

Alma Mater Studiorum - Università di Bologna

DOTTORATO DI RICERCA IN
SCIENZA E CULTURA DEL BENESSERE E DEGLI STILI DI VITA

Ciclo 35

Settore Concorsuale: 05/G1 - FARMACOLOGIA, FARMACOLOGIA CLINICA E
FARMACOGNOSIA

Settore Scientifico Disciplinare: BIO/14 - FARMACOLOGIA

ANTICANCER DRUG DISCOVERY USING ARTIFICIAL INTELLIGENCE: AN
APPLICATION IN PHARMACOLOGICAL ACTIVITY PREDICTION

Presentata da: Valentina Pellicioni

Coordinatore Dottorato

Carmela Fimognari

Supervisore

Carmela Fimognari

Co-supervisore

Gustavo Marfia

Esame finale anno 2023

Abstract

Hematological cancers are a heterogeneous family of diseases that can be divided into leukemias, lymphomas, and myelomas, often called "liquid tumors". Since they cannot be surgically removable, chemotherapy represents the mainstay of their treatment. However, it still faces several challenges like drug resistance and low response rate, and the need for new anticancer agents is compelling. The drug discovery process is long-term, costly, and prone to high failure rates. With the rapid expansion of biological and chemical "big data", some computational techniques such as machine learning tools have been increasingly employed to speed up and economize the whole process. Machine learning algorithms can create complex models with the aim to determine the biological activity of compounds against several targets, based on their chemical properties. These models are defined as multi-target Quantitative Structure-Activity Relationship (mt-QSAR) and can be used to virtually screen small and large chemical libraries for the identification of new molecules with anticancer activity.

The aim of my Ph.D. project was to employ machine learning techniques to build an mt-QSAR classification model for the prediction of cytotoxic drugs simultaneously active against 43 hematological cancer cell lines. For this purpose, first, I constructed a large and diversified dataset of molecules extracted from the ChEMBL database. Then, I compared the performance of different ML classification algorithms, until Random Forest was identified as the one returning the best predictions. Finally, I used different approaches to maximize the performance of the model, which achieved an accuracy of 88% by correctly classifying 93% of inactive molecules and 72% of active molecules in a validation set. This model was further applied to the virtual screening of a small dataset of molecules tested in our laboratory, where it showed 100% accuracy in correctly classifying all molecules. This result is confirmed by our previous *in vitro* experiments.

1. Introduction.....	6
1.1 Cancer.....	6
1.2 Hematological cancers.....	6
1.3 Chemotherapy.....	8
1.4 Problems in the use of anticancer drugs.....	13
2. The drug discovery process.....	14
2.1 Computer-aided drug discovery.....	17
2.1.1 QSAR models.....	18
2.2 ML.....	20
3. Aim of the study.....	24
4. Research methods and instruments.....	26
4.1 Data acquisition and dataset construction.....	26
4.1.1 Data curation.....	27
4.1.2 Cutoff value.....	28
4.1.3 Standardization.....	29
4.1.4 Molecular descriptors calculation.....	30
4.2 Model development.....	34
4.2.1 Dataset division.....	35
4.2.2 Box-Jenkins approach.....	36
4.3 Predictive model generation.....	39
4.3.1 ML algorithms.....	40
4.3.1.1 <i>k-NN</i>	40
4.3.1.2 <i>SVC</i>	41
4.3.1.3 <i>RF</i>	42
4.3.1.4 <i>GB</i>	43
4.3.1.5 <i>MLP</i>	43
4.4 Performance evaluation metrics.....	46
4.5 Dimensionality reduction.....	49
4.5.1 Principal Component Analysis.....	50
4.5.2 Linear Discriminant Analysis.....	51
4.5.3 Information theory-based feature selection.....	52
4.5.4 Genetic Algorithm- <i>k-NN</i> approach.....	53
4.6 Applicability Domain.....	55
5. Results.....	57
5.1 Dataset construction.....	57
5.2 Model development and optimization.....	60
5.3 VS.....	71
6. Conclusion and future perspectives.....	77
7. References.....	79

1. Introduction

1.1 Cancer

The term cancer, tumor, or neoplasm refers to a broad category of diseases that can affect any part of the body. More than 200 different cell types are present in the human body, and potentially each of them can become a cancer cell. Therefore, hundreds of cancers are known to date and are named according to the tissue, organ, or cell type where they originate. A fundamental characteristic of cancer is the rapid transformation, through a multi-stage process, of normal cells into cancer cells that grow beyond their limits and can invade parts of the body adjacent to the site of tumor formation and disseminate to other organs [1].

Cancer has a major social impact, it represents one of the leading causes of human morbidity and mortality, being second only to cardiovascular disease. In 2018, 18.1 million new cases of cancer occurred worldwide, and deaths attributable to it were 9.6 million. These statistics are expected to increase, estimated to occur annually by 2030 22 million new cancer cases and 13 million cancer-related deaths [2].

In addition, cancer has also an important economic impact on public health costs, ranking first in terms of global spending according to therapeutic class (\$91 billion in 2013) [3].

1.2 Hematological cancers

Hematological cancers are a heterogeneous group of malignancies so-called since originate from cells involved in the hematopoiesis process (Fig. 1), whereby the formation of blood cellular components takes place [4]. All blood cells arise from a common pluripotent hematopoietic stem cell, which can differentiate into a common myeloid precursor or a common lymphoid precursor. Erythrocytes, polymorphonuclear lymphocytes, monocytes, platelets, eosinophils, and basophils

originate from the common myeloid precursor [5], whereas B plasma cells, T cells, or natural killer cells originate from the common lymphoid precursor [5].

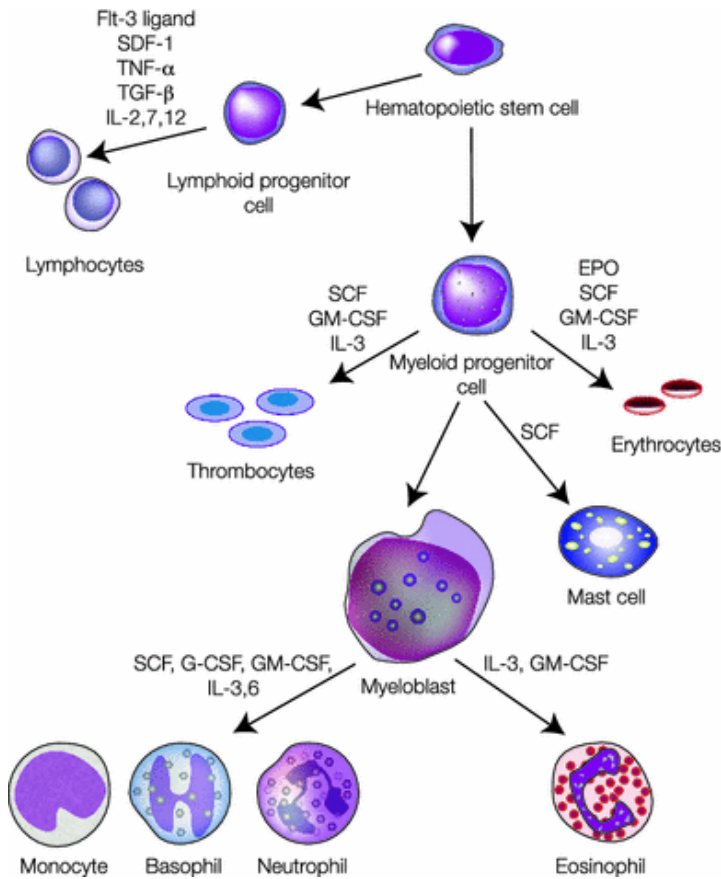


Figure 1. Schematic representation of the hematopoietic process [6].

Hematological cancers are divided according to the cell type of origin into leukemias, lymphomas, or myelomas, each encompassing several subtypes.

Leukemias arise as a result of abnormal proliferation of myeloid or lymphoid precursors and an aberrant accumulation in the bone marrow of cells called blasts.

Leukemias can be classified according to the primary cell line of origin, myeloid or lymphoid, and the disease onset modality and progression, acute or chronic [5].

Consequently, four main types of leukemia can be identified: acute lymphoblastic leukemia (ALL), chronic lymphoblastic leukemia (CLL), acute myeloid leukemia (AML), chronic myeloid leukemia (CML)

However, considering the less common forms of leukemia, hundreds of different types are known. [5].

Lymphomas arise in lymphoid cells of lymph nodes or other lymphoid tissues and can be divided into Hodgkin lymphoma (HL) and non-Hodgkin lymphoma (NHL). [7].

Although HL is the best-characterized lymphoma, it has a lower incidence than NHL. Macroscopically, HL can be divided into classical and non-classical types, and NHL into B-cell, T-cell, and natural killer cell types. In addition, lymphomas can be classified for clinical purposes according to their severity into high grade if they are aggressive, or low grade if they are indolent. [7].

Myelomas comprise several conditions resulting from aberrant proliferation in the bone marrow of plasma cells that produce a paraprotein, i.e. a single species of an immunoglobulin molecule, therefore defined as monoclonal, named M-protein.

Myelomas embrace conditions of benign and malignant nature, the latter including multiple myelomas, Waldenström's macroglobulinemia, plasma cell leukemia, and plasmacytomas [4].

Effective cancer treatment requires the elimination of all cancer cells, whether the tumor is confined to the primary site or if metastases are present in other regions of the body. For solid tumors, the main anticancer therapies are surgery and radiotherapy when the tumor is circumscribed, and chemotherapy, especially when the tumor has disseminated outside the primary site. Hematological cancers are often described as liquid tumors because they do not form nodules or masses surgically removable, in contrast to solid tumors. For this reason, chemotherapy represents the centerpiece of hematological cancer treatment [4].

1.3 Chemotherapy

Chemotherapy is the most widely used therapeutic approach in the treatment of solid and hematological tumors and can be administered alone or in combination with other forms of therapy, such as surgery or radiotherapy. Combination therapy aims to reduce the risk of relapses. Conventional antineoplastic drugs are cytotoxic agents that induce cell damage and cause tumor cell death through a direct action on the genome or through the interference with the replicative processes of the proliferating cell [8]. No drug, among those available, is devoid of toxicity. All are characterized by a low

therapeutic index. Therefore, the selection of a drug is subject to a careful evaluation of the risk/benefit assessment [9]. Cytotoxic agents belong to different therapeutic classes and are classified according to their mechanism of action.

DNA damaging agents

- Polyfunctional alkylating agents, such as cisplatin and busulfan, covalently bind DNA by transferring an alkyl group [8]. This binding requires metabolic activation of the drug, which involves the formation of an electrophilic group (carbocation), interacting with electron-dense species such as sulphhydryls, amines, phosphates, and other cellular nucleophiles. Alkylating agents preferentially bind the guanine 7 nitrogen of a DNA strand; if they possess two reactive groups, they simultaneously attack both DNA strands (cross-linking) [8]. They exert their maximal effect in replicating cells, when the DNA is partially unwound and more accessible. The interaction occurs during the S-phase, blocking cells in the G2 phase [10].

- Anti-cancer antibiotics, such as bleomycin and mitomycin, bind to DNA by intercalating between bases and preventing the synthesis of RNA, DNA, or both nucleic acids. This results in the fragmentation of one or both strands and interference with cell replication [10].

Antimetabolites

Antimetabolites such as methotrexate, 6-mercaptopurine, and 5-fluorouracil, interfere with metabolic processes essential for cancer cells' survival and proliferation, such as the synthesis of nucleotides or nucleic acids, and thus with DNA replication[8]. They are synthetic molecules structurally analogous to physiological metabolites and act as false enzyme substrates (purine and pyrimidine analogues) or as reversible or irreversible inhibitors of key enzymes (folic acid antagonists, inhibitors of the enzyme dihydrofolate reductase) [8].

Antimitotics

- Microtubule-targeting drugs alter the functionality of the mitotic spindle by binding to specific sites on tubulin- β and preventing the polymerization and rupture of microtubules, such as vincristine, or hindering their depolymerization, such as paclitaxel. This blocks the cell cycle in metaphase and triggers apoptosis [8].

- Topoisomerase inhibitors inhibit the catalytic activity of topoisomerase I and II enzymes. They form a ternary complex with the DNA and the enzyme, stabilizing the 'cleavage complex'. The cleaved DNA cannot be repaired and the fragmented strands trigger the apoptotic process. Topoisomerase I inhibitors, such as irinotecan, generate single-strand breaks that are not repaired; the cell cycle proceeds, and the damaged DNA 'collides' with the replication fork, producing double-strand breaks that arrest the cell cycle in the G2 phase. Topoisomerase II inhibitors, e.g. anthracyclines, can intercalate between DNA base pairs and prevent nucleic acid synthesis by inhibiting transcription and replication processes. The action is cell cycle-specific and directed at proliferating tumor cells [10].

Targeted therapy

Targeted therapy arises from the increased understanding of the molecular pathobiological basis of tumors. Many oncogenes and oncosuppressor genes produce oncoproteins that are involved in signal transduction pathways and induce alterations promoting cellular proliferation, inhibition of apoptosis, invasiveness, and metastasis [11].

The most important family of oncoproteins activating oncogenic signal transduction are tyrosine kinases (TKs), which are located in the cytoplasm and can be classified into TKs, serine-threonine kinases, and tyrosine-threonine kinases. TKs are further subdivided into nonreceptor tyrosine kinases (NRTKs), such as ABL, FES, JAK, ACK, SYK, TEC, FAK, SRC, and CSK families [12,13], and receptor TKs (RTKs). RTKs include TKs associated with receptors for epidermal growth factor (EGF), platelet-derived growth factor (PDGF), vascular endothelial growth factor (VEGF), hepatocyte growth factor, insulin-like growth factor (IGF-1) [14], stem cell growth factor (SCF or c-Kit) [15]. NRTKs

and RTKs are currently the most important targets of innovative anticancer drugs [16]. Pharmacological intervention consists of chemical compounds that penetrate the cytoplasmic membrane and inhibit NRTKs and RTKs or monoclonal antibodies directed against RTKs or their receptors.

- Signal transduction inhibitors

The first selective TKs inhibitor approved for clinic use was imatinib. Imatinib acts by competitively inhibiting ATP binding to its binding site in the TK domain and blocking cell proliferation [17]. This drug has been shown to be a potent inhibitor of Bcr-Abl, constitutively present in leukemias expressing the Philadelphia chromosome. The Philadelphia chromosome is a shortened chromosome 22 resulting from a reciprocal translocation of parts of chromosomes 22 and 9, which replaces the first exon of ABL gene with sequences of the BCR gene, producing the BCR-ABL oncogene. This oncogene expresses an enzyme that has a constitutive, abnormal tyrosine kinase activity. The chromosome is the hallmark of CML and is found in more than 95% of CML patients [18]. Accordingly, imatinib currently represent the first line treatment for CML [19]. Imatinib is also capable of binding PDGF and c-Kit [20]. Despite its potent activity, tumors treated with imatinib can develop resistance, mainly induced by mutations at Bcr-Abl kinase, PDGF receptor, and c-Kit [21]. The occurrence of these mutations led to the development of new TKs inhibitors, such as dasatinib and nilotinib, which can also be used to treat tumors with mutated forms of Bcr-Abl [22]. Other TKs inhibitors are gefitinib, erlotinib, and lapatinib, which inhibit the EGF receptor, and sunitinib and sorafenib, which inhibit the VEGF receptor [23]. Since the VEGF receptor has important functions in the regulation of angiogenesis, sunitinib and sorafenib inhibit angiogenesis and tumor growth. Also belonging to the signal transduction inhibitor drugs are the serine/threonine kinase mTor (mammalian receptor of rapamycin) inhibitors, temsirolimus and everolimus, and the proteasome inhibitor bortezomib [10].

- Monoclonal antibodies

Monoclonal antibodies (mAbs) are humanized or chimeric antibodies produced by clones of a unique B cell, which can selectively bind specific target either unique to or overexpressed by cancer cells, such as RTKs and their receptors [24]. Their therapeutic effect may depend on several mechanisms. However, it is believed that mAb can act through more than one of the following mechanisms: preventing ligand-receptor interaction by binding to the ligand or to the receptor, disrupting receptor internalization, promoting receptor internalization or release of the extracellular portion of the receptor, blocking receptor dimerization and activation, and inducing apoptosis [21]. The induction of apoptosis can occur through several mechanisms, such as complement-dependent cytotoxicity (CDC), antibody-dependent cell cytotoxicity (ADCC), antibody-dependent cell phagocytosis (ADCP) or by altering the signal transduction of cancer cells [25]. MAbs are advantageous since they allow to both directly kill tumor cells while simultaneously develop long-lasting immune responses against the tumor [24]. The first mAbs approved was rituximab in 1997 for the treatment of NHL. Rituximab binds to the surface antigen CD20, a protein overexpressed on cancerous B cells of NHL, but absent on healthy immature B cells, [24]. Since the approval of rituximab, more than 30 cytotoxic antibodies targeting antigens identified as overexpressed on tumor cells have entered clinical development [25]. Trastuzumab is an mAbs targeting Human Epidermal Grow factor 2 (HER2), a transmembrane receptor of the EGFR family, resulting in inhibition of cell growth. HER2 is overexpressed in 20% of breast cancer [26] and is associated with aggressive tumors. Cetuximab and panitumumab are antibodies targeting EGFR, involved in the processes of tumor cell proliferation, migration, and invasion and overexpressed in many cancers. These two mAbs block ligand binding and receptor dimerization, inducing apoptosis in cancer cells [24]. Bevacizumab targets VEGF, which is found overexpressed in many types of human cancers. Bevacizumab acts by preventing VEGF from interacting with its receptor. Recently, the approach involving the use of mAb is no longer based solely on targeting tumor antigens, but also includes stimulating T cell antitumor immunity. For this purpose, bispecific T Cell Engager (BiTE) antibodies have been developed that both target a tumor antigen and activate receptor on T cells [24]. BiTEs directly target tumor cells and at the same time recruit cytotoxic T cells into the tumor microenvironment. These drugs have been shown to be highly effective in

inducing tumor regression and can be administered at doses three orders of magnitude lower than classical mAbs. One of these mAbs is blinatumomab, which was approved in 2017 by the FDA for the treatment of ALL [24].

1.4 Problems in the use of anticancer drugs

The main issue of chemotherapy is still toxicity: cytotoxic drugs are not selective for cancer cells but affect also normal tissues highly proliferating, such as mucous membranes and bone marrow [8]. For many drugs, there is a need to establish dose limits that must not be exceeded because they are associated with an unacceptable risk of toxicity. The toxicity of chemotherapeutic drugs can be acute or chronic. Acute toxic reactions, such as gastrointestinal toxicity, myelotoxicity, and alopecia, may be frequently observed but are often reversible and can be managed with the administration of antiemetic drugs, bone marrow growth factors, and hydration. Chronic toxic reactions occur late as a result of the cumulative effects of multiple administrations. These reactions are the most dangerous since they are irreversible or only partially reversible [8].

Another frequent consequence that can occur using chemotherapeutic agents with a wide range of action, is the development of secondary malignancies, despite the treatment of primary cancer. Moreover, it may happen that some neoplastic cells can survive cancer chemotherapy [8]. Resistance is another important limitation to the therapeutic efficacy of chemotherapeutic drugs. To date, resistance to anticancer therapy can occur through many molecular mechanisms, such as decreased drug uptake, increased drug inactivation, alterations in drug targets, increased ability to repair DNA, and cell death escape. However, changes in stroma and tumor microenvironment and local immunity can also contribute to the development of resistance. These molecular mechanisms of resistance are the result of somatic mutations that make cancer cells less sensitive to the drug. Given the ability to develop different mutations, cancers consist of heterogeneous cells that give them the possibility of multiple resistance to different drugs. Consequently, the ideal treatment should be based on a combination of different drugs (polychemotherapy) that are not cross-resistant [8].

2. The drug discovery process

The introduction of a new drug into the market may essentially begin in three ways: serendipity, the me-too method, and drug discovery.

Serendipity refers to the accidental discovery of a drug, such as the observation of a drug biological activity when that specific activity was not being investigated. An example is the discovery of sildenafil (Viagra), a drug developed for the treatment of angina that was instead approved for the treatment of erectile dysfunction and was later shown to be cytotoxic to cancer cells [27].

Conversely, me-too is a method aimed at producing compounds with the same mechanism of action and structurally related to first-in-class compounds but chemically different enough to assure a more favorable pharmacokinetic and pharmacodynamic profile [28]. For years, this method has been the most used by pharmaceutical companies.

Lastly, drug discovery is the most rational among these methods. Indeed, it relies on solid knowledge of the cellular and molecular events that characterize a disease to create or find new drugs capable of modulating the altered processes underlying the disease [29].

The drug discovery process is very long and complex. From more than 10000 molecules that enter the first preclinical phases, only 1 or 2 reach the clinical trials and even less the marketing authorization [30].

It consists of several steps which aim to guarantee drug efficacy and safety. This is the reason why it is time-consuming and expensive.

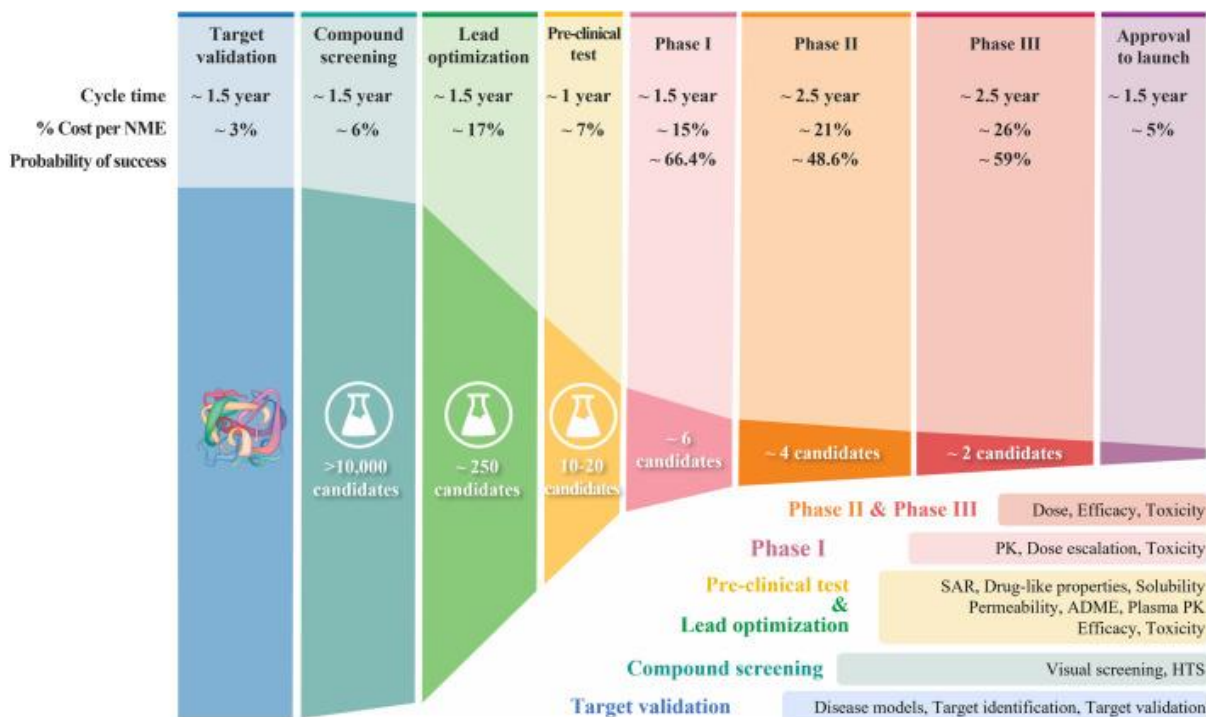


Figure 2. The various stages of the drug discovery process. For each stage, the time required to complete the stage and the success rate of drug candidates are illustrated [30].

The drug discovery process can be divided into two main stages: preclinical (stages one, two, three, and four) where molecules are tested *in vitro* on cell lines or *in vivo* on animals, and clinical (stages five, six, seven and eight) where molecules are tested on humans (Fig. 2).

The first stage of drug discovery regards target identification. Here, the target on which the drug in development should act to fight the disease is decided.

The second stage is target validation, whereby an assay is developed to test if a small molecule interacts with the designated target. The assay is conducted on a large scale by high throughput screening (HTS). HTS is a brute-force automated technology-driven approach that employs miniaturized assay systems and robotics to test large amounts of small molecules against single or multiple targets in a limited time. HTS aims to identify large subsets of molecules capable of eliciting the desired activity, known as hits, from large collections of compounds (10^5 - 10^6) [31]. Since the number of small molecules tested can range from thousands to millions, the volume of experimental data produced by HTS can be enormous.

The third stage is the lead generation, also called the hit-to-lead phase. Leads are drug-like molecules that are derived from hits after optimization at the end of which they have acceptable specificity, affinity, and selectivity for the target.

The fourth stage is the lead optimization. Here, drug candidates are generated by optimizing the lead structure through chemical modifications [32]. At the end of this stage, the lead candidate should possess a better pharmacokinetic and pharmacodynamic profile, and a lower toxicity than the starting lead.

Stages five, six, seven, and eight represent clinical trials. Phase I trial is conducted on a few healthy volunteers (10-100) to assess the safety and tolerability of the candidate drug. Phase II trial is conducted on patients (50-500) to assess for the first time clinical efficacy. Furthermore, dosage regimens are determined and safety confirmed. Phase III clinical trials evaluate the treatment on a large scale, with thousands of patients, to confirm the efficacy and tolerability of the drug. Phase IV trials begin after approval and commercialization of the drug. The aim of this phase, also called post-marketing surveillance, is to detect rare or long-term adverse effects, and drug-drug or drug-disease interactions that could not be found during the early stages of clinical trials [32].

Notwithstanding impressive technological advances and further understanding of cancer biology, the discovery and development of new anticancer drugs is still a challenging, time-consuming, and expensive process: the average time to develop a new anticancer drug is 7.3 years (range: 5.8-15.2 years) and the average cost is \$648.0 million (range: \$157.3 million to \$1950.8 million) [3]. In addition to time and cost issues, anticancer drugs are subject to high failure rates. Indeed, their success rate in clinical development is less than 10% [33]. The reasons for this failure are complex and can involve every stage of the drug discovery process. The identification of a specific target for a lead compound remains a challenge. Due to the limited and incomplete knowledge of cancer-related proteins involved in the development of human malignancies, the identification of validated anticancer drug targets is difficult [34]. Target identification could be the result of intensive HTS, which is mainly hypothesis driven. However, it is difficult to predict whether the interaction between the proposed target and the lead compound will induce the expected therapeutic effect

before investing in a complete drug-discovery program, and this could result in failure in the later stages of drug discovery [34]. Another key challenge for preclinical development of anticancer drugs is the limited ability of *in vitro* and *in vivo* models to mimic all the characteristics of a patient's tumor [34]. Indeed, HTS assays are cell based and most of them are performed on bi-dimensional (2D) cell cultures. 2D cell cultures fail to reproduce tumor complexity and thus to predict drug efficacy. Therefore, compounds with clear antitumor activity in 2D cell models often demonstrate lack of activity once the drug is tested on *in vivo* models or in clinical trials [35].

Another key point in drug discovery failure is the screening of new anticancer agents on rodents. Mice are the most commonly used animal models; however, they have a limited ability to accurately mimic most human diseases. Therefore, safety and efficacy identified in animal models fail to become translated through clinical trials [36]. For the reasons outlined so far, the adoption of approaches making the drug-discovery process more rational, rapid, and cost-effective is crucial.

2.1 Computer-aided drug discovery

With the development of computer science, it has become increasingly popular and simple to apply computational techniques to the field of chemistry, leading to the coining of the term chemoinformatics. Computer-aided drug discovery (CADD) or *in silico* drug discovery is a set of techniques that can be utilized to optimize all phases of the drug discovery process.

Specifically, through CADD it is possible to perform hit-to-lead selection, optimize the absorption, distribution, metabolism, excretion, and toxicity profile, and limit safety issues [37]. However, these stages are computationally performed and candidate compounds must be subjected to *in vitro/in vivo* experiments for confirmation [38]. CADD tools are increasingly appreciated because they can reduce the number of small molecules to be experimentally evaluated while increasing the success rate by early detection of inactive or potentially toxic compounds [38]. Several approved drugs were discovered using CADD. Some examples include saquinavir, indinavir, ritonavir (HIV

protease inhibitors) [39] and tirofiban (fibrinogen receptor antagonist) [40]. CADD methods can be broadly classified as structure-based (SBDD) or ligand-based (LBDD). SBDD technique is based on knowledge of the 3D structure of the target obtained by X-ray crystallography or NMR spectroscopy. SBDD techniques are used to identify or draw a ligand based on a specific target. The most used is molecular docking, where molecules can be drawn based on complementarity with the structure of the target protein.

In contrast, LBDD is based on the similarity between ligands. Ligand-based techniques are used when the target is unknown or when it is known but its 3D structure is not available. Prior knowledge of active drugs, such as structural, physical, and chemical properties is employed in LBDD methods to predict new molecules with similar biological effects. Part of the LBDD methods is pharmacophore modeling, similarity search, and quantitative structure-activity relationship (QSAR) models [41].

2.1.1 QSAR models

First suggested by Hansch and Fujita in 1964, QSAR modeling is a method based on the assumption that the chemical structure of compounds determines their physical, chemical, and biological properties. This theory has two consequences: 1) it is possible to describe the mathematical relationship between molecular structure and a specific property (bioactivity, toxicity, etc.) of a set of molecules, and 2) different molecular structures have different chemical properties while similar molecular structures have similar molecular properties [42].

QSAR modeling relies on data analysis and statistical methods to build models that could predict biological activities or chemical properties of unknown compounds based on their molecular structure [43]. In order to build a QSAR model, it is necessary to represent the properties of a molecule in a certain way. This can be done using molecular descriptors, which are quantities that numerically quantify molecular properties. The objective of QSAR models is to find a trend in the descriptor values that explain the trend in biological activity [43]. Initially, molecular descriptors were determined experimentally, so they were few in number and encoded chemical-

physical properties. As knowledge and computational methods have progressed, hundreds of molecular descriptors capable of encoding even the biological properties of molecules have been calculated using algorithms [44]. Molecular descriptors can be classified according to the dimensionality of molecular representation required to depict them as 0-dimensional (0D), 1-dimensional (1D), 2-dimensional (2D), 3-dimensional (3D), and 4-dimensional (4D) (they are described in detail in section 4.1.4). Accordingly, the resulting type of QSAR model will be called 0D, 1D, 2D, 3D, or 4D [45].

Two different types of QSAR models exist: regression models and classification models. Regression models aim to estimate the relationship between a set of independent variables (X, the molecular descriptors) and the dependent variable (Y, the outcome), to predict a continuous value, i.e. numeric value that have an infinite number of values between any two values. Classification models define the relationship between independent variables (X) and the discrete dependent variable (Y) to predict the class (or label) for a given input data [46].

QSAR models have been employed in drug discovery for a long time. However, in the early days of QSAR modeling, experimental data to construct a model, the database, were difficult to obtain. Thus, QSAR models were generated from small and congeneric sets of molecules, active against a single target in order to investigate and improve their chemical and physical properties through mechanistic interpretation.

Consequently, the statistical methods used to create these models were simple and unsophisticated [47]. However, the increasing use of HTS in drug discovery has resulted in an explosion of experimental biological and chemical "big data," which have been digitized and are freely available in public online datasets, such as ChEMBL, Pubchem, ZINC, etc., some of which are updated daily. The availability of large volumes of data makes it easy to create extremely complex datasets that can be used in QSAR modeling. However, to handle such complexity, QSAR models abandoned simple statistical methods and adopted computational tools that include more complex and sophisticated statistical methods, such as machine learning (ML) algorithms.

2.2 ML

ML is a branch of artificial intelligence that uses computational algorithms to parse data and learn complex patterns from them, with the aim to make a determination or prediction on a new dataset without being directly programmed [48]. For this reason, ML is applied to solve problems for which a large amount of data and variables are available but the way to relate them is unknown. ML techniques have been introduced in the field of drug discovery where they have the opportunity to be applied in every stage of the process [48].

ML algorithms can be divided into three major types: supervised, unsupervised, and reinforcement learning [49].

In supervised learning methods, data are provided to the model in the form of labeled inputs with the respective desired outputs. Supervised algorithms aim to extract a general rule that associates the input with the correct output. Depending on the independent variables, supervised learning can be used to solve two types of problems: classification or regression. In classification, the outputs are divided into two (binary classification) or more (multi-class classification) classes and the learning algorithm must create a model that assigns unseen inputs to one or more classes. In regression, the model aims to predict a continuous value. Supervised learning algorithms, such as Random Forest (RF), K-Nearest Neighbors (K-NN), Gradient Boosting (GB), and Multilayer Perceptron (MLP) are typically employed for classification or regression problems [49].

Unsupervised learning methods identify hidden patterns or intrinsic structures among unknown input data that allow clustering them according to common characteristics not specified by the user. Unsupervised algorithms, such as clustering algorithms, association rules, or dimensionality reduction are used for explanatory purposes [48].

Reinforcement learning is a hybrid of the previous approaches. Here the model interacts with a dynamic environment in which it tries to achieve a goal without the intervention of a supervisor to guide it. Thus, the model learns by trying and failing [50].

Moreover, a further approach can be placed between supervised and unsupervised learning: semi-supervised learning. Here, an incomplete dataset is provided to train the model, where only for part of the data the desired output is specified [51].

The above-mentioned methods vary in the task they can perform, computational speed, and number of variables they can handle. It is important to choose the appropriate algorithm suitable for the problem and the amount and type of data available to construct reliable ML models. [48]. Despite the variety, supervised learning algorithms are commonly employed in the field of drug discovery, and have also been employed to construct QSAR models [52].

Since a QSAR classification model has been built in this study, the following explains how the construction of an ML-based QSAR model for classification is accomplished.

A QSAR classification model is generally outlined in four parts. First, a dataset of molecules labeled as active or inactive is created and molecular descriptors are generated. Second, different ML models are constructed to establish the relationship between the descriptors and the biological activity of interest. Third, models are validated and their predictive performance is compared to select those that show the best ones. Finally, the model is applied to an external data set to verify the ability to correctly classify new samples.

Once this model has been created and validated, it can be used to search for new drugs with the desired activity in small or large chemical databases. This procedure can be seen as a computational method to perform HTS, since both find new hits, and it is called virtual screening (VS) [53]. In particular, searching for new compounds with similar activity to known molecules is called ligand based VS [54]. Although HTS and VS have the same purpose, the philosophy behind their approach is different [55]. Actually, HTS aims to test every single compound of a large collection using an automated plate-based experimental assay. On the other hand, VS is a computational knowledge-based approach that rationalizes the identification of new compounds with desired activity to reduce the number of drug candidates to be tested experimentally. VS based on QSAR models allows predicting the biological activity of compounds not yet synthesized, and this offers substantial advantages. Since the time to perform VS of

large databases of molecules is much less than that which would be required to synthesize and test them, VS allows the drug discovery process to be speeded up [56]. In addition, since laboratory instruments, chemical reagents, and biological materials are not employed, VS economizes the drug discovery process [56].

As already anticipated, ML algorithms enable QSAR models to solve increasingly complex problems. As one of the most advantageous properties of these powerful algorithms is their high predictive capacity. Initially, QSAR models were able to handle a small homogeneous dataset of molecules with a single target, using relatively simple algorithms to identify only linear relationships between variables. With the adoption of ML algorithms, QSAR models became capable of handling complex datasets, capturing non-linear relationships between molecular descriptors and biological properties. A complex dataset could include thousands of molecules tested against multiple targets. QSAR model that can predict the activity of molecules against multiple targets is defined as multi-target QSAR (mt-QSAR) [57].

Mt-QSAR models have been successfully applied in anticancer drug discovery relying on two main approaches [58]. The first approach performs large-scale prediction of growth inhibition patterns employing hundreds to thousands of molecules against tens to hundreds of cancer cell lines belonging to many different cancers [59–65]. This complex problem requires either chemical information and/or cell line profiling data.

The second approach is based on mt-QSAR models in which the multiple targets are represented by different cell lines belonging to the same type of cancer [58]. This approach is leading to the identification of new compounds active against one type of cancer. Consequently, new drugs identified on the basis of these mt-QSAR models could prove to be highly active against a tumor type. Several of these models have been developed since the 2010s to predict through VS and/or to allow the design of compounds active against several cancer cell lines belonging to connective tissue [66], prostate [67], breast [68], brain [69], colorectal tract [70], and bladder [71].

These above-mentioned models rely on ML algorithms and statistical approaches that make it possible to perform transformations of the molecular descriptors so that they can encode the biological activity of molecules not only on the basis of their molecular

structures but also in relation to the target for which they have been found to be active. One of these approaches, the Box-Jenkins moving average approach, was used in the present study and is discussed in detail in section 4.2.2.

3. Aim of the study

The extremely expensive cost and time of drug discovery have increased the need for methods to identify new anticancer drugs more rapidly and inexpensively. Thanks to the development of computer science and the rapid development of biological and chemical “big data” obtained from HTS, ML methods emerged as effective tools for all phases of drug discovery to speed up and economize the process. One of the most interesting applications of ML is in QSAR modeling, allowing the creation of predictive models based on complex libraries of compounds tested against multiple targets. Such mt-QSAR models can correlate the chemical structure of compounds dataset with a complex biological endpoint such as cytotoxicity against cancer cells. By using them as VS technique, it is possible to identify whether new unknown molecules are cytotoxic towards cancer cells based on their structural similarity to the compounds dataset that generated the model.

The aim of my Ph.D. project was to employ ML techniques to generate and optimize an mt-QSAR classification model that can be used for VS purposes and to identify new drugs potentially effective against several cancer cell lines belonging to leukemias, lymphomas, and myelomas. This model could represent the first mt-QSAR model for the prediction of molecules active against several hematological cancer cell lines. Indeed, to the best of our knowledge, there are no mt-QSAR models for the prediction of cytotoxic compounds for hematological cancers reported in the literature.

To accomplish this aim, I built a dataset of cytotoxic molecules tested on 43 different hematological cancer cell lines, extracted from the database ChEMBL. In order to describe the activity of the molecules, molecular descriptors were calculated for each molecule. Then, a transformation of the molecular descriptors was operated through the Box-Jenkins approach to enable the model to discriminate the activity of molecules according to their target.

Furthermore, I applied different ML classification algorithms to generate a predictive model, including RF, k-NN, Support Vector Classifier (SVC), GB, and MLP.

Once the mt-QSAR model has been built, I investigated different approaches to enhance the predictive ability, among which dimensionality reduction techniques and the variation of some properties of the dataset such as the cutoff value and the number of experimental conditions.

Finally, in order to validate the reliability of the model, I applied the best predictive model for the VS of a small dataset of molecules tested in our laboratory, to compare the outcome with the experimentally obtained results.

4. Research methods and instruments

4.1 Data Acquisition and Dataset Construction

The chemical and biological data necessary for the construction of the dataset were retrieved from two public online databases: Cellosaurus and ChEMBL.

Cellosaurus [72] is a knowledge resource on cell lines available on the ExpASY server (<https://web.expasy.org/cellosaurus/>) used to search for leukemia, lymphoma and myeloma cell lines employed in biomedical research. This search resulted in a list of 71 hematological cancer cell lines. Subsequently, the ChEMBL database (<https://www.ebi.ac.uk/chembl/db>) was sought for molecules with cytotoxic activity assayed against those cell lines. ChEMBL is a publicly available database of compounds and bioactivities from multiple sources, curated by the European Molecular Biology Laboratory – European Bioinformatics Institute (EMBL-EBI). ChEMBL version 31 contains more than 19 million bioactivity values, over 2.3 million unique compounds, and 15000 targets. The majority of the information is retrieved from more than 85000 scientific publications, but also both nonprofit and commercial organizations' deposited data sets [73]. The research was performed by inserting the name of each cell line found on Cellosaurus as a target in the ChEMBL query toolbar. When information on molecules active against that target is available, a section called 'Activity Charts' appears in the menu. Here the molecules are listed with their bioactivity indices (IC_{50} , GI_{50} , EC_{50} , CC_{50} , etc.). Since cytotoxic activity is usually reported in terms of IC_{50} (concentration capable of inhibiting 50% of cell viability), only molecules with a corresponding IC_{50} value were selected and downloaded in comma-separated values (.csv) format. This resulted in 71 separate datasets, each comprising a single cell line, their respective bioactive compounds, and information.

Utilizing Python version 3.9, an open-source programming language popular for scientific computing, and Jupyter Notebook (<https://docs.anaconda.com/ae-notebooks/user-guide/basic-tasks/apps/jupyter/index.html>), a Python-based web interactive computing platform for big data processing and ML, the 71 individual datasets were fused to build a single one consisting of 66787 molecules and 19 columns containing information about each molecule.

Subsequently, the number of columns was reduced to 10, retaining only information essential to finalize the construction of the dataset. The dataset can be visualized as a matrix of rows and columns. The columns contain information about the molecules, and the rows represent each molecule with all the information about it and are defined as samples. The columns of this initial dataset are listed below and a brief description is provided for each one.

- Molecule ChEMBL ID: is a univocal alphanumeric code used to identify molecules within the ChEMBL database.
- SMILES: Simplified Molecular Input Line Entry System (SMILES) format, is the 2D notation through which the structure of the molecules is represented [74].
- Standard Type: is the index indicating the bioactivity of the molecule, such as IC₅₀.
- Standard Relation: mathematical symbol determining the relationship between Standard Type and Standard Value, such as “=, <, >” etc.
- Standard Value: numerical value corresponding to the concentration of substance required to reach the IC₅₀.
- Standard Units: the concentration unit
- Assay Description: text string describing the conditions of the assay which determined the bioactivity of the molecule
- Assay Cell Type: the tumor cell line on which the molecule was assayed
- Target Name: target name of the molecule
- Target Type: target type of the molecule

On the dataset thus composed, the data curation process was undertaken.

4.1.1 Data Curation

The successful predictive performance of a ML model depends on at least 80% of data quality and 20% of the algorithm applied [48]. Therefore, a higher volume of data does not always correspond to better predictions. To achieve this, the data must be high

quality, meaning it must be curated to be as much accurate and complete as possible. An ideal training set should contain data systematically generated, that are complete and contain good annotations [75]. In real cases, very often happens that data are obtained from multiple sources and are highly variable and with inhomogeneous quality. The dataset used in my research project belongs to the latter case. As the data reported on ChEMBL are manually extracted from the scientific literature, they could be prone to the aforementioned issues. Therefore, before applying any ML algorithm, it is necessary to pre-process the data and maximize its usability. This process is called data curation and represents a crucial step in dataset construction.

There are no absolute rules for increasing the data quality of a dataset; the actions performed for data curation depend mainly on the size of the dataset and the type of information needed to solve the problem, which should be preserved as much as possible.

Analyzing the dataset and checking the values in each column revealed several data homogeneity issues, they are described in detail in section 5.1.

Considering the large scale of the dataset and that any information contained in the columns at this stage of construction is essential, samples that had non-compliant values were eliminated. Despite the vast size of the dataset, the elimination of samples required a brief amount of time; in fact, by coding in Python on Jupyter Notebook it was possible to filter the dataset according to specific search keys and remove samples that did not satisfy or had the wrong values.

4.1.2 Cutoff value

Once the samples composing the dataset were defined, the next step was to determine a criterion that would allow the model to classify the molecules, therefore termed the cutoff value. That criterion was identified in the activity value contained in the Standard Value column, allowing molecules to be discriminated based on their cytotoxicity. Since there is no consensus regarding the optimal value to be used as a cutoff value in drug discovery problems, during the model development step this was chosen arbitrarily corresponding to 1 μM . In this way, all molecules with an activity

value less than or equal to 1 μM were considered active, while those with an activity value higher than 1 μM were considered inactive. To make the classification task easier for the algorithm that will create the predictive model, the molecules were assigned a Boolean variable, i.e. a variable that can take only two values, in this case, 1 for active molecules, and -1 for inactive ones. In the case of two categories, the classification problem is called binary classification or two-class classification. This procedure was carried out in Excel, where the dataset was sorted according to increasing Standard Values, a new column called Toxicity was created, and the appropriate Boolean variables were inserted for each molecule.

A cutoff value of 1 μM was considered advantageous because once the model is built, validated, and used for VS, it can allow new active molecules to be identified at submicromolar or at most micromolar concentrations. Thus, this approach could allow very early detection of potentially highly active drugs.

4.1.3 Standardization

Chemical structures used for ML problems in drug discovery can be represented in multiple ways [76]. Consequently, when a model has to be generated with a dataset containing thousands of molecules, they should be standardized. Standardizing chemical structures is about making the molecular representations uniform, clean, and comparable, improving the quality of the data.

In addition, the standardization process is crucial for the good calculation of molecular descriptors, which is based on the assumption that the molecular structure to which the mathematical algorithms are applied is correct [77]. Incomplete chemical structures, the presence of secondary molecules such as solvents or salts, the presence of charges, etc. can prevent or obstacle the calculation of certain molecular descriptors. Therefore, standardization is a mandatory process for constructing a dataset in the context of QSAR modeling.

In my project, the structures of the compounds underwent a double standardization process, first using ChemAxon Standardiser version 21.2.0 and then Open Babel, both free software dedicated to this purpose [78]. To do so, only the Molecule ID and

SMILES columns were selected from the dataset and transferred to a new file to be submitted to the software. The double standardization made it possible to transform implicit hydrogen atoms into explicit ones, calculate the two-dimensional atom coordinate, calculate the three-dimensional atom coordinate, represent all aromatic rings in the same way, neutralize charged molecules, remove fragments of the molecule by keeping only the largest one if the chemical structure was multi-fragment, and remove predefined solvent and salt fragments from multi-fragment molecules. At the end of the standardization process, the molecules were converted into structure data files (SD files). SD files encode chemical structure data using the molfile connection table format, which represents chemical structures using a block of text listing atoms, bonds, connectivity, and coordinates for the 3 dimensions of space: x, y, and z. Chemical software for the calculation of molecular descriptors can interpret SD file format and translate the content into a graphic chemical structure and data table [79]. Through this conversion, molecules went from being represented through the SMILES format that encodes 2D information to the SD files format that can also encode 3D information.

4.1.4 Molecular descriptors calculation

ML algorithms need to be fed with numerical inputs. Thus, once the data has been curated and standardized, the structural information included in the chemical files needs to be converted into numerical values that can be used as input for model building [77]. Since the inputs are the information that the ML algorithm employs to train the model, they must be informative numbers, not simply arbitrary or sequential. Many different numerical representations have been proposed to represent the molecules and are overall defined molecular descriptors. In this perspective, molecular descriptors are the meeting point between ML algorithms and QSAR modeling. Molecular descriptors can be classified based on the level of molecular representation required to represent them in 0D, 1D, 2D, 3D, 4D (Fig. 3).

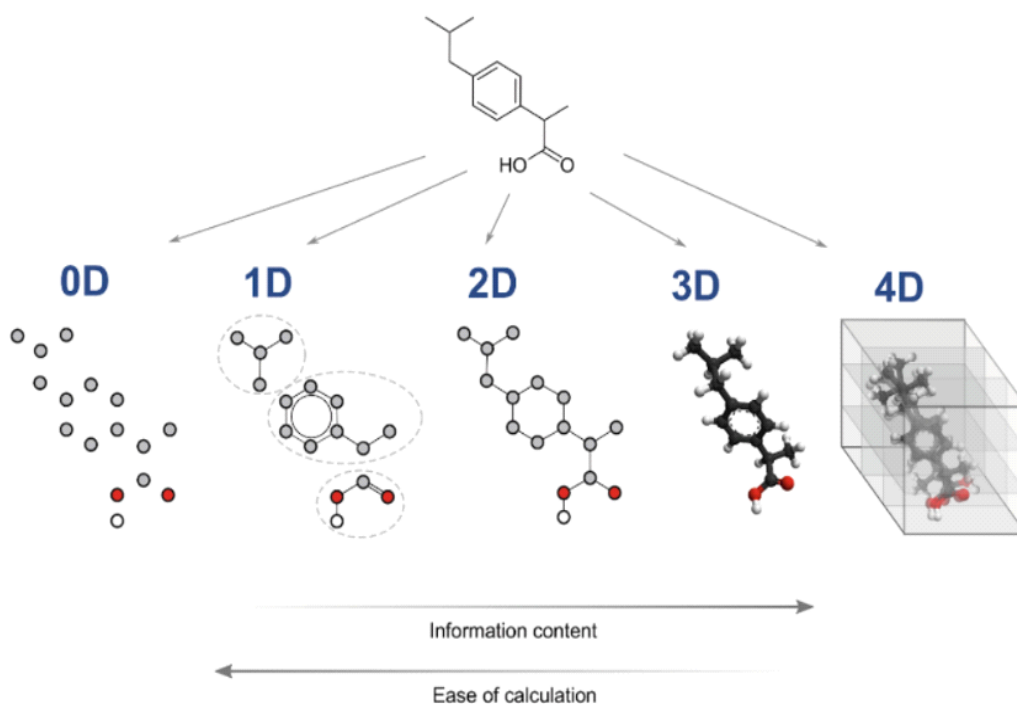


Figure 3. Representation of the type of information that five classes of theoretical descriptors can encode about the same molecular structure [80].

0D descriptors do not require optimization of the molecular structure and are independent of conformational problems, they can be always easily calculated and interpreted. 0D descriptors do not provide any information about the entire molecular structure, they encode information about atom and bond counts, as well as the sum or average of the atomic properties, which makes them useful to model only physicochemical properties [81].

1D descriptors can be calculated from the substructure of the molecules and are often called fingerprints. Fingerprints are binary vectors where the presence of a substructure is indicated with 1 and the absence with 0. Like 0D descriptors, fingerprints do not require optimization of the molecular structure and are independent of conformational problems, they can be easily calculated and interpreted. Different from 0D descriptors, fingerprints can model both physicochemical and biological properties. Since these two types of descriptors encode very simple information, they are often used in combination with more complex descriptors [80].

2D descriptors are based on a graph representation (often an H-depleted molecular graph) of the molecule and represent graph-theoretical properties. They can be also called topological descriptors since they provide information on molecular topology based on the graph representation. Topological descriptors are usually divided into two categories: topostructural and topochemical indices [82]. Topostructural indices encode information on the adjacency and distance of atoms in the molecular structure, while topochemical indices describe both topological and chemical information [81]. Typical 2D descriptors are the adjacency matrix, the Coulomb matrix, or the distance matrix. Since topological descriptors are sensitive to the structural characteristics of the molecule, such as size, shape, symmetry, branching, and cyclicity, they are commonly employed in QSAR modeling.

3D descriptors are called geometrical descriptors since derived from the geometrical representation of the molecules. For this property, 3D descriptors can define the molecule as an object in space, in terms of the atom types and their spatial coordinates x-y-z, bearing high information and discrimination content [80,81]. However, due to their complexity, the geometrical descriptors can be time-consuming to calculate and can increase the complexity of the classification problem [81]. For these reasons, topological descriptors, fingerprints, and 0D descriptors are usually preferred to describe large databases of molecules [81]. 4D descriptors, also called grid-based descriptors, are based on geometrical representation but introduce a fourth dimension in order identify and characterize quantitatively the interactions between the molecule and their target. These descriptors can also be used to describe the multiple conformational states of the molecule. 4D descriptors contain more information than any other molecular descriptor, but due to their extreme complexity are difficult and time-consuming to calculate [80].

Several software packages, both commercial and free, are available to compute molecular descriptors, such as PaDEL-Descriptor [83], Mordred [84], CDK [85], RDKit [<http://www.rdkit.org>], alvaDesc [86], ChemoPy [87], PyDPI [88], and RcpI [89]. These software packages provide thousands of descriptors encoding a broad spectrum of pharmacodynamics, pharmacokinetic and toxicological properties, among others.

In my project, molecular descriptors were calculated using the software alvaDesc version 2.2. AlvaDesc is a commercial software that enables the calculation of more than 5666 different molecular descriptors divided into 30 logical blocks [86] covering from 0D to 3D descriptors. It takes in input SD files format and from the representations of the molecules allows a plethora of algorithms to be applied to calculate the descriptors which best fit the problem to be addressed.

During model development, only molecular descriptors from 0D to 2D were used to describe the molecules of the dataset. On the other hand, in the model optimization phase, another dataset with 0D-3D was built to test how the type of information encoded by the different descriptors affected predictive performance.

Since alvaDesc allows the calculation of a large number of descriptors, it implemented methods to remove non-informative descriptors, which might not accurately describe molecular properties or constitute redundant information [86]. Precisely, through these methods, it was possible to exclude those descriptors for which there was at least one missing value, those with a correlation of 95%, and those with a variance of less than 0.001. After this operation, the descriptors were inserted into the dataset.

4.2 Model development

Generalization is the aim of every good ML model and consists in finding statistical patterns in the training set that allows it to correctly predict new unseen data [48]. To be able to generalize, the trained model needs to discriminate signal, i.e. relevant information, from noise, i.e. irrelevant information.

When the model is unable to generalize correctly, two conditions can occur, underfitting and overfitting [90,91].

Underfitting occurs when a model is unable to capture the signal from the data. This condition arises when the dataset used to train the model contains few samples, and the data are not varied enough to describe the problem [90].

Model overfitting occurs when the model learns both signal and noise, producing predictions that are very accurate on the training set but significantly less accurate on new unseen data. This condition arises when the model is trained on an overly noisy dataset, i.e. one with many uninformative information. For determining whether the model has generalized properly on the training set data, two additional sets are needed, termed validation and test set [91].

The division into training, test, and validation sets (Fig.4) is conducted before training the model. Training and test sets are obtained by dividing the dataset set into two parts. The largest part, with the most data, represents the training set and is used to train the model. The smallest part represents the test set and is employed to assess the generalization performance of the model. The validation set is obtained as a smaller subset of the training set, is independent of it but follows the same probability distribution, and is employed to tune the parameter of the trained model.

Once the dataset has been divided, the process leading to the generalization assessment proceeds as follows: the model is trained using the training set. During the training phase, the model parameters are tuned and different models with different parameter combinations are evaluated by the validation set. Training and tuning are iterative processes, which continue until the combination of model parameters that achieves the best predictive performance is identified. Once the predictive model is obtained, its generalization is evaluated through the test set. Since the test set does

not participate in the training and tuning process, can be considered an external dataset and can be used to assess the generalization performance of the model.

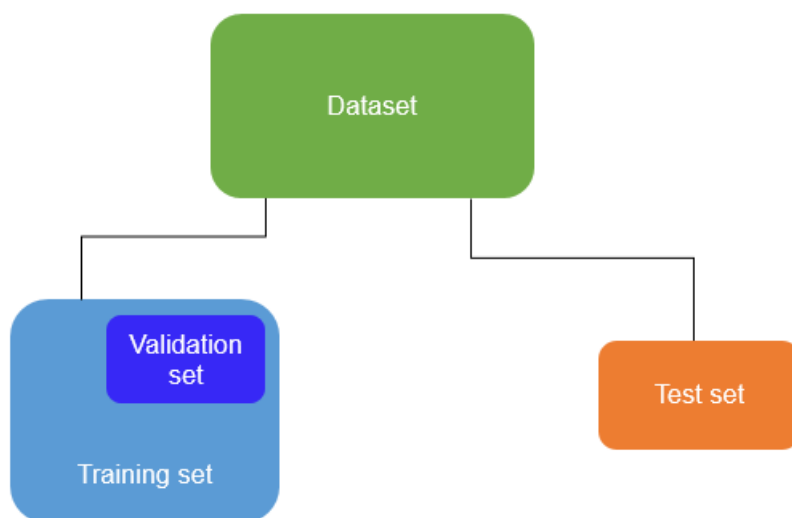


Figure 4. Schematic representation of the dataset division in training, test, and validation sets.

4.2.1 Dataset division

The dataset division in my project was conducted employing QSAR-Co-X software.

QSAR-Co-X is an open-source toolkit developed by the research group of professor Maria Natalia Dias Soeiro Cordeiro to support the generation of mt-QSAR models based on the Box-Jenkins moving average approach [92]. QSAR-Co-X is divided into four modules, it allows the use of a wide variety of utilities that follow the QSAR modelling rules dictated by the Organization for Economic Cooperation and Development (OECD) [93] and was used in almost all phases of my project.

Module 1 allows the division of the dataset according to three methodologies; the one employed in my project is random division. In random division, the dataset is shuffled and samples are randomly selected and inserted into the training, validation, or test set according to the percentage specified by the user. Therefore, since the 70:30 proportion is commonly used to divide the training from the test set [94], by setting the percentages in QSAR-Co-X, the training set (70% of the data from the initial

dataset) and the test set (30% of the data from the initial dataset) were generated. To understand how the third set is generated, it is first necessary to discuss the Box-Jenkins moving average approach.

4.2.2 Box-Jenkins approach

The Box-Jenkins moving average approach is a mathematical model originally used for forecasting data following time series, which measures the robustness of a dependent variable in relation to changing variables. This approach is fundamental for developing mt-QSAR models. However, in mt-QSAR modeling, the Box-Jenkins approach is not applied to time series, but rather to experimental and/or theoretical conditions [95].

Normally, molecular descriptors encode the physicochemical and biological properties of the compound exclusively according to their structure. Consequently, they will not be able to discriminate the cytotoxicity of a molecule when tested against different hematological cancer cell lines. The Box-Jenkins approach represents a solution allowing the calculations of moving averages, here represented by deviation descriptors, calculated by input descriptors [96]. Such deviation descriptors can encode also information about experimental and theoretical conditions under which the biological activity manifested itself [95].

The QSAR-Co-X module 1 allows the calculation of deviation descriptors according to four different equations. Here, it has been used the first equation, which is represented as follows:

$$\Delta(D_i)c_j = D_i - avg(D_i)c_j \tag{1}$$

Where $\Delta(D_i)c_j$ is the new descriptor coding the experimental conditions. $\Delta(D_i)c_j$ represents the standard deviation obtained by subtracting the value of the original descriptor D_i , namely the input descriptor calculated by alvaDesc, from the average value of that descriptor $avg(D_i)c_j$ concerning an experimental condition [97]. In the

model-building phase, only one experimental condition relating to the target cell line was included in the dataset. To do this, the Target Name column containing the name of the cell line to which the molecules were cytotoxic was reinserted into the dataset. After that, $avg(D_i)c_j$ of the molecules tested against a specific cell line was calculated. This process was repeated for all cell lines in the dataset. Finally, once $\Delta(D_i)c_j$ relative to all cell lines have been obtained, the model can use these deviation descriptors to discriminate the cytotoxicity of molecules according to different cell lines.

Indeed, once the new $\Delta(D_i)c_j$ descriptors have been calculated, the ML algorithm will rely exclusively on them to create the predictive model, and no longer on the input descriptors.

During the optimization phase of the model, following the same procedure, two additional experimental conditions were included in the dataset, relating to the time point and the type of assay used to assess cytotoxicity. To do this, two new columns were created, named Cd2 (time point) and Cd3 (assay type) respectively, and short alphanumeric values corresponding to each time point or bioactivity assay were assigned to these columns.

Equal time points or equal bioactivity assays have equal alphanumeric values.

Within the dataset, the number of new deviation descriptors $num. \Delta(D_i)c_j$ will be represented by the number of input descriptors nD_i multiplied by the number of experimental conditions included in the model (k):

$$nD_i \times k = num. \Delta(D_i)c_j \tag{2}$$

It is therefore intuitive that the more experimental conditions to be included in the model, the greater the number of deviation descriptors will be.

To accomplish the calculation of deviation descriptors, equation 1 was applied to the training set. Subsequently, the training set with its deviation descriptors was randomly subdivided in training set and test set according to the percentage specified by the user. In this phase, 80% of the data were kept within the training set while 20% were

used to form a test set, which in this case represents a calibration set [92]. At this point, the $avg(D_i)c_j$ values obtained in the training set were used to calculate the deviation descriptors of the previous test set, the one obtained from the initial splitting of the dataset, renamed validation set to avoid misunderstandings (Fig. 5). This validation set can be effectively used to evaluate the generalization performance of the model since the data composing it participates neither in the development phase of the model nor in the calculation of the deviation descriptors [92].

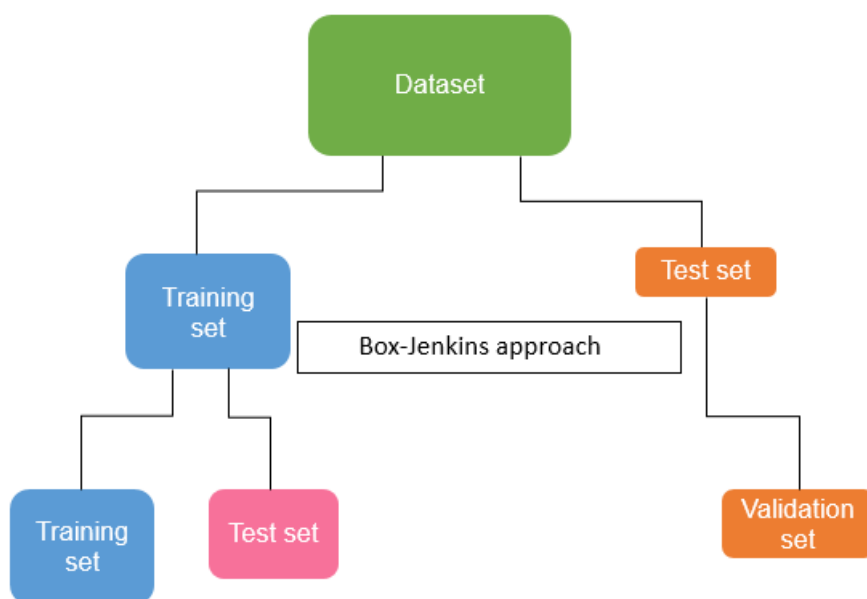


Figure 5. Schematic representation of the dataset division according to QSAR-Co-X.

Thus, at the end of the data division and after applying the Box-Jenkins approach, the dataset is divided into training, test, and validation sets (Fig. 5). For the sake of clarity, contrary to what was previously explained, the various sets will be named according to the division made by QSAR-CO-X, where the training set represents the set used to train the data, the test set represents the test for tuning the model parameters and the validation set represents the set to determine the model generalization. Once the calculation of the deviation descriptors and the division of the dataset had been completed, it was possible to build the predictive model.

4.3 Predictive model generation

The construction of the predictive model involved the application of different ML algorithms, to identify the one capable of generating the model with the best predictive capabilities. To do this, was used QSAR-Co-X module 2, which allows to generate predictive non-linear models by applying different ML algorithms implemented in the scikit-learn library [98], and tuning their parameters. Most ML algorithms have some parameters (usually more than one) called hyperparameters, that need to be optimized to maximize the predictive performance of the model. The process of optimization is called hyperparameter tuning and consists in searching for the best combination of hyperparameter values. Several methods have been proposed for hyperparameter tuning, the one implemented in QSAR-Co-X and employed in my project is called grid search, which represents the optimization standard [99]. Grid search provides an exhaustive search among a subset of hyperparameters of the algorithm being used to generate the model. The subset may be user-defined [99]. The algorithms used for the construction of the mt-QSAR model were RF, GB, SVC, K-NN, MLP. For the tuning of their hyperparameters, grid search was performed using a file containing a subset of parameters and their values included in the QSAR-Co-X package, and evaluated with 5-fold cross-validation. In 5-fold cross-validation, the training set is randomly divided into 5 subsets of the same size [100]. One of the subsets is used to evaluate the performance obtained by training the remaining 4 subsets with a combination of hyperparameters. This validation is repeated 5 times, each time using a different subset for validation and the remaining 4 subsets to train the model with the same combination of hyperparameters. At the end of the training, an average of the 5 iterations is performed and model evaluation metrics are returned. The entire 5-fold cross-validation is repeated several times, each time using different combinations of hyperparameter values [100]. Afterward, the performance of all 5-fold cross validations is compared using the test set and the model with the combination of hyperparameters that resulted in the highest scores in the evaluation metrics is selected. Finally, the best predictive model is evaluated using the validation set to assess its generalization performance. All these operations are performed via QSAR-Co-X, where it is necessary to upload the training set, the test set, and the file with the hyperparameters for the model of interest. The model is then trained and the best-

performing combination of hyperparameters is selected. At the end of this process, the results of the trained model are returned, and it is validated by uploading the validation set.

The algorithms used to build the mt-QSAR model are listed below, for each one a brief description is provided.

4.3.1 ML algorithms

4.3.1.1 k-NN

k-NN is an algorithm that predicts the class to which a data point belongs based on proximity. k-NN represents each element of the dataset as a point in a d-dimensional space where d represents the number of features [101]. K-NN is based on the assumption that similar data can be found adjacent to each other. Thus, given input data, its class label is determined by comparing it with its nearest neighbor data points. The classification process operated through k-NN consists of the following stages: 1) during the training process k-NN store all the training data. When the algorithm is asked to predict new data, it compares the new data with the training data. Finally, the new data is assigned to the class based on a majority vote [102], i.e. is assigned to the class that is most represented in the k- nearest neighbors, where k is a hyperparameter [102]. Particular attention must be taken when choosing the value of k. Low values may make the classifier susceptible to overfitting, while high values may lead the classifier to incorrectly predict a label because points that are very far have been counted in the neighborhood.

The distance between data points can be evaluated in different ways, often Euclidean distance is employed [101].

4.3.1.2 SVC

Many ML models are linear, meaning they can only linearly represent the classes of the data of a problem, which makes them too simple for many applications. SVC, instead, is a linear model that can be used to represent non-linear problems [103]. To solve the classification problem SVC aims to find a hyperplane in a d -dimensional space (where d represents the number of features) that can separate the elements of that space into two classes [104].

Many hyperplanes can separate the two classes of data points. The objective of SVC is to find the one with the maximum margin, i.e. the maximum distance between the data points of both classes [104]. Maximizing the distance margin allows future data points to be classified with higher confidence.

Hyperplanes are decision boundaries whose shape depends on the number of features. When the number of features describing a data point is 2, the hyperplane is represented by a line. If the features are 3, the hyperplane is represented as a 2D plane. When the number of features is more than 3, the hyperplane becomes very complex and can no longer be represented. Consequently, the higher the number of features in a dataset, hence its dimensions, the more difficult and time-consuming will be the task of finding a hyperplane.

To determine the best hyperplane, SVC uses the data points closest to the hyperplane, which are called support vectors, and influence its position and orientation. By using these support vectors, the classifier maximizes the margin [104].

However, in many cases, not all data points can be separated by a hyperplane.

Through the adjustment of a specific hyperparameter, termed parameter C , it is possible to allow some data points (as few as possible) to cross the decision boundary, making the model more flexible but more prone to classification errors [104].

In other cases, however, the separation of data with hyperplanes is not possible at all. Therefore, the SVC algorithm provides another hyperparameter that enables the separation of non-linear patterns. This is achieved through the use of a transformation function that maps in a new linear space, a non-linear model [104].

Thus, a non-linear model is elevated to a higher dimension to make it linearly separable.

4.3.1.3 RF

RF is an algorithm based on the concept of ensemble learning, in which several individually weak (with poor accuracy) classifiers are combined to generate a strong (with high accuracy) classifier [105]. The combination is successful only when the individual classifiers are, at least partially, independent of each other, which means they do not make the same errors [106].

RF is thus called because it is a combination of multiple decision tree classifiers. A decision tree (DT) is a non-parametric algorithm with a hierarchical tree structure. This structure begins with a root node from which outgoing branches flow into internal nodes, also known as decision nodes [107]. The nodes without a descendant are called terminal nodes, or leaves. The root and internal nodes represent a test over a given feature of the dataset, while branches represent the decision rules [107]. In the classification problem, terminal nodes correspond to the predicted class.

In a DT, to predict the class of a data point the algorithm starts from the root node and descends through the branches according to the results of the feature test. The classification task is accomplished when it reaches a terminal node.

In RF independence between the classifiers can be achieved by training the DT on different portions of the training set, for this reason, RF is termed a Bagging algorithm. To obtain a single predictive model, the individual DT outputs its own decision (the class of the data point), and all the decisions about that specific data point are then merged at some level of the classification process. Specifically, each decision is combined through a fusion method. The fusion method employed by RF is the majority vote rule, in which each DT votes for a class and the data point is assigned to the class with the most votes.

4.3.1.4 GB

GB is another classification algorithm that belongs to the group of ensemble learning algorithms. Unlike the Bagging method where each classifier is independent and the final solution is dictated by majority voting, Boosting aims to seek continuous improvement [108]. In this way, this method also attempts to transform weak classifiers into strong classifiers. Boosting is also known as a sequential ensemble since weak classifiers are produced sequentially during the training phase, and not in parallel as in RF in which all classifiers are independent. Like RF, GB uses multiple DTs as weak learners. Here, the performance of the model is improved by assigning a greater weight (a real number that determines the importance of input data for the output) to the incorrectly classified samples at each iteration of Boosting [108].

Therefore, the principle behind the operation of the Boosting algorithm is the generation of multiple weak classifiers and the combination of their predictions to form a strong rule [108]. This is done by generating weak rules on different distributions of the data set at each iteration of the training phase of the algorithm. Finally, the weak classifiers are combined to form a strong classifier that predicts a more accurate result from a stronger set of rules.

Thus, Boosting method relies on incremental learning, an iterative process in which a new classifier is added after each iteration and trained on the data points incorrectly classified by the previous classifier [109].

Therefore, weak classifiers built in subsequent iterations focus more on the examples that the previous classifiers were unable to classify correctly.

4.3.1.5 MLP

The building blocks of neural networks are artificial neurons, so-called because they resemble biological neurons. Biological neurons are the basic elements of the nervous system. They consist of the soma, which is the cell body of the neuron from which originate minor fibers called dendrites, and a main fiber called the axon. Dendrites collect input from afferent neurons and propagate it to the soma, which propagates it to the axon to transmit the output from one neuron to another, even if they are very distant. The axon terminates with minor branching fibers, which make contact and

propagate information to a second neuron through an electrochemical process involving the release of neurotransmitters.

The simplest and oldest model of an artificial neuron is the perceptron, derived from the model of Frank Rosenblatt in 1958 [110]. The perceptron can be considered a non-linear function that transforms inputs (x_1, x_2, \dots, x_n) into an output y , and consists of two types of nodes: input nodes, representing the attributes, and output nodes, representing the output of the system. Each input node is connected via a weighted link to the output node, where the weight of the connection determines the strength of the connection [110]. The configuration of the weights is used to optimize the connections to achieve a good correlation between the input and output of the model. A perceptron determines the output value by summing all input values multiplied by their weights, then removing a bias factor (correction) from the sum, and finally analyzing the sign of the result through an activation function. During the training phase, the weights are repeatedly adjusted until the output of the system becomes consistent with the output of the training data, i.e. with the expected results. This is referred to as the learning rate. Perceptrons can take various binary inputs to produce a single binary output.

Optimization and enhancement of the perceptron enabled the creation of artificial neural networks (ANNs) which consist of several neurons organized in layers [111]. The simplest type of neural network is the MLP, which comprises 3 layers: input layer, hidden layer, and output layer.

In MLP, each neuron in one layer is connected to each neuron in the next layer, which is why the network is defined as fully connected [111].

Traditional ML algorithms require to be performed by users with an in-depth knowledge of the topic related to the problem that is being solved. The quality of an ML model depends mainly on the quality of the data set, which requires meticulous data curation work by an expert in the problem field. On the other hand, ANNs, such as MLP, do not require the intervention of experts because they can learn the features directly from the data and learn how to represent the data, providing predictive models that can be even better than those obtained with ML algorithms.

ANNs can be built in a variety of sizes, with complexity increasing proportionally as the size increases. Complexity can greatly influence the model performance, for this reason, is essential to choose an appropriate number of hidden layers and neurons in each layer. ANNs with two or more hidden layers are called deep neural networks (DNNs), in which first hidden layers answer very simple and specific questions about the input data, and later layers compute a more abstract, high-level representation of the data [112].

4.4 Performance evaluation metrics

Performance evaluation is an essential step during model development that can be performed through evaluation metrics. Evaluation metrics allow not only to assess the prediction performance of a model but also allows to compare predictive performances across models.

Evaluation metrics used in this project were computed by QSAR-Co-X. All of them are derived from the confusion matrix, a well-known evaluation parameter for binary classification problems. A confusion matrix for a binary classification problem is a 2 x 2 matrix in which the rows constitute the actual class and the columns the predicted class of data (Fig. 6). It can be represented as a table with four combinations of possible results:

		Actual Class	
		Positive (P)	Negative (N)
Predicted Class	True (T)	True Positive (TP)	False Positive (FP)
	False (F)	False Negative (FN)	True Negative (TN)
		$P=TP+FN$	$N=FP+TN$

Figure 6. Representation of the confusion matrix for a binary classification problem.

True positive (TP): the model predicts a positive result (active compound) and it is actually positive

True negative (TN): the model predicts a negative result (inactive compound) and it is actually negative

False positive (FP): the model predicts a positive result but it is actually negative

False negative (FN): the model predicts a negative result but it is actually positive

In this manner, the orange diagonal of the matrix indicates correct prediction, while the blue diagonal indicates the incorrect ones.

The outputs deriving from the confusion matrix are utilized to calculate several other classification metrics such as:

- Sensitivity (recall or true positive rate): is the ratio of TP predictions and total P predictions. Sensitivity can assume values between 1 (best value) and 0 (worst value).

$$Sensitivity = \frac{TP}{TP + FN} = \frac{TP}{P} \quad (3)$$

- Specificity (true negative rate): is the ratio between TN predictions and total N predictions.

$$Specificity = \frac{TN}{TN + FP} = \frac{TN}{N} \quad (4)$$

- Accuracy (ACC): is the result of the all correct predictions divided by all the predictions of the model.

$$ACC = \frac{TP + TN}{TP + TN + FN + FP} = \frac{TP + TN}{P + N} \quad (5)$$

SN, SP, and ACC can assume values between 1 (best value) and 0 (worst value).

- Matthews Correlation Coefficient (MCC): is a measure that takes into account every predictions of the confusion matrix and all their combinations.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (6)$$

The MCC can return a value between -1 and $+1$, where $+1$ represents a perfect accordance between observed and predicted values, 0 indicates random prediction and -1 represents total disagreement between observed and predicted values.

- Area Under the Curve – Receiver Operating Characteristic (AUC – ROC) curve: ROC is a probability curve and AUC represents the degree of separability between two classes. The ROC curve is built with TP on the y-axis against the FP on the x-axis. AUC-ROC can assume values between 1 (best value) and 0 (worst value), with 0.5 corresponding to random prediction in balanced dataset.

Since there is no perfect method to describe the confusion matrix, all these metrics were considered when comparing the various models. However, particular attention was given to ACC and MCC since they consider both positive and negative categories. ACC is one of the most frequently utilized metrics in binary classification problems. However, when the dataset is unbalanced it could provide an over-optimistic evaluation of the predictive capacity of the model since it does not consider the proportion of positive and negative elements [113]. On the other hand, MCC is a more trustworthy statistical factor as it returns a higher value the more correct the predictions are in all four categories of the confusion matrix. In addition, it considers the proportion between positive and negative elements [113].

4.5 Dimensionality reduction

As previously explained, a dataset for classification is composed of a matrix of rows (samples) and columns. Once the molecular descriptors, and lately the deviation descriptors are calculated and integrated into the dataset, each descriptor constitutes a new column. These columns are also called features or dimensions. Therefore, the dataset will contain as many dimensions/features as the descriptors calculated to describe the problem.

Features can be compared as variables in a scientific experiment since they are characteristics of the phenomenon under observation that can be quantified or measured. When features are fed into a ML algorithm the network tries to discern relevant patterns between them to generate the outputs. The outputs of a classification problem are the classes belonging to the compounds. Thus, features become the inputs that the model utilizes to make predictions. For this reason, their quality must be as high as possible to make a good predictive model.

Due to the large number of descriptors that can potentially be generated using different software, datasets can reach a huge dimensionality. It is led to think that a higher number of features correspond to more information and better predictive performance. However, this is not always the case. Generally, features can be categorized according to their influence on the output as 1) relevant, if they influence the output, 2) irrelevant, if they have no influence on the output, and 3) redundant if two or more features encode the same type of information about the data [114]. Thus, a large number of features can lead to various problems, such as difficult data analysis and visualization, and difficulties in training the ML model. When the latter condition arises, a very common problem known as the “curse of dimensionality” occurs [115]. High dimensionality represents a problem because as the number of the dimension increase, the number of data required to generalize accurately grows exponentially.

Some techniques, called dimensionality reduction techniques, are used to adequately combine the features of a high dimensional input space into a lower-dimensional subspace, maintaining their relevant information. The aim is to discard irrelevant or redundant data insignificant to the problem [116].

Operating in lower dimensionality spaces makes it easier to train ML algorithms and can improve model performance. Dimensionality reduction can be performed according to two main approaches, feature extraction techniques, and feature selection techniques [116].

Feature Selection techniques aim to select only those features that contain the relevant information for solving the problem. The ideal number of features is the smallest one that most contributes to accurately describing the problem. Feature Extraction, instead, operates a transformation of the input space onto a low-dimensional subspace that preserves most of the relevant information [117]. Feature extraction and selection methods can be used isolated or in combination to improve predictive performance.

In my Ph.D. project, during the optimization phase of the model, four different dimensionality reduction techniques were used singularly and in different phases. Two of them are feature extraction techniques and correspond to Principal Component Analysis (PCA), and Linear Discriminant Analysis (LDA). The other two methods are feature selection techniques and correspond to information theory-based feature selection and Genetic Algorithm-k-NN (GA-k-NN).

4.5.1 Principal Component Analysis

To execute the dimensionality reduction PCA perform an unsupervised linear mapping of the initial features space and employs an orthogonal transformation to convert them into a smaller subset of uncorrelated artificial features called principal components (PCs) [118]. In this way, PCA extracts information from several redundant features in a smaller number of unrelated features. Therefore, the generation of the predictive model will not rely on the initial descriptors anymore but on the PCs. The purpose of PCA is to preserve the initial maximal variance in a lower-dimensional space [119], as the variance reflects the information in the data. The PCs are ranked with the first bearing the highest percentage of variance, followed by other PCs with variance values in descending order. Thus, most of the information contained in the original variables is compressed into the first PCs [118]; it is sufficient to eliminate all PCs that

contain little variance to remove the noise and improve model performances. The number of PCs selected to create the model is arbitrarily determined. Usually, PCs are chosen to express 90% of the initial variance. When two or three PCs are chosen, data distribution can be visualized.

A common method for determining the number of PCs to be used for model building is a graphical representation known as a scree plot. A scree plot is a simple line-segment graph showing the PCs and their respective variance content. The graph presents an ascending or descending curve, which starts on the left, ascends or descends rapidly, and then flattens out. The PCs are selected using the elbow rule, i.e. the point on the curve just before the line flattens is searched for, known as the 'elbow', and all PCs within the point are considered.

Since PCA is not implemented in QSAR-Co-X, to perform it a code in Python language was created, using the PCA algorithm implemented in scikit-learn.

4.5.2 Linear Discriminant Analysis

LDA is a feature extraction technique that performs a supervised linear mapping of the initial data and projects it on a straight line to maximize the proximity of the projection points of the interclass samples and maximize the distance between the projection points of the inclass samples. LDA not only reduces the dimensions but also tries to maximize the discriminatory information between classes. Thus, when classifying a new sample, it is projected on the same straight line, and its classification is determined by the position of the projected point. Unlike PCA, which prioritizes dimensions that best represent a pattern, LDA prioritizes dimensions that best discriminate patterns [119].

To perform the LDA, was used the model implemented in QSAR-Co-X called forward stepwise LDA (FS-LDA), which can be accessed via module 1 of the software. FS-LDA combines feature selection techniques with LDA.

FS is an iterative method of feature selection, that starts with an empty set, where variables are added one by one according to the lowest p-value. The p-value indicates

the significance of the feature for the target variable, the lower its value, the more significant the feature. Then, after the linear transformations of the starting features have been performed via LDA, they are stored in a new dataset according to their increasing p-values.

4.5.3 Information theory-based feature selection

Information theory relies on a work conducted in 1948 by Claude Shannon [120], whose purpose is to quantify how much information a communication system can transmit relying on probabilistic techniques. Information theory is based on the measure of information entropy, called Shannon entropy (SE), which is the average level of information contained in a random variable, and can be represented with the following equation [121]:

$$SE = - \sum p_i \log_2 p_i \quad (7)$$

where 'p' is the probability that a data point falls within a specific data interval 'i', while \log_2 can be interpreted as a scaling factor allowing SE to be considered as an information content metric [121].

The concept of information defined for information theory embraces various fields that can be far from telecommunications systems for which it was born, such as chemistry, statistics, biology, neuroscience, behavioral science, and statistical mechanics [122].

For instance, in chemoinformatics, the main applications of information theory concepts include:

- quantifying the chemical information contained in different representations of small molecules [122].
- its use in statistical analysis, data mining, and ML, which have become fundamental tools for chemoinformatics research [122].

A feature selection method based on information theory was used here to perform dimensionality reduction. For this purpose, was employed a free software with a graphical user interface, named IMMAN [123].

IMMAN is the acronym for Information theory-based CheMoMetrics ANalysis, and it allows performing 20 feature selection different approaches. The approach used in my project is a supervised method that allows obtaining a ranking of the first k features according to their differential Shannon entropy (DSE) values, where k is a number decided by the user. Feature selection based on DSE has previously proven to be successful for the construction of mt-QSAR models [124,125]

DSE can be calculated as follow:

$$DSE = SE_{1,2,3\dots n} - (SE_1 + SE_2 + SE_3 + \dots SE_n)/n \quad (8)$$

Where $SE_{1,2,3\dots n}$ usually is the SE measured for n combination of compound databases [121,123], while in feature selection task represent n -class partition based (in this project a 2-class partition based).

Thus, DSE is an extension of SE that allows the comparison of the information content of descriptors in different class-based partitions, even if the differences in the distribution of their values are difficult to identify [121].

4.5.4 Genetic Algorithm-k-NN approach

GA is one of the most advanced algorithms for feature selection, it is a stochastic method inspired by human genetics and biological evolution, and is therefore called an evolutionary algorithm [126]. Indeed, similarly to how in-nature genes evolve in successive generations to better adapt to the environment, GA operates on a population of starting features to produce better feature subsets.

GA mimics what occurs during gamete replication, particularly in the first meiotic division. During meiosis, in gametes, there are pairs of chromosomes called homologs, one inherited from the father and one from the mother. In the initial stage of meiosis, called prophase, the two pairs of homologous chromosomes come so close that they exchange chromosome portions of the same size, in other words, equal amounts of genetic material [127]. In this way, chromosomes different from their parent

chromosomes are created. This exchange process is called crossing-over and its function is to enhance the genetic diversity of meiotic products [128].

The research of the best subset of features by the GA employs strings of values called chromosomes. Each chromosome represents a specific combination of features, the parameters to be optimized, and consists of a string no longer than the total number of features in the dataset. Chromosomes consist of individual genes and each gene represents a feature indicated by binary values (0,1), which will determine the absence (0) or presence (1) of that feature in the chromosome. A combination of chromosomes for a particular dataset is called a population.

The GA process begins by generating a population of random chromosomes large enough to ensure adequate diversity in the space of feature subsets [126]. The purpose of the process is to advance only the fittest chromosomes into the next generation. Thus, the fitness of each chromosome is estimated by employing a fitness function, a measure of performance that determines how much the chromosome contributes to the good performance of the model. A fitness score is assigned to each chromosome, the higher this score is, the higher the probability that the chromosome will be selected for breeding in the next generation. At the end of this process, the most promising chromosomes are selected to produce the next generation through procedures of reproduction, cross-over, and mutations.

Reproduction involves preserving the best chromosomes in the next population, while the others are modified through crossover and mutation procedures to form new chromosomes [126].

During cross-over, two chromosomes exchange portions of genes of the same size and position. In this way, two new chromosomes are formed to replace the ones in the initial population with poor fitness scores [126].

In contrast, during mutation, the arrangement of genes in the chromosome is altered to produce an entirely new chromosome.

Once a new generation has been created, it is evaluated, and the entire procedure is repeated for a predefined number of repetitions until a final solution is obtained [126].

For the dimensionality reduction approach adopted in my project, GA was employed in conjunction with k-NN. Hence, each time GA produces a generation, it is evaluated in terms of classification accuracy using k-NN. The use of k-NN is chosen because it is a very fast classifier, which is a considerable advantage when it is necessary to evaluate many generations. To apply this approach, a software called GA-KNN, implemented by the research group of professor Cordeiro and not yet distributed, was used. This software rely on the GA-k-NN methodology implemented in the Python based scikit-learn-genetic program (<https://github.com/manuel-calzolari/sklearn-genetic>), which uses a “deep” function to execute the GA.

Here, an initial population of 100 chromosomes was selected, and from it 100 generations were created and evaluated by k-NN. The final number of features cannot be specified by the user, but the number of features that gave the best performance with k-NN was saved and stored in a new dataset. Then, the dataset with a reduced number of features was evaluated with RF, and after that was submitted to the GA-KNN software again. This iterative process continued until it was not possible to reduce the number of features. In this way, several datasets with a reduced number of features were created and compared.

4.6 Applicability domain

One of the most relevant problems in QSAR analysis is establishing the applicability domain (AD) of a model. AD represents the region in space defined by the nature of molecular structures present in the training set, through their molecular descriptors [129]. Therefore, it is not possible to use only one QSAR model to correctly predict the activity of any compound, no matter how robust, significant, and validated the model is [130]. To avoid incorrect predictions, the domain within the QSAR model can make predictions should be restricted to AD [131]. Defining the border of the AD may then be viewed as detecting outliers. Several approaches have been proposed to estimate AD, with none emerging as the best [130]. All these techniques aim to reject outliers that would lead the model to make wrong predictions, increasing its reliability [131].

In my Ph.D. project an approach called “confidence estimation” was used to define the AD of the mt-QSAR model. This method relies on the information of the class labels of the training set to estimate the level of confidence of new predictions [131]. The approach is implemented in QSAR-Co-X. Here, whenever the model makes a prediction two columns are created in the results datasheet, one showing the probability of the compound being negative (-1), and the other the probability of the compound being positive (1). These probabilities can take values between 0 and 1. The closer to 1, the compound will be considered positive, and negative otherwise. Also, the absolute difference between the two probabilities is considered, indicating how much confidence the model has in that prediction. The higher such a value, the more confidence the model has in making that prediction. A threshold of 0.5 was applied to the absolute difference value. Therefore, if the absolute difference between the two probabilities was greater than 0.5 the compound was considered in, otherwise it was considered an outlier.

5. Results

5.1 Dataset construction

The research for hematological cancer cell lines in Cellosaurus and molecules tested against them in ChEMBL, resulted in a dataset of 66787 molecules and 71 hematological cancer cell lines. During the data curation process, by analyzing the dataset and checking data quality, several data homogeneity issues were founded. To improve the quality of the dataset samples with non-compliant values were eliminated. A list of the issues encountered is listed below:

- Presence of missing values: this condition occurs very frequently in datasets, the values corresponding to some columns are not recorded or available. The abbreviation "N/A" is chosen to indicate them.
- Standard Value equals 0: samples with this condition *de facto* are like having a missing value. It is impossible for a molecule to be active at the concentration corresponding to the value 0, this value did not provide any information.
- Target Type value different from cell line: although the search on ChEMBL was conducted by filtering only compounds active against hematologic cancer cell lines, some samples exhibited, in the corresponding column, a target different from the cell line, often a protein. Although the mt-QSAR model can include different targets, it is important that these are all represented by cancer cell lines.
- Standard Type different from IC₅₀: although the ChEMBL search was conducted by filtering only those molecules with the activity value expressed in terms of IC₅₀, there were some molecules for which the Standard Type was expressed with other indices, frequently GI₅₀ (concentration causing 50% cell growth inhibition).
- Standard Units not expressed in molar concentration: In order for molecule activities to be comparable and for the model to learn correctly from the data, it was important that all molecules had the same concentration unit expressed in μM . However, there were some concentrations expressed in $\mu\text{g/mL}$. One solution could have been to convert the data to μM concentration, but

considering the time required to perform the conversion and the number of samples that would be gained, it was preferred to eliminate the samples.

- Standard Relation not expressed with "=" symbol: some Standard Relation were expressed with the symbol $>$, $<$, \geq , \leq . These symbols do not express an exact concentration but rather a range. Since it is very important for subsequent construction steps that molecules have an unambiguous activity value, samples with this issue were eliminated.

Another very important step of data curation, involved eliminating molecules based on the Assay Description column. Although only molecules with activity expressed via IC_{50} were retained, analyzing the Assay Description column it became apparent that not all the assays reported were related to the cytotoxic activity of the molecules. Indeed, in a more comprehensive sense, IC_{50} can be defined as the concentration capable of inhibiting 50% of the activity under investigation. Thus, not all the molecules in the dataset were evaluated for their cytotoxic activity against cancer cells, but they had different activities, which could only be inferred from the description of the biological assay used to evaluate them. The most represented activities after cytotoxicity were cytostatic, anti-HIV, and anti-inflammatory activities. Therefore, it was necessary to carefully check the assay description of each molecule. 2317 different Assay Descriptions were present in the whole dataset, and due to the high variability of their text string, it was not possible to filter the dataset as described in section 4.1.1 for the previous cases. Therefore, it was necessary to proceed by inspecting the assay for a single molecule at a time, consuming some days.

Samples that presented assays for anti-HIV and anti-inflammatory activities were eliminated. Samples that showed cytostatic activity, on the other hand, were used to build a second dataset that could be useful for future applications. Instead, only samples with cytotoxic activity were retained in the dataset used for my research project.

A further step of data curation involved the elimination of duplicate molecules. It is important to emphasize that some molecules appear in the dataset more than once because they may have been tested on different hematologic cancer cell lines, these molecules are not considered duplicates but rather are defined as a case. In fact, for a

molecule to be considered a duplicate it must have the same target and be active at the exact same concentration. In order to make the identification of duplicates faster and easier, this step was performed using Excel. Here molecules were sorted by increasing Standard Value, so molecules with identical activity values were found to be consecutive. After that, an operation was performed to identify duplicates based on the SMILES, Standard Value, and Target Name columns. Indeed, molecules with identical values in all three of these columns are completely identical and their presence represented redundant information. Molecules found to be duplicates were eliminated all but one.

After the dataset curation, a cutoff value of 1 μM was defined, according to the procedure and motivation explained in section 4.1.2.

After assigning the cutoff value, it was necessary to eliminate duplicate molecules again. The procedure used is identical to the previous one, but in this stage were considered identical those molecules having the same SMILES, Target Value, and Toxicity value, i.e., having the same molecular structure, that were active against the same cell line at identical concentration. These duplicate molecules were eliminated all but one.

A final data curation step became necessary after the introduction of the cutoff value, which is the elimination of molecules, called discordant, that were both active and inactive against the same target. With an operation conducted in Excel, similar to that used to identify duplicates, all those molecules having the same SMILES notation, and the same Target Value but possessing both values 1 and -1 in the Toxicity column were found and eliminated. Since it was impossible to determine what the real activity of these molecules was, all discordant molecules were eliminated.

Within this specific dataset, the 1 μM cutoff value results in a ratio of active to inactive molecules of 30% and 70%, respectively, leading to the formation of an unbalanced dataset. During the optimization phase of the model, the impact that a change in the cutoff value could make on its performance was evaluated. For this purpose, through the same procedure performed to construct the dataset with a cutoff value of 1 μM , several datasets with the following cutoff values and respective ratios of active to inactive molecules were generated:

- 0.3 μM : \approx 15% active and 85% inactive molecules
- 0.5 μM : \approx 20% active and 80% inactive molecules
- 1 μM : \approx 30% active and 80% inactive molecules
- 2.5 μM : \approx 37% active and 73% inactive molecules
- 5 μM : \approx 50% active and 50% inactive molecules
- 7.5 μM : \approx 57% active and 43% inactive molecules
- 10 μM : \approx 62% active and 38% inactive molecules

It is important to evaluate the performance of the model on a more or less balanced dataset. In fact, many classifiers exhibit mispredictions in minority classes as they try to optimize overall accuracy without considering the distribution of each class. [132].

Comparing the performance of models with different cutoff values is used to identify the cutoff that allows the classifier to make the most accurate predictions.

After the data curation process and elimination of duplicate and discordant molecules, the dataset was composed of 11704 molecules, and the number of hematological cancer cell lines comprised was reduced to 43.

On the molecules in this dataset using alvaDesc, the descriptors from 0D to 2D were calculated, after which the deviation descriptors were calculated using QSAR-Co-X. The result is a dataset consisting of 11704 molecules, 43 hematological cancer cell lines which represent the one experimental condition included in the dataset, and 1640 deviation descriptors from 0D to 2D.

5.2 Model development and optimization

The above-mentioned dataset was subjected to five different classification algorithms to identify the one that yielded the best predictive model. As reported in Table 1, the algorithm that produced the best performance was RF, with an ACC value of over 86% in the training, test, and validation sets and an MCC of 0.635 in the test set and 0.625 in the validation set. This means that the model correctly classified 86% of the molecules in the dataset, specifically 6426 samples out of 7458 in the training set,

1137 samples out of 1317 in the test set, and 2521 out of 2926 in the validation set, for a total of 10084 correctly classified molecules out of 11704 in the entire dataset.

Focusing on the validation set, the model performs significantly better on negative samples, correctly classifying 94% of the total (sensitivity), while it performs worse on positive samples, correctly classifying only 63% of the total (specificity). This is not surprising considering that negative samples are the most represented in the training set, so the model is better trained to recognize them.

Table 1. Comparison of the prediction performance of four classification algorithms. RF, random forest; MLP, multilayer perceptron; GB, gradient boosting; K-NN, k-nearest neighbor; ACC, accuracy; MCC, Matthews correlation coefficient; ROC AUC, Area Under Receiver Operating Characteristic curve; TP, true positive; TN, true negative; FP, false positive; FN, false negative.

Algorithm	Set	ACC %	MCC	ROC AUC	Sensitivity%	Specificity%	TP	TN	FP	FN
RF	<i>Training</i>	86.128			94.865	62.722	1272	5154	279	756
	<i>Test</i>	86.333	0.635	0.798	66.0	93.692	231	906	61	119
	<i>Validation</i>	86.159	0.625	0.789	63.554	94.246	490	2031	124	281
MLP	<i>Training</i>	82.429			88.441	66.322	1345	4805	628	683
	<i>Test</i>	82.764	0.573	0.795	72.571	86.453	254	836	131	96
	<i>Validation</i>	83.527	0.561	0.769	62.776	90.951	484	1960	195	287
GB	<i>Training</i>	81.383			95.233	44.280	898	5174	259	1130
	<i>Test</i>	81.473	0.481	0.703	46.571	94.106	163	910	57	187
	<i>Validation</i>	81.066	0.464	0.694	44.747	94.060	345	2027	128	426
K-NN	<i>Training</i>	79.413			88.478	55.128	1118	4807	626	910
	<i>Test</i>	80.410	0.491	0.742	60.857	87.487	213	846	121	137
	<i>Validation</i>	78.298	0.424	0.705	54.086	86.961	417	1874	281	354

The model obtained with MLP has an ACC value that does not deviate much from that of the RF and reaches 83% in the validation set. However, when comparing the MCC values, these are significantly lower in both the test set (0.573) and the validation set (0.561), demonstrating a significant deterioration in performance. The same trend was recorded with GB.

Poor results were obtained with k-NN, while no results were recorded for SVC. In fact, even ten days after applying the algorithm, no results could be obtained on the dataset.

Once RF has been identified as the algorithm that returned the best predictive model, the optimization phase began. In order to investigate whether the predictive performance of the model could be improved, changes were made to the composition of the dataset.

Initially, it was assessed whether the addition of 3D descriptors to the dataset would improve the performance of the RF model. Indeed, 3D descriptors can encode structural information that is missing in 2D descriptors and could facilitate the algorithm to perform a more accurate classification.

Table 2. Comparison of the prediction performance between RF2D and RF3D. ACC, accuracy; MCC, Matthews correlation coefficient; ROC AUC, area under receiver operating characteristic curve; TP, true positive; TN, true negative; FP, false positive; FN, false negative.

Dataset	Set	ACC %	MCC	ROC AUC	Sensitivity%	Specificity%	TP	TN	FP	FN
RF2D	<i>Training</i>	86.128			94.865	62.722	1272	5154	279	756
	<i>Test</i>	86.333	0.635	0.798	66.0	93.692	231	906	61	119
	<i>Validation</i>	86.159	0.625	0.789	63.554	94.246	490	2031	124	281
RF3D	<i>Training</i>	85.213			95.204	60.550	1210	5200	261	788
	<i>Test</i>	86.115	0.625	0.781	60.795	95.341	214	921	45	138
	<i>Validation</i>	85.442	0.610	0.769	58.271	95.585	465	2035	94	333

Comparing the results of the model with 0D to 3D descriptors (RF3D) with the previous one with 0D to 2D descriptors (RF2D), it is immediately evident that there is not a big difference in their performance (Table 2). However, RF2D proves to be a better predictor than RF3D, which has a lower ACC (86%) in both the training and validation set, a lower MCC (0.625 in the test set and 0.610 in the validation set), and especially a worse classification of positive samples. Indeed, its sensitivity corresponds to 58.271% (compared to 63.554% for RF2D). The first observation that can be made is that the

addition of the 3D descriptors does not introduce any new structural information that improves the discrimination between the two classes. Therefore, it might seem that the topological descriptors are sufficient to adequately describe the problem. The second observation is that the increase in the number of descriptors, and thus the complexity of the model, may have introduced irrelevant or redundant information that prevents the algorithm from accurately discerning the signal, worsening the generalization task. To examine the impact that a reduced number of descriptors may have on model performance, two different dimensionality reduction techniques were employed on the RF3D dataset: PCA and LDA. Since PCA is not implemented in any module of QSAR-Co-X, it was executed by coding in the Python programming language. The first stage of the PCA execution involved the creation of a scree plot. The interpretation of the scree plot (Fig.7), through the elbow rule, established that 5 PCs were required to preserve as much variance in the data as possible. Since the number of PCs was greater than 3, it was not possible to display the distribution of the data.

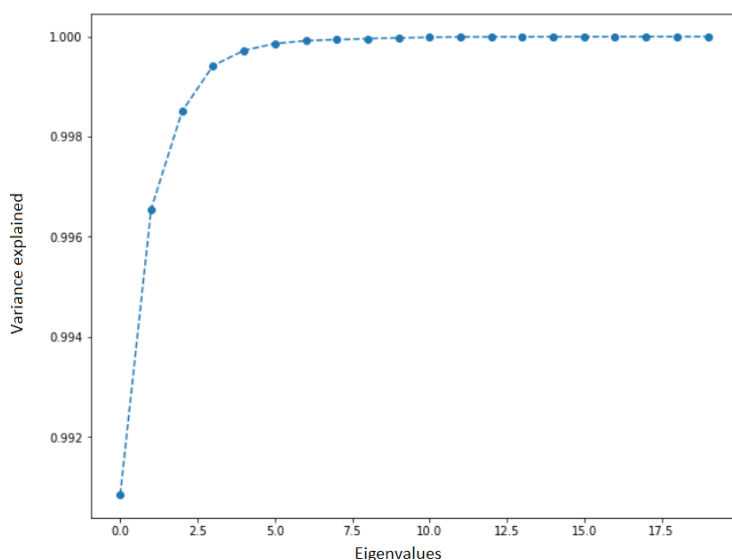


Figure 7. Visualization of the scree plot executed on RF3D dataset.

Table 3. ACC values of the RF3D generated with 5 PCs. ACC, accuracy

Dataset	RF3D with 5PCs		
Set	<i>Training</i>	<i>Test</i>	<i>Validation</i>
ACC %	99.293	71.695	70.179

The second step of the PCA execution involved the creation of an RF model using the 5 PCs as features. The results were reported in terms of ACC only, and are reported in Table 3.

The large reduction in dimensionality and the use of the PCs significantly worsened the performance of the model. In fact, the ACC values of the test and validation sets did not exceed 71%, although the value of the training set was very high and exceeded 99%. Such a good performance on the training set and the decreasing of performance in the test and the validation set indicate the presence of overfitting. Thus, the deterioration in performance is explained by the fact that the 5 PCs are not sufficient to allow the model to generalize accurately.

Instead, through the application of LDA, a dataset with 284 descriptors was obtained. In order to find the most suitable number of descriptors to increase the performance of the model, a code was created using the Python programming language. This code made it possible to automatically generate an RF model for each number of descriptors, from 2 to 284, reporting the results in terms of MCC value (Table 4). The best model was obtained with 239 descriptors, as demonstrated by the MCC values 0.610, 0.627, and 0.619 for the training, test, and validation sets, respectively. However, these values are lower than the results of the RF2D model. Therefore, the initial 1640 descriptors were retained in the subsequent steps of model optimization.

Table 4. MCC values of the RF3D model build with 239 descriptors. MCC, Matthews correlation coefficient

Dataset	RF3D with 239 descriptors		
Set	<i>Training</i>	<i>Test</i>	<i>Validation</i>
MCC	0.610	0.627	0.619

The process of model optimization was continued by modifying the activity cutoff value, thus varying the ratio between active and inactive compounds in the dataset. Comparing the results of the various validation sets, represented in Table 5, for cutoff values smaller than 1 μ M, the ACC values increased proportionally as the cutoff value decreased. Actually, the values increase from 85.308% in the dataset with cutoff value

0.75 μM , to 87.329% in the dataset with cutoff value 0.5 μM , to 89.322% in the dataset with cutoff value 0.3 μM .

Initially, it could be thought that the performance of the model is improving. However, when looking at the MCC values, these show a significant decrease, reporting values of 0.581 (0.75 μM), 0.590 (0.5 μM) and 0.584 (0.3 μM) respectively. This phenomenon can be explained by observing the specificity and sensitivity values. In fact, the model correctly classifies 93% positive and 59% negative samples in the 0.75 μM cutoff value data set, 94% positive and 58% negative samples in the 0.5 μM data set, and 95% positive and 56% negative samples for the 0.3 μM data set. Therefore, by decreasing the cutoff value, the model specializes in the correct classification of negative samples and commits more errors in the classification of positive samples.

Table 5. Comparison of the prediction performance on the validation set of the RF2D model with different cutoff values. ACC, accuracy; MCC, Matthews correlation coefficient; ROC AUC, area under receiver operating characteristic curve; TP, true positive; TN, true negative; FP, false positive; FN, false negative.

Cutoff value of RF2D	Set	ACC %	MCC	ROC AUC	Sensitivity %	Specificity %	TP	TN	FP	FN
0.3 μM	Validation	89.322	0.584	0.763	56.849	95.799	332	2805	123	252
0.5 μM	Validation	87.329	0.590	0.797	58.640	94.923	431	2636	141	304
0.75 μM	Validation	85.308	0.581	0.787	59.746	93.686	518	2478	167	349
1 μM	Validation	86.159	0.625	0.789	63.554	94.246	490	2031	124	281
2.5 μM	Validation	82.916	0.637	0.805	69.265	91.686	942	1941	176	418
5 μM	Validation	80.125	0.602	0.801	79.976	80.278	1426	1388	341	357
7.5 μM	Validation	79.784	0.584	0.789	84.291	73.695	1701	1101	393	317
10 μM	Validation	78.986	0.558	0.774	85.266	69.602	1794	980	428	310

Since the ACC only considers correct predictions over total predictions, it increased as the correct classification of negative samples increased. However, the MCC, which evaluates correct performance in all elements of the confusion matrix, does not

undergo the same increase, proving to be a more reliable metric when the ratio of active to inactive compounds becomes dramatically unbalanced.

On the other hand, for cutoff values greater than 1 μM , both ACC and MCC values decrease proportionally as the cutoff value increases, demonstrating an overall deterioration in model performance. Noteworthy are the specificity and sensitivity values on the 5 μM data set, where the ratio of positive to negative cases is 50% and both classes are equally represented. In this case, the model correctly classified about 80% of both negative and positive samples. However, the overall metrics are lower compared to the 1 μM data set, as demonstrated by ACC (80.125%) and MCC (0.602). Collectively, these data show that the 1 μM cutoff value allows the model to perform a more accurate classification. From now on, I used 1 μM cutoff value for all generated models.

The model optimization process continued by adding to the RF2D dataset (cutoff value 1 μM , one experimental condition, and 1640 deviation descriptors) two additional experimental conditions. Their relative deviation descriptors were calculated through the Box-Jenkins approach. In this manner, two new datasets were generated. The first one, having two experimental conditions (2COND RF), relative to the cell line and the time point, and the second one having three experimental conditions (3COND RF) relative to the cell line, time point, and type of assay.

Because of the Box-Jenkins moving average approach, both datasets contain a higher number of molecular descriptors than the previous RF2D, 3280 and 4920 descriptors, respectively.

As reported in Table 6, the introduction of a second set of deviation descriptors related to the time point condition does not significantly improve the ability of the classifier. In fact, if in the training and test set the accuracy exceeds 89%, in the validation set it returns to 86%, exactly as in the dataset with only one condition. The same trend concerns the MCC, which was 0.734 in the test set and 0.636 in the validation set, and ROC AUC, 0.854 in the test set and 0.791 in the validation set. This difference in metrics indicates the presence of overfitting.

Table 6. Comparison of the performance between RF2D, 2COND RF and 3COND RF. ACC, accuracy; MCC, Matthews correlation coefficient; ROC AUC, area under receiver operating characteristic curve; TP, true positive; TN, true negative; FP, false positive; FN, false negative.

Dataset	Set	ACC %	MCC	ROC AUC	Sensitivity%	Specificity%	TP	TN	FP	FN
RF2D (one condition)	<i>Training</i>	86.128			94.865	62.722	1272	5154	279	756
	<i>Test</i>	86.333	0.635	0.798	66.0	93.692	231	906	61	119
	<i>Validation</i>	86.159	0.625	0.789	63.554	94.246	490	2031	124	281
2COND RF	<i>Training</i>	89.149			94.872	72.915	1233	4451	246	458
	<i>Test</i>	89.714	0.734	0.854	76.072	94.830	337	1119	61	106
	<i>Validation</i>	86.195	0.636	0.791	63.389	94.843	606	2391	130	350
3COND RF	<i>Training</i>	87.115			94.313	67.208	1158	4494	271	565
	<i>Test</i>	88.293	0.693	0.835	73.441	93.697	318	1115	75	115
	<i>Validation</i>	88.151	0.690	0.832	72.484	93.905	677	2388	155	257

In contrast, the introduction of the second and third set of deviation descriptors related to the time point and type of assay significantly improved the predictive performance of the RF, which achieves an ACC of 88% (Table 6). Moreover, ACC values of the training, test, and validation set (87.115%, 88.293%, and 88.151% respectively), and the MCC values of the test and validation set (0.693 and 0.690 respectively) are more consistent with each other, indicating that there is no presence of overfitting. While specificity remains almost unchanged compared to RF2D, sensitivity gains 10% points. Thus, while negative sample classification remains stable, the model increases the number of positive samples it can correctly classify to 72% (compared to 63% for RF2D). This is reflected in a better ability to separate the two classes of samples, which is also shown by the increase in the ROC AUC value to 0.832 (versus 0.789 in RF2D). Therefore, 3COND RF was used for the subsequent optimization steps.

The high number of molecular descriptors in the 3COND RF model required the application of dimensionality reduction techniques once again in order to understand whether its performance could be improved. The techniques investigated in this phase were information theory-based feature selection and GA-k-NN.

Table 7. Comparison of the performance of the model 3COND RF with different numbers of descriptors selected through information theory-based feature selection. ACC, accuracy; MCC, Matthews correlation coefficient; ROC AUC, area under receiver operating characteristic curve; TP, true positive; TN, true negative; FP, false positive; FN, false negative.

Number of descriptors of 3COND RF	Set	ACC %	MCC	ROC AUC	Sensitivity%	Specificity%	TP	TN	FP	FN
20	Training	82.522			93.473	52.245	900	4454	311	823
	Test	83.857	0.566	0.763	60.046	92.521	260	1101	89	173
	Validation	82.168	0.517	0.735	54.818	92.214	512	2345	198	422
15	Training	84.541			93.956	58.503	1008	4477	288	715
	Test	85.521	0.6121	0.785	63.510	93.529	275	1113	77	158
	Validation	84.757	0.590	0.769	59.850	93.905	559	2388	155	375
10	Training	82.922			93.242	54.382	937	4443	322	786
	Test	83.980	0.564	0.754	57.044	93.781	247	1116	74	186
	Validation	82.427	0.521	0.732	53.212	93.158	497	2369	174	437
5	Training	79.162			91.752	44.341	764	4372	393	959
	Test	80.653	0.465	0.705	48.729	92.269	211	1098	92	222
	Validation	79.321	0.427	0.686	45.503	91.742	425	2333	210	509

The reduction of descriptors through information theory-based feature selection resulted in four datasets consisting of 20, 15, 10, and 5 descriptors. The results of all four datasets (Table 7) exhibit lower metrics than the model with 4920 descriptors. The best results were obtained on the dataset with 15 descriptors, which showed an ACC of 84.541%, 85.521% and 84.757% in the training, testing and validation set, respectively. In contrast, the MCC values are 0.612 and 0.590 for the test and validation sets. Although this model does not match the performance of 3COND RF with 4920 descriptors, the level of ACC and MCC it achieved is remarkable considering

the drastic reduction in the number of descriptors.

Through the GA-k-NN approach, several datasets were generated with 1619, 1350, 1220, 691, 597, 202 and 86 descriptors.

Table 8. Comparison of the performance of the model 3COND RF with different numbers of descriptors selected through GA-k-NN. ACC, accuracy; MCC, Matthews correlation coefficient; ROC AUC, area under receiver operating characteristic curve; TP, true positive; TN, true negative; FP, false positive; FN, false negative.

Number of descriptors of 3COND RF	Set	ACC %	MCC	Incorrect predictions	ROC AUC	Sensitivity %	Specificity %	TP	TN	FP	FN
4920	Validation	88.151	0.690	412	0.832	72.484	93.905	677	2388	155	257
1619	Validation	88.151	0.690	412	0.830	72.055	94.062	673	2392	151	261
1350	Validation	88.093	0.688	414	0.831	72.270	93.905	675	2388	155	259
1220	Validation	88.093	0.581	414	0.831	72.270	93.686	673	2390	153	261
691	Validation	87.949	0.684	419	0.830	72.163	94.246	674	2384	159	260
597	Validation	87.863	0.681	422	0.826	71.306	93.944	666	2389	154	268
202	Validation	88.007	0.685	417	0.827	71.306	94.140	666	2394	149	268
86	Validation	87.403	0.666	438	0.813	68.094	94.950	636	2403	140	298

Reducing the number of descriptors from 4920 of 3COND RF to 1619 does not lead to any significant change in performance. Indeed, as can be seen from the results obtained on the validation set (Table 8), ACC (88.151%) and MCC (0.690) are completely identical to those of 3COND RF. This result indicates that the 3COND RF dataset contains at least 3301 descriptors that are completely irrelevant for classification purposes.

Examining the results obtained on the validation sets with 1350 and 1220 descriptors, it may be noted that both models lead to identical results for ACC (88.093%) and MCC (0.688), which are only slightly lower than those of the previous model. Given the similarity of the results of these first three models, to facilitate the identification of the best predictive model, the total number of incorrect predictions was considered. The model with 1619 descriptors misclassified 412 samples out of 3477, while both models with 1350 and 1220 descriptors had 414 misclassified samples. The model generated with 1619 descriptors misclassified 412 samples out of 3477, while both models with 1350 and 1220 descriptors had 414 misclassified samples.

The models generated with 691, 597, and 86 descriptors decreased slightly in performance, although their overall ACC never fell below 86%. This result is very successful considering the drastic reduction in features. However, the classification errors increase, corresponding to 419, 422 and 438 respectively out of 3477 samples. The model generated with 202 descriptors has ACC (88.007%) and MCC (0.685) values that are closest to the models with 1619, 1350 and 1220 descriptors. However, it commits more classification errors (417). Based on these results, it was decided to use the model with 1619 descriptors. Although the performance of the model with 202 descriptors was similar and the reduced number of descriptors would certainly speed up calculation operations, the model with 1619 descriptors preserves performance to a maximum and has a number of descriptors that still allows calculation operations to be performed in a fairly efficient time.

The validation set used to evaluate the performance of the finalized mt-QSAR model was also used to evaluate the applicability domain (AD) of the model. Through the confidence estimation approach, 757 outlier compounds were identified, approximately 22% of the total (3477). By rejecting those compounds and evaluating the model again, the performance has significantly improved. Indeed, as reported in Table 9, before the implementation of AD the ACC of the validation set was 88.151%, while after AD it increased to 94.301%. Similarly, the MCC increased from 0.689 before AD to 0.817 after AD. There was also a significant increase in sensitivity, which varied from 72.055% (before AD) to 80.256% (after AD). Although the increase in

performance came at the expense of a reduction in molecules, the model can be said to be robust and reliable.

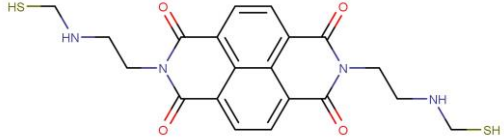
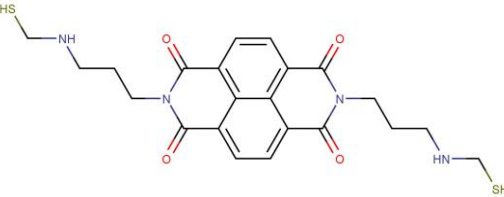
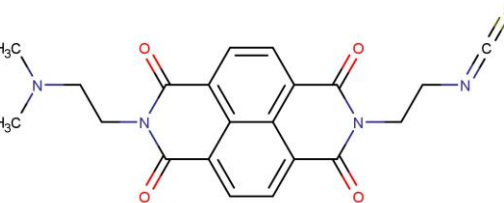
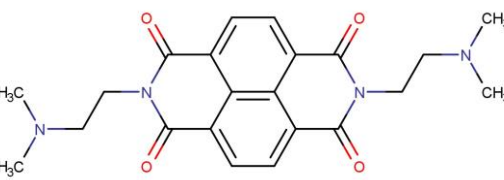
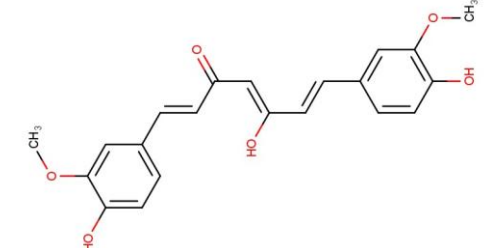
Table 9. Comparison of the performance of the finalized mt-QSAR model before and after the application of AD. ACC, accuracy; MCC, Matthews correlation coefficient; ROC AUC, area under receiver operating characteristic curve; TP, true positive; TN, true negative; FP, false positive; FN, false negative.

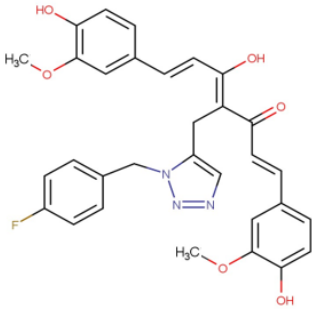
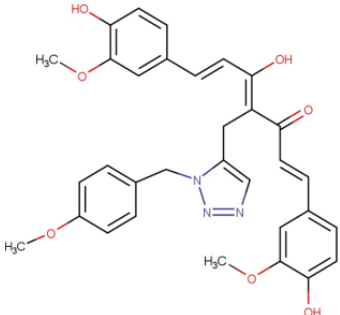
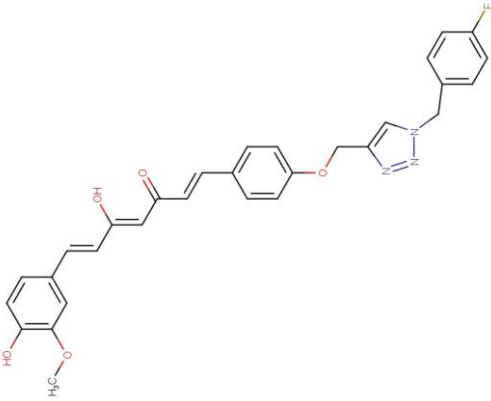
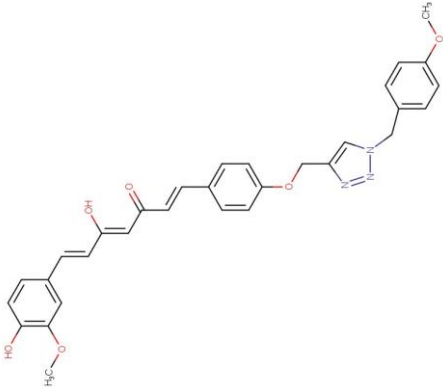
Finalized mt-QSAR	Set	ACC %	MCC	Incorrect predictions	ROC AUC	Sensitivity %	Specificity %	TP	TN	FP	FN
Before AD	Validation	88.151	0.690	412	0.830	72.055	94.062	673	2392	151	261
After AD	Validation	94.301	0.817	155	0.890	80.256	97.837	439	2126	47	108

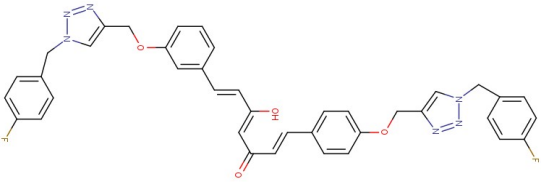
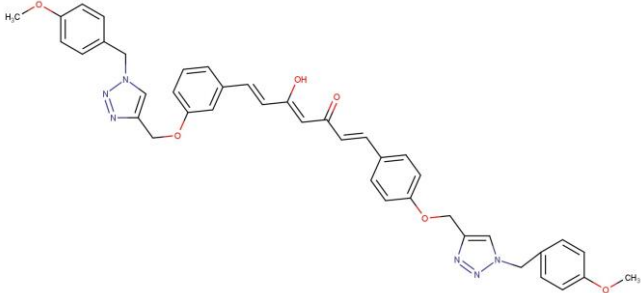
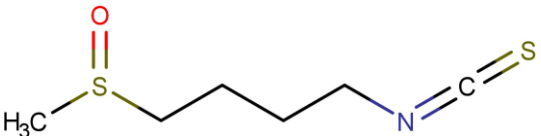
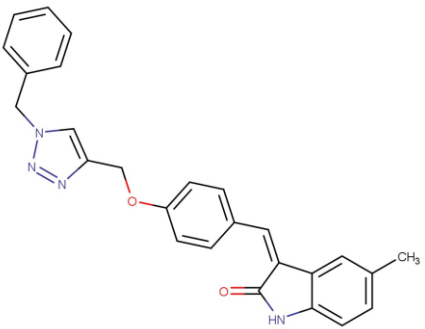
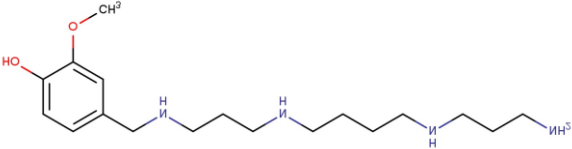
5.3 VS

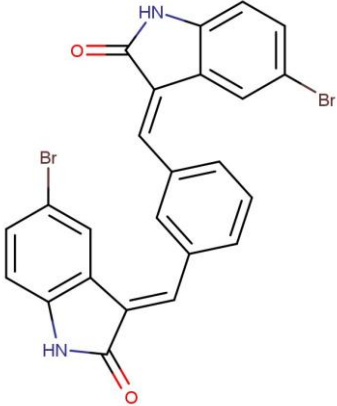
The purpose of my research project was not only the construction and optimization of a mt-QSAR classification model, but also confirmation of its generalization performance on an external validation set for VS purposes. For a final validation, an external dataset of 36 cases was assembled. Actually, the number of different molecules composing the dataset (reported in Table 10) is 15, but since some of them were tested against more than one cell line or at different time points, this results in 36 different cases. Although the dataset has small dimension, it ensures structural variety. All the molecules have been previously evaluated by our research group *in vitro* in term of cytotoxic activity against one or more hematological cancer cells, and for some of them the results have already been published [133–137]. The activity range of these compounds varies from 0.43 μM to 142.79 μM , and only 2 molecules were found to be active at concentrations less than or equal to 1 μM , while 34 were active at concentrations higher than 1 μM , and therefore considered inactive. Neither of the molecules was present in the training set.

Table 10. Molecular representation of the 15 compounds used to build the dataset employed for VS.

Structure	Target (cell line)	Time point	Reference
	<ul style="list-style-type: none"> Jurkat 	<ul style="list-style-type: none"> 24h 	[133]
	<ul style="list-style-type: none"> Jurkat 	<ul style="list-style-type: none"> 24h 	[133]
	<ul style="list-style-type: none"> Jurkat 	<ul style="list-style-type: none"> 24h 	[133]
	<ul style="list-style-type: none"> Jurkat 	<ul style="list-style-type: none"> 24h 	[133]
	<ul style="list-style-type: none"> CCRF-CEM 	<ul style="list-style-type: none"> 24h 48h 	[134]

	<ul style="list-style-type: none"> • CCRF-CEM 	<ul style="list-style-type: none"> • 24h • 48h 	[134]
	<ul style="list-style-type: none"> • CCRF-CEM 	<ul style="list-style-type: none"> • 24h • 48h 	[134]
	<ul style="list-style-type: none"> • CCRF-CEM 	<ul style="list-style-type: none"> • 24h • 48h 	[134]
	<ul style="list-style-type: none"> • CCRF-CEM 	<ul style="list-style-type: none"> • 24h • 48h 	[134]

	<ul style="list-style-type: none"> • CCRF-CEM 	<ul style="list-style-type: none"> • 24h • 48h 	[134]
	<ul style="list-style-type: none"> • CCRF-CEM 	<ul style="list-style-type: none"> • 24h • 48h 	[134]
	<ul style="list-style-type: none"> • MOLM-13 • MV4-11 • OCI-AML3 • U-937 	<ul style="list-style-type: none"> • 24h • 48h • 72h 	[135]
	<ul style="list-style-type: none"> • CEM • Jurkat 	<ul style="list-style-type: none"> • 24h • 48h 	[136]
	<ul style="list-style-type: none"> • HL-60 	<ul style="list-style-type: none"> • 24h 	Not published

	<ul style="list-style-type: none"> • Jurkat 	<ul style="list-style-type: none"> • 24h 	[137]
---	--	---	-------

To assemble the dataset, molecular structure of the compounds were drawn with Marvin Sketch version 21.1.0, ChemAxon (<https://www.chemaxon.com>), and merged in one single SD format file through BIOVIA Discovery Studio Visualizer (<https://discover.3ds.com/discovery-studio-visualizer-download>). The file was submitted to alvaDesc version 2.2 where molecular descriptors from 0D to 2D were calculated for each case. The column Toxicity was then added at the dataset, and Boolean variables corresponding to the actual activity of the molecules were assigned at each case. Three further columns were added, concerning the three experimental conditions included in the mt-QSAR model dataset, i.e. cell line, time point and type of assay used for the *in vitro* assays. It should be specified that no new conditions values were introduced in this validation set, all of them were present in the training set conditions.

Finally, the dataset built was subjected to QSAR-Co-X for the calculation of the deviation descriptors. Since this dataset for the VS contains 3 experimental conditions, 3 series of the input descriptors were obtained, corresponding to 4920 deviation descriptors. However, in order to employ the finalized mt-QSAR model to predict the activities of the compounds, the VS dataset and the mt-QSAR model must contain the same number and type of descriptors. Therefore, of the 4920 deviation descriptors calculated, only the 1619 contained in the finalized mt-QSAR model were selected.

For the VS task, the model correctly classified all active cases (2) and all inactive cases (34), returning perfect metrics, as reported in Table 10. The result obtained can be considered impressive according to the complexity of the modeled biological response and the structural diversity of the molecules of the dataset.

Since the molecules had already been tested for their cytotoxic activity against hematological cancer cells, our assays confirm that the prediction of the model are 100% accurate.

Table 11. Results of the VS performed with the finalized mt-QSAR on a laboratory dataset. ACC, accuracy; MCC, Matthews correlation coefficient; ROC AUC, area under receiver operating characteristic curve; TP, true positive; TN, true negative; FP, false positive; FN, false negative.

Laboratory dataset	ACC %	MCC	ROC AUC	Sensitivity %	Specificity %	TP	TN	FP	FN
	100	1.0	1.0	100	100	2	34	0	0

6. Conclusions and future perspectives

The aim of my Ph.D. project was to build an mt-QSAR model to identify novel compounds with cytotoxic activity against several hematological cancer cells in order to speed up the early stages of the drug discovery process.

A large and diverse dataset including 11704 molecules tested against 43 hematological cancer cell lines was constructed and used to train a predictive model. 0D to 2D molecular descriptors were calculated to describe the activities of the molecules, and a cutoff value was used to allow the model to discriminate between active and inactive molecules. Additionally, the Box-Jenkins moving average approach was applied, allowing the model to evaluate the cytotoxicity of the molecules not only in relation to their physicochemical properties but, also, according to their target, i.e. the hematological cancers cell line. Five classification ML algorithms were utilized, identifying RF as the one capable of generating the best predictive model. Different approaches, including dimensionality reduction methods such as PCA, LDA, GA-k-NN, and information theory-based feature selection were employed to reduce the dimensionality of the dataset and improve the mt-QSAR model predictive performance. Through the GA-k-NN approach, it was possible to reduce the model dimensions to 1619, improving its performance. Additionally, during model optimization, further approaches were employed to modify certain characteristics of the dataset. The first involved a modification of the cutoff value, which nevertheless confirmed that the value of 1 μ M chosen initially was the best one for performance purposes. The second approach involved the calculation of two further sets of deviation descriptors, using the Box-Jenkins approach, which also enabled the model to discriminate the activity of molecules according to the time point and the type of assay used to assess their biological activity.

All these efforts led to a good final mt-QSAR classification model which can predict the biological activity of molecules based on their behavior towards the target, a time point and an assay type. The goodness of the model was confirmed by its ACC which

reaches 88% in the validation set and increases to 94% if the model is applied within its AD.

This classification model was used for the VS of a small laboratory dataset where it correctly classified all molecules, both active and inactive, laying the groundwork for the prediction of cytotoxic molecules against hematological cancer cells through the use of artificial intelligence methods. Indeed, this model is the first (to our knowledge) capable of discriminating molecules active against 43 hematological cancer cell lines and two further additional experimental conditions.

Despite the encouraging result of VS, it would be desirable to evaluate the performance of the model in classifying a larger and balanced dataset, namely one with a ratio of active to inactive molecules close to 50%. Therefore, as a future work, the laboratory dataset will be expanded by adding more positive molecules, either tested in our laboratories or appeared in recent scientific publications. Afterwards, the expanded dataset will be virtually screened again by the mt-QSAR model.

In case the model would confirm the accuracy of its predictions, its use will be extended for VS of larger databases, both on laboratory and commercial compounds.

For this purpose, a dataset of molecules tested in our laboratories on cell lines different from hematological cancers is being ultimate. The mt-QSAR model will be used to predict the activity of the molecules, and only those eventually found to be active will be tested on hematological cancer cells through wet laboratory experiments to unequivocally assess the real performance of the model.

7. References

1. Hanahan, D.; Weinberg, R.A. Hallmarks of Cancer: The Next Generation. *Cell* **2011**, *144*, 646–674.
2. Ferlay, J.; Soerjomataram, I.; Dikshit, R.; Eser, S.; Mathers, C.; Rebelo, M.; Parkin, D.M.; Forman, D.; Bray, F. Cancer Incidence and Mortality Worldwide: Sources, Methods and Major Patterns in GLOBOCAN 2012. *Int J Cancer* **2015**, *136*, E359–386.
3. Prasad, V.; Mailankody, S. Research and Development Spending to Bring a Single Cancer Drug to Market and Revenues After Approval. *JAMA Intern Med* **2017**, *177*, 1569–1575.
4. Mughal, T.I.; Mughal, T.; Goldman, J.; Goldman, J.M.; Mughal, S.T.; Mughal, S. *Understanding Leukemias, Lymphomas and Myelomas*; CRC Press, 2013;
5. Burke, V.P.; Startzell, J.M. The Leukemias. *Oral Maxillofac Surg Clin North Am* **2008**, *20*, 597–608.
6. Sanganalmath, S.K.; Abdel-Latif, A.; Bolli, R.; Xuan, Y.-T.; Dawn, B. Hematopoietic Cytokines for Cardiac Repair: Mobilization of Bone Marrow Cells and Beyond. *Basic Res Cardiol* **2011**, *106*, 709–733.
7. Mugnaini, E.N.; Ghosh, N. Lymphoma. *Prim Care* **2016**, *43*, 661–675.
8. *Basic & Clinical Pharmacology*; Katzung, B.G., Vanderah, T.W., Eds.; A Lange medical book; Fifteenth edition.; McGraw-Hill: New York Chicago San Francisco Athens London Madrid Mexico City Milan New Delhi Singapore Sydney Toronto, 2021;
9. Minami, H.; Kiyota, N.; Kimbara, S.; Ando, Y.; Shimokata, T.; Ohtsu, A.; Fuse, N.; Kuboki, Y.; Shimizu, T.; Yamamoto, N.; et al. Guidelines for Clinical Evaluation of Anti-cancer Drugs. *Cancer Sci* **2021**, *112*, 2563–2577.
10. Chu, E. Cancer Chemotherapy. In *Basic & Clinical Pharmacology*, 14e; Katzung, B.G., Ed.; McGraw-Hill Education: New York, NY, 2017.
11. Sever, R.; Brugge, J.S. Signal Transduction in Cancer. *Cold Spring Harb Perspect Med* **2015**, *5*, a006098–a006098.
12. *Tyrosine Kinases as Druggable Targets in Cancer*; Ren, H., Ed.; IntechOpen, 2019;
13. Azevedo, A.; Silva, S.; Rueff, J. Non-Receptor Tyrosine Kinases Role and Significance in Hematological Malignancies. In *Tyrosine Kinases as Druggable Targets in Cancer*; Ren, H., Ed.; IntechOpen, 2019.
14. Sudhesh Dev, S.; Zainal Abidin, S.A.; Farghadani, R.; Othman, I.; Naidu, R. Receptor Tyrosine Kinases and Their Signaling Pathways as Therapeutic Targets of Curcumin in Cancer. *Front Pharmacol* **2021**, *12*, 772510.
15. Cardoso, H.J.; Figueira, M.I.; Socorro, S. The Stem Cell Factor (SCF)/c-KIT Signalling in Testis and Prostate Cancer. *J Cell Commun Signal* **2017**, *11*, 297–307.
16. Sochacka-Ćwikła, A.; Mączyński, M.; Regiec, A. FDA-Approved Small Molecule Compounds as Drugs for Solid Cancers from Early 2011 to the End of 2021. *Molecules* **2022**, *27*, 2259.
17. Hantschel, O.; Grebien, F.; Superti-Furga, G. The Growing Arsenal of ATP-Competitive and Allosteric Inhibitors of BCR–ABL. *Cancer Res* **2012**, *72*, 4890–4895.
18. Shepherd, P.; Suffolk, R.; Halsey, J.; Allan, N. Analysis of Molecular Breakpoint and M-RNA Transcripts in a Prospective Randomized Trial of Interferon in Chronic Myeloid Leukaemia: No Correlation with Clinical Features, Cytogenetic Response, Duration of Chronic Phase, or Survival. *Br J Haematol* **1995**, *89*, 546–554.
19. Gambacorti-Passerini, C.; Piazza, R. Imatinib—A New Tyrosine Kinase Inhibitor for First-Line Treatment of Chronic Myeloid Leukemia in 2015. *JAMA Oncol* **2015**, *1*, 143.
20. Keretsu, S.; Ghosh, S.; Cho, S.J. Molecular Modeling Study of C-KIT/PDGFR α Dual Inhibitors for the Treatment of Gastrointestinal Stromal Tumors. *Int J Mol Sci* **2020**, *21*, 8232.
21. Sierra, J.R.; Cepero, V.; Giordano, S. Molecular Mechanisms of Acquired Resistance to Tyrosine Kinase Targeted Therapy. *Mol Cancer* **2010**, *9*, 75.

22. Ciarcia, R.; Damiano, S.; Puzio, M.V.; Montagnaro, S.; Pagnini, F.; Pacilio, C.; Caparrotti, G.; Bellan, C.; Garofano, T.; Polito, M.S.; et al. Comparison of Dasatinib, Nilotinib, and Imatinib in the Treatment of Chronic Myeloid Leukemia. *J Cell Physiol* **2016**, *231*, 680–687.
23. Reardon, D.A.; Wen, P.Y.; Mellingshoff, I.K. Targeted Molecular Therapies against Epidermal Growth Factor Receptor: Past Experiences and Challenges. *Neuro-Oncol* **2014**, *16*, viii7–viii13.
24. Zahavi, D.; Weiner, L. Monoclonal Antibodies in Cancer Therapy. *Antibodies* **2020**, *9*, 34.
25. Gauthier, L.; Vivier, E. Boosting Cytotoxic Antibodies against Cancer. *Cell* **2020**, *180*, 822–824.
26. Rimawi, M.F.; Schiff, R.; Osborne, C.K. Targeting HER2 for the Treatment of Breast Cancer. *Annu Rev Med* **2015**, *66*, 111–128.
27. Prasad, S.; Gupta, S.C.; Aggarwal, B.B. Serendipity in Cancer Drug Discovery: Rational or Coincidence? *Trends Pharmacol Sci* **2016**, *37*, 435–450.
28. Aronson, J.K.; Green, A.R. Me-too Pharmaceutical Products: History, Definitions, Examples, and Relevance to Drug Shortages and Essential Medicines Lists. *Br J Clin Pharmacol* **2020**, *86*, 2114–2122.
29. Hughes, J.; Rees, S.; Kalindjian, S.; Philpott, K. Principles of Early Drug Discovery. *Br J Pharmacol* **2011**, *162*, 1239–1249.
30. Sun, D.; Gao, W.; Hu, H.; Zhou, S. Why 90% of Clinical Drug Development Fails and How to Improve It? *Acta Pharm. Sin. B* **2022**, *12*, 3049–3062.
31. Carnero, A. High Throughput Screening in Drug Discovery. *Clin Transl Oncol* **2006**, *8*, 482–490.
32. Sinha, S.; Vohora, D. Drug Discovery and Development. In *Pharmaceutical Medicine and Translational Clinical Research*; Elsevier, 2018.
33. Hait, W.N. Anticancer Drug Development: The Grand Challenges. *Nat Rev Drug Discov* **2010**, *9*, 253–254.
34. Liu, Z.; Delavan, B.; Roberts, R.; Tong, W. Lessons Learned from Two Decades of Anticancer Drugs. *Trends Pharmacol Sci* **2017**, *38*, 852–872.
35. Smalley, K.S.M.; Lioni, M.; Herlyn, M. Life Isn't Flat: Taking Cancer Biology to the next Dimension. *Vitro Cell Dev Biol Anim* **2006**, *42*, 242.
36. Mak, I.W.; Evaniew, N.; Ghert, M. Lost in Translation: Animal Models and Clinical Trials in Cancer Treatment. *Am J Transl Res* **2014**, *6*, 114–118.
37. Kapetanovic, I.M. Computer-Aided Drug Discovery and Development (CADD): In Silico-Chemico-Biological Approach. *Chem Biol* **2008**, *171*, 165–176.
38. Shaker, B.; Ahmad, S.; Lee, J.; Jung, C.; Na, D. In Silico Methods and Tools for Drug Discovery. *Comput Biol Med* **2021**, *137*, 104851.
39. Eagling, V.A.; Back, D.J.; Barry, M.G. Differential Inhibition of Cytochrome P450 Isoforms by the Protease Inhibitors, Ritonavir, Saquinavir and Indinavir. *Br J Clin Pharmacol* **1997**, *44*, 190–194.
40. Hartman, G.D.; Egbertson, M.S.; Halczenko, W.; Laswell, W.L.; Duggan, M.E.; Smith, R.L.; Naylor, A.M.; Manno, P.D.; Lynch, R.J.; Zhang, G. Non-Peptide Fibrinogen Receptor Antagonists. 1. Discovery and Design of Exosite Inhibitors. *J Med Chem* **1992**, *35*, 4640–4642.
41. Smith, J.S.; Roitberg, A.E.; Isayev, O. Transforming Computational Drug Discovery with Machine Learning and AI. *ACS Med Chem Lett* **2018**, *9*, 1065–1069.
42. Akamatsu, M. Current State and Perspectives of 3D-QSAR. *Curr Top Med Chem* **2002**, *2*, 1381–1394.
43. Tropsha, A. Best Practices for QSAR Model Development, Validation, and Exploitation. *Mol Inf.* **2010**, *29*, 476–488.
44. Todeschini, R.; Consonni, V. *Handbook of Molecular Descriptors*; Wiley-VCH: Weinheim, 2011;

45. Andrade, C.H.; Pasqualoto, K.F.M.; Ferreira, E.I.; Hopfinger, A.J. 4D-QSAR: Perspectives in Drug Design. *Molecules* **2010**, *15*, 3281–3294.
46. Roy, K.; Kar, S.; Das, R. *A Primer on QSAR/QSPR Modeling: Fundamental Concepts*; Springer, 2015;
47. *Drug Design: Structure- and Ligand-Based Approaches*; Merz, Jr, K.M., Ringe, D., Reynolds, C.H., Eds.; 1st ed.; Cambridge University Press, 2010;
48. Vamathevan, J.; Clark, D.; Czodrowski, P.; Dunham, I.; Ferran, E.; Lee, G.; Li, B.; Madabhushi, A.; Shah, P.; Spitzer, M.; et al. Applications of Machine Learning in Drug Discovery and Development. *Nat Rev Drug Discov* **2019**, *18*, 463–477.
49. Alloghani, M.; Al-Jumeily, D.; Mustafina, J.; Hussain, A.; Aljaaf, A.J. A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science. In *Supervised and Unsupervised Learning for Data Science*; Berry, M.W., Mohamed, A., Yap, B.W., Eds.; Unsupervised and Semi-Supervised Learning; Springer International Publishing: Cham, 2020.
50. Dayan, P.; Niv, Y. Reinforcement Learning: The Good, The Bad and The Ugly. *Curr Opin Neurol* **2008**, *18*, 185–196.
51. Hady, M.F.A.; Schwenker, F. Semi-Supervised Learning. In *Handbook on Neural Information Processing*; Bianchini, M., Maggini, M., Jain, L.C., Eds.; Intelligent Systems Reference Library; Springer Berlin Heidelberg: Berlin, Heidelberg, 2013; Vol. 49.
52. Mitchell, J.B.O. Machine Learning Methods in Chemoinformatics. *WIREs Comput Mol Sci* **2014**, *4*, 468–481.
53. Gimeno, A.; Ojeda-Montes, M.; Tomás-Hernández, S.; Cereto-Massagué, A.; Beltrán-Debón, R.; Mulero, M.; Pujadas, G.; Garcia-Vallvé, S. The Light and Dark Sides of Virtual Screening: What Is There to Know? *Int J Mol Sci* **2019**, *20*, 1375.
54. Hamza, A.; Wei, N.-N.; Zhan, C.-G. Ligand-Based Virtual Screening Approach Using a New Scoring Function. *J Chem Inf Model* **2012**, *52*, 963–974.
55. Polgar, T.; M. Keseru, G. Integration of Virtual and High Throughput Screening in Lead Discovery Settings. *Comb Chem* **2011**, *14*, 889–897.
56. Carracedo-Reboredo, P.; Liñares-Blanco, J.; Rodríguez-Fernández, N.; Cedrón, F.; Novoa, F.J.; Carballal, A.; Maojo, V.; Pazos, A.; Fernandez-Lozano, C. A Review on Machine Learning Approaches and Trends in Drug Discovery. *Comput Struct Biotechnol J* **2021**, *19*, 4538–4558.
57. Speck-Planche, A.; Cordeiro, M.N.D.S. Multi-Target QSAR Approaches for Modeling Protein Inhibitors. Simultaneous Prediction of Activities Against Biomacromolecules Present in Gram-Negative Bacteria. *Curr Top Med Chem* **2015**, *15*, 1801–1813.
58. Kleandrova, V.V.; Scotti, M.T.; Scotti, L.; Nayarisseri, A.; Speck-Planche, A. Cell-Based Multi-Target QSAR Model for Design of Virtual Versatile Inhibitors of Liver Cancer Cell Lines. *SAR QSAR Env. Res* **2020**, *31*, 815–836.
59. Kutalik, Z.; Beckmann, J.S.; Bergmann, S. A Modular Approach for Integrative Analysis of Large-Scale Gene-Expression and Drug-Response Data. *Nat Biotechnol* **2008**, *26*, 531–539.
60. Riddick, G.; Song, H.; Ahn, S.; Walling, J.; Borges-Rivera, D.; Zhang, W.; Fine, H.A. Predicting *in Vitro* Drug Sensitivity Using Random Forests. *Bioinformatics* **2011**, *27*, 220–224.
61. Menden, M.P.; Iorio, F.; Garnett, M.; McDermott, U.; Benes, C.H.; Ballester, P.J.; Saez-Rodriguez, J. Machine Learning Prediction of Cancer Cell Sensitivity to Drugs Based on Genomic and Chemical Properties. *PLoS ONE* **2013**, *8*, e61318.
62. Ammad-ud-din, M.; Georgii, E.; Gönen, M.; Laitinen, T.; Kallioniemi, O.; Wennerberg, K.; Poso, A.; Kaski, S. Integrative and Personalized QSAR Analysis in Cancer by Kernelized Bayesian Matrix Factorization. *J Chem Inf Model* **2014**, *54*, 2347–2359.
63. Cortés-Ciriano, I.; van Westen, G.J.P.; Bouvier, G.; Nilges, M.; Overington, J.P.; Bender, A.; Malliavin, T.E. Improved Large-Scale Prediction of Growth Inhibition Patterns Using the NCI60 Cancer Cell Line Panel. *Bioinformatics* **2015**, btv529.

64. Wang, L.; Li, X.; Zhang, L.; Gao, Q. Improved Anticancer Drug Response Prediction in Cell Lines Using Matrix Factorization with Similarity Regularization. *BMC Cancer* **2017**, *17*, 513.
65. Bediaga, H.; Arrasate, S.; González-Díaz, H. PTML Combinatorial Model of ChEMBL Compounds Assays for Multiple Types of Cancer. *ACS Comb Sci* **2018**, *20*, 621–632.
66. Speck-Planche, A.; Kleandrova, V.V.; Luan, F.; Cordeiro, M.N.D.S. Fragment-Based QSAR Model toward the Selection of Versatile Anti-Sarcoma Leads. *Eur J Med Chem* **2011**, *46*, 5910–5916.
67. Speck-Planche, A.; Kleandrova, V.V.; Luan, F.; Cordeiro, M.N.D.S. Multi-Target Drug Discovery in Anti-Cancer Therapy: Fragment-Based Approach toward the Design of Potent and Versatile Anti-Prostate Cancer Agents. *Bioorg Med Chem* **2011**, *19*, 6239–6244.
68. Speck-Planche, A.; Kleandrova, V.V.; Luan, F.; Cordeiro, M.N.D.S. Chemoinformatics in Anti-Cancer Chemotherapy: Multi-Target QSAR Model for the in Silico Discovery of Anti-Breast Cancer Agents. *Eur J Pharm Sci* **2012**, *47*, 273–279.
69. Speck-Planche, A.; Kleandrova, V.V.; Luan, F.; Cordeiro, M.N.D.S. Chemoinformatics in Multi-Target Drug Discovery for Anti-Cancer Therapy: In Silico Design of Potent and Versatile Anti-Brain Tumor Agents. *Anticancer Agents Med Chem* **2012**, *12*, 678–685.
70. Speck-Planche, A.; Kleandrova, V.V.; Luan, F.; Cordeiro, M.N.D.S. Rational Drug Design for Anti-Cancer Chemotherapy: Multi-Target QSAR Models for the in Silico Discovery of Anti-Colorectal Cancer Agents. *Bioorg Med Chem* **2012**, *20*, 4848–4855.
71. Planche, A.S.-; Kleandrova, V.V.; Luan, F.; Cordeiro, M.N.D.S. Unified Multi-Target Approach for the Rational in Silico Design of Anti-Bladder Cancer Agents. *Anti Cancer Agents Med Chem* **2013**, *13*, 791–800.
72. Bairoch, A. The Cellosaurus, a Cell-Line Knowledge Resource. *J Biomol Tech* **2018**, *29*, 25–38.
73. Gaulton, A.; Bellis, L.J.; Bento, A.P.; Chambers, J.; Davies, M.; Hersey, A.; Light, Y.; McGlinchey, S.; Michalovich, D.; Al-Lazikani, B.; et al. ChEMBL: A Large-Scale Bioactivity Database for Drug Discovery. *Nucleic Acids Res.* **2012**, *40*, D1100–D1107.
74. Weininger, D. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *J Chem Inf Model* **1988**, *28*, 31–36.
75. Xu, Y.; Goodacre, R. On Splitting Training and Validation Set: A Comparative Study of Cross-Validation, Bootstrap and Systematic Sampling for Estimating the Generalization Performance of Supervised Learning. *J Anal Test* **2018**, *2*, 249–262.
76. Warr, W.A. Representation of Chemical Structures. *WIREs Comput Mol Sci* **2011**, *1*, 557–579.
77. Fourches, D.; Muratov, E.; Tropsha, A. Trust, But Verify: On the Importance of Chemical Structure Curation in Cheminformatics and QSAR Modeling Research. *J Chem Inf Model* **2010**, *50*, 1189–1204.
78. O'Boyle, N.M.; Banck, M.; James, C.A.; Morley, C.; Vandermeersch, T.; Hutchison, G.R. Open Babel: An Open Chemical Toolbox. *J Cheminform* **2011**, *3*, 33.
79. Dalby, A.; Nourse, J.G.; Hounshell, W.D.; Gushurst, A.K.I.; Grier, D.L.; Leland, B.A.; Laufer, J. Description of Several Chemical Structure File Formats Used by Computer Programs Developed at Molecular Design Limited. *J Chem Inf Comput Sci* **1992**, *32*, 244–255.
80. Grisoni, F.; Ballabio, D.; Todeschini, R.; Consonni, V. Molecular Descriptors for Structure–Activity Applications: A Hands-On Approach. In *Computational Toxicology*; Nicolotti, O., Ed.; Methods in Molecular Biology; Springer New York: New York, NY, 2018; Vol. 1800.
81. Todeschini, R.; Consonni, V.; Gramatica, P. Chemometrics in QSAR. In *Comprehensive Chemometrics*; Elsevier, 2009.
82. Basak, S.C.; Gute, B.D.; Grunwald, G.D. Use of Topostructural, Topochemical, and Geometric Parameters in the Prediction of Vapor Pressure: A Hierarchical QSAR Approach. *J Chem Inf Comput Sci* **1997**, *37*, 651–655.
83. Yap, C.W. PaDEL-Descriptor: An Open Source Software to Calculate Molecular Descriptors and Fingerprints. *J Comput Chem* **2011**, *32*, 1466–1474.

84. Moriwaki, H.; Tian, Y.-S.; Kawashita, N.; Takagi, T. Mordred: A Molecular Descriptor Calculator. *J Cheminform* **2018**, *10*, 4.
85. Steinbeck, C.; Han, Y.; Kuhn, S.; Horlacher, O.; Luttmann, E.; Willighagen, E. The Chemistry Development Kit (CDK): An Open-Source Java Library for Chemo- and Bioinformatics. *J Chem Inf Comput Sci* **2003**, *43*, 493–500.
86. Mauri, A. AlvaDesc: A Tool to Calculate and Analyze Molecular Descriptors and Fingerprints. In *Ecotoxicological QSARs*; Roy, K., Ed.; Methods in Pharmacology and Toxicology; Springer US: New York, NY, 2020.
87. Cao, D.-S.; Xu, Q.-S.; Hu, Q.-N.; Liang, Y.-Z. ChemoPy: Freely Available Python Package for Computational Biology and Chemoinformatics. *Bioinformatics* **2013**, *29*, 1092–1094.
88. Cao, D.-S.; Liang, Y.-Z.; Yan, J.; Tan, G.-S.; Xu, Q.-S.; Liu, S. PyDPI: Freely Available Python Package for Chemoinformatics, Bioinformatics, and Chemogenomics Studies. *J Chem Inf Model* **2013**, *53*, 3086–3096.
89. Cao, D.-S.; Xiao, N.; Xu, Q.-S.; Chen, A.F. Rcpri: R/Bioconductor Package to Generate Various Descriptors of Proteins, Compounds and Their Interactions. *Bioinformatics* **2015**, *31*, 279–281.
90. Cunningham, P.; Delany, S.J. Underestimation Bias and Underfitting in Machine Learning. In *Trustworthy AI - Integrating Learning, Optimization and Reasoning*; Heintz, F., Milano, M., O'Sullivan, B., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, 2021; Vol. 12641.
91. Ying, X. An Overview of Overfitting and Its Solutions. *J Phys Conf Ser* **2019**, *1168*, 022022.
92. Halder, A.K.; Dias Soeiro Cordeiro, M.N. QSAR-Co-X: An Open Source Toolkit for Multitarget QSAR Modelling. *J Cheminform* **2021**, *13*, 29.
93. OECD *Guidance Document on the Validation of (Quantitative) Structure-Activity Relationship [(Q)SAR] Models*; OECD Series on Testing and Assessment; OECD, 2014;
94. Ripley, B.D. *Pattern Recognition and Neural Networks*; 1st ed.; Cambridge University Press, 1996;
95. Halder, A.K.; Moura, A.S.; Cordeiro, M.N.D.S. Moving Average-Based Multitasking In Silico Classification Modeling: Where Do We Stand and What Is Next? *Int J Mol Sci* **2022**, *23*, 4937.
96. Kleandrova, V.V.; Ruso, J.M.; Speck-Planche, A.; Dias Soeiro Cordeiro, M.N. Enabling the Discovery and Virtual Screening of Potent and Safe Antimicrobial Peptides. Simultaneous Prediction of Antibacterial Activity and Cytotoxicity. *ACS Comb Sci* **2016**, *18*, 490–498.
97. Speck-Planche, A.; Cordeiro, M.N.D.S. Advanced In Silico Approaches for Drug Discovery: Mining Information from Multiple Biological and Chemical Data Through Mtk- QSBER and Pt-QSPR Strategies. *Curr Med Chem* **2017**, *24*.
98. Hao, J.; Ho, T.K. Machine Learning Made Easy: A Review of *Scikit-Learn* Package in Python Programming Language. *J Educ Behav Stat* **2019**, *44*, 348–361.
99. Belete, D.M.; Huchaiah, M.D. Grid Search in Hyperparameter Optimization of Machine Learning Models for Prediction of HIV/AIDS Test Results. *Int J Comput Appl* **2022**, *44*, 875–886.
100. Wong, T.-T.; Yeh, P.-Y. Reliable Accuracy Estimates from k -Fold Cross Validation. *IEEE Trans Knowl Data Eng* **2020**, *32*, 1586–1594.
101. Kramer, O. k -Nearest Neighbors. In *Dimensionality Reduction with Unsupervised Nearest Neighbors*; Intelligent Systems Reference Library; Springer Berlin Heidelberg: Berlin, Heidelberg, 2013; Vol. 51.
102. Cover, T.; Hart, P. Nearest Neighbor Pattern Classification. *IEEE Trans Inf. Theory* **1967**, *13*, 21–27.
103. Piccialli, V.; Sciandrone, M. Nonlinear Optimization and Support Vector Machines. *Ann Oper Res* **2022**, *314*, 15–47.
104. Pisner, D.A.; Schnyer, D.M. Support Vector Machine. In *Machine Learning*; Elsevier, 2020.

105. Rokach, L. *Pattern Classification Using Ensemble Methods*; World Scientific Pub. Co.: Singapore, 2010;
106. Schonlau, M.; Zou, R.Y. The Random Forest Algorithm for Statistical Learning. *Stata J.* **2020**, *20*, 3–29.
107. Kingsford, C.; Salzberg, S.L. What Are Decision Trees? *Nat Biotechnol* **2008**, *26*, 1011–1013.
108. Natekin, A.; Knoll, A. Gradient Boosting Machines, a Tutorial. *Front Neurorobot* **2013**, *7*.
109. Zhang, C.; Zhang, Y.; Shi, X.; Alpanidis, G.; Fan, G.; Shen, X. On Incremental Learning for Gradient Boosting Decision Trees. *Neural Process Lett* **2019**, *50*, 957–987.
110. Rosenblatt, F. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychol. Rev.* **1958**, *65*, 386–408.
111. Taud, H.; Mas, J.F. Multilayer Perceptron (MLP). In *Geomatic Approaches for Modeling Land Change Scenarios*; Camacho Olmedo, M.T., Paegelow, M., Mas, J.-F., Escobar, F., Eds.; Lecture Notes in Geoinformation and Cartography; Springer International Publishing: Cham, 2018.
112. Bengio, Y. Learning Deep Architectures for AI. *FNT Mach. Learn.* **2009**, *2*, 1–127.
113. Chicco, D.; Jurman, G. The Advantages of the Matthews Correlation Coefficient (MCC) over F1 Score and Accuracy in Binary Classification Evaluation. *BMC Genomics* **2020**, *21*, 6.
114. Afshar, M.; Usefi, H. Optimizing Feature Selection Methods by Removing Irrelevant Features Using Sparse Least Squares. *Expert Syst Appl* **2022**, *200*, 116928.
115. Verleysen, M.; François, D. The Curse of Dimensionality in Data Mining and Time Series Prediction. In *Computational Intelligence and Bioinspired Systems*; Cabestany, J., Prieto, A., Sandoval, F., Eds.; Lecture Notes in Computer Science; Springer Berlin Heidelberg: Berlin, Heidelberg, 2005; Vol. 3512.
116. Huang, X.; Wu, L.; Ye, Y. A Review on Dimensionality Reduction Techniques. *Int J Patt Recogn Artif Intell* **2019**, *33*, 1950017.
117. Khalid, S.; Khalil, T.; Nasreen, S. A Survey of Feature Selection and Feature Extraction Techniques in Machine Learning. In Proceedings of the 2014 Science and Information Conference; IEEE: London, UK, August 2014; pp. 372–378.
118. Karamizadeh, S.; Abdullah, S.M.; Manaf, A.A.; Zamani, M.; Hooman, A. An Overview of Principal Component Analysis. *JSIP* **2013**, *04*, 173–175.
119. Martinez, A.M.; Kak, A.C. PCA versus LDA. *IEEE Trans Pattern Anal Mach. Intel* **2001**, *23*, 228–233.
120. Shannon, C.E. A Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423.
121. Godden, J.W.; Bajorath, J. Chemical Descriptors with Distinct Levels of Information Content and Varying Sensitivity to Differences between Selected Compound Databases Identified by SE-DSE Analysis. *J Chem Inf Comput Sci* **2002**, *42*, 87–93.
122. Vogt, M.; Wassermann, A.M.; Bajorath, J. Application of Information—Theoretic Concepts in Chemoinformatics. *Information* **2010**, *1*, 60–73.
123. Urias, R.W.P.; Barigye, S.J.; Marrero-Ponce, Y.; García-Jacas, C.R.; Valdes-Martini, J.R.; Perez-Gimenez, F. IMMAN: Free Software for Information Theory-Based Chemometric Analysis. *Mol Divers* **2015**, *19*, 305–319.
124. Stahura, F.L.; Godden, J.W.; Bajorath, J. Differential Shannon Entropy Analysis Identifies Molecular Property Descriptors That Predict Aqueous Solubility of Synthetic Compounds with High Accuracy in Binary QSAR Calculations. *J Chem Inf Comput Sci* **2002**, *42*, 550–558.
125. Speck-Planche, A. Multicellular Target QSAR Model for Simultaneous Prediction and Design of Anti-Pancreatic Cancer Agents. *ACS Omega* **2019**, *4*, 3122–3132.

126. Murray-Smith, D.J. *Modelling and Simulation of Integrated Systems in Engineering: Issues of Methodology, Quality, Testing, and Application*; Woodhead Publishing: Oxford ; Philadelphia, 2012;
127. Griswold, M.D.; Hunt, P.A. Meiosis. In *Brenner's Encyclopedia of Genetics*; Elsevier, 2013.
128. Jones, G.H.; Franklin, F.C.H. Meiotic Crossing-over: Obligation and Interference. *Cell* **2006**, *126*, 246–248.
129. Gramatica, P. Principles of QSAR Models Validation: Internal and External. *QSAR Comb Sci* **2007**, *26*, 694–701.
130. Tropsha, A.; Golbraikh, A. Predictive QSAR Modeling Workflow, Model Applicability Domains, and Virtual Screening. *Curr Pharm Des* **2007**, *13*, 3494–3504.
131. Mathea, M.; Klingspohn, W.; Baumann, K. Chemoinformatic Classification Methods and Their Applicability Domain. *Mol Inf* **2016**, *35*, 160–180.
132. Rahman, M.M.; Davis, D.N. Addressing the Class Imbalance Problem in Medical Datasets. *Int J Mach Learn Comput* **2013**, 224–228.
133. Minarini, A.; Milelli, A.; Tumiatti, V.; Ferruzzi, L.; Marton, M.-R.; Turrini, E.; Hrelia, P.; Fimognari, C. Design, Synthesis and Biological Evaluation of New Naphtalene Diimides Bearing Isothiocyanate Functionality. *Eur J Med Chem* **2012**, *48*, 124–131.
134. Seghetti, F.; Di Martino, R.M.C.; Catanzaro, E.; Bisi, A.; Gobbi, S.; Rampa, A.; Canonico, B.; Montanari, M.; Krysko, D.V.; Papa, S.; et al. Curcumin-1,2,3-Triazole Conjugation for Targeting the Cancer Apoptosis Machinery. *Molecules* **2020**, *25*, 3066.
135. Greco, G.; Schnekenburger, M.; Catanzaro, E.; Turrini, E.; Ferrini, F.; Sestili, P.; Diederich, M.; Fimognari, C. Discovery of Sulforaphane as an Inducer of Ferroptosis in U-937 Leukemia Cells: Expanding Its Anticancer Potential. *Cancers* **2021**, *14*, 76.
136. Das, A.; Greco, G.; Kumar, S.; Catanzaro, E.; Morigi, R.; Locatelli, A.; Schols, D.; Alici, H.; Tahtaci, H.; Ravindran, F.; et al. Synthesis, in Vitro Cytotoxicity, Molecular Docking and ADME Study of Some Indolin-2-One Linked 1,2,3-Triazole Derivatives. *Comput Biol Chem* **2022**, *97*, 107641.
137. Morigi, R.; Catanzaro, E.; Locatelli, A.; Calcabrini, C.; Pellicioni, V.; Leoni, A.; Fimognari, C. Synthesis and Biological Evaluation of New Bis-Indolinone Derivatives Endowed with Cytotoxic Activity. *Molecules* **2021**, *26*, 6277, doi:10.3390/molecules26206277.