Introduction to Inhibitors and Their Biological Applications. *EMD Millipore Corporation*, 1–48. 15

MYSQL (2018). What is mysql. *(2018-11-06)*, https://dev.mysql.com/doc/refman/8.0/en/what–is–mysql.html. 43

NANTASENAMAT, C., ISARANKURA-NA-AYUDHYA, C. & PRACHAYASITTIKUL, V. (2010). Advances in computational methods to predict the biological activity of compounds. *Expert Opin Drug Discov*, **5**, 633–654. 3

NEUBIG, R.R., SPEDDING, M., KENAKIN, T. & CHRISTOPOULOS, A. (2003). International Union of Pharmacology Commitee on Receptorn Nomenclature and Drug Classification. *Pharmacological Reviews*, **55**, 597–606. 14, 15

NGUYEN, D., LEMOS, C., WORTMANN, L., EIS, K., HOLTON, S.J., BOEMER, U., MOOSMAYER, D., EBERSPAECHER, U., WEISKE, J., LECHNER, C., PRECHTL, S., SUELZLE, D., SIEGEL, F., PRINZ, F., LESCHE, R., NICKE, B., NOWAK-REPPEL, K., HIMMEL, H., MUMBERG, D., VON NUSSBAUM, F., NISING, C.F., BAUSER, M. & HAEGEBARTH, A. (2018). Discovery and Characterization of the Potent and Highly Selective (Piperidin-4-yl)pyrido[3,2-d]pyrimidine based in vitro Probe BAY-885 for the Kinase ERK5. *Journal of Medicinal Chemistry*, acs.jmedchem.8b01606. 52

PHARMACEUTICAL, D.P. (2018). About drug development. *(2018-08-02)*, http://www.ppdi.com/About/About–Drug–Discovery–and–Development. 3

PHARMACEUTICAL RESEARCH AND MANUFACTURERS OF AMERICA (2016). 2016 Biopharmaceutical Research Industry Profile. *Pharmaceutical Research and Manufacturers of America*, 86. 1

PHPMYADMIN (2018). Bringing mysql to the web. *(2018-11-06)*, https://www.phpmyadmin.net/. 43

POLISHCHUK, P. (2017). Interpretation of Quantitative Structure-Activity Relationship Models: Past, Present, and Future. *Journal of Chemical Information and Modeling*, **57**, 2618–2639. 14

ROGERS, D. & HAHN, M. (2010). Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.*, 742–754. 10, 11

Roy, K. (2004). Topological descriptors in drug design and modeling studies. *Molecular Diversity*, 321–323. 10

Roy, K., Kar, S. & Narayan Das, R. (2015). *Understanding the basics of QSAR for applications in pharmaceutical sciences and risk assessment*. Elsevier Inc., Chennai, 1st edn. 9

Shockley, K.R. (2016). Estimating Potency in High-Throughput Screening Experiments by Maximizing the Rate of Change in Weighted Shannon Entropy. *Scientific Reports*, **6**, 1–10. 15

Skinnider, M.A., Dejong, C.A., Franczak, B.C., Mcnicholas, P.D. & Magarvey, N.A. (2017). Comparative analysis of chemical similarity methods for modular natural products with a hypothetical structure enumeration algorithm. *Journal of Cheminformatics*, 1–15. 11

Sterling, T. & Irwin, J.J. (2015). ZINC 15 – Ligand Discovery for Everyone. *Journal of Chemical information and modeling*, **55**, 2324–2337. 16

Todeschini, R., Consonni, V., Xiang, H., Holliday, J., Buscema, M. & Willet, P. (2012). Similarity Coefficients for Binary Chemoinformatics Data: Overview and Extended Comparison Using Simulated and Real Data Sets. *Journal of Chemical Informaton and Modeling*, **52**, 2884–2901. 11, 12, 13

Tosco, P., Stiefl, N. & Landrum, G. (2014). The integration of Open3DTOOLS into the RDKit and KNIME. *Journal of Cheminformatics*, **6**, P8. 24

United, N. (2017). World population prospects 2017. *(2018-08-02)*, https://esa.un.org/unpd/wpp/. 1

Vilar, S. & Costanzi, S. (2012). Predicting the biological activities through QSAR analysis and docking-based scoring. *Methods in Molecular Biology*, **914**, 271–284. 4

Waley, S.G. (1982). A quick method for the determination of inhibition constants. *Biochem. J.*, **205**, 631–633. 15

Warr, W.A. (2011). Representation of chemical structures. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, **1**, 557–579. 7

Warr, W.A. (2015). Many InChIs and quite some feat. *Journal of Computer-Aided Molecular Design*, **29**, 681–694. 8, 9

Weininger, D. (1988). SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *Chem. Inf. Comput. Sci.*, **28**, 31–36. 7

Wishart, D.S., Feunang, Y.D., Guo, A.C., Lo, E.J., Marcu, A., Grant, R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., Assempour, N., Iynkkaran, I., Liu, Y., Maciejewski, A., Gale, N., Wilson, A., Chin, L., Cummings, R., Le, D., Pon, A., Knox, C. & Wilson, M. (2018). DrugBank 5 . 0 : a major update to the DrugBank database for 2018. *Nucleic Acids Research*, **46**, 1074–1082. 14

Appendix A

	1,000								
	Clusters			Son Clusters			Time (in seconds)		
	First	Second	Third	First	Second	Third	First	Second	Third
Random 1	479	446	495	191	221	204	1	1	2
Random 2	465	450	488	221	225	239	2	1	2
Random 3	473	441	481	211	232	224	2	1	1
Random 4	472	446	495	204	237	218	1	1	1
Random 5	454	446	494	232	238	214	1	1	1

Mean	468	220	1.27
Max	495	239	2
Min	441	191	1

Table A.1: Result of algorithm applied to 3 sets (1,000 molecules) with 5 different random order

	2,000								
	Clusters			Son Clusters			Time (in seconds)		
	First	Second	Third	First	Second	Third	First	Second	Third
Random 1	752	733	764	526	518	525	4	3	4
Random 2	746	745	755	532	510	530	3	4	3
Random 3	753	739	758	525	499	526	4	3	3
Random 4	735	746	769	565	500	523	3	3	3
Random 5	751	754	756	525	504	537	3	3	3

Mean	750	523	3.27
Max	769	565	4
Min	733	499	3

Table A.2: Result of algorithm applied to 3 sets (2,000 molecules) with 5 different random order

	5,000								
	Clusters			Son Clusters			Time (in seconds)		
	First	Second	Third	First	Second	Third	First	Second	Third
Random 1	1,324	1,294	1,320	1,517	1,455	1,551	11	11	11
Random 2	1,340	1,289	1,299	1,503	1,490	1,561	11	14	11
Random 3	1,345	1,308	1,318	1,519	1,516	1,514	11	12	11
Random 4	1,317	1,325	1,287	1,496	1,465	1,516	11	11	11
Random 5	1,323	1,318	1,298	1,529	1,447	1,542	11	11	11

Mean	1,313	1,508	11.27
Max	1,345	1,561	14
Min	1,287	1,447	11

Table A.3: Result of algorithm applied to 3 sets (5,000 molecules) with 5 different random order

	10,000								
	Clusters			Son Clusters			Time (in seconds)		
	First	Second	Third	First	Second	Third	First	Second	Third
Random 1	1,969	1,968	1,950	3,107	3,233	3,141	27	28	27
Random 2	2,012	1,992	1,962	3,077	3,212	3,104	28	28	27
Random 3	2,018	1,985	1,941	3,091	3,208	3,194	28	28	27
Random 4	1,992	1,955	1,970	3,139	3,226	3,113	28	27	27
Random 5	2,001	1,982	1,971	3,111	3,299	3,067	28	28	27

Mean	1,978	3,152	27.53
Max	2,018	3,299	28
Min	1,941	3,067	27

Table A.4: Result of algorithm applied to 3 sets (10,000 molecules) with 5 different random order

	20,000								
	Clusters			Son Clusters			Time (in seconds)		
	First	Second	Third	First	Second	Third	First	Second	Third
Random 1	2,900	2,881	2,874	6,310	6,188	6,124	66	67	64
Random 2	2,883	2,888	2,847	6,193	6,229	6,132	66	66	65
Random 3	2,854	2,913	2,840	6,296	6,253	6,219	66	67	65
Random 4	2,872	2,873	2,837	6,201	6,189	6,197	67	66	64
Random 5	2,856	2,809	2,805	6,233	6,304	6,173	64	66	63

Mean	2,862	6,216	65.47
Max	2,913	6,310	67
Min	2,805	6,124	63

Table A.5: Result of algorithm applied to 3 sets (20,000 molecules) with 5 different random order

	50,000								
	Clusters			Son Clusters			Time (in seconds)		
	First	Second	Third	First	Second	Third	First	Second	Third
Random 1	4,422	4,447	4,408	14,230	14,039	14,337	196	194	196
Random 2	4,450	4,439	4,439	14,159	14,072	14,207	197	193	192
Random 3	4,399	4,461	4,430	14,113	13,996	14,267	196	195	196
Random 4	4,408	4,478	4,407	14,110	13,892	14,303	195	194	193
Random 5	4,420	4,453	4,388	14,229	14,013	14,239	196	196	197

Mean	4,430	14,147	195.07
Max	4,478	14,337	197
Min	4,388	13,892	192

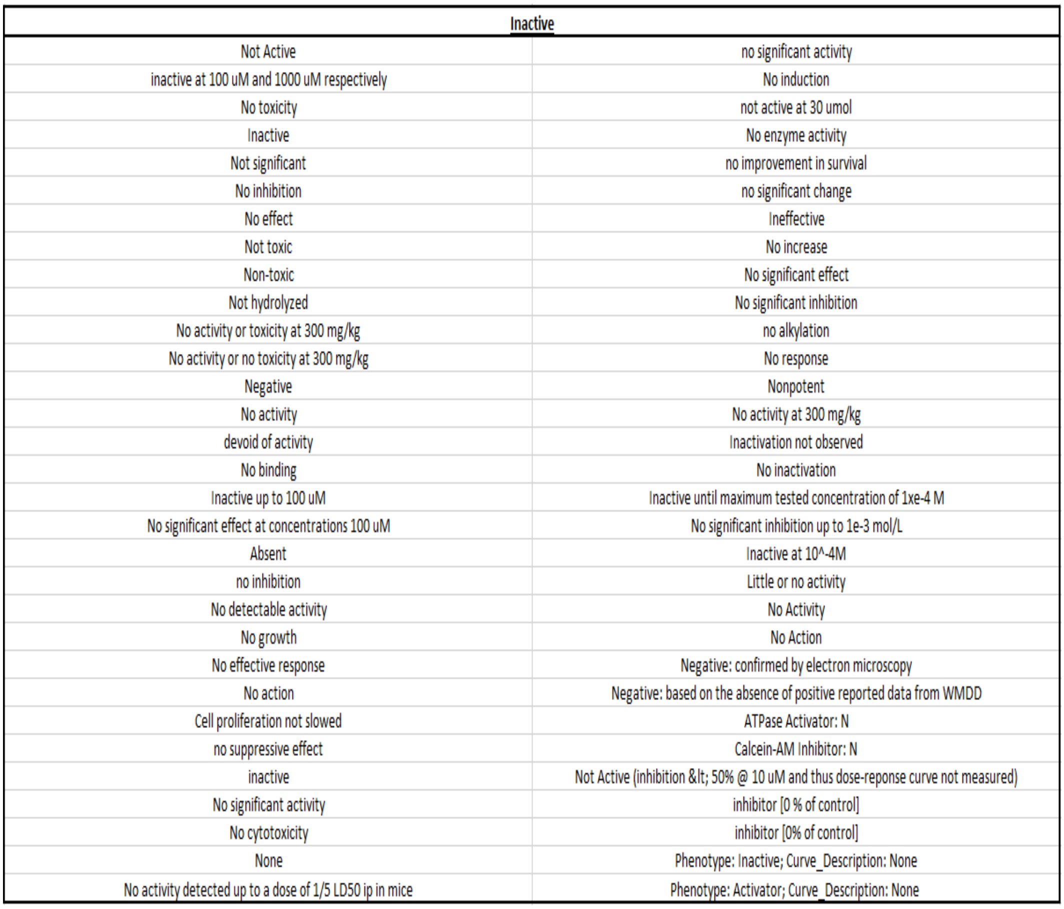
Table A.6: Result of algorithm applied to 3 sets (50,000 molecules) with 5 different random order

	100,000								
	Clusters			Son Clusters			Time (in seconds)		
	First	Second	Third	First	Second	Third	First	Second	Third
Random 1	6,031	6,019	6,025	25,369	25,213	25,328	440	525	431
Random 2	6,047	5,997	5,981	25,640	25,306	25,211	439	431	425
Random 3	6,051	5,942	6,024	25,279	25,185	25,306	438	434	432
Random 4	6,064	6,036	5,971	25,510	25,319	25,348	442	434	433
Random 5	6,022	5,953	6,019	25,329	25,209	25,310	511	431	439

Mean	6,012	25,324	445.67
Max	6,064	25,640	525
Min	5,942	25,185	425

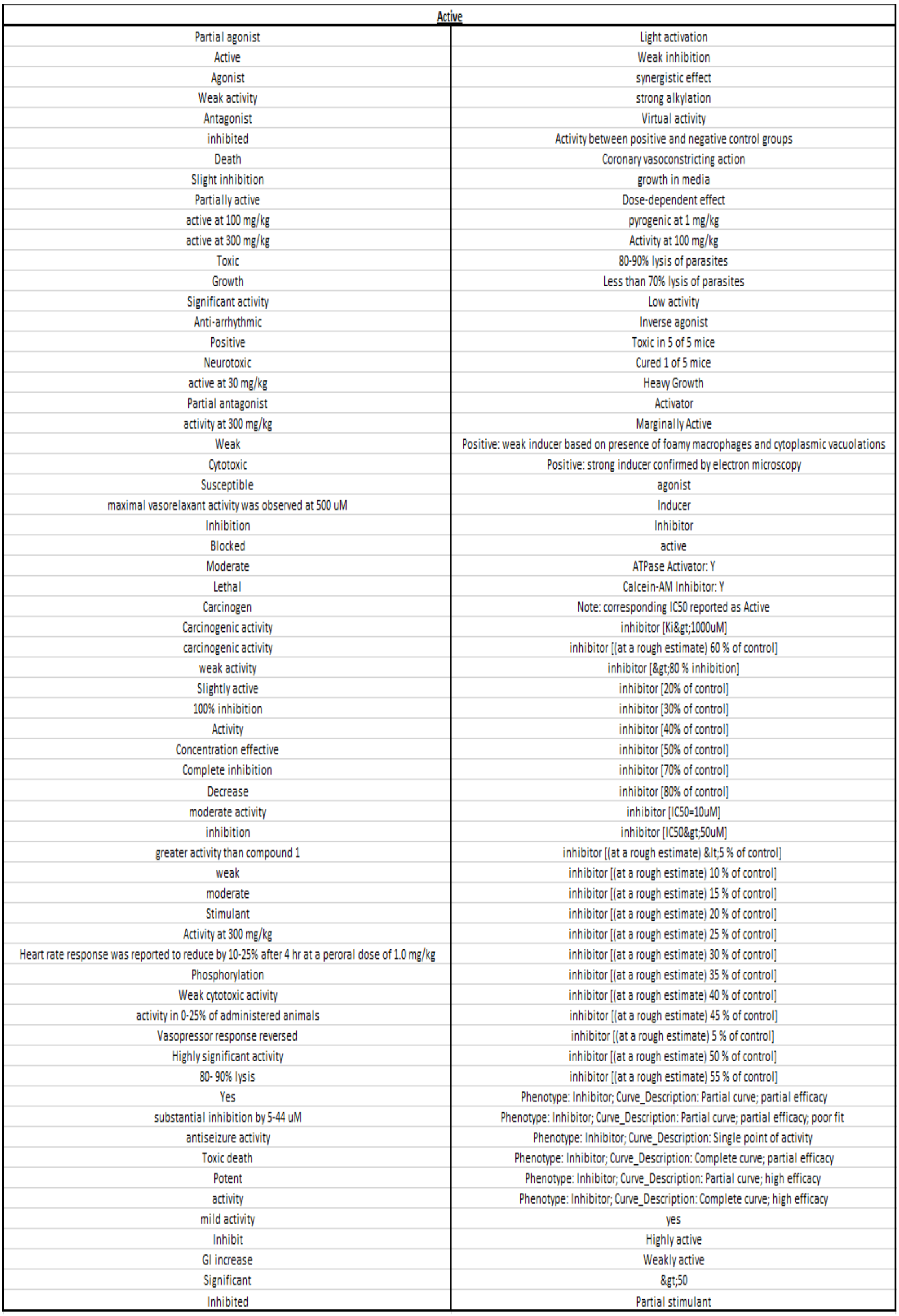
Table A.7: Result of algorithm applied to 3 sets (100,000 molecules) with 5 different random order

Appendix B



Inactive	
Not Active	no significant activity
inactive at 100 uM and 1000 uM respectively	No induction
No toxicity	not active at 30 umol
Inactive	No enzyme activity
Not significant	no improvement in survival
No inhibition	no significant change
No effect	Ineffective
Not toxic	No increase
Non-toxic	No significant effect
Not hydrolyzed	No significant inhibition
No activity or toxicity at 300 mg/kg	no alkylation
No activity or no toxicity at 300 mg/kg	No response
Negative	Nonpotent
No activity	No activity at 300 mg/kg
devoid of activity	Inactivation not observed
No binding	No inactivation
Inactive up to 100 uM	Inactive until maximum tested concentration of 1xe-4 M
No significant effect at concentrations 100 uM	No significant inhibition up to 1e-3 mol/L
Absent	Inactive at 10^-4M
no inhibition	Little or no activity
No detectable activity	No Activity
No growth	No Action
No effective response	Negative: confirmed by electron microscopy
No action	Negative: based on the absence of positive reported data from WMDD
Cell proliferation not slowed	ATPase Activator: N
no suppressive effect	Calcein-AM Inhibitor: N
inactive	Not Active (inhibition &lt; 50% @ 10 uM and thus dose-reponse curve not measured)
No significant activity	inhibitor [0 % of control]
No cytotoxicity	inhibitor [0% of control]
None	Phenotype: Inactive; Curve_Description: None
No activity detected up to a dose of 1/5 LD50 ip in mice	Phenotype: Activator; Curve_Description: None

Figure B.1: Inactive Dictionary.



Active	
Partial agonist	Light activation
Active	Weak inhibition
Agonist	synergistic effect
Weak activity	strong alkylation
Antagonist	Virtual activity
inhibited	Activity between positive and negative control groups
Death	Coronary vasoconstricting action
Slight inhibition	growth in media
Partially active	Dose-dependent effect
active at 100 mg/kg	pyrogenic at 1 mg/kg
active at 300 mg/kg	Activity at 100 mg/kg
Toxic	80-90% lysis of parasites
Growth	Less than 70% lysis of parasites
Significant activity	Low activity
Anti-arrhythmic	Inverse agonist
Positive	Toxic in 5 of 5 mice
Neurotoxic	Cured 1 of 5 mice
active at 30 mg/kg	Heavy Growth
Partial antagonist	Activator
activity at 300 mg/kg	Marginally Active
Weak	Positive: weak inducer based on presence of foamy macrophages and cytoplasmic vacuolations
Cytotoxic	Positive: strong inducer confirmed by electron microscopy
Susceptible	agonist
maximal vasorelaxant activity was observed at 500 uM	Inducer
Inhibition	Inhibitor
Blocked	active
Moderate	ATPase Activator: Y
Lethal	Calcein-AM Inhibitor: Y
Carcinogen	Note: corresponding IC50 reported as Active
Carcinogenic activity	inhibitor [Ki>1000uM]
carcinogenic activity	inhibitor [(at a rough estimate) 60 % of control]
weak activity	inhibitor [>80 % inhibition]
Slightly active	inhibitor [20% of control]
100% inhibition	inhibitor [30% of control]
Activity	inhibitor [40% of control]
Concentration effective	inhibitor [50% of control]
Complete inhibition	inhibitor [70% of control]
Decrease	inhibitor [80% of control]
moderate activity	inhibitor [IC50=10uM]
inhibition	inhibitor [IC50>50uM]
greater activity than compound 1	inhibitor [(at a rough estimate) <5 % of control]
weak	inhibitor [(at a rough estimate) 10 % of control]
moderate	inhibitor [(at a rough estimate) 15 % of control]
Stimulant	inhibitor [(at a rough estimate) 20 % of control]
Activity at 300 mg/kg	inhibitor [(at a rough estimate) 25 % of control]
Heart rate response was reported to reduce by 10-25% after 4 hr at a peroral dose of 1.0 mg/kg	inhibitor [(at a rough estimate) 30 % of control]
Phosphorylation	inhibitor [(at a rough estimate) 35 % of control]
Weak cytotoxic activity	inhibitor [(at a rough estimate) 40 % of control]
activity in 0-25% of administered animals	inhibitor [(at a rough estimate) 45 % of control]
Vasopressor response reversed	inhibitor [(at a rough estimate) 5 % of control]
Highly significant activity	inhibitor [(at a rough estimate) 50 % of control]
80- 90% lysis	inhibitor [(at a rough estimate) 55 % of control]
Yes	Phenotype: Inhibitor; Curve_Description: Partial curve; partial efficacy
substantial inhibition by 5-44 uM	Phenotype: Inhibitor; Curve_Description: Partial curve; partial efficacy; poor fit
antiseizure activity	Phenotype: Inhibitor; Curve_Description: Single point of activity
Toxic death	Phenotype: Inhibitor; Curve_Description: Complete curve; partial efficacy
Potent	Phenotype: Inhibitor; Curve_Description: Partial curve; high efficacy
activity	Phenotype: Inhibitor; Curve_Description: Complete curve; high efficacy
mild activity	yes
Inhibit	Highly active
GI increase	Weakly active
Significant	>50
Inhibited	Partial stimulant

Figure B.2: Active Dictionary.