Improving a Database of Cyanobacterial Bioactive Compounds that can be used for Therapeutic Approaches in Human Diseases

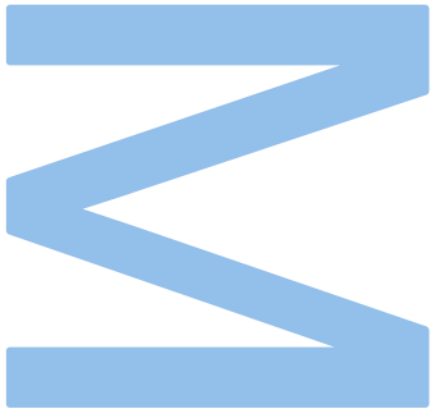
Renato Soares
Masters in bioinformatics and computational Biology
Department in Computer Sciences | FCUP
2023

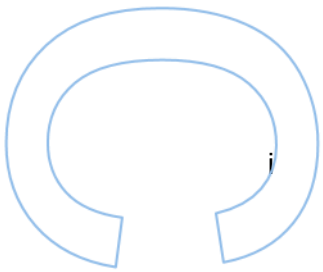
Supervisor
João Carneiro, Researcher, Interdisciplinary Centre of Marine and Environmental Research (CIIMAR), University of Porto, Porto, Portugal

Cosupervisor
Sérgio Sousa, Category, UCIBIO – Applied Molecular Biosciences Unit, BioSIM – Department of Biomedicine, Faculty of Medicine

Cosupervisor
Diogo Pratas, IEETA/DETI/LASI, University of Aveiro, 3810-193 Aveiro, Portugal


U.PORTO
FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO





SWORN STATEMENT

I, Renato Jorge Sousa Soares, enrolled in the master's degree Bioinformatics and Computational Biology at the Faculty of Sciences of the University of Porto hereby declare, in accordance with the provisions of paragraph a) of Article 14 of the Code of Ethical Conduct of the University of Porto, that the content of this dissertation reflects perspectives, research work and my own interpretations at the time of its submission.

By submitting this dissertation, I also declare that it contains the results of my own research work and contributions that have not been previously submitted to this or any other institution.

I further declare that all references to other authors fully comply with the rules of attribution and are referenced in the text by citation and identified in the bibliographic references section. This dissertation does not include any content whose reproduction is protected by copyright laws.

I am aware that the practice of plagiarism and self-plagiarism constitute a form of academic offense.

Renato Jorge Sousa Soares

30/06/2023

Acknowledgements

This master's project would have been a lot harder if not for the wonderful people I had by my side and those working with me.

I want to give special thanks to professor João Carneiro for the help and energy provided during this journey that was this project, thank you for being so thoughtful and such a good person and always believing in me and being so positive and forward thinking not letting me get stuck in the difficulties I had during the thesis.

I would also like to thank Professor Sergio Sousa, Diogo Pratas and Sérgio Crisóstomo for their input and help during the project.

Additionally, a mention to all the people working at P2A.06 office at CIIMAR for being so supporting in everything.

I would also like to thank my friend, Software Engineer João Bernardo, for also providing his opinion on my work and being an example of a hard and dedicated worker.

Finally, I want to also thank my family, my father Armando, my mom Silvina, my brothers Hugo and Nuno, my sister-in-law Sonia, and my well-behaved nephews for always supporting me during my life.

Resumo

Os metabolitos secundários encontrados em blooms de cianobactérias têm sido relatados em diversos estudos.

Esses compostos bioativos de cianobactérias podem ter aplicações em muitas áreas incluindo cosméticos naturais, antienvelhecimento, abordagens terapêuticas anticancerígenas, atividade antibacteriana, obesidade e biofilmes. Neste projeto foram abordadas duas hipóteses principais: 1) Podemos construir um banco de dados de compostos bioativos de cianobactérias que inclua todas as informações moleculares e químicas de diferentes fontes online; 2) Implementação de modelos de aprendizagem de maquina usando o banco de dados final para prever alvos de proteínas com aplicação em terapêutica humana. Para abordar esta hipótese, os bancos de dados online moleculares e químicos mais recentes que incluem compostos de cianobactérias foram fundidos em um banco de dados final. As bases de dados foram unidas usando como chave primaria as InchIKeys que são uma hash dos InchI (Internacional Chemical Identifier). Além disso um algoritmo de aprendizagem de máquina foi implementado usando os descritores moleculares e químicos calculados pelos softwares PaDEL-descriptor, Mordred e DrugTax para cada um dos compostos bioativos de cianobactérias armazenados no banco de dados final.

O banco de dados final contém para cada composto o respetivo SMILES (isomérico), InchIKeys, taxonomia, fonte original do banco de dados, bioensaio experimental, IC50 e alvo. Os descritores químicos foram calculados e adicionados a cada registo de composto bioativo usando o descritor PaDEL, Mordred e DrugTax. Esses descritores moleculares e químicos permitiram a determinação dos compostos de cianobactérias mais relevantes para fins terapêuticos em humanos usando uma implementação de aprendizagem de máquina. O banco de dados final está disponível em um servidor online em https://cyanobioactivedb.jcresearchteam.com/.

Abstract

The secondary metabolites found in cyanobacteria blooms have been reported in several studies.

These cyanobacteria bioactive compounds can have applications in many fields, including natural antiaging cosmetics, anticancer therapeutic approaches, antibacterial activity, obesity, and biofilms. In this project, two main hypotheses were addressed: 1) Can we build a cyanobacteria bioactive compounds database that includes all molecular and chemical information from different online sources? 2) Can we implement machine learning models using the final database to predict protein targets with application in human therapeutics? To address this hypothesis, the most recent molecular and chemical online databases that include cyanobacteria compounds were merged in a final database. The databases were merged using as primary Key the InchIKeys of these compounds, which are a hash of the INCHI (International Chemical Identifier). Furthermore, a machine learning algorithm was implemented using the molecular and chemical descriptors calculated by PaDEL-descriptor, Mordred and DrugTax software for each of the bioactive cyanobacteria compounds stored in the final database.

The final database contains for each compound the respective SMILES (isomeric), InchIKeys, taxonomy, database original source, experimental bioassay, IC50, and target. Chemical descriptors were calculated and added to each bioactive compound record using the PaDEL-descriptor, Mordred and DrugTax. These molecular and chemical descriptors allowed the determination of the most relevant cyanobacteria compounds for therapeutic purposes in humans using a machine learning implementation. The final database is available on an online webserver at https://cyanobioactivedb.jcresearchteam.com/.

Keywords: Cyanobacteria; Bioactive compounds; Molecular and chemical descriptors; Machine learning; Online Database

Index

List of Figures

List of Charts

List of Tables

List of Abbreviations

INCHI	International Chemical Identifier
APIs	Application Programming Interface
N_2	Nitrogen
NH_3	Ammonia
NRPs	Nonribosomal peptides
NRPSs	Nonribosomal peptide synthetases
PCP	Peptide carrier protein
PKS	Polyketide synthases
AT	Acyltransferase
KS	Ketosynthase
mcy	Microcystin
RiPPs	Ribosomally synthesized and post translationally modified peptides.
PRPS	Post-ribosomal peptide synthesis
MEP	Methylerythritol-phosphate
IDP	Isopentenyl diphosphate
DMADP	Dimethylallyl diphosphate
GGDP	Geranyl geranyl diphosphate
PHAs	Polyhydroxyalkanoates
PHB	Polyhydroxybutyrate
VS	Virtual screening
SMILES	Simplified molecular-input line-entry system
NPAtlas	Natural Products Atlas

ADMET	Absorption, Distribution, Metabolism, Excretion and Toxicity
QSAR	Quantitative structure-activity relationship
GUI	Graphical user interface
CDK	Chemistry Development Kit
ML	Machine learning
KNN	K-Nearest Neighbours
AUC	Area under curve
CA	Classification accuracy
MCC	Mathews Correlation Coefficient
FAIR	Findability, Accessibility,Interoperability and Reusability
ROC	Receiver operating characteristic
CBCs	Cyanobacteria bioactive compounds
CID	PubChem compound identifier
CMS	Content management system

1. Introduction

1.1. Cyanobacteria

Cyanobacteria are gram-negative photoautotrophic prokaryotes capable of photosynthesis, playing a significant role in providing atmospheric oxygen, fixing nitrogen (N_2) from the atmosphere and turning it into ammonia (NH_3). Being found in a wide array of environmental conditions, they inhabit a broad range of habitats all over the world and have the adaptability to survive in extreme environments from deserts to thermal springs, hypersaline valleys and volcanic substrates (Seckbach, 2007).

The taxonomy of the cyanobacteria phylum can be established using both morphologic and genomic information. Considering the genomic information, a recent study used the 16S rRNA gene to determine the order classification. The phylogenetic tree based on the 16S rRNA allowed the division of the cyanobacteria phylum into 19 orders, as shown in the adapted Figure 1 (Strunecký, Ivanova, & Mareš, 2023).

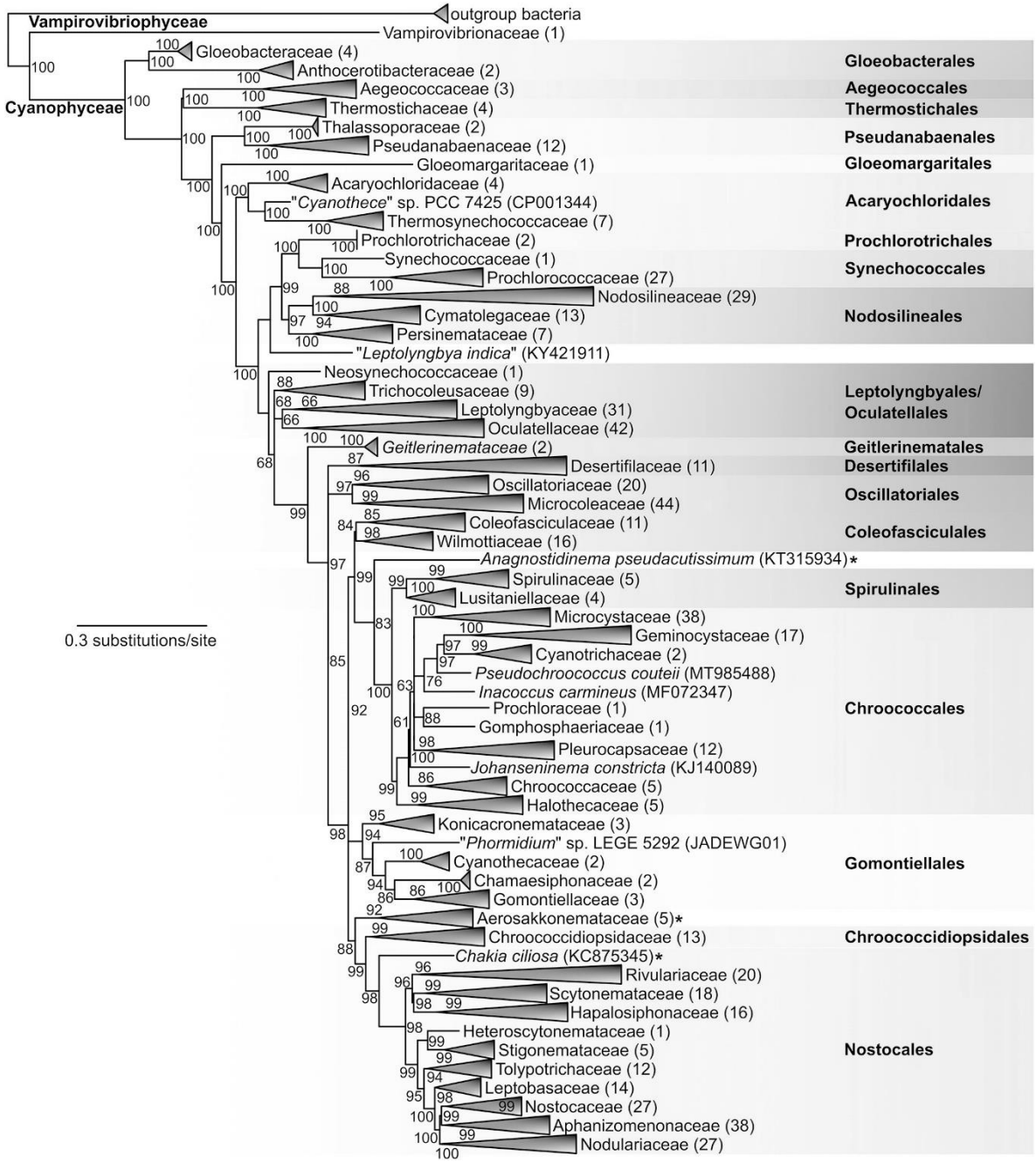


Figure 1- The 16S rRNA phylogenetic tree of cyanobacterial families and orders calculated with a maximum likelihood algorithm. Adapted from Strunecký et al. 2022.

Although monophyletic, physical differences exist between them, allowing for this group of prokaryotes to be divided into five subsections regarding their morphological characteristics. (Table 1).

Types of Compounds	Characteristics
Subsection I (Order: Chroococcales)	**Single cells or aggregates that reproduce by binary fission.**
Subsection II (Order: Pleurocapsales)	**Similar to subsection I but can undergo multiple fission producing small, dispersed cells named baeocytes.**
Subsection III (Order: Oscillatoriales)	**Filamentous cyanobacteria of this subsection have only vegetative cells. Made up of a chain of cells called trichomes, their reproduction is done through these trichome which break producing fragments called hormogonia.**
Subsection IV (Order: Nostocales)	**Have cell differentiation instead of being composed of only vegetative cells like subsection III. This specialization allows to some of the cells to acquire new functions for example, heterocysts are cells specialized in the fixations of N_2 but without producing O_2. Nitrogen fixation is an incompatible process with photosynthesis since nitrogenase is inactivated by oxygen. Among other types of specialized cells are akinetes and hormogonia.**
Subsection V (Order: Stigonematales)	**Additionally, have cell differentiation instead of being composed of only vegetative cells like subsection III. Their filaments can form heterocysts, akinetes which are nonmotile dormant cells that are resistant to cold and desiccation and hormogonia which motile filaments of cells that occur during vegetative reproduction.**

Table 1- Different subsections of Cyanobacteria and their characteristics(Kumar, Mella-Herrera, & Golden, 2010; Melinda L. Micallef, D'Agostino, Sharma, Viswanathan, & Moffitt, 2015; Tomitani, Knoll, Cavanaugh, & Ohno, 2006).

1.2. Cyanobacteria and their secondary metabolites

Cyanobacteria are one of the most promising groups of microorganisms that could play a role in pharmacology or biotechnological applications (Shaden A. M. Khalifa et al., 2021). Cyanobacterial metabolites can possess antimicrobial, antifungal, antiprotozoal, and anti-inflammatory properties that could be used to develop new drugs that could be beneficial to humans (R. K. Singh, Tiwari, Rai, & Mohapatra, 2011). The value of the discovery of natural compounds that could be used in just oncology drugs of marine

cyanobacterial origin is estimated to be anywhere from 37 to 181.5 billion dollars back in 2010 (Erwin, López-Legentil, & Schuhmann, 2010; Ramos et al., 2018).

Secondary metabolites produced by cyanobacteria are compounds that are produced by the organism but are not necessarily needed for its primary metabolism. These natural products are often produced in response to environmental stress, either abiotic or biotic, providing them with an advantage over other species. There exists an overlap between what is considered a primary and secondary metabolite since some compounds are required for primary metabolism, but since they are only synthesized by certain species, they are considered secondary metabolites. Since cyanobacteria exist in harsh environments and their metabolites originate mostly from environmental stress, cyanobacteria represent a rich source of natural products.

1.3. Types of secondary metabolites and synthesis

1.3.1. Nonribosomal peptides

Secondary metabolites that are commonly found in cyanobacteria are nonribosomal peptides (NRPs). These NRPs are produced by enzymes with multiple domains that create natural products, called nonribosomal peptide synthetases (NRPSs), through reactions that allow the assembly of proteinogenic and nonproteinogenic amino acids in a modular way. These synthetases are organized into different domains: condensation (C), responsible for catalyzing peptide formation and chain elongation; adenylation (A), responsible for substrate selection; peptide carrier protein (PCP), responsible for binding the substrate to the enzymatic complex; and thioesterase domains. To realize any future modifications, since these three domains only synthesize the raw peptide in their respective biosynthetic pathways, there can also exist additional domains that are activated, providing structural diversity. (Bethan & Carole, 2018; Marahiel, 2009; M. L. Micallef, D'Agostino, Al-Sinawi, Neilan, & Moffitt, 2015)

1.3.2. Polyketides

Polyketides are secondary metabolites such as NRPS that are produced through polyketide synthases (PKS), which utilize acetyl-CoA to produce these natural products. Furthermore, similar to NRPS, PKS also has domains including an acyltransferase (AT) domain, an acyl carrier protein domain and a ketosynthase (KS) domain as well as other domains that allow for further modification. There are three different classes of PKSs responsible for biosynthesis. Type I PKSs are more commonly found in cyanobacteria and are large multifunctional enzyme complexes that contain noninteractively acting domains. Types II and III PKSs are rarely found in cyanobacteria. Type II PKSs contain

only a single interactively acting domain, and type III PKSs are homodimeric enzymes with interactively acting domains.(M. L. Micallef et al., 2015)

In addition to these two already described secondary metabolites, they can also be NRPS and PKS hybrids, which are composed of the attachment of polyketide utilizing PKS to nonribosomal peptides in a combinatory biosynthetic pathway producing different chemical structures with specific roles and bioactivity. Various gene clusters that have NRPS/PKS and hybrid genes have been identified through whole-genome sequencing. For example, the microcystin (mcy) gene cluster encodes two PKSs, three NRPSs and a hybrid NRPS/PKS gene, and it has been found in nine Microcystis aeruginosa genomes among other species of cyanobacteria. (Bethan & Carole, 2018; Fiore et al., 2013; M. L. Micallef et al., 2015; Tanabe, Hodoki, Sano, Tada, & Watanabe, 2018)

1.3.3. Ribosomal products

Cyanobacteria also produce a group of peptides synthesized in the ribosome that, similar to NRPs, can be modified after translation. The natural products resulting from this process are named RiPPs, and they are synthesized by the postribosomal peptide synthesis (PRPS) pathway. These compounds produced by the PRPS come from a precursor peptide that is then posttranslationally altered to create the final RiPP. (Arnison et al., 2013; M. L. Micallef et al., 2015). There are various families of RiPPs whose gene clusters are reported to be in cyanobacterial genomes. For example, bacteriocins were reported to be found in up to 115 genomes of 131 cyanobacterial genomes analysed (M. L. Micallef et al., 2015). Microviridins are another family of RiPPS known to be able to inhibit proteases that have been found in 26 cyanobacterial genomes, mainly from the genera Microcystis and Planktothrix (Arnison et al., 2013; M. L. Micallef et al., 2015; Philmus, Christiansen, Yoshida, & Hemscheidt, 2008). Finally, cyanobactins are small cyclic RiPPs that are made of proteinogenic amino acids and have been identified in 32 genomes, although in some cases, they might not be functional. Cyanobactins exhibit a wide range of activities, including cytotoxic effects and drug-reversing properties. They have demonstrated potential as bioactive compounds with diverse pharmacological characteristics. Consequently, cyanobactins are an area of interest in the field of drug discovery and development.

1.3.4. Alkaloids

Alkaloids are nitrogen-containing compounds that normally have toxic properties. There are two major alkaloids produced by cyanobacteria, saxitoxin, which is a neurotoxin known for causing paralytic shellfish poisoning. In contrast to NRPS and PKS biosynthesis, saxitoxin is created through a series of monofunctional enzymes encoded

by its respective gene cluster that build the core of the saxitoxin and allow further tailoring reactions. The other major alkaloid group is the hapalindole family, which is a group of isoprenoid indole alkaloids that are found exclusively in Stigonematales (subsection V) (Bethan & Carole, 2018; M. L. Micallef et al., 2015).

1.3.5. Isoprenoids

Additionally, known as terpenoids, isoprenoids represent a large family of compounds, including carotenoids, tocopherol, phytol, sterols, and hormones. A large variety of terpenoids are produced by cyanobacteria, and these compounds play a role in the growth and survival of photosynthetic organisms. They play an important role in the conversion of light into chemical energy and in the assembly and function of photosynthetic reaction centres such as chlorophylls bacteriochlorophylls, rhodopsins, and carotenoids (Bethan & Carole, 2018; Pattanaik & Lindberg, 2015).

In cyanobacteria, they are produced through the methylerythritol-phosphate (MEP) pathway, utilizing glyceradehyde 3-phosphate and pyruvate produced by photosynthesis as substrates. This pathway leads to the creation of the five-carbon building blocks of isopentenyl diphosphate (IDP) and dimethylallyl diphosphate (DMADP), which represent the essential components for the formation of all terpenoids. A variety of these compounds have properties that are of interest in the pharmaceutical and nutritional fields and potentially as biofuels.

One group of tetraterpenes of particular importance to cyanobacteria is carotenoid pigments, a family of compounds with wide structural diversity. They are formed by two C20 geranyl geranyl diphosphate (GGDP) molecules in a head-to-head condensation reaction. These natural products play an important role in the synthesis of structural components in membranes and are essential in photosynthetic membranes and antioxidative effects and provide protection against the negative effects of free radicals, stabilizing photosynthetic reaction centers (Pattanaik & Lindberg, 2015).

1.4. Applications of cyanobacterial secondary metabolites that are currently in use.

There already exists a diverse range of cyanobacterial metabolites that serve different purposes around the world. They can play a vital role in the sustainability of the environment, agriculture, and industry. In the future, with the ever-growing human population and the usage of non-renewable energy presenting an ongoing threat, these natural products might help reduce the carbon footprint and increase sustainability as

well as improve overall human wellbeing (Pathak et al., 2018; J. S. Singh, Kumar, Rai, & Singh, 2016).

Considering the European Union's project for carbon neutrality by 2050 (https://climate.ec.europa.eu/eu-action/climate-strategies-targets/2050-long-term-strategy_en), there are potential applications of cyanobacteria that could help in this goal, especially their potential as biofuel, providing another renewable energy source and in the creation of biodegradable plastics. The same applies for Portugal's own goals in the search for carbon neutrality and its transition from fossil fuel to other renewable energy sources (https://www.portugal.gov.pt/pt/gc23/comunicacao/noticia?i=portugal-esta-em-condicoes-de-antecipar-neutralidade-carbonica-para-2045).

1.4.1. Cyanobacteria as bioremediators

These prokaryotes have the potential to be used as bioremediators since they have a few advantages over other microorganisms. In addition to being photoautotrophs and being able to capture atmospheric N_2, they have a high multiplication rate, making them adaptable and self-sufficient in surviving in highly polluted environments (J. S. Singh et al., 2016).

As such, they can be used as a tertiary treatment for agroindustry or urban effluents, helping to reduce the eutrophication of such areas. Currently, cyanobacteria are used as a low-cost tool in treating residual water from barns since they contain high amounts of N_2 and phosphorus, allowing for the conversion of these minerals into biomass. Species such as *Synechococcus elongatus, Anacystis nidulans* and *Microcystis aeruginosa* have been shown to be able to degrade many organophosphate-based insecticides and organochlorines in aquatic systems. (J. S. Singh et al., 2016; Subramaniyan, 2012)

1.4.2. Cyanobacteria as biofuel

Another application of cyanobacteria is as a source of bioenergy, such as biodiesel, or as producers of biohydrogen. The latter would be the ideal energy source to replace fossil fuels, but it is still not a profitable source of income due to the low rate of production of H_2. It can also be a source of biogas through the exploration of the anaerobic processes of these organisms, such as fermentation(J. S. Singh et al., 2016; Tiwari & Pandey, 2012).

1.4.3. Cyanobacteria as food supplements

In the form of pills or liquids, cyanobacteria can be used as food supplements due to their digestibility and richness in nutrients. In countries such as Mexico, Chile, Peru, and the Philippines, certain species are used for human nutrition, such as *Nostoc, Anabaena*

and *Spirulina. Arthrospira platensis* contains a high level of proteins, ranging over 60%, while being very rich in B_{12} and beta carotenes (Chittora, Meena, Barupal, Swapnil, & Sharma, 2020; Prasanna et al., 2010; J. S. Singh et al., 2016).

1.4.4. Cyanobacteria as an anti-pathogenic agent

Some cyanobacteria secondary compounds have a variety of potential antibacterial, antifungal, antiviral and antialgal properties. The antialgal properties perturb the physiological and metabolic activities of pathogens, inhibiting their growth (Dahms, Ying, & Pfeiffer, 2006) and allowing them to be used as biocontrol agents. For example, the compounds can help inhibit the incidence of *Botrytis cinerea*, which is a necrotrophic fungus that causes “gray mold” in strawberries (J. S. Singh et al., 2016; Swain, Paidesetty, & Padhy, 2017).

1.4.5. A source of bioplastics

Cyanobacteria have the ability to produce biopolymer polyhydroxyalkanoates (PHAs) and other copolymers, such as polyhydroxybutyrate (PHB). This PHB is a material that exhibits properties similar to polypropylene, which is commonly derived from fossil fuels such as petroleum, but unlike this material, PHB is biodegradable. This material could be implemented as an alternative to conventional plastics and reduce the worldwide impact of the nonbiodegradability of plastics as well as reduce fossil fuel dependency. Certain genera of cyanobacteria are capable of being a bio factory for bioplastics, such as *Anabaena, Synechocystis, Nostoc muscorum, and Spirulina* (Agarwal et al., 2022; A. K. Singh, Sharma, Mallick, & Mala, 2017).

1.4.6. Cyanobacteria applications in the medical and pharmaceutical field

The diverse range of compounds produced by cyanobacteria leads to the synthesis of powerful toxins as well as compounds that are also very important for their anticancer, antibiotic, anti-inflammatory and immunosuppressant effects. Furthermore, cyanobacteria have many applications in nanotechnology, either as nanoparticles of different types of metals or through the nanobiotechnological processing of their bioactive compounds. For example, *Anabaena, Calothrix, and Leptolyngbya* can modify the shape of gold, silver or palladium nanoparticles, allowing them to possess antimicrobial effects against various bacteria. In addition, silver nanoparticles also play a substantial role in impregnating medical equipment, such as surgical masks and

insertable devices, with high antimicrobial effectiveness (S. A. M. Khalifa et al., 2021; Patel, Berthold, Puranik, & Gantar, 2015; Rajeshkumar et al., 2013).

Computational techniques such as virtual screening (VS) and docking can be used to find ligand target interactions based on the binding of the system. Docking is used to predict the binding of small molecules, called ligands, with a target protein and has been previously used on cyanobacterial compounds in search of inhibitors (Sahu, Mishra, Kesheri, Kanchan, & Sinha, 2023). Virtual screening is applied to large libraries of compounds in search of compounds with drug-like properties and is used to help predict the possibility of binding to a target protein, helping in the identification of potential compounds for experimental validation.

1.5. Cyanobacteria compound databases

The number of available online sources related to cyanobacteria bioactive compounds has increased exponentially over the last years. The following databases include several pieces of information regarding these compounds.

1.5.1. CyanoMetDB

CyanoMetDB is a database developed with the goal of joining disparate information regarding cyanobacterial secondary metabolites with bioactivity potential. A freely accessible database (https://zenodo.org/record/7922070#.ZHYCbXbMJPY) was provided that contained a wide range of compounds and associated them with their chemical structure. To complete this objective, it utilized its own in-house libraries as well as other public access databases, such as CyanoMetMass (Le Manach et al., 2019), The Natural Products Atlas (van Santen et al., 2019), and Microcystins_Miles (Bouaïcha et al., 2019; R. Singh et al., 2017). This database is presented in a flat-file database containing various fields, including a compound identifier, the compound class, a simplified molecular-input line-entry system (SMILES) string and both an Internacional Chemical Identifier (InChI) and a hashed version of the InChI called InchIKey. These last three entries can be used as a textual identifier of each compound (Heller et al., 2015). In addition to these fields, some optional fields were also completed. The field “Nuclear magnetic resonance spectroscopy (NMR) used” is meant to indicate which compounds were subjected to nuclear magnetic resonance spectroscopy to confirm their structure. Fields such as “genus”, “species”, and “strain” have the purpose of providing an overview of the type of sample and its origin. “Field sample” refers to compounds that were identified in samples from blooms of one or a series of different cyanobacteria species. The field “Notes” refers to any additional information regarding the compound’s origin or

structure. All isomeric SMILES sequences present in the database were also only included when proper evidence of the compound stereochemistry was provided.

1.5.2. Natural Products Atlas 2.0

The Natural Products Atlas (NPAtlas) 2.0 is a database that continues the work of the original NPAtlas database and now contains up to 33 373 different compounds. This database can also be downloaded in a csv file format (https://www.npatlas.org/download). Each entry in NPAtlas has the chemical structure, the original isolation reference, and the organism from which it originated. Since it regards all natural products and not only those that come from cyanobacteria, this database also contains compounds from different types of bacteria and from fungi (van Santen et al., 2019; van Santen et al., 2021).

1.5.3. Pubchem and Wikidata

PubChem is a public chemical database that serves as a repository consisting of three primary databases regarding substances, compounds, and bioassay information. If a description of a chemical substance is submitted to the Substance database, their unique chemical structures are then automatically processed through structure standardization and saved into the compound database. All the data originated from hundreds of contributors and were organized into various collections organized by record type. The Substance and Bioassay are data archives, while the Compound collection is more similar to a knowledgebase for chemicals. All these data can be accessed interactively through their website (https://pubchem.ncbi.nlm.nih.gov/) (Hähnke, Kim, & Bolton, 2018; Kim et al., 2020).

To obtain any missing information that might not be available or extractable from PubChem, we utilized Wikidata since it is a common reference for many compounds in PubChem (https://pubchem.ncbi.nlm.nih.gov/source/23756).

1.5.4. ChEMBL

ChEMBL is an open large-scale bioactivity database (https://www.ebi.ac.uk/chembl/) containing information regarding the binding, functional and absorption, distribution, metabolism, excretion and toxicity (ADMET) information regarding many compounds. This data is obtained by manually extracting them from the medical literature and then procedurally curated and standardized. ChEMBL as of 2017 held approximately 1.6 million different compound structures, and 14 million activity values were provided from 1.2 million assays regarding approximately 11 thousand targets. (Gaulton et al., 2012; Gaulton et al., 2017)

For our work, ChEMBL was used to retrieve bioassay information for the compounds in our database. The values we retrieved were related to four different types of bioactivities:

a) IC50 represents the concentration of the substances required to inhibit 50% of the enzyme's activity; if the value of IC50 is low, the potency of the substance is higher.
b) Ki, which is the inhibition constant, represents the measure of the binding affinity of an enzyme and its inhibitor and represents the concentration of the inhibitor required to have 50% inhibition of an enzyme's activity. The lower the Ki value is, the stronger the binding between the enzyme and the inhibitor.
c) INH stands for inhibition presented as a percentage representing a substance that can bind to a biological molecule and decrease or inhibit its activity.
d) MIC is the minimum inhibitory concentration, which means it is the lowest concentration at which an antimicrobial agent inhibits the growth of a microorganism.

1.5.5. ChemSpider and Octaparse

ChemSpider is a completely free, online chemical database that offers its users access to different characteristics of its compounds, including its physical and chemical properties, its molecular structure and even its nomenclature. It contains up to almost 25 million unique chemical compounds that were collected from nearly 400 different data sources (Pence & Williams, 2010).

As a possibly important and popular database in the field of chemistry and in scientific research in general, the need for compounds to be correctly linked to their corresponding webpages across a different database is paramount to help users ease of use.

A way to accomplish such a task may rely on web scraping tools such as Octaparse (https://www.octoparse.com/); this tool allows its users to extract data from various websites with the benefit of not needing any coding knowledge. Its ease of use comes from its point and select system that allows its users to click on the information we want to collect and allows for the use of loops through a series of keys for mass extraction of data.

1.6. Molecular descriptors and fingerprints

1.6.1. Molecular representations and relation to descriptors

The data stored in the different previously described databases can show several features of the chemical cyanobacterial compound molecular structure. In this context,

different representations of each molecule can be used. Depending on the representation of the molecule, the kind of molecular descriptor produced is different; for example, a chemical formula represents the simplest form of representation of a molecule; as such, it is referred to as a 0D molecular descriptor. Molecular descriptors originating from a substructure list are defined as 1D molecular descriptors; they are easy to understand and are normally used for similarity/diversity analysis and virtual screening of large chemical databases. 2D molecular descriptors, also known as topological descriptors, are the most popular two-dimensional representations that contain the atomic composition and information regarding the connection of the atoms in the molecule. Another form of descriptors is 3D molecular descriptors that contain spatial information regarding the position of the atoms as well as their connection and allow for each atom to have a defined position on a three-dimensional scale.

1.6.2. Molecular descriptors and fingerprints

Molecular fingerprints are used to represent the structure of a molecule in an encoded way, normally represented in a series of binary digits that indicate the presence or lack of any structure in the molecule. There are two main categories of molecular descriptors: experimental descriptors, which represent all the experimental data, such as the octanol-water partition coefficient, polarizability and other physiochemical characteristics obtained through the specified experimental procedure. The other category is theoretical descriptors obtained by specified molecular algorithms applied to a molecular representation.

1.7. Implementation of PaDEL, Mordred and Drugtax descriptor calculators

1.7.1. PaDEL, an open-source descriptor and fingerprint calculator

PaDEL-Descriptor is open-source software that can calculate molecular descriptors and fingerprints and is able to calculate up to 1875 different descriptors for each compound, including 3D descriptors as well as 10 types of fingerprints. These descriptors are determined with the aim of providing a model for predicting the biological activity of new compounds.(Moriwaki, Tian, Kawashita, & Takagi, 2018; Yap, 2011)

This software was developed through Java language and is composed of two different components. The library component is self-contained, can function on its own and is capable of being integrated into other quantitative structure-activity relationship (QSAR) software. There is also the interface component, which provides a graphical user interface (GUI) and command line interface allowing the user to select program options and even individual types of descriptors and fingerprints to be calculated as required by

the library component. For a more specific and detailed description of how the software works, check (Yap, 2011). The reasons for selecting PaDEL-Descriptor were that it has both an easy-to-use GUI interface that simplifies the work being done and supports various platforms and a wide range of molecular file formats. Allowing us to cut the time of various conversions of file formats. Other advantages of PaDEL-Descriptor are its speed, which is faster than similar descriptor calculators such as the Chemistry Development Kit (CDK), and the ability to calculate up to ten different fingerprints more than its direct competitors. Finally, the greatest advantage of choosing this software is that it is free.

1.7.2. Mordred descriptor calculator

Mordred is a descriptor calculator capable of calculating up to 1800 different descriptors per compound with the aim of being easily installed, having a high calculation speed and including automated tests. To facilitate its own installation, Mordred uses a range of libraries that, aside from RDKIT (open-source cheminformatics library written in C++ with Python bindings) and Numpy, are coded in Python to simplify its installation, and these two libraries, although not coded in Python, are widely used in Python libraries and can be easily installed. The high number of base descriptors present in Mordred along with the ability to add more through passing parameters for new descriptors makes it a valid descriptor calculator to be used as an alternative QSAR study. Regarding performance when compared to other similar software tests, Mordred was two times faster than PaDEL-Descriptor software and as such was shown to be much more efficient.(Moriwaki et al., 2018)

The basis for picking Mordred software was its easy installation, the flexibility it provides and the relative speed at which it calculates its descriptors, allowing for quick results when running various tests.

1.7.3. DrugTax

DrugTax is a Python package for the characterization of small molecules. Utilizing SMILES as an input, we extract the taxonomic information and up to 163 features of the compounds. Some of these features will be presented in the form of categorical values that refer to different chemical definitions. The determination of whether one compound is organic or inorganic depends on it having one carbon atom, which leads to it being categorized as an organic compound, although there exists some exceptions to this rule.

1.8. Machine learning models and Orange software

Machine learning (ML) methods have been widely used to establish predictive tools considering the molecular and chemical characteristics of different types of compounds

(Keith et al., 2021). Furthermore, since the primary hypothesis of this thesis is the development of a curated database of cyanobacteria bioactive compounds and a complete set of their chemical features to be used in developing ML models. These models would serve as a prediction for possible compound-target interactions that could prove helpful in developing human therapeutic approaches.

Machine learning has been previously used in predicting binding affinity utilizing biochemical descriptors and targets of compounds in training datasets (D'Souza, Prema, & Balaji, 2020; Seko, Togo, & Tanaka, 2018). A series of different ML algorithms can be applied utilizing the chemical descriptors.

In this work, we utilized the KNN, Random Forrest, Gradient Boosting and AdaBoost ML algorithms. KNN (K-Nearest-Neighbours) is a machine learning algorithm utilized with both classification and regression that utilizes proximity to make its predictions and classifications, classifying an instance depending on most of a class of its K nearest neighbours. This algorithm can be computationally expensive when utilized on large datasets (Taunk, De, Verma, & Swetapadma, 2019; Zhang, 2016).

Random forest is an ML algorithm that utilizes decision trees to make predictions. Each of these trees is trained utilizing a random subset of data and features. Its final prediction is an averaging of each individual tree. It is an algorithm that is robust to overfitting and is able to handle large datasets but is hard to interpret (Breiman, 2001; Cutler, Cutler, & Stevens, 2011).

Gradient boosting is an ensemble learning method that utilizes decision trees, but in this case, they are used as weak learners that are combined to form a strong learner. Each new model that is trained is used to reduce the errors made by the previous tree. Overfitting more easily occurs, but it also has the potential to be more accurate (Natekin & Knoll, 2013; Xuan et al., 2019).

AdaBoost is also an ensemble learning algorithm working on the same base of addition in stages, utilizing several weak learners to obtain a strong learner. It gives higher weights to samples that are misclassified and adjusts the weights each time it utilizes a weak classifier, giving more emphasis to difficult examples (Gu, Xie, He, & Zhang, 2018; Schapire, 2013).

The implementation of ML algorithms onto calculated biochemical descriptors can be done with programs such as Orange (https://orangedatamining.com/), which is a machine learning and data mining program built on Python scripting. Similarly with Octaparse despite being built on Python it does not need any code writing skills to use

and implement. Orange offers a user-friendly design that allows users to complete a series of different tasks, including machine learning, by supporting a high number of different ML models and allowing for data preprocessing that permits the selection of the columns of most interest for our models. Additionally, it also allows us to validate and analyse our results through its test, train and validation functions.

To measure the results obtained from the ML algorithms employed above, we utilized a series of different values, including the area under the curve (AUC), classification accuracy (CA), F1 score, precision, recall and Mathews correlation coefficient (MCC).

The AUC value is the value under a receiver operating characteristic (ROC) curve and is seen as a good measure for evaluating the performance of classifiers (Melo, 2013). The classification accuracy represents the percentage of predictions our model correctly predicted. The F1 score is a harmonic mean of the precision and recall, allowing it to penalize extreme values of both. Precision is the ratio of correctly classified positive samples compared to the total number of classified positive samples. Recall is utilized to measure how well the model detects positive samples; hence, the higher the recall is, the more positive samples were detected. The final evaluation metric is the Matthews correlation coefficient; the higher the value is, the better the classification performance. It considers true positives, true negatives, false positives, and false negatives to rate the model's performance.

1.9. Main hypothesis

In short, the main question we try to answer with this work is whether it is possible to create a freely available cyanobacteria compound database that contains their chemical descriptors. Additionally, the final database will be accessible, obtainable, and reusable for other models and studies, including for example, machine learning applied to the descriptors that have been calculated. The two main hypotheses of this project are: 1) To build a free online cyanobacteria bioactive compound database that includes all chemical information from different online sources and 2) To implement machine learning models using the final database to predict protein targets with applications in human therapeutics.

2. Objectives

The primary objective of this project was to create an online database that focuses on natural cyanobacterial compounds. The database was designed to be curated, regularly updated, searchable, downloadable, and aligned with the aim of meeting the principles of Findability, Accessibility, Interoperability and Reusability (FAIR). This database includes chemical descriptors, fingerprints, and associated bioassay targets. Additionally, we have successfully accomplished several other objectives throughout the development process, which include:

1. Establish a curated database specifically dedicated to the most recent bioactive compounds derived from cyanobacteria (CBCs).
2. Develop a semiautomated workflow that utilizes data mining techniques to retrieve cyanobacteria-derived compounds from various sources, including scientific articles and databases.
3. Utilize up-to-date software tools to calculate molecular descriptors and fingerprints for each CBC.
4. Create an online database that adheres to FAIR principles by incorporating the collected information.
5. Design and implement a machine learning (ML) algorithm capable of predicting potential targets for compounds present in the database, utilizing the calculated molecular descriptors.

The overarching aim of this work is to integrate in a free online final database (CyanoBioactiveDB) the existing cyanobacterial databases and their chemical descriptors, as well as any cyanobacteria-derived compounds available in publicly accessible databases such as PubChem or ChEMBL. By calculating descriptors and fingerprints for these compounds, we intend to prepare them for future implementation in virtual screening and molecular docking calculations. Our objective is to unify and add new molecular and chemical features of each compound by identifying common elements and establishing connections that facilitate the merging of relevant information between different existing cyanobacteria databases.

By achieving these objectives, our work aims to contribute to the knowledge and research on cyanobacteria natural compounds, facilitate their discovery and exploration, and potentially enable the development of new industrial applications and human therapeutic approaches.

3. Materials and Methods

3.1. Retrieval of cyanobacteria bioactive compounds

We initially developed a workflow to retrieve the highest amount of information for each already described bioactive compound from different cyanobacteria online databases. We used the following databases considering their relevance in the field: CyanoMetDB (Jones et al., 2021), PubChem (Kim et al., 2020), NPAtlasDB (van Santen et al., 2019; van Santen et al., 2021) and ChEMBL (Mendez et al., 2018). We developed a procedure considering the different column fields in each database as follows:

1) Downloading the CyanoMetDB database file named “CyanoMetDB_v02_2023.csv” from the link https://zenodo.org/record/7922070#.ZHYQ5XbMJPZ, the file contains 2605 different compounds:
   a) To optimize our database structure, we utilize a Python script on a Jupyter notebook to curate CyanoMetDB:
      i) The number of references present in the database is not necessary for our goals, and some of the columns will not be used, so we remove them with the following command:

```
cyanodb=cyanodb.drop(["Reference_Text_No1 Title; Journal; Vol,;
Issue; pages; year; type; DOI; author1; authors2; etc.","Bibtex
No1","Ref_ID_No2","DOI_No2","Url_No2",                    "Ref-
erence_Text_No2 No Title; Journal; Vol,; Issue; pages; year;
type; DOI; author1; authors2; etc.","PubMed ID No2" ,"Bibtex
No2","Ref_ID_No3","DOI_No3", "Url_No3",
"Reference_Text_No3 Title; Journal; Vol,; Issue; pages; year;
type; DOI; author1; authors2; etc.",                    "Pub-
Med ID No3","Bibtex No3","Build block string", "Strain","Spe-
cies","Field sample","PubMed ID No1",                    "Nu-
clear magnetic resonance spectroscopy (NMR) used ","Notes
","Ref_ID_No1"],axis=1)
```

      Nevertheless, this information can be recovered using the primary key to merge the original source database information.
      ii) We resolved the problem of string values marked as "n.a." by substituting them with appropriate missing values. This adjustment avoids any conflicts that might arise during future debugging tests or analysis.
      iii) Additionally, we addressed the concern of using a semicolon (";") as a separator in the "Genus" column, which could interfere with saving the file in CSV format. To overcome this issue, we replaced the semicolon with a slash ("/") as the separator, ensuring smooth file saving in the desired format.
      To accomplish these modifications (ii and iii), we utilized the following code:

```
cyanodb=cyanodb.replace("n.a.",np.nan)
cyanodb["Genus"]=cyanodb["Genus"].str.replace(";","/")
cyanodb.to_csv("CyanometDB_Curated.csv",sep=";",index=False)
```

2) Download of the Natural Products Atlas Database (NPAtlasDB) file from their website (https://www.npatlas.org/download):
   a) Upon acquiring the database, our next step involved curating NPAtlasDB due to the presence of 33 372 compounds that were not exclusively derived from cyanobacteria:
      i) To accomplish this, we utilized a Python script that specifically selected compounds with "Bacterium" listed in the "origin_type" column. This filtering process effectively narrowed down the compounds to only those originating from bacteria.
      ii) Furthermore, we generated a comprehensive list of cyanobacteria genera and employed it to filter the "genus" column in the database. This filtering step ensured that only compounds of cyanobacterial origin remained in the dataset. As a result, we obtained a final file containing approximately 1965 distinct compounds.

3) To obtain cyanobacteria compounds, we accessed the PubChem database by visiting the PubChem website (https://pubchem.ncbi.nlm.nih.gov/) and conducting a search for "cyanobacteria compounds". After retrieving the search results, we specifically selected the compounds of interest and proceeded to download the corresponding database.

3.2. Merging databases and collecting missing information

1) Before proceeding with merging, we create for each database a column named after itself containing a string value “True” to describe if the compound is present in that database.
2) To merge the databases, we utilize the InchIKeys present in a column in each database, and as such, it is optimal to rechange the names of the columns to the same name, so the merge is done within only this one column. Utilizing the panda module merge function, it is important to define the merge to be an “outer” merge, so all the information is saved, and the corresponding rows are connected, but we do not lose the ones with no corresponding values between databases. The databases were merged in the following order:

i) CyanoMetDB merges with PubChem to form the Cyano_PubChem table containing 2922 compounds and 60 different columns.
ii) Cyano_PubChemDB then proceeded to merge with NPAtlasDB, creating the Cyano_Pubchem_Atlas data frame containing a total of 3537 compounds and 92 different columns.

3) At this stage, we create a duplicate of the Cyano_Pubchem_Atlas database called "all merges". In this copy, we address the missing values in columns related to compound presence across the databases by filling them with the value "False". Subsequently, we proceed to generate a series of new columns by merging existing columns within the database and filling any missing values with the available information. For instance, we combine data from two different columns to populate the "Isomeric Smiles" column. If a row already contains a value, we prioritize the columns originating from CyanoMetDB as the base column, as it provides the most reliable and validated data pertaining to cyanobacteria.

4) Using an Octaparse workflow, we searched the PubChem/Wikidata and ChemSpider databases to find information such as the PubChem CID and information regarding the genus and isomeric SMILES of the compounds in our database. In addition, we utilize a similar workflow for the extraction of compounds that are also present in ChemSpider. The workflow loops through the InchIKeys of the compounds in our database, making an individual search through these websites retrieving the information about each compound.
   a) PubChem information:
      i) Due to the way both Octaparse collects information and the way it is stored on PubChem, the resulting file needs to be curated. For that purpose, we created a python script that loads the file retrieved from Octaparse and curates the data by dividing the file into different tables regarding isomeric and canonical SMILES, taxa, and the PubChem compound identifier (CID), and then they are joined into one data table under the variable “curated”. The code shown below omits certain parts present in the script to help interpret the process.

```
ISO1=pubchem_octo.loc[:,"Text2"]=="isomeric SMILES"
isomeric1=pubchem_octo[ISO1].copy()

isomeric1["Isomeric Smiles"]=isomeric1["Text3"]
isomeric1["InChI"]=isomeric1["Text12"]
isomeric1["InChIKey"]=isomeric1["Text14"]
isomeric1=isomeric1.drop(isomeric1.columns[0:18],axis=1)


Curated=pd.concat([isomeric1,isomeric2])
Curated=pd.concat([Curated,isomeric3])
Curated=pd.concat([Curated,isomeric4])


CID=pubchem_octo.loc[:,"Text15"]=="PubChem CID"
ID=pubchem_octo[CID].copy()
ID["InChIKey"]=ID["Text14"]
ID["PubChem CID"]=ID["Text16"]
ID=ID.drop(ID.columns[0:18],axis=1)

Curated=pd.merge(Curated,ID,on="InChIKey",how="outer")


found_taxon1=pubchem_octo.loc[:,"Text6"]=="found in taxon"
taxon1=pubchem_octo[found_taxon1].copy()

taxon1["Genus1"]=taxon1["Text7"]
taxon1["InChIKey"]=taxon1["Text14"]
taxon1=taxon1.drop(taxon1.columns[0:18],axis=1)


all_taxa=pd.merge(taxon1,taxon2,how="outer",on="InChIKey")
all_taxa["Genus1"]=all_taxa["Genus1"].fillna(all_taxa["Genus2"])
all_taxa=all_taxa.drop(all_taxa.columns[2],axis=1)


Curated=pd.merge(Curated,all_taxa,on="InChIKey",how="outer")
Curated[["Genus","Species"]]=Curated["Genus1"].str.split(" ",1,ex-
pand=True)


CAN1=pubchem_octo.loc[:,"Text2"]=="canonical SMILES"
canonical1=pubchem_octo[ISO1].copy()


canonical1["Canonical Smiles"]=canonical1["Text3"]
canonical1["InChIKey"]=canonical1["Text14"]
canonical1=canonical1.drop(canonical1.columns[0:18],axis=1)


all_canonical=pd.concat([canonical1,canonical2])
all_canonical=pd.concat([all_canonical,canonical3])
all_canonical=pd.concat([all_canonical,canonical4])
all_canonical.drop_duplicates()

Curated=pd.merge(Curated,all_canonical,on="InChIKey",how="outer")
```

b) ChemSpider Information:

Regarding the information obtained from ChemSpider, it must be more readily obtained than PubChem and allows for a simple concatenation of the obtained file from our Octaparse workflow and the compound database.

3.3. Collecting the bioactivity information where available

Considering the importance of discriminating the number of compounds with already experimental validation using different methodologies, we updated the database to include this information. To obtain the data and merge it with the database, we used three different Python scripts:

1) The first script, “Obtain_Chembl_ID”, obtains the ChEMBL IDs of the compounds from our database by utilizing InchIKeys as a search term. To accomplish this, we built a function called “inchikey_search” that utilizes the pandas and csv modules as well as the “new_client” function from the “chembl_web_resource_client” module that is available at https://github.com/chembl/chembl_webresource_client. We utilize this function by selecting the “InchIkey_final” column as a variable and running it as a list of targets of the function.

```
def inchikey_search(targets):
    mollist = []
    df = pd.DataFrame()
    for target in targets:
        molecule = new_client.molecule
        mol = molecule.filter(molecule_structures__standard_inchi_key=target)
        if len(mol) == 0:
            new_row = {'molecule_structures': {'standard_inchi_key': target}}
            new_row = pd.json_normalize(new_row, max_level=2)
            df = df.append(new_row)
        else:
            mollist.append(pd.json_normalize(mol, max_level=2))
            df = pd.concat(mollist)
    df.to_csv("Compound_Chembl_id.csv", sep=";", index=False)
```

2) The second script, “Obtain_Chembl_Activity”, was created with the goal of utilizing newly acquired ChEMBL IDs as a search variable for a custom-built function with the goal of obtaining the respective bioactivity information. For this script, we utilized the requests, pandas, json and csv modules as well as the URL provided with the ChEMBL web services API live documentation (https://www.ebi.ac.uk/chembl/api/data/docs).

```
def find_activity_CHEMBL(target_list):
    mollist = []
    df = pd.DataFrame()
    for target in target_list:
        print(target)
        headers = {'accept': 'application/json'}
        url_stem = "https://www.ebi.ac.uk"  # This is the stem of the url
        url_full_string = url_stem + "/chembl/api/data/activity/search?q=" + target
        response = (requests.get(url_full_string,headers=headers))

        if response.status_code == 404 or response.status_code == 500:
            return print("url failed")

        else:
            response_text = response.text
            response_info = json.loads(response_text)
            print(response_info.keys())
            moleculas = response_info["activities"]
            mollist.append(pd.json_normalize(moleculas, max_level=2))

        df = pd.concat(mollist)
    df.to_csv("Compound_Chembl_Activity.csv", sep=";", index=False)
```

3) Finally, with the "Mergin_Chembl_activity" script, we load the "Compound_Chembl_Activity" csv file and create a function called "create_activity_column" that takes in 2 variables. The first is the name we want our new column to have, and the second variable is which type of activity was targeted and subsequently creates a column with only data values from that type of activity. We created the columns IC50, KI, INH, and MIC.

```
def create_activity_column(name_of_column,type_of_activity):

    activities[str(name_of_column)]=activities.query('type=
    =@type_of_activity')["value"]+activities["units"]

    activities[str(name_of_column)]=activities[str(name_of_column)]
    .fillna("None")

    activities[str(name_of_column)]=activities[str(name_of_column)]
    .astype("object")
```

```
create_activity_column("IC50","IC50")
create_activity_column("Ki","Ki")
create_activity_column("INH","INH")
create_activity_column("MIC","MIC")
```

From there, we only want to keep the activities of these four types of activity, and so we restrict our data table to only the bioassay activities regarding these 4 types. The "target_chembl_id" column is then split into two columns, one containing only the integer part of the target, so we can sort the targets properly in ascending order. The lines are then grouped by their individual "molecule_chembl_id", giving a separator for each entry. These compounds are then merged with the compound ID file retrieved in step i) and finally merge the columns of interest with our compound database through InchIKeys.

3.4. Molecular and chemical descriptor calculation

To analyse the molecular and chemical properties of each cyanobacterial compound, we employed three software programs: a) PaDEL-descriptor, b) Mordred, and c) Drugtax. Through the following workflow, thousands of chemical descriptors were calculated for the compounds:

a) PaDEL-descriptor methodology
   i) Conversion of the database to SD format was performed through DataWarrior (Sander, Freyss, von Korff, & Rufener, 2015). This process is easily performed by opening DataWarrior, creating a new file from there, creating a text column and then simply copying and appending the data from your csv/excel file to Datawarrior.
   ii) With the data loaded on DataWarrior, there were certain rows of compounds that were incorrectly displayed. They ‘re easily found by selecting the compound column and ordering in inverse order and returning to regular order. To solve this problem, we simply open the csv file in a program such as Notepad+ and search for the SMILES of the compounds wrongly displayed and proceed to remove the spaces and new lines from these rows. Then, we simply reload the csv file in Datawarrior as previously mentioned.
   iii) After successfully copying the information, select file>save special and save as an SD file. From here, select the column containing the Isomeric Smile structure, “Structure of Smile_Isomeric” and the compound name as the “Compound” column for proper identification and save the file.
   iv) Utilizing the Open Babel GUI program (O'Boyle et al., 2011), load the SD-file onto the program, ensure that the input format is the same as the file type (SD-file) and define the output format as a MOL file type.
   v) Load the MOL-file to the PaDEL-descriptor software to define an output directory and then run the PaDEL-descriptor. If at any point PaDEL fails to conduct its normal process, it is advised to proceed from step i) but load only 500 compounds into DataWarrior and create several SD files.

b) Mordred molecular descriptor calculator methodology (https://github.com/mordred-descriptor/mordred):
   i) To utilize Mordred, we need to first install it through our Anaconda command prompt with the command:

```
conda install -y -c rdkit -c mordred-descriptor mordred
```

ii) Utilizing a python script, we imported both the pandas module and Mordred. We create a descriptor calculator with the following command:

```
calc =Calculator(descriptor, ignore_3D=True)
```

iii) The database is then loaded and divided into a subset that only contains the rows containing isomeric smiles, and the index is reset.

```
cyanoter2=cyanoter.dropna(subset=["Smiles_Isomeric"])
cyanoter2=cyanoter2.reset_index(drop=True)
cyanoter2
```

iv) From there, the isomeric smiles are converted into mol values with the command:

```
mols = [Chem.MolFromSmiles(smi) for smi in cyanoter2.Smiles_Isomeric]
```

v) We then utilize the pandas module and the descriptor calculator created in step i) to calculate the descriptors of smiles in a data frame. After the calculation is complete, we join the Mordred results with the compound database:

```
df = calc.pandas(mols2)

cyanoter2_descriptors=pd.concat([cyanoter2,df],axis=1,join="inner")

cyanoter2_descriptors
```

c) DrugTax Python module methodology:

i) We initially imported the drugtax and pandas module and created a class called “drugtax_of_database” with an initializer function that referred to the file that was called, the column of the file that we wanted to analyse and the name we wanted for the output file. Another function called “run_drugtax” can be divided into two parts. The first utilizes the drugtax module on each SMILE and appends them to a list. The second part turns this list into a text file in a way that it can later be turned into a proper csv file.

```
class drugtax_of_database:
    def __init__(self,nome_do_ficheiro,indice_do_ficheiro,nome_do_output):
        self.nome_do_ficheiro=nome_do_ficheiro
        self.indice_do_ficheiro =indice_do_ficheiro
        self.nome_do_output = nome_do_output

    def run_drugtax(self):
        file=pd.read_csv(self.nome_do_ficheiro,sep=";",encoding="UTF-8")
        smiles=file[self.indice_do_ficheiro]
        count=0
        features_list=[]
        for x in smiles:
            while count<1:
                count+=1
                molecule1=drugtax.DrugTax(x)
                features=molecule1.features
                for feature in features:
                    features_list.append(str(feature))

        output=open(self.nome_do_output,"w")
        output.write("SMILE;Kingdom;Superclasses;")
        for feature in features_list:
            output.write(feature +";")
        output.write("\n")
        for smile in smiles:
            molecule=drugtax.DrugTax(str(smile))
            results = str(molecule)
            features = molecule.features
            results2 = results.replace("\n", ";")
            results3 = results2.replace("SMILE:", "")
            results4 = results3.replace("Kingdom:", "")
            results5 = results4.replace("Superclasses:", "")
            output.write(results5)
            for feature in features:
                output.write(str(features[feature]) + ";")
            output.write("\n")
        output.close()
        return
```

ii) For our database, we utilized the following code utilizing the above class:

```
final_version=drugtax_of_database("Cyanoter_V2_5.csv","Smiles_Isomeric
","Results_final_Drugtax.txt")

final_version.run_drugtax()
```

iii) The next command in the script simply turns the text file created by the class and function above into a csv file.

```
file=pd.read_csv("Results_final_Drugtax.txt",sep=";")

file.to_csv("Results_final_Drugtax.csv",sep=";",index=False,
encoding="utf-8")
```

3.5. Statistics

Due to the nature of the data table and the way information is stored, we created a new file in each entry of a genus or of a target that is correctly registered. For that purpose, we created the Python script (“Creation_of_graphics_and_targets”) that separates each line entry by the number of genera or different targets present for each compound and changes the format of the columns referring to the presence of the

compounds in the original databases. These files will allow us to build the charts utilized on our website.

```
cyano=pd.read_csv("Cyanoter_V2.csv",sep=";")

cyano2=cyano.replace("",np.nan)
cyano2=cyano2.replace("none",np.nan)
cyano2=cyano2.replace("n.a.",np.nan)

cyano2["CyanoMet_DB"]=cyano2["CyanoMet_DB"].apply(lambda x: 1 if str(x)=="True" else 0 )
cyano2["PubChem_DB"]=cyano2["PubChem_DB"].apply(lambda x: 1 if str(x)=="True" else 0 )
cyano2["NPAtlas_DB"]=cyano2["NPAtlas_DB"].apply(lambda x: 1 if str(x)=="True" else 0 )
cyano2["ChEMBL_DB"]=cyano2["ChEMBL_DB"].apply(lambda x: 1 if str(x)=="True" else 0 )

cyano2.to_csv("Cyanoter_2_for_graphics_compounds.csv",sep=";",index=False)

cyano2["Genus_of_origin"]=cyano2["Genus_of_origin"].str.replace(";","/")

cyano3=(cyano2.assign(Genus=cyano2["Genus_of_origin"].str.split('/')).explode("Ge-
nus").reset_index(drop=True))

cyano3.pop("Genus_of_origin")
cyano3.insert(2,"Genus",cyano3.pop("Genus"))
cyano3["Genus"]=cyano3["Genus"].str.replace(" ","")
cyano3.to_csv("Cyanoter_2_for_graphics_genus.csv",sep=";",index=False)


cyano4=(cyano2.assign(Individual_targets=cyano2["tar-
get_chembl_IDs"].str.split("\|\|")).explode("Individual_targets").reset_in-
dex(drop=True))

cyano4.to_csv("Cyanoter_2_for_graphics_targets.csv",sep=";",index=False)
```

3.6. Machine learning-based target file.

To make this file, similar techniques as above were implemented, including the “create_activity_column”, but in this case, only applying it to the IC50 values and their respective targets since this metric has the most associated bioassays values and as such gives us the most information possible without mixing evaluation metrics. From these IC50 bioassays we obtain a series of targets that are molecules involved in pathways leading to various diseases. Only this column was created, so we only selected compounds and targets where this inhibition metric occurred. We completed the same process as done before, but the purpose of the file created (“OnlyIC50.csv”) is to be utilized as a base file where we attach the calculated descriptors of the various programs and proceed to its proper preparation to be applied in an ML workflow.

3.7. Machine learning implementation

For implementation of the base target file in a predictive machine learning workflow, we first must separate our list of targets into a new column (“Individual_targets”) containing only one target per row. These targets represent biological complexes or cell lines, normally related to diseases,possessing an active binding site that can allow drug-like compounds to interact with them. Utilizing the targets as categorical variable for our machine learning model we implementing the descriptors as features of the model predict possible new target-compound interactions. After we reduced the number of

classes (individual targets), our file was limited to only those that appeared in a minimum of 25 instances in our database. This process resulted in11 final categorical targets. Since there was a large discrepancy between the minority and majority classes, we implemented an oversampling approach to our data, utilizing the "imblearn.over_sampling" module to make it more balanced. This approach resulted in a total of 2750 different instances of compound-target interactions. The 11 classes are represented by the 11 different targets: CHEMBL3419, CHEMBL364, CHEMBL382, CHEMBL384, CHEMBL387, CHEMBL3879801, CHEMBL391, CHEMBL396, CHEMBL398, CHEMBL399, CHEMBL612545.

Regarding the calculated descriptors, some did not produce any kind of interpretable result since at times certain values were divided by 0 through the descriptor calculator software and returned a text response, so they were removed from the database. Descriptors that had less than 80% interpretable values were removed, considerably reducing their number in both PaDEL and Mordred.

3.8. Machine learning models

To determine if the final curated database with molecular and chemical descriptors can be used to predict the putative binding affinity to the different types of targets, we implemented a bioinformatics machine learning workflow. The procedure used in another study (Carneiro et al., 2023) allowed the implementation of 4 different machine learning (ML) models: Gradient boosting, Random Forest, AdaBoost and KNN. The automatic update of the molecular and chemical descriptors allows for correct classification of the associated types of targets used in different bioassays. The methodology we used to build an ML model based upon the IC50 bioassay targets was as follows:

1. Loading of the created csv file onto the workflow and properly selecting the "Individual_targets" column as the target column and as a classifier.
2. Selection of the cyanobacterial compound molecular and chemical features that allowed the highest information gain using an Orange workflow that ranks the descriptors through 4 properties: info. Gain, Gain ratio, Gini and X^2.
3. Training of the database of cyanobacteria with 80% of the compounds from the initial database.
4. Validation of the ML models using a testing dataset (20% of the initial database).
5. A 20-fold cross-validation procedure with all the data. A 10-fold cross-validation procedure is more commonly used, but a higher number may allow for a better performance of the model.
6. ROC analysis was utilized to verify the quality of the models.

7. The top 25 descriptors of Mordred, PaDEL and Drugtax have also been saved in new data tables.

3.9. Free online Linux hosting database implementation

A free online webserver was used to make the curated cyanobacteria bioactive compounds database freely available to everyone. The website was implemented using WordPress, which is a free open-source website creation platform that is one of the easiest and most powerful website builders. WordPress websites utilize WordPress as its content management system (CMS), powering both the backend and frontend of the website. For the uploading and displaying of the data tables on our website, the installation and use of two plug-ins, the first being Big File Uploads, allows us to define our own size limit to the files that can be uploaded to the WordPress media library (https://wordpress.org/plugins/tuxedo-big-file-uploads/). The second plug-in is wpDataTable, which allows us to create graphs of the data tables and easily displays our data (https://wordpress.org/plugins/wpdatatables/). For the development of the CyanoBioactiveDB website, we selected the riverbank theme as our foundation and customized it to suit our requirements. This allowed us to effectively showcase the unique characteristics and features of our database. Editing the website is done entirely through the WordPress Gutenberg editor. This platform utilizes blocks to build upon the website so we can add any type of information in the form we wish and preview what it looks like on a PC, tablet, and phone screen. To incorporate the data tables into the media library for utilization with the wpDataTable plugin, it is necessary to upload the files in Excel format to ensure appropriate column formatting.

The availability of all source code and databases from the following methodology is available at https://cyanobioactivedb.jcresearchteam.com/cyanoterdb/download/.

4. Results

4.1. Cyanobacteria bioactive compound database

To obtain the cyanobacteria putative bioactive compounds final database (CyanoBioactiveDB), the methods to merge information between different online cyanobacteria databases were applied. Exactly 2605 compounds were obtained from the CyanoMetDB, another 403 from the PubChem database and 1965 from the NPAtlasDB. From the merging of these three databases, we ended up with a database of 3436 different compounds since some of them were the same between databases. Table 1 shows the first ten compounds in our data table. We limited it to only four columns of the database because of formatting issues caused when importing it in Microsoft Word due to the SMILES and other columns.

Compound	Compound_name	Classes_of_compounds	Genus_of_origin
Cyano_0001	Anhydrohapaloxindole B	other linear nonpeptide	Hapalosiphon
Cyano_0002	Columbamide B	other linear nonpeptide	Lyngbya/Moorea
Cyano_0003	Columbamide C	other linear nonpeptide	Lyngbya/Moorea
Cyano_0004	Columbamide D	other linear nonpeptide	Moorea
Cyano_0005	Columbamide E	other linear nonpeptide	Moorea
Cyano_0006	Companeramide A	other cyclic peptide	Leptolyngbya
Cyano_0007	Companeramide B	other cyclic peptide	Leptolyngbya
Cyano_0008	Floridamide	other cyclic peptide	Lyngbya/Moorea
Cyano_0009	Malyngamide Y	other linear nonpeptide	Lyngbya/Moorea
Cyano_0010	Laucysteinamide A	other linear nonpeptide	Caldora

Table 1- Table containing the first ten entries in the database and the first 4 columns.

From these source databases, we observed that CyanoMetDB and NPAtlasDB contained the highest number of retrieved bioactive compounds, as described in Chart 1.

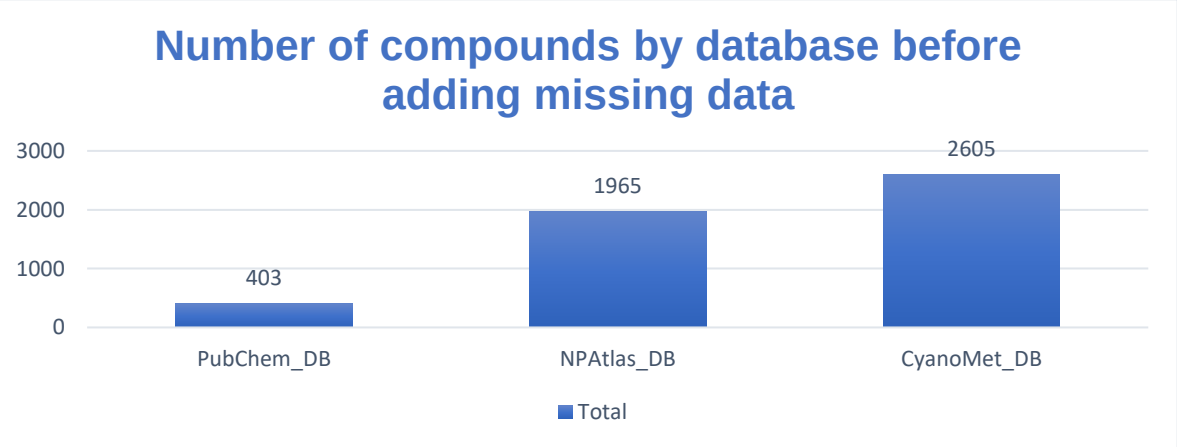


Chart 1-Distribution of the number of cyanobacteria bioactive compounds by original database source.

Nevertheless, utilizing InchIKeys as a search term for the bioactive compounds allowed us to obtain missing information not contained in the original merge and add to it the bioassay values for the compounds that were present in ChEMBL. As a result of this process, we found that many compounds were present in PubChem despite not being initially obtained through the PubChem search (Chart 2). Additionally, we confirmed that only 724 compounds had any bioassay information regarding the selected types (Chart 2). The reduction of the obtained number of numbers in both CyanoMetDB and NPAtlasDB can be explained by repeated entries that were removed and compounds not containing InchIKeys.

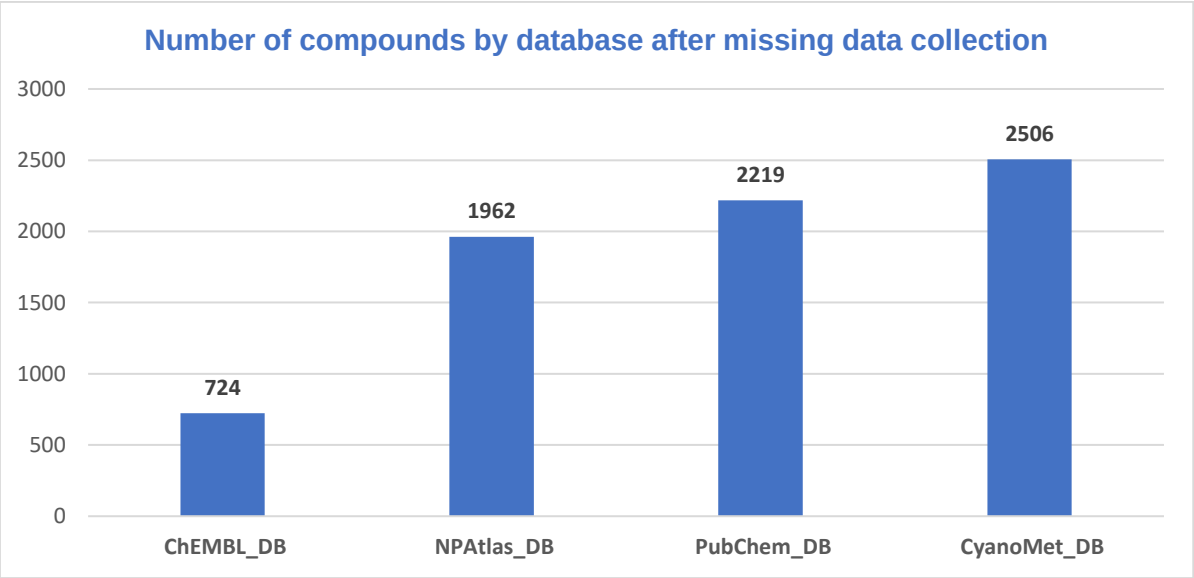


Chart 2-- Distribution of the number of cyanobacteria bioactive compounds after collecting missing information and bioassay experimental values. Total represents the sum of all occurrences of the compound in each database.

The phylogenetic distribution, considering the genus, of the number of compounds is shown in Chart 3. The existence of higher numbers of Microcystis compounds in the database can be explained considering that hepatotoxic microcystins are the most widespread class of cyanotoxins, and as such, a higher number of studies have been conducted in this genus. Cyanobacterial blooms can be accompanied by various cyanotoxins, one of which is microcystins, produced by species of *Microcystis* that can promote tumours and are hepatoxic, making them extremely dangerous(Pham & Utsumi, 2018; Svirčev et al., 2017).

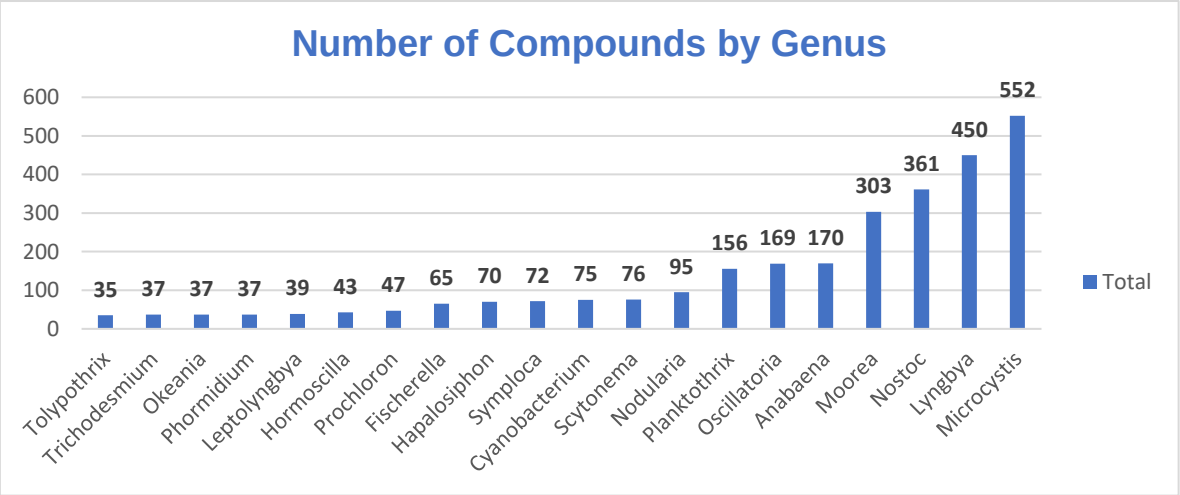


Chart 3- Distribution of the number of compounds by each genus across the cyanobacteria phylum considering 35 occurrences as the lower threshold.

Regarding the classes of compounds in Chart 4, we found a high number of nondefined cyclic and linear peptides in our database, followed by microcystins, which are the most widespread group of cyanotoxins capable of causing cyanobacterial poisoning. These toxins behave like hepatotoxins and are able to inhibit protein phosphatases, which in turn will create phosphorylated proteins and damage liver cells (Lopes, Silva, & Vasconcelos, 2022). Cyanopeptolins are nonribosomal peptides produced by different genera of cyanobacteria, such as *Microcystis* or *Anabaena*. These compounds have been found to be toxic to aquatic organisms and exhibit enzyme inhibiting capabilities, more specifically serine proteases. (Mazur-Marzec et al., 2018). Anabaenopetin inhibited carboxypeptidase-A, protein phosphatase 1, and elastase. Its primary application lies in its potential as an inhibitor of thrombin activatable fibrinolysis inhibitor, making it a promising candidate for antithrombotic mechanisms (Monteiro, do Amaral, Siqueira, Xavier, & Santos, 2021; Vercauteren, Gils, & Declerck, 2013). Aeruginosins are powerful inhibitors of serine proteases even at low concentrations. Its structural characteristics allow it to have affinity for binding with trypsin and other serine proteases. These proteases are related to the development of tumours and metastasis (Ahmed et al., 2021; Ersmark, Del Valle, & Hanessian, 2008; Martin & List, 2019; Tagirasa & Yoo, 2022).

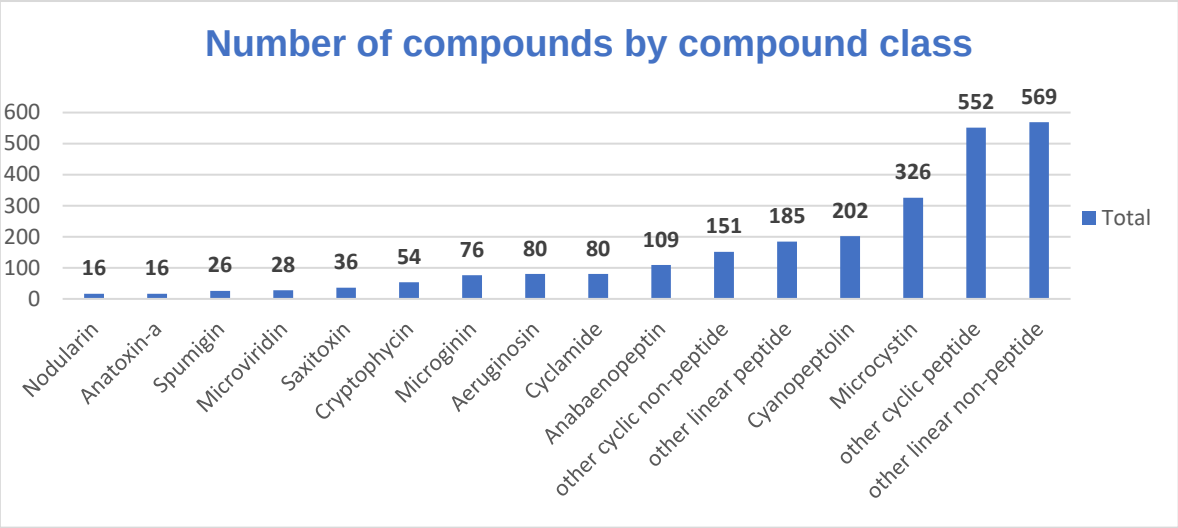


Chart 4- Distribution of the number of compounds by their chemical class in the cyanobacteria final database (CyanoBioactiveDB).

In Chart 5, we can verify which targets were most used in experimental assays regarding our compounds. The field containing the information for the experimentally validated bioactive compounds allowed the discrimination of the different targets, as shown in Chart 5, which displays the fifteen most tested targets on compounds in our website. The targets “CHEMBL261” (n=190), “CHEMBL205” (n=300) and “CHEMBL3594” (n=38) refer to carbonic anhydrases I, II and IX, respectively, and carbonic anhydrases II and IX have been linked to human cancer development (Haapasalo et al., 2007; Pastorekova & Gillies, 2019). “CHEMBL3932” also refers to carbonic anhydrase in *Methanosarcina thermophila*.

In addition to these targets, “CHEMBL382” (n=44), “CHEMBL387” (n=43), “CHEMBL384” (N=56), “CHEMBL394” (n=32), “CHEMBL396” (n=40), “CHEMBL399” (n=54) and “CHEMBL399” (n=44) also refer to the cell lines CCRF-CEM, MCF7, HT-29, HTC-116 NCI-H460, KB and HeLa, which are important for cancer studies. The first refers to a cell line used in oncology as well as human acute lymphoblastic leukemia T cells, the second is important in studying breast cancer, the third and fourth refer to human colorectal cancer, and the fifth refers to lung cancer. The sixth and seventh cell lines are both cervical carcinoma cell lines.(Elemam, Al-Jaderi, Hachim, & Maghazachi, 2019; Lee, Oesterreich, & Davidson, 2015; Martínez-Maqueda, Miralles, & Recio, 2015; Moore, Weise, Zawydiwski, & Thompson, 1985; Qamar et al., 2021; Townsend et al., 2017; Vaughan, Glänzel, Korch, & Capes-Davis, 2017). The “CHEMBL204” (n=35) target refers to thrombin, a serine protease that is important in platelet aggregation (A. R. Anas et al., 2012). “CHEMBL3419” (n=42) is a protease known as carboxypeptidase B2 isoform A related to fibrinolysis (Halland et al., 2015). The data collected can be used to help further improve human therapeutic approaches in cancer, which can be achieved

by performing virtual screening of the compounds stored in the database to search the compounds with higher binding affinity to these targets.

The remaining targets were not directly related to human molecules, and some did not contain any information readily available, such as “CHEMBL612545” (n=712) or “CHEMBL3879801” (n=85), which is described only as nonmolecular.

Other targets still refer to pathogens that affect humans. Some of the more important ones are “CHEMBL352” (n=66), which refers to the organism *Staphylococcus aureus,* which is a common human pathogen and causes a series of infectious diseases, including endocarditis osteomyelitis and even lethal pneumonia (Guo, Song, Sun, Wang, & Wang, 2020). The target “CHEMBL364” (n=60) is also an organism *Plasmodium falciparum* known for causing malaria in humans transmitted mostly by an *Anopheles* mosquito bite. (Joste et al., 2019). “CHEMBL354” (n=34) is the ChEMBL id for *Escherichia coli,* a well-known bacillus that can cause intestinal and other diseases in humans (Braz, Melchior, & Moreira, 2020). “CHEMBL368” (n=31) refers to *Trypanosoma cruzi,* a parasite that is known to cause Chagas disease in humans (A. R. J. Anas et al., 2012). “CHEMBL391” (n=32) refers to *Chlorocebus sabaeus* and was used for testing cytotoxicity in the Vero cell line (Carina, Elisabete, & Elsa, 2013).

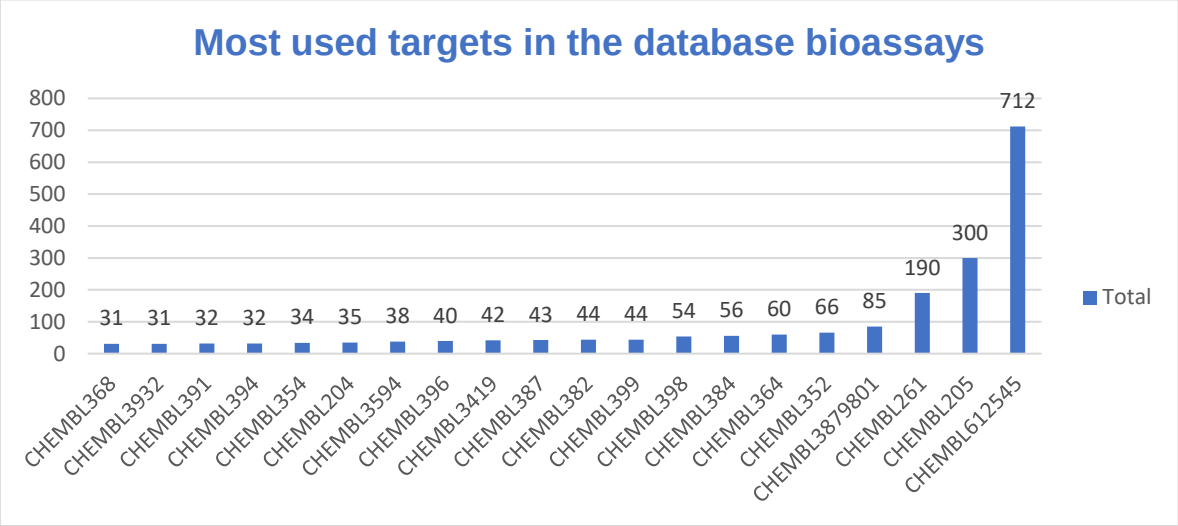


Chart 5- Distribution of the occurrences of each target in the retrieved bioassay values of our database (Top 20 targets by number of occurrences)

Regarding the obtained bioactivity information (Chart 6), we were able to obtain 6067 different bioassay results, of which 3538 were of the desired test type. Of these results, we verified that 1604 were from IC50 assays, 1165 from Ki, 397 from INH and 372 from MIC. As such, we verified that 45% of the bioassays with obtained values were from IC50 assays, 33% were from Ki assays, and 11% were from MIC and INH assays (Chart 7).

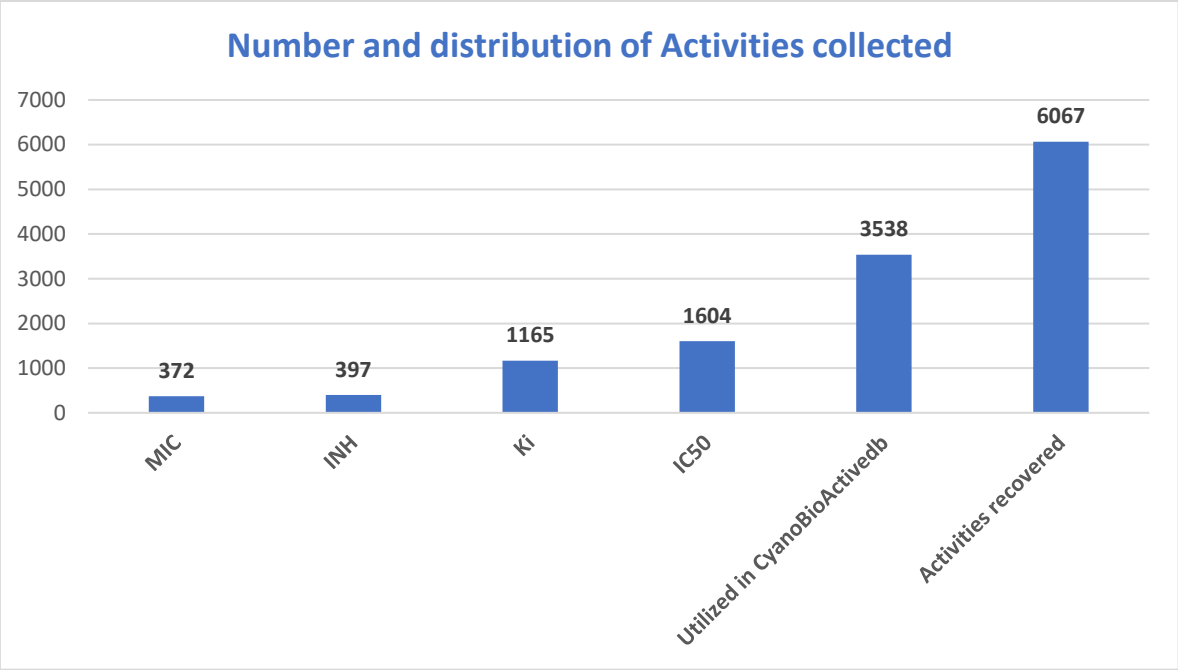


Chart 6- Distribution of the type of methodology used to ascertain the bioactive potential of the cyanobacterial bioactive compounds.

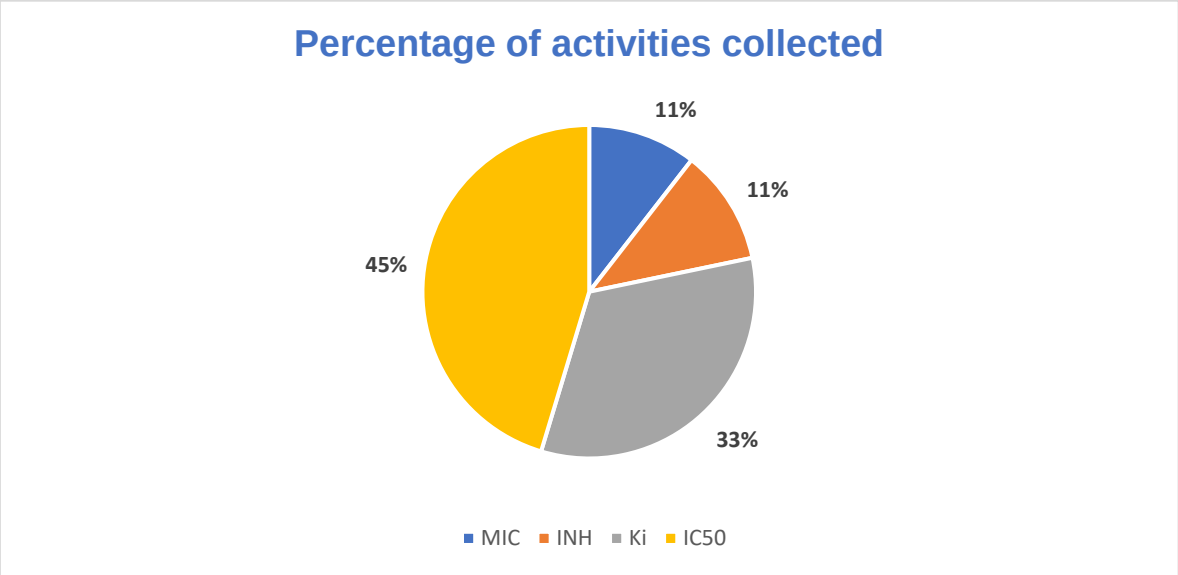


Chart 7- Percentage of distribution of the 4 types of methods to measure bioactive potential in CyanoBioactiveDB.

4.2. Machine learning model implementation and evaluation

Considering our second hypothesis concerning a machine learning model implementation to address the search for putative protein targets of cyanobacteria bioactive compounds, the developed workflow was executed. We employed the machine learning model workflow depicted in Figure 2 to determine the classification of our

compounds based on protein targets used in various bioassays. Initially, our prepared database was loaded, and we utilized the “Individual_target” column and assigned it the categorical label to enable accurate prediction of similar chemical compounds using distinct chemical descriptors from PaDEL-descriptor, Mordred, and Drugtax. The structure of the workflow remains the same, only differing which files are loaded.

To identify the most informative descriptors, we selected 25 descriptors with the highest “info.gain” value ranked through Orange3’s own rank function. Subsequently, the data were sampled and subjected to four different machine learning algorithms (gradient boosting, random forest, AdaBoost and kNN) in both the test and training sets (shown in the right part of Figure 2), along with a 20-fold cross-validation (shown in the left side of Figure 2). The outcomes of the training, testing, and cross-validation were analysed for the Mordred descriptors, PaDEL descriptors, and DrugTax, as depicted in Figure 2.

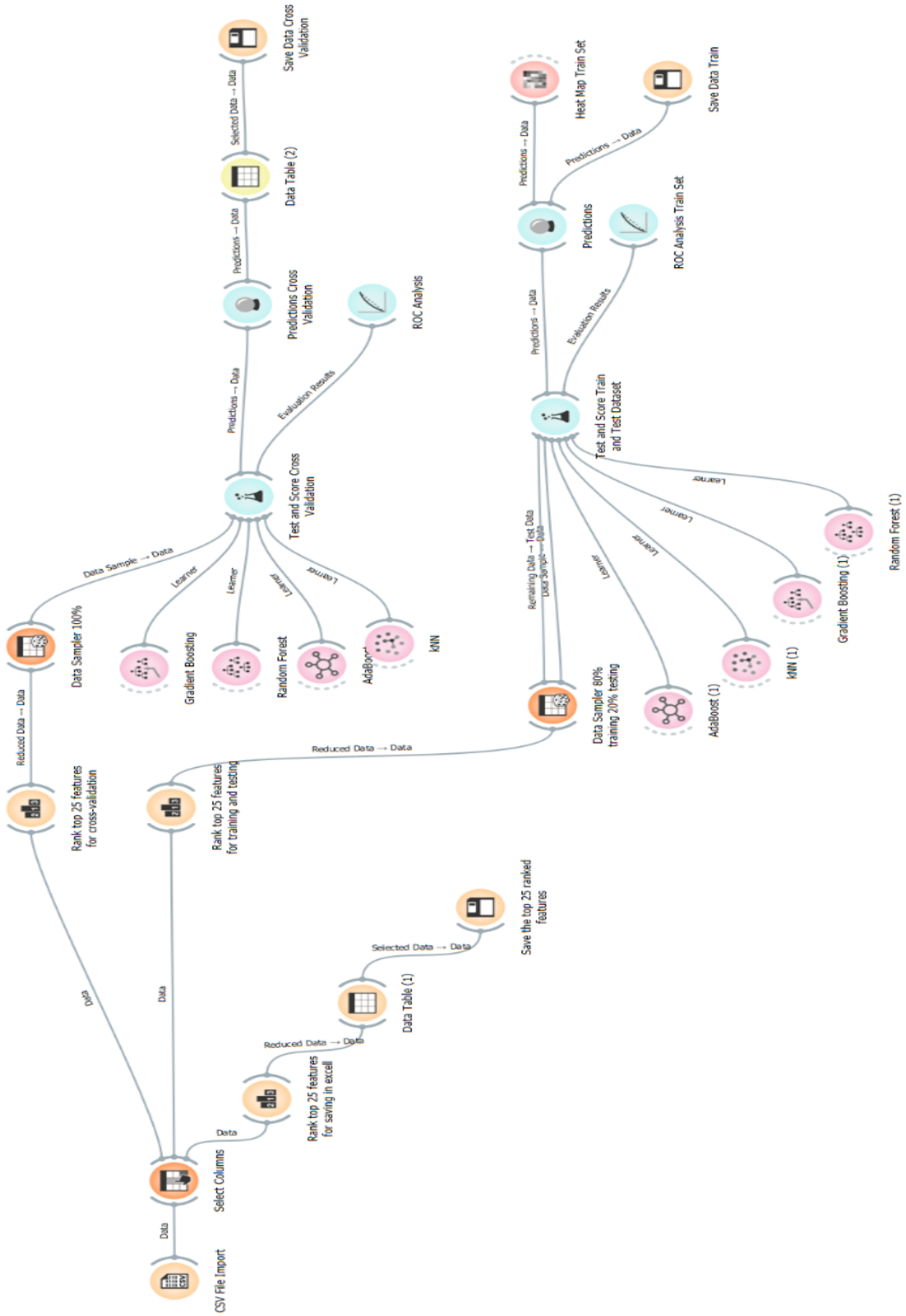


Figure 2-Orange software workflow used to implement the machine learning algorithms with the descriptors as features and the bioassay proteins as targets.

4.2.1. Feature selection

Regarding the features selected, the Oranges3 rank function allowed us to select the top 25 features by the “info. Gain” metric. This function selected the following features in Mordred: ATS7dv, SssO, PEOE_VSA3, NssO, ATS8DV, piPC6, VSA_Estate9, Xc-

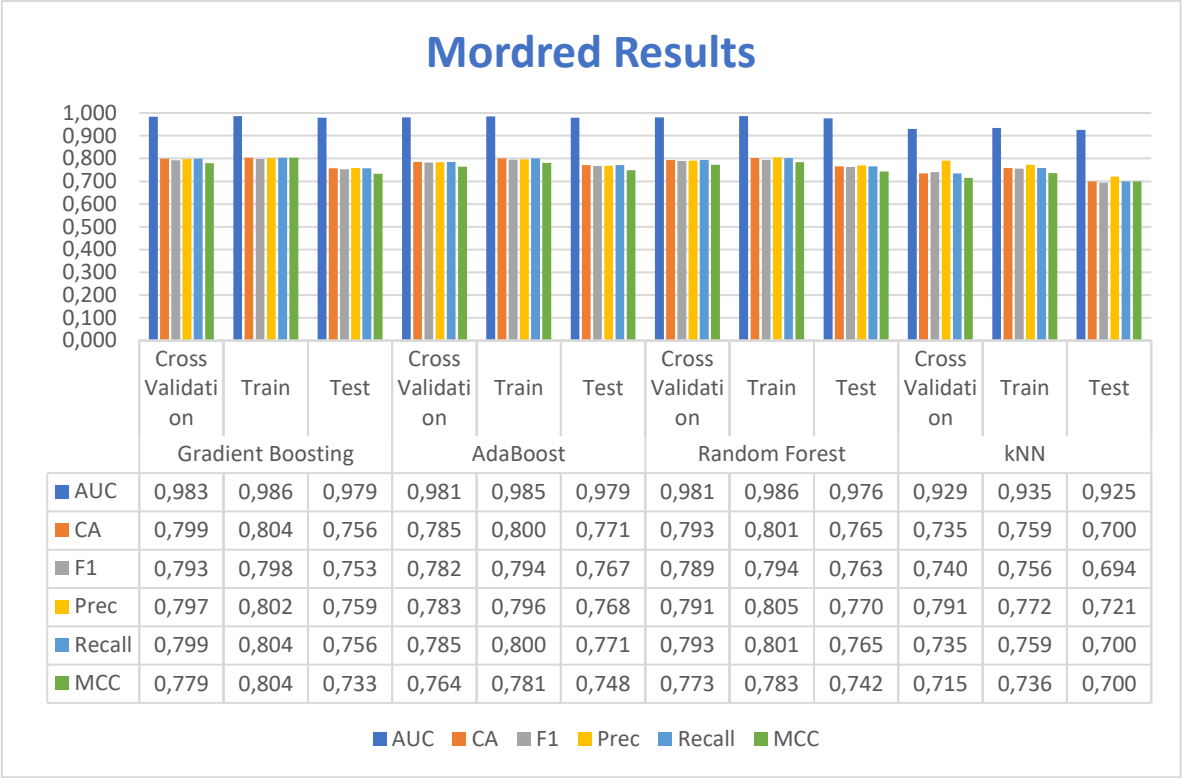
6dv, piPC5, nO, VSA_Estate5, piPC7, IC3, ATSC5se, NdssC, piPC8, NssNH, ATSC5pe, piPC10, SlogP_VSA1, nHetero and IC4.

For PaDEL, the features selected were MDEN-22, nTG12Ring, nAtomLAC, nHBAcc, C4SP3, nHBAcc2, MDEN-12, nHBAcc_Lipinski, nHBDon_Lipinski, nBondsD, nBondsD2, nBondsS3, nHBDon, nBondsM, C1SP3, nTRing, AATSC5c, nN, AATSC2c, MDEO-12, nBondsS2, nBondsS, piPC10, nAtomP and GATS2c.

DrugTax's top 25 features were char_C, char_N, char_Og, char_-, char_., char_[, char_#, char_B, char_\, organophosphorus, organosulfur, char_Si, carboxyl, phenylpropanoids_and_polyketides, char_=, negative, char_@, hydrocarbon, char_+, positive, char_P, aromatic_rings, benzenoid, organic_nitrogen and organic_salt.

4.2.2. Mordred results

Utilizing the Mordred calculated descriptors, we obtained the values in chart 8 and table 2 regarding the area under the ROC curve (AUC), classification accuracy, F1, precision, recall and Matthews correlation coefficient. These metrics are represented in values between 0 and 1. Chart 8 shows that the values of these evaluation metrics were generally higher in the training dataset, although this good performance does not necessarily be a good indication of the models' applicability with new data. The test dataset results are better to see the model's generalization when utilizing new data.. Despite this fact, the test dataset results are still promising, with AUCs between 0.93 and 0.983, classification accuracies between 0.7 and 0.771, F1 scores from 0.694 to 0.763, precision scores between 0.721 and 0.771, recall values between 0.7 and 0.771 and MCC values varying from 0.7 to 0.742. With the goal of obtaining a better estimate of the model's performance, a 20-fold cross-validation test was also performed. In general, the algorithms that performed better utilizing these descriptors were Gradient Boosting, which presented better overall values, followed by AdaBoost and Random Forest, which had similar results, while kNN underperformed in both datasets and in cross-validation and presented worse results across all evaluation metrics.



	Gradient Boosting			AdaBoost			Random Forest			kNN		
	Cross Validation	Train	Test	Cross Validation	Train	Test	Cross Validation	Train	Test	Cross Validation	Train	Test
AUC	0,983	0,986	0,979	0,981	0,985	0,979	0,981	0,986	0,976	0,929	0,935	0,925
CA	0,799	0,804	0,756	0,785	0,800	0,771	0,793	0,801	0,765	0,735	0,759	0,700
F1	0,793	0,798	0,753	0,782	0,794	0,767	0,789	0,794	0,763	0,740	0,756	0,694
Prec	0,797	0,802	0,759	0,783	0,796	0,768	0,791	0,805	0,770	0,791	0,772	0,721
Recall	0,799	0,804	0,756	0,785	0,800	0,771	0,793	0,801	0,765	0,735	0,759	0,700
MCC	0,779	0,804	0,733	0,764	0,781	0,748	0,773	0,783	0,742	0,715	0,736	0,700

Chart 8- Results of the implementation of the 4 different types of machine learning algorithms built using the top 25 Mordred descriptors as features on the training, testing and 20-fold cross-validation datasets.

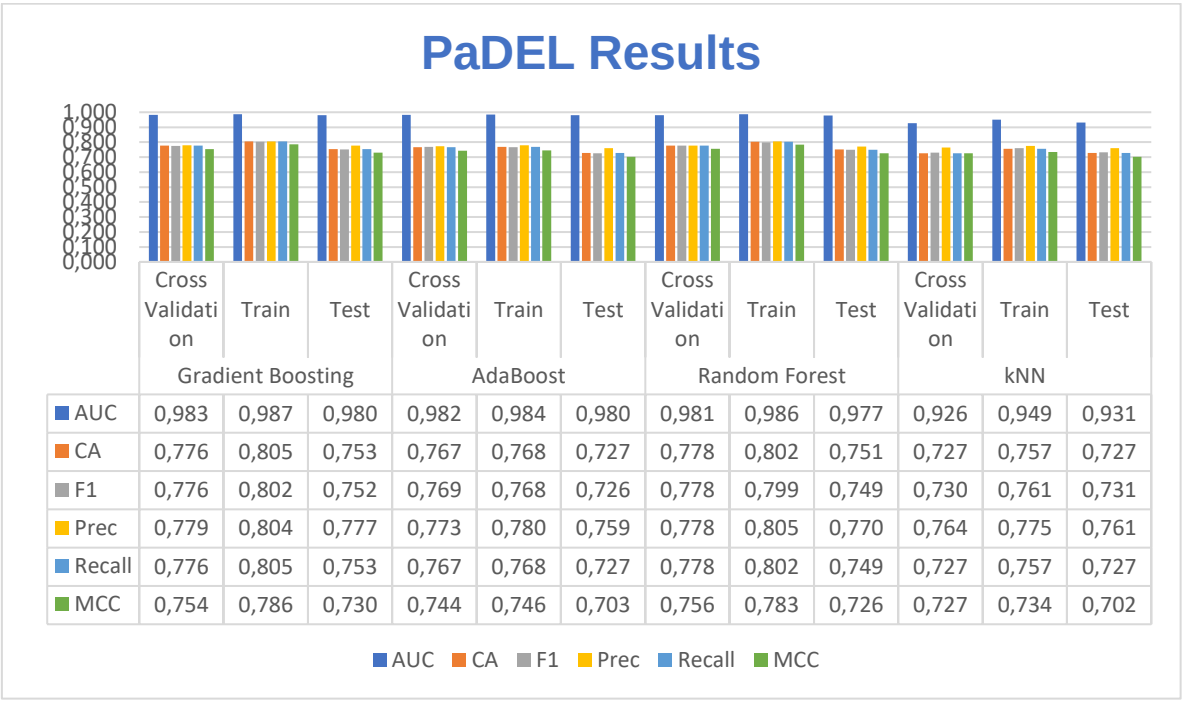
Model	Type of test	AUC	CA	F1	Prec	Recall	MCC
Gradient Boosting	**Average**	**0,983**	**0,786**	**0,781**	**0,786**	**0,786**	**0,772**
	Cross Validation	0,983	0,799	0,793	0,797	0,799	0,779
	Train	0,986	0,804	0,798	0,802	0,804	0,804
	Test	0,979	0,756	0,753	0,759	0,756	0,733
AdaBoost	**Average**	**0,982**	**0,785**	**0,781**	**0,783**	**0,785**	**0,764**
	Cross Validation	0,981	0,785	0,782	0,783	0,785	0,764
	Train	0,985	0,800	0,794	0,796	0,800	0,781
	Test	0,979	0,771	0,767	0,768	0,771	0,748
Random Forest	**Average**	**0,981**	**0,787**	**0,782**	**0,789**	**0,787**	**0,766**
	Cross Validation	0,981	0,793	0,789	0,791	0,793	0,773
	Train	0,986	0,801	0,794	0,805	0,801	0,783
	Test	0,976	0,765	0,763	0,770	0,765	0,742
kNN	**Average**	**0,930**	**0,731**	**0,730**	**0,761**	**0,731**	**0,717**
	Cross Validation	0,929	0,735	0,740	0,791	0,735	0,715

	Train	0,935	0,759	0,756	0,772	0,759	0,736
	Test	0,925	0,700	0,694	0,721	0,700	0,700

Table 2- Values of the test, training and 20-fold cross-validation databases utilizing the top 25 features in Mordred. The values in bold represent the average value of the models.

4.2.3. PaDEL descriptors

Utilizing the Padel calculated descriptors, we obtained the values in chart 9 and table 3 regarding the area under the ROC curve (AUC), classification accuracy, F1, precision, recall and Matthew's correlation coefficient. These metrics are represented in values between 0 and 1. As before, the values of these evaluation metrics were generally higher in the training dataset. The test dataset presented an AUC between 0,931 and 0,980, classification accuracy values from 0,727 and 0,753, F1 scores from 0.726 to 0.752, precision between 0.759 and 0.777, recall values between 0.727 and 0.753 and MCC values varying from 0.702 to 0.730. The algorithms that performed better utilizing these descriptors were Gradient Boosting and Random Forest, while AdaBoost underperformed in every metric except for AUC. kNN underperformed all metrics of evaluation in both datasets and in cross-validation. In all applications of the kNN algorithm, the utilization of the descriptors calculated through PaDEL provided the best results.



	Gradient Boosting			AdaBoost			Random Forest			kNN		
	Cross Validation	Train	Test	Cross Validation	Train	Test	Cross Validation	Train	Test	Cross Validation	Train	Test
AUC	0,983	0,987	0,980	0,982	0,984	0,980	0,981	0,986	0,977	0,926	0,949	0,931
CA	0,776	0,805	0,753	0,767	0,768	0,727	0,778	0,802	0,751	0,727	0,757	0,727
F1	0,776	0,802	0,752	0,769	0,768	0,726	0,778	0,799	0,749	0,730	0,761	0,731
Prec	0,779	0,804	0,777	0,773	0,780	0,759	0,778	0,805	0,770	0,764	0,775	0,761
Recall	0,776	0,805	0,753	0,767	0,768	0,727	0,778	0,802	0,749	0,727	0,757	0,727
MCC	0,754	0,786	0,730	0,744	0,746	0,703	0,756	0,783	0,726	0,727	0,734	0,702

Chart 9-Results of the implementation of the 4 different types of machine learning algorithms built using the top 25 PaDEL descriptors as features on the training, testing and 20-fold cross-validation datasets.

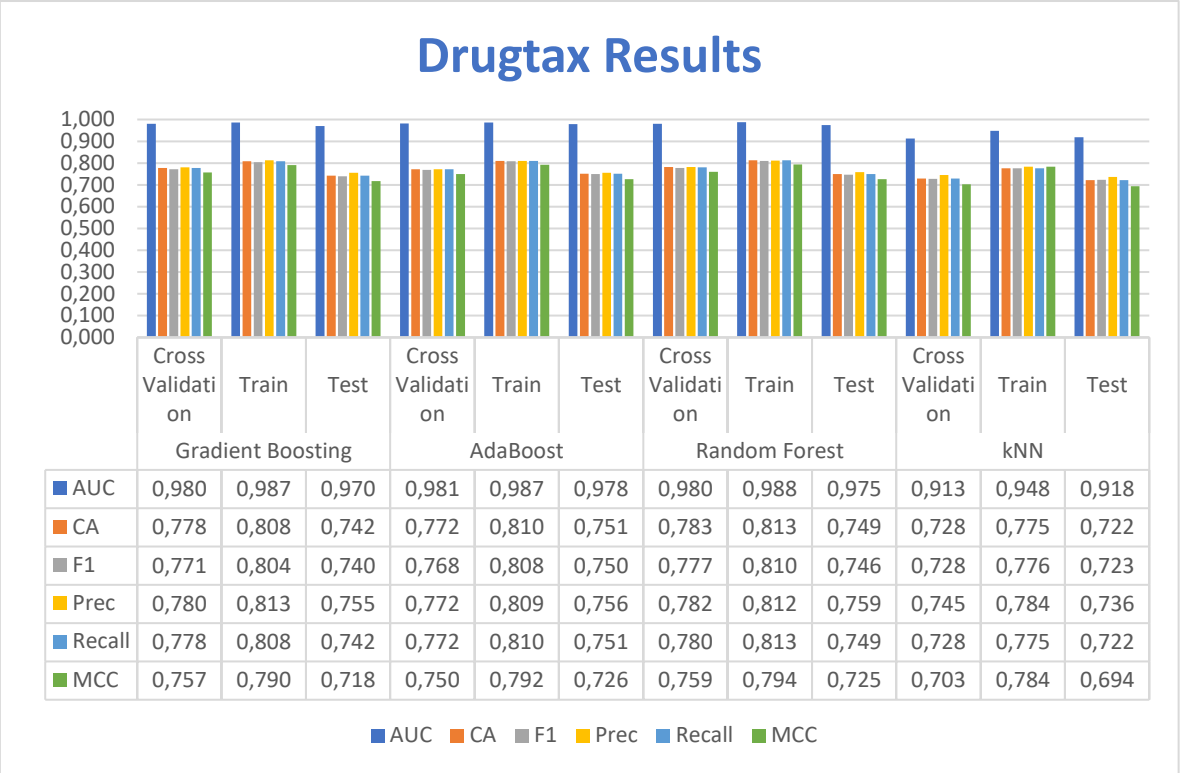
Model	Type of test	AUC	CA	F1	Prec	Recall	MCC
Gradient Boosting	**Average**	**0,983**	**0,778**	**0,777**	**0,787**	**0,778**	**0,757**

	Cross Validation	0,983	0,776	0,776	0,779	0,776	0,754
	Train	0,987	0,805	0,802	0,804	0,805	0,786
	Test	0,980	0,753	0,752	0,777	0,753	0,730
AdaBoost	**Average**	**0,982**	**0,754**	**0,754**	**0,771**	**0,754**	**0,731**
	Cross Validation	0,982	0,767	0,769	0,773	0,767	0,744
	Train	0,984	0,768	0,768	0,780	0,768	0,746
	Test	0,980	0,727	0,726	0,759	0,727	0,703
Random Forest	**Average**	**0,982**	**0,777**	**0,776**	**0,784**	**0,776**	**0,755**
	Cross Validation	0,981	0,778	0,778	0,778	0,778	0,756
	Train	0,986	0,802	0,799	0,805	0,802	0,783
	Test	0,977	0,751	0,749	0,770	0,749	0,726
kNN	**Average**	**0,935**	**0,737**	**0,741**	**0,767**	**0,737**	**0,721**
	Cross Validation	0,926	0,727	0,730	0,764	0,727	0,727
	Train	0,949	0,757	0,761	0,775	0,757	0,734
	Test	0,931	0,727	0,731	0,761	0,727	0,702

Table 3- Values of the test, training and 20-fold cross-validation databases utilizing the top 25 features in PaDEL. The values in bold represent the average value of the models.

4.2.4. DrugTax

Utilizing the Drugtax calculated descriptors, we obtained the values in chart 10 and table 4 regarding previously mentioned evaluation metrics. As before, the values of these evaluation metrics were generally higher in the training dataset. The test dataset presented an AUC between 0,918 and 0,978, classification accuracy with values from 0,722 and 0,751, F1 scores from 0.723 to 0.750, precision between 0.736 and 0.759, recall values between 0.722 and 0.751 and MCC values varying from 0.694 to 0.726. The algorithm that performed best utilizing these descriptors was random forest, which slightly outperformed Gradient Boosting, AdaBoost and kNN on both datasets and in cross-validation.



	Gradient Boosting			AdaBoost			Random Forest			kNN		
	Cross Validation	Train	Test	Cross Validation	Train	Test	Cross Validation	Train	Test	Cross Validation	Train	Test
AUC	0,980	0,987	0,970	0,981	0,987	0,978	0,980	0,988	0,975	0,913	0,948	0,918
CA	0,778	0,808	0,742	0,772	0,810	0,751	0,783	0,813	0,749	0,728	0,775	0,722
F1	0,771	0,804	0,740	0,768	0,808	0,750	0,777	0,810	0,746	0,728	0,776	0,723
Prec	0,780	0,813	0,755	0,772	0,809	0,756	0,782	0,812	0,759	0,745	0,784	0,736
Recall	0,778	0,808	0,742	0,772	0,810	0,751	0,780	0,813	0,749	0,728	0,775	0,722
MCC	0,757	0,790	0,718	0,750	0,792	0,726	0,759	0,794	0,725	0,703	0,784	0,694

Chart 10- Results of the implementation of the 4 different types of machine learning algorithms built using the top 25 Drugtax descriptors as features on the training, testing and 20-fold cross-validation datasets.

Model	Type of test	AUC	CA	F1	Prec	Recall	MCC
Gradient Boosting	**Average**	**0,979**	**0,776**	**0,772**	**0,783**	**0,776**	**0,755**
	Cross Validation	0,980	0,778	0,771	0,780	0,778	0,757
	Train	0,987	0,808	0,804	0,813	0,808	0,790
	Test	0,970	0,742	0,740	0,755	0,742	0,718
AdaBoost	**Average**	**0,982**	**0,778**	**0,776**	**0,779**	**0,778**	**0,756**
	Cross Validation	0,981	0,772	0,768	0,772	0,772	0,750
	Train	0,987	0,810	0,808	0,809	0,810	0,792
	Test	0,978	0,751	0,750	0,756	0,751	0,726
Random Forest	**Average**	**0,981**	**0,847**	**0,845**	**0,784**	**0,781**	**0,760**
	Cross Validation	0,980	0,777	0,771	0,782	0,780	0,759
	Train	0,988	0,813	0,810	0,812	0,813	0,794
	Test	0,975	0,749	0,746	0,759	0,749	0,725
kNN	**Average**	**0,927**	**0,742**	**0,742**	**0,755**	**0,742**	**0,727**
	Cross Validation	0,913	0,728	0,728	0,745	0,728	0,703
	Train	0,948	0,775	0,776	0,784	0,775	0,784
	Test	0,918	0,722	0,723	0,736	0,722	0,694

Table 4- Values of the test, training and 20-fold cross-validation databases utilizing the top 25 features in Drugtax. The values in bold represent the average value of the models.

During our work, we discovered that Drugtax might not properly calculate compounds containing hydrogen since the SMILES it outputs after the initial input does

not contain their corresponding hydrogens. As such, the information and model obtained from this descriptor must be taken with this information into account and be handled with caution and displaying this overarching issue.

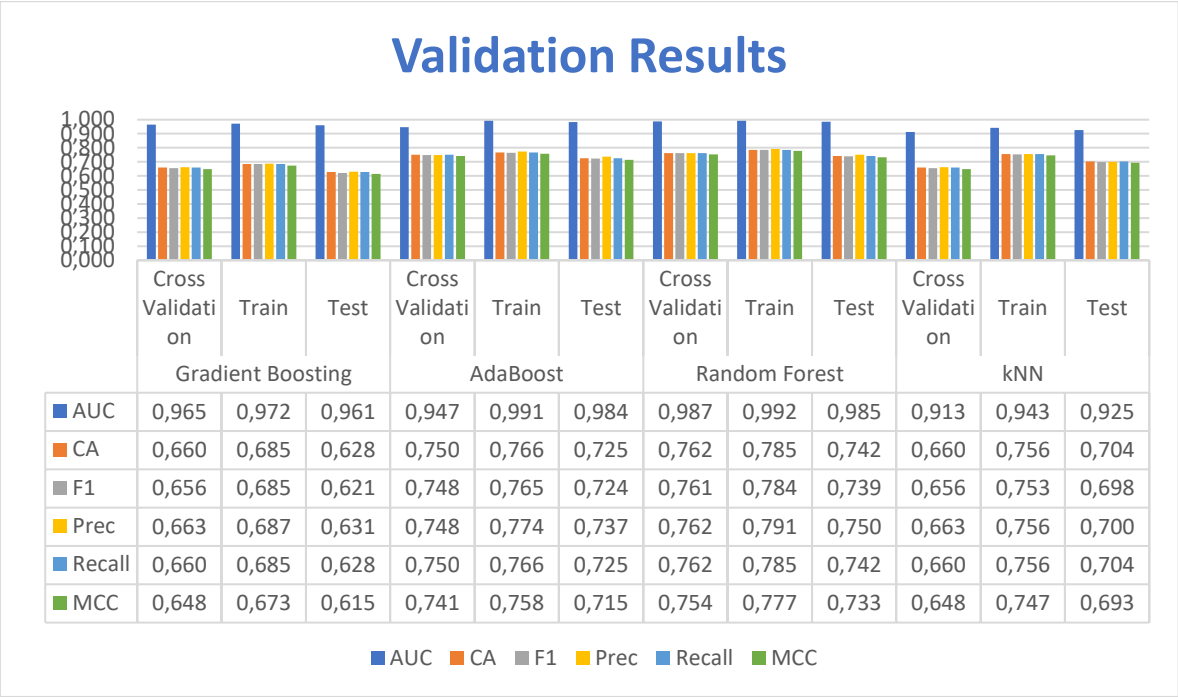
In summary, utilizing the Mordred descriptors Gradient Boosting, AdaBoost and Random Forrest presented their best overall results when considering all models and tests, while kNN presented its worst result of all implementations. When using the PaDEL-calculated descriptors, its best models were Gradient Boosting and kNN, which had results similar to the Drugtax descriptors model. Finally, the Drugtax model presented the best results for the kNN algorithm.

4.3. Validation test

To obtain a better assessment of the performance of our Mordred machine learning workflow, we utilized a new database obtained from BindingDB, and the data we used were drawn from PubChem and its 2D structures. This is done since this new acquired data only has compounds with known binding capability to its targets and has such allows for a proper analysis of the model's predictive capabilities with known compound-target connections. For this database, the same methodology was applied to limit the number of prediction classes and descriptors. Utilizing the chemical structures in this file and Datawarrior, we are able to determine their SMILES, which are then run through Mordred to obtain their respective descriptors. This database also had a high number of individual entries totalling 41148 different interactions, with *Homo sapiens* as its target species, from which we utilized Datawarrior to calculate the smiles from the compound structures. This process allows us to calculate the Mordred descriptors that can then be applied in our models as features for predicting possible targets for compound-target interaction.

In this database the targets are identified by their Unitprot primary ID and were as follows: O75116, P00519, P00533, P03951, P07858, P07900, P07949, P10696, P17612, P21453, P21462, P25090, P28562, P28566, P30305, P34949, P35398, P36888, P42858, P61981, Q01196, Q07817, Q13285, Q15139 Q16548, Q99500, V9GZ37. Since our limit is 25 occurences and this database contained more data the number of targets that were present increased from 11 to 27. From there, we applied same oversampling technique to the database and then used it in our Mordred model since it provided the best results in the previous test, and we obtained the results in Chart 11 and Table 5. From the test data we can verify that there is big decrease in the evaluation metrics for gradiant boosting all the metrics besides AUC had their values between 0.615 and 0.628 and where comparatively lower than with our CyanoBioactiveDB. The Random Forrest algorithm presented the best results overall in

this database followed by Adaboost and then kNN. These algorithms had generally lower results also when compared to their implementation on the CyanoBioactiveDB but with a lesser difference of values than what is seen in gradiant boosting.



	Gradient Boosting			AdaBoost			Random Forest			kNN		
	Cross Validation	Train	Test	Cross Validation	Train	Test	Cross Validation	Train	Test	Cross Validation	Train	Test
AUC	0,965	0,972	0,961	0,947	0,991	0,984	0,987	0,992	0,985	0,913	0,943	0,925
CA	0,660	0,685	0,628	0,750	0,766	0,725	0,762	0,785	0,742	0,660	0,756	0,704
F1	0,656	0,685	0,621	0,748	0,765	0,724	0,761	0,784	0,739	0,656	0,753	0,698
Prec	0,663	0,687	0,631	0,748	0,774	0,737	0,762	0,791	0,750	0,663	0,756	0,700
Recall	0,660	0,685	0,628	0,750	0,766	0,725	0,762	0,785	0,742	0,660	0,756	0,704
MCC	0,648	0,673	0,615	0,741	0,758	0,715	0,754	0,777	0,733	0,648	0,747	0,693

Chart 11- Results of the implementation of the 4 different types of machine learning algorithms built using the top 25 Mordred descriptors as features on the training, testing and 20-fold cross-validation datasets from the validation database.

Model	Type of test	AUC	CA	F1	Prec	Recall	MCC
Gradient Boosting	**Average**	**0,966**	**0,658**	**0,654**	**0,660**	**0,658**	**0,645**
	Cross Validation	0,965	0,660	0,656	0,663	0,660	0,648
	Train	0,972	0,685	0,685	0,687	0,685	0,673
	Test	0,961	0,628	0,621	0,631	0,628	0,615
AdaBoost	**Average**	**0,974**	**0,747**	**0,746**	**0,753**	**0,747**	**0,738**
	Cross Validation	0,947	0,750	0,748	0,748	0,750	0,741
	Train	0,991	0,766	0,765	0,774	0,766	0,758
	Test	0,984	0,725	0,724	0,737	0,725	0,715
Random Forest	**Average**	**0,988**	**0,763**	**0,761**	**0,768**	**0,763**	**0,755**
	Cross Validation	0,987	0,762	0,761	0,762	0,762	0,754
	Train	0,992	0,785	0,784	0,791	0,785	0,777
	Test	0,985	0,742	0,739	0,750	0,742	0,733
kNN	**Average**	**0,927**	**0,707**	**0,702**	**0,706**	**0,707**	**0,696**
	Cross Validation	0,913	0,660	0,656	0,663	0,660	0,648
	Train	0,943	0,756	0,753	0,756	0,756	0,747
	Test	0,925	0,704	0,698	0,700	0,704	0,693

Table 5- Values of the test, training and 20-fold cross-validation databases utilizing the top 25 features calculated by Mordred in the validation dataset. The values in bold represent the average value of the models.

4.4. Free online database

The main hypothesis was validated by creating the CyanoBioactiveDB database and making it accessible through a free online database website. To ensure easy access for academic and corporate research companies, scientists and the general public, a WordPress website was developed. The homepage of the website (Figure 3) provides an overview of the database contents, including statistical analysis. Users can explore detailed information about the compounds and their molecular and chemical descriptors (Figures 4 and 8). Finally, we allow any visitor to search for any type of information in the database and allow the download of that information (Figure 9). The data stored on the website follow the FAIR principles, which allows a fast search of the cyanobacterial compounds of interest and export the data in several formats. These formats allow the reuse of the data and interoperability with other computational tools.

On or main page we highlight the main characteristics of our database and the information you can find on our website from the number of compounds, classes, and unique targets. Additionally, we also highlight the descriptors calculated from our database and the ML algorithms we utilized these descriptors for their calculations.

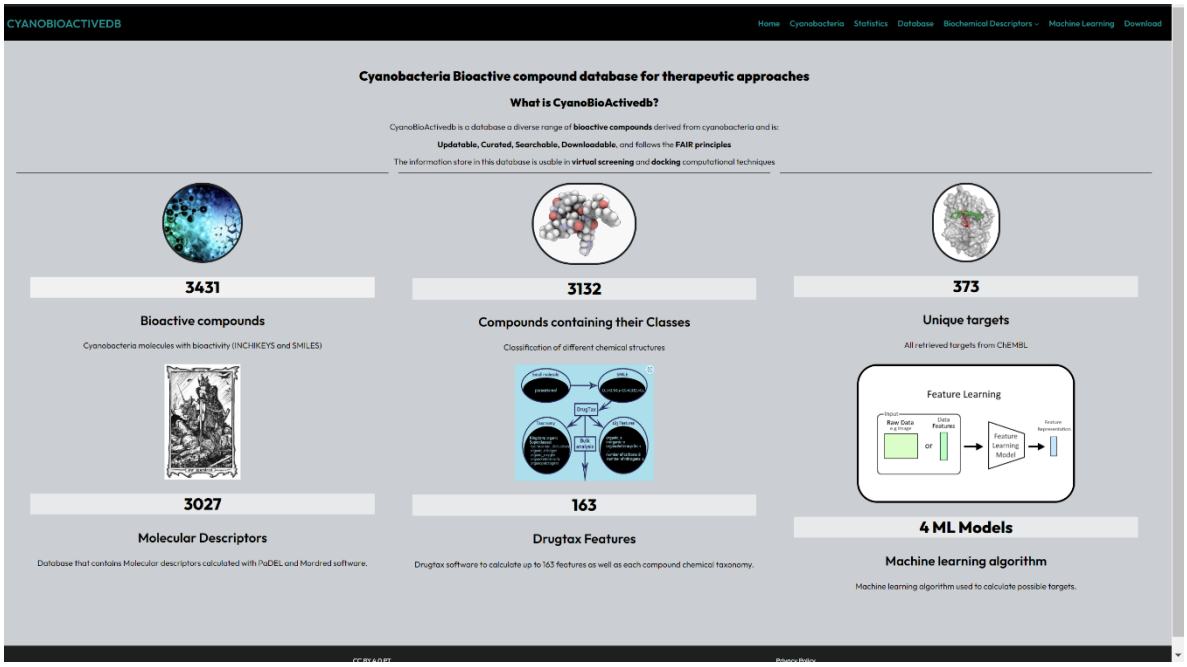


Figure 3-Front page of the CyanobioactiveDB website, which contains a brief description of this database and its purpose as well as its main characteristics.

The webpage “Cyanobacteria” (Figure 4) of the website contains a series of submenus with a summary of what cyanobacteria are, their applications, a description of their compounds and links to other databases utilized to create this database.

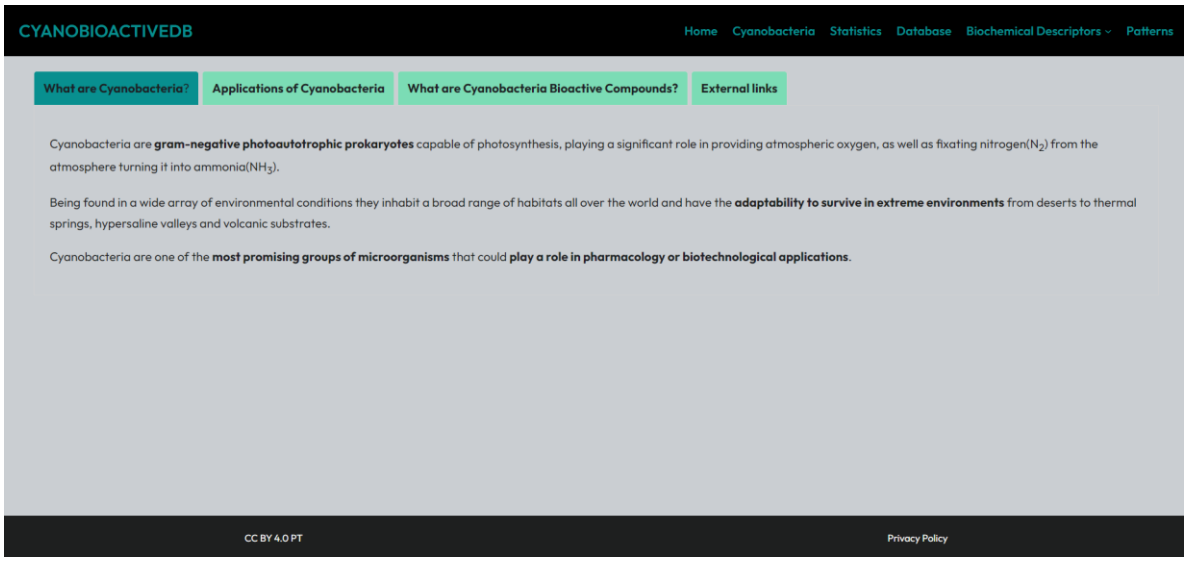


Figure 4-Cyanobacteria page of the CyanobioactiveDB website.

On the Statistics page (Figure 5), we created several submenus containing different distributions of key elements in our database. The charts and analysis focused on highlighting the sources of the compounds in our database, as well as the distribution of the top 20 most occurring genera and targets.

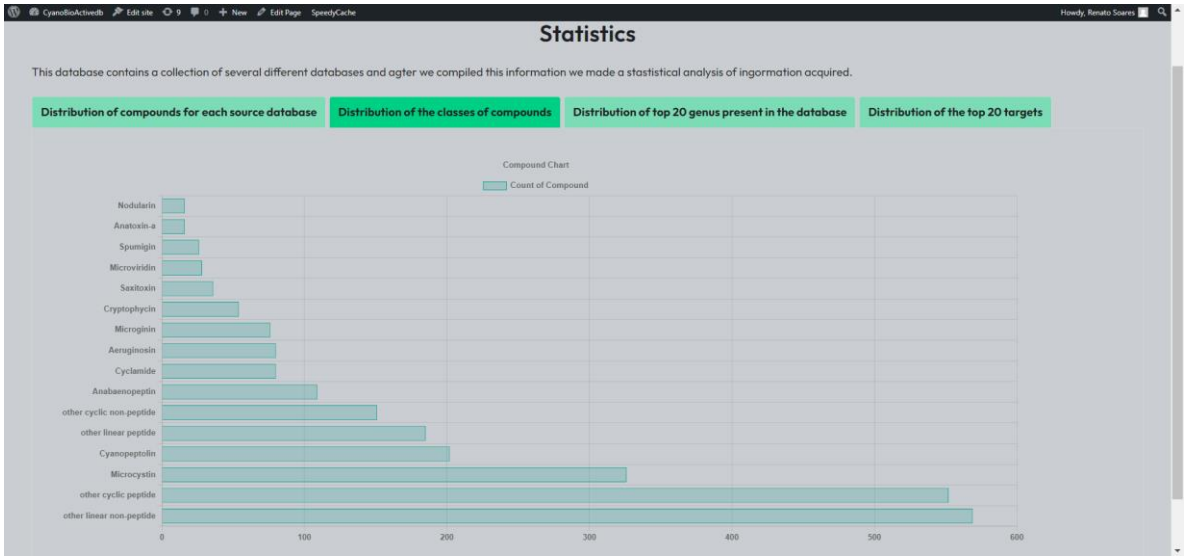
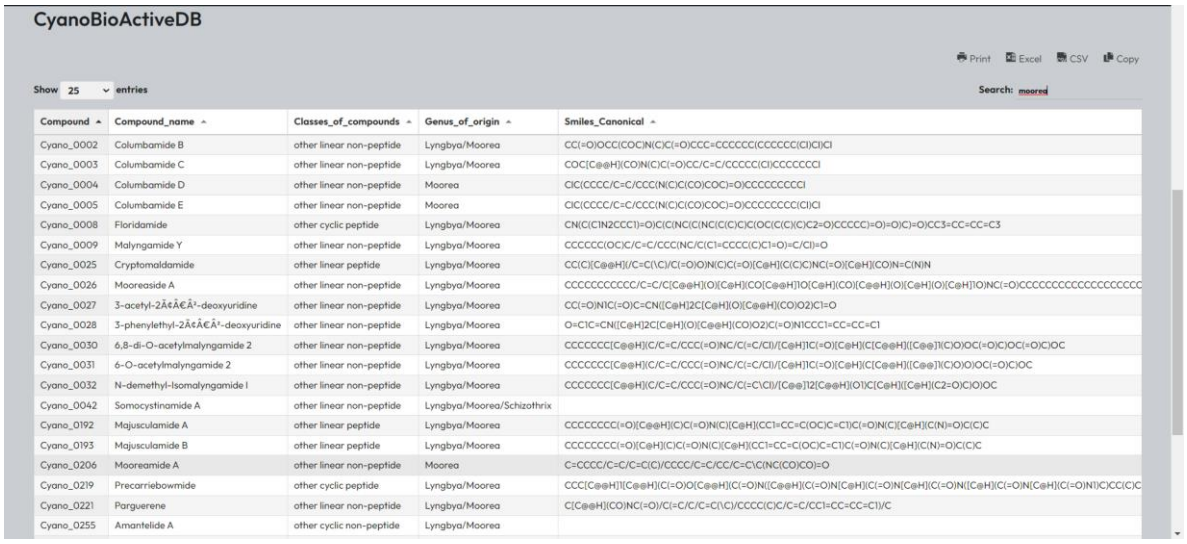


Figure 5-Statistics page of the CyanobioactiveDB website.

In Figure 6, the website permits a search for a variety of terms, including SMILES, InchIKeys and other information from the database columns, and then the download will be only of the selected rows from the search parameter. These csv files can be reused to calculate new descriptors or utilize the database already containing descriptors for use in other ML models.



Compound	Compound_name	Classes_of_compounds	Genus_of_origin	Smiles_Canonical
Cyano_0002	Columbamide B	other linear non-peptide	Lyngbya/Moorea	CC(=O)OCC(COC)N(C)C(=O)CCC=CCCCCC(CCCCCC(Cl)Cl)Cl
Cyano_0003	Columbamide C	other linear non-peptide	Lyngbya/Moorea	COC[C@@H](CO)N(C)C(=O)CC/C=C/CCCCC(Cl)CCCCCCCl
Cyano_0004	Columbamide D	other linear non-peptide	Moorea	ClC(CCCC/C=C/CCC(N(C)C(CO)COC)=O)CCCCCCCCCl
Cyano_0005	Columbamide E	other linear non-peptide	Moorea	ClC(CCCC/C=C/CCC(N(C)C(CO)COC)=O)CCCCCCCCC(Cl)Cl
Cyano_0008	Floridamide	other cyclic peptide	Lyngbya/Moorea	CN(C(C1N2CCC1)=O)C(C(NC(C(NC(C(C)C)C)OC(C(C)C)C2=O)CCCCC)=O)=O)C)=O)CC3=CC=CC=C3
Cyano_0009	Malyngamide Y	other linear non-peptide	Lyngbya/Moorea	CCCCCC(OC)C/C=C/CCC(NC/C(C1=CCCC(C)C1=O)=C/Cl)=O
Cyano_0025	Cryptomaldamide	other linear peptide	Lyngbya/Moorea	CC(C)[C@@H](/C=C(\C)/C(=O)O)N(C)C(=O)[C@H](C(C)C)NC(=O)[C@H](CO)N=C(N)N
Cyano_0026	Mooreaside A	other linear non-peptide	Lyngbya/Moorea	CCCCCCCCCCC/C=C/C[C@@H](O)[C@H](CO[C@@H]1O[C@H](CO)[C@@H](O)[C@H](O)[C@H]1O)NC(=O)CCCCCCCCCCCCCCCCCCC
Cyano_0027	3-acetyl-2Â€Â€Â²-deoxyuridine	other linear non-peptide	Lyngbya/Moorea	CC(=O)N1C(=O)C=CN([C@H]2C[C@H](O)[C@@H](CO)O2)C1=O
Cyano_0028	3-phenylethyl-2Â€Â€Â²-deoxyuridine	other linear non-peptide	Lyngbya/Moorea	O=C1C=CN([C@H]2C[C@H](O)[C@@H](CO)O2)C(=O)N1CCC1=CC=CC=C1
Cyano_0030	6,8-di-O-acetylmalyngamide 2	other linear non-peptide	Lyngbya/Moorea	CCCCCCC[C@@H](C/C=C/CCC(=O)NC/C(=C/Cl)/[C@H]1C(=O)[C@@H](C[C@@H]([C@@]1(C)O)OC(=O)C)OC(=O)C)OC
Cyano_0031	6-O-acetylmalyngamide 2	other linear non-peptide	Lyngbya/Moorea	CCCCCCC[C@@H](C/C=C/CCC(=O)NC/C(=C/Cl)/[C@H]1C(=O)[C@@H](C[C@@H]([C@@]1(C)O)O)OC(=O)C)OC
Cyano_0032	N-demethyl-Isomalyngamide I	other linear non-peptide	Lyngbya/Moorea	CCCCCCC[C@@H](C/C=C/CCC(=O)NC/C(=C\Cl)/[C@@]12[C@@H](O1)C[C@H]([C@H](C2=O)C)O)OC
Cyano_0042	Somocystinamide A	other linear non-peptide	Lyngbya/Moorea/Schizothrix	
Cyano_0192	Majusculamide A	other linear peptide	Lyngbya/Moorea	CCCCCCCC(=O)[C@@H](C)C(=O)N(C)[C@H](CC1=CC=C(OC)C=C1)C(=O)N(C)[C@H](C(N)=O)C(C)C
Cyano_0193	Majusculamide B	other linear peptide	Lyngbya/Moorea	CCCCCCCC(=O)[C@H](C)C(=O)N(C)[C@H](CC1=CC=C(OC)C=C1)C(=O)N(C)[C@H](C(N)=O)C(C)C
Cyano_0206	Mooreamide A	other linear non-peptide	Moorea	C=CCCC/C=C/C=C(C)/CCCC/C=C/CC/C=C\C(NC(CO)CO)=O
Cyano_0219	Precarriebowmide	other cyclic peptide	Lyngbya/Moorea	CCC[C@@H]1[C@@H](C(=O)O[C@@H](C(=O)N[C@@H](C(=O)N[C@H](C(=O)N[C@H](C(=O)N[C@@H](C(=O)N1)C)CC(C)C
Cyano_0221	Parguerene	other linear non-peptide	Lyngbya/Moorea	C[C@@H](CO)NC(=O)/C(=C/C/C=C(\C)/CCCC(C)C/C=C/CC1=CC=CC=C1)/C
Cyano_0255	Amantelide A	other cyclic non-peptide	Lyngbya/Moorea	

Figure 6-Database page of the CyanobioactiveDB website. In this example, we searched for any row that originated from the "moorea" genus, and the database returned the following entries that allowed for their download.

In Figure 7, when selecting "Biochemical Descriptors", a submenu is presented, listing the names of the different descriptor calculators used. Each name serves as a link to an individual dataset containing the descriptors.

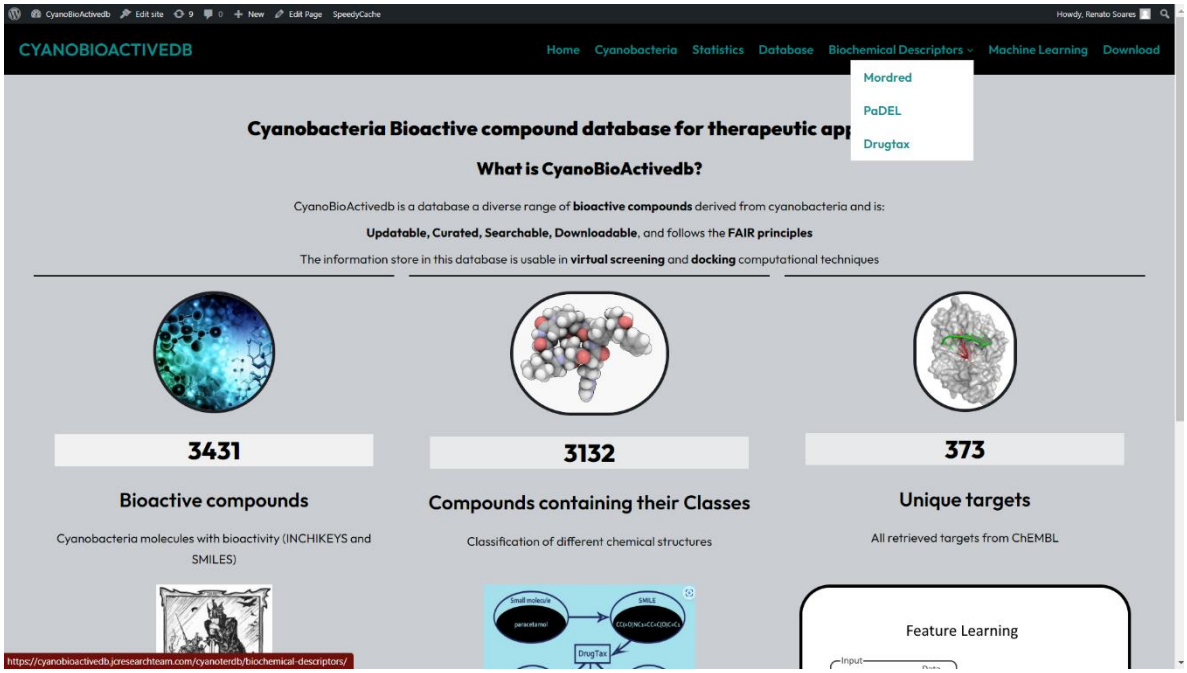


Figure 7- Main page of CyanoBioactivedb selecting the submenu of the Biochemical Descriptors tab.

These datasets can be downloaded and accessed, allowing users to view the main molecular and chemical descriptors when applied to an ML algorithm. Alternatively, users can click on the "Biochemical Descriptors" option, which takes them to a page providing a description of what a descriptor is (Figure 8).

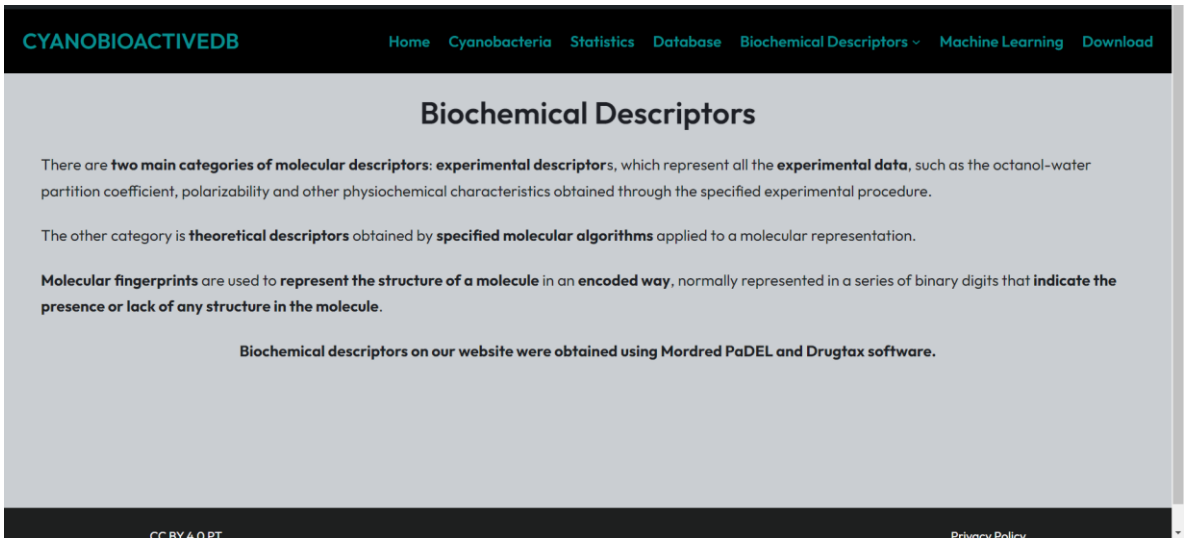


Figure 8- Biochemical descriptors page of CyanoBioactivedb presenting a short description of what biochemical descriptors are.

This page likely explains the concept of descriptors and their significance in the context of the database and ML algorithms. By providing both the option to access individual descriptor datasets and offering an explanation of descriptors themselves, the website ensures that users have the resources and information they need to understand and utilize the molecular and chemical descriptors effectively (Figures 9, 10 and 11).

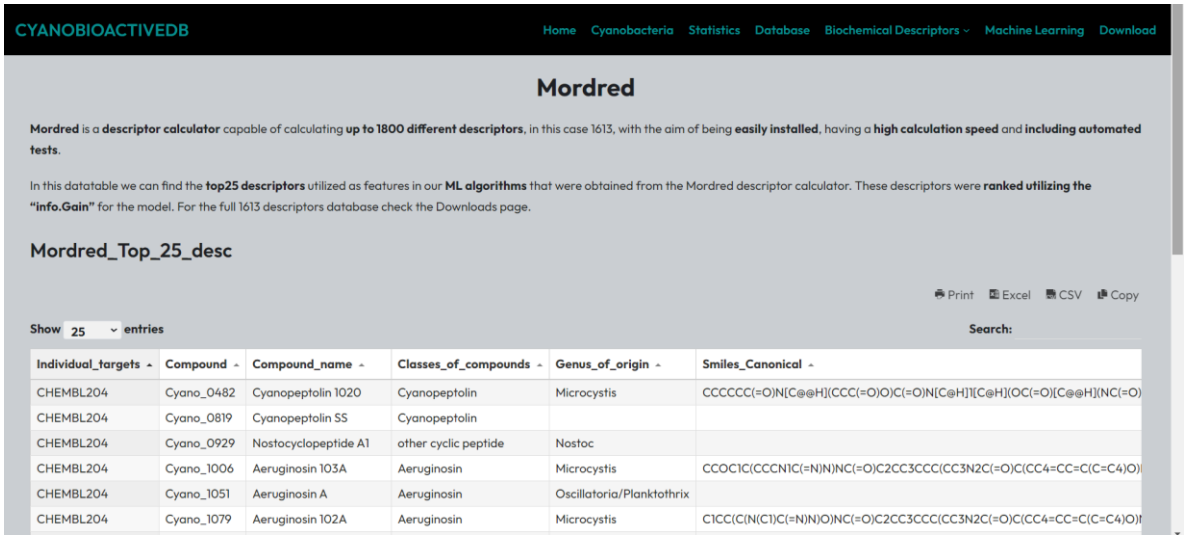


Figure 9-Mordred page of CyanoBioativedb website containing a database of the top 25 descriptors from the Mordred descriptor calculator when ranking descriptors using info. Gain, Gain ratio, Gini and X^2.

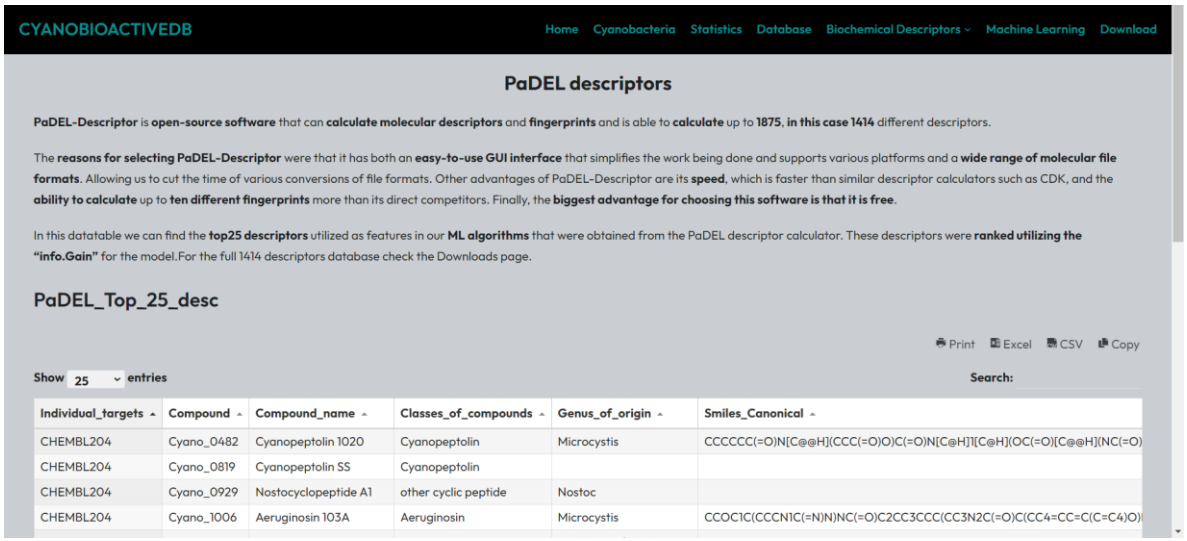


Figure 10-PaDEL page of CyanoBioativedb website containing a database of the top 25 descriptors from Padel when ranking descriptors using info. Gain, Gain ratio, Gini and X^2.

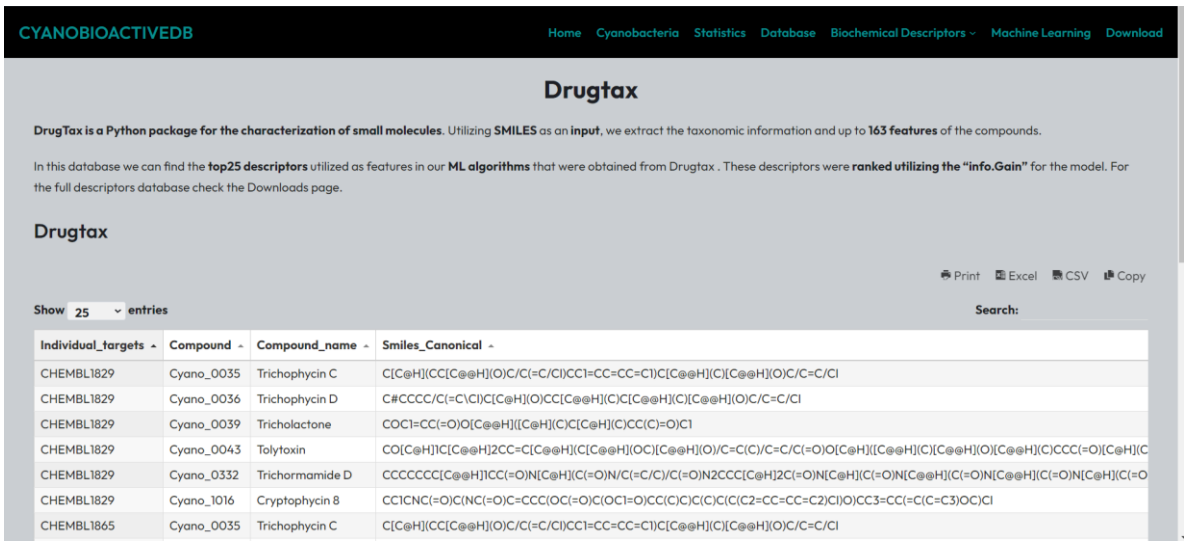


Figure 11- DrugTax page of the CyanoBioactive website containing a database of the top 25 descriptors from the DrugTax program when ranking descriptors using info. Gain, Gain ratio, Gini and X^2.

For users who wish to check our ML workflow and wish to apply it to their own databases or use ours in their own workflows or with other algorithms besides the ones we have chosen, all the needed files are contained in the "Machine Learning" page (Figure 12).

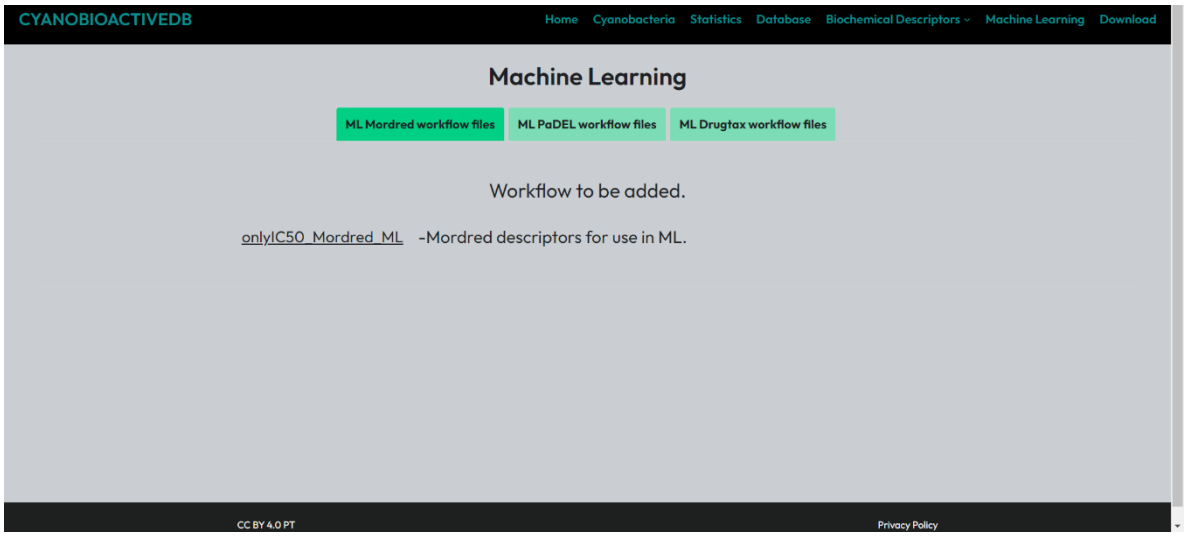


Figure 12-Machine Learning page of CyanoBioativedb website that contains the files regarding our applied machine learning algorithm including the descriptors used and the workflow itself.

Finally, for all the available downloadable content in our database and the process utilized in creating it, the download page contains a series of different databases, the ML orange workflows and the Python files utilized in forming it (Figure 13).

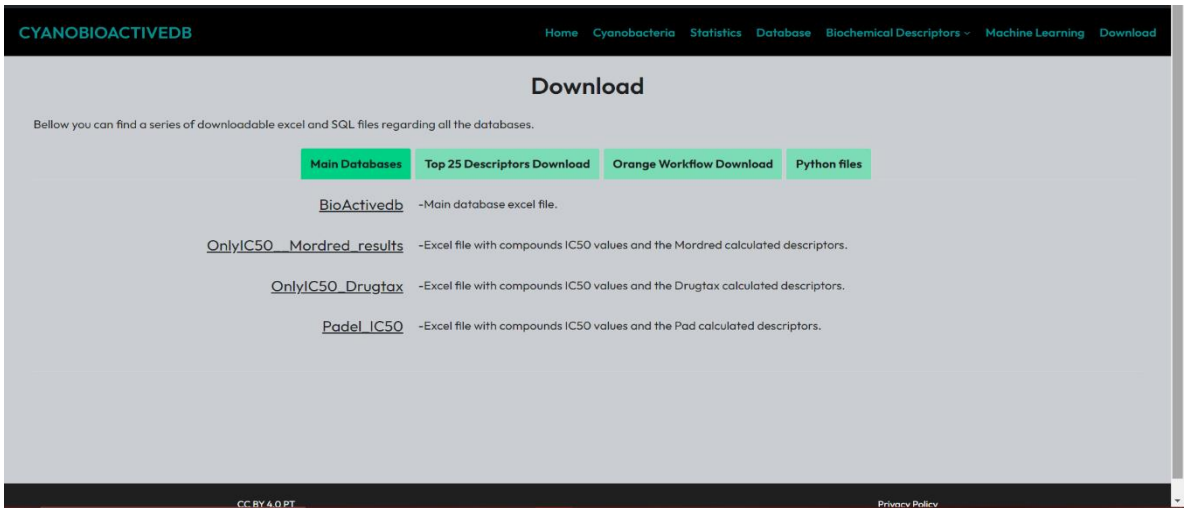


Figure 13- Download page of CyanoBioativedb website containing all the major data tables regarding the website.

Taking into account the online availability of the CyanoBioactiveDB database, our initial hypothesis has been completely addressed. We successfully created a comprehensive database of bioactive compounds derived from cyanobacteria, encompassing molecular and chemical information gathered from various online sources. Additionally, we fulfilled our second hypothesis by utilizing machine learning models that leverage the final database to predict protein targets with potential applications in human therapeutics.

5. Conclusion

In conclusion, we were able to fully address the specific objectives derived from our two main hypotheses. The specific objectives fully achieved were as follows: 1) We established a curated free online database dedicated to cyanobacteria bioactive compounds and their molecular and chemical descriptors. 2) We implemented a semiautomated workflow utilizing data mining techniques to retrieve cyanobacteria-derived compounds from various sources, including articles and databases. 3) Utilizing the most up-to-date software tools, we were able to calculate molecular descriptors for compounds containing isomeric SMILES with the possibility of including calculations from canonical SMILES in the future. 4) We created an online database that adheres to FAIR principles by incorporating the collected information. 5) We designed and implemented a machine learning (ML) algorithm capable of predicting potential targets for compounds present in the database, utilizing the calculated molecular descriptors. This ML algorithm has been minimally validated through the several metrics we utilized to measure in evaluating its performance. Our models showed that regarding the algorithms when utilizing Mordred descriptors, Gradient Boosting, AdaBoost and random forest obtain better results, while the kNN algorithm favors the Drugtax descriptors. The use of our validation test also proves the possible applications of this model in large datasets.

In the future, this work includes the possibility of collecting the compounds of this database for utilization in docking and virtual screening approaches, from which this database can provide some information regarding other fields, such as drug discovery. The ongoing collection of bioassay information will also help in the future to improve the ML algorithm by providing more data entries that can help improve results. In addition, the implementation of other ML algorithms in addition to those tested might lead to better results.

6. Output of this work

Directly from this work, the main scientific contributions were as follows:

i. An online database available at our website called CyanoBioActiveDB (https://cyanobioactivedb.jcresearchteam.com/) contains cyanobacteria compounds with their bioassay information and their chemical and molecular descriptors. The website also allows for a series of different data tables to be downloaded containing the full database, the molecular descriptors of the compounds utilizing three different descriptor calculators: PaDEL, Mordred and Drugtax and a data table containing the "top 25 "descriptors from our ML workflow for each of them.

ii. A scientific poster presented at IJUP 2023 at Rectory University of Porto (http://dx.doi.org/10.13140/RG.2.2.25674.75208).

iii. A machine learning algorithm also available at our website that utilized the descriptors as features was used to create three databases containing their top 25 descriptors when applied to the different origins of the database.

7. Bibliography

Agarwal, P., Soni, R., Kaur, P., Madan, A., Mishra, R., Pandey, J., . . . Singh, G. (2022). Cyanobacteria as a Promising Alternative for Sustainable Environment: Synthesis of Biofuel and Biodegradable Plastics. *Frontiers in Microbiology, 13*. doi:10.3389/fmicb.2022.939347

Ahmed, M. N., Wahlsten, M., Jokela, J., Nees, M., Stenman, U.-H., Alvarenga, D. O., . . . Fewer, D. P. (2021). Potent Inhibitor of Human Trypsins from the Aeruginosin Family of Natural Products. *ACS Chemical Biology, 16*(11), 2537-2546. doi:10.1021/acschembio.1c00611

Anas, A. R., Kisugi, T., Umezawa, T., Matsuda, F., Campitelli, M. R., Quinn, R. J., & Okino, T. (2012). Thrombin inhibitors from the freshwater cyanobacterium Anabaena compacta. *J Nat Prod, 75*(9), 1546-1552. doi:10.1021/np300282a

Anas, A. R. J., Kisugi, T., Umezawa, T., Matsuda, F., Campitelli, M. R., Quinn, R. J., & Okino, T. (2012). Thrombin Inhibitors from the Freshwater Cyanobacterium Anabaena compacta. *Journal of Natural Products, 75*(9), 1546-1552. doi:10.1021/np300282a

Arnison, P. G., Bibb, M. J., Bierbaum, G., Bowers, A. A., Bugni, T. S., Bulaj, G., . . . van der Donk, W. A. (2013). Ribosomally synthesized and posttranslationally modified peptide natural products: overview and recommendations for a universal nomenclature. *Nat Prod Rep, 30*(1), 108-160. doi:10.1039/c2np20085f

Bethan, K., & Carole, L. (2018). Secondary Metabolites in Cyanobacteria. In V. Ramasamy & S. S. R. Suresh (Eds.), *Secondary Metabolites* (pp. Ch. 2). Rijeka: IntechOpen.

Bouaïcha, N., Miles, C. O., Beach, D. G., Labidi, Z., Djabri, A., Benayache, N. Y., & Nguyen-Quang, T. (2019). Structural Diversity, Characterization and Toxicology of Microcystins. *Toxins, 11*(12), 714. Retrieved from https://www.mdpi.com/2072-6651/11/12/714

Braz, V. S., Melchior, K., & Moreira, C. G. (2020). Escherichia coli as a Multifaceted Pathogenic and Versatile Bacterium. *Frontiers in Cellular and Infection Microbiology, 10*. doi:10.3389/fcimb.2020.548492

Breiman, L. (2001). Random Forests. *Machine Learning, 45*(1), 5-32. doi:10.1023/A:1010933404324

Carina, M., Elisabete, V. r., & Elsa, D. (2013). The Kidney Vero-E6 Cell Line: A Suitable Model to Study the Toxicity of Microcystins. In G. Sivakumar (Ed.), *New Insights into Toxicity and Drug Testing* (pp. Ch. 2). Rijeka: IntechOpen.

Carneiro, J., Magalhães, R. P., de la Oliva Roque, V. M., Simões, M., Pratas, D., & Sousa, S. F. (2023). TargIDe: a machine-learning workflow for target identification of molecules with antibiofilm activity against Pseudomonas aeruginosa. *Journal of Computer-Aided Molecular Design, 37*(5), 265-278. doi:10.1007/s10822-023-00505-5

Chittora, D., Meena, M., Barupal, T., Swapnil, P., & Sharma, K. (2020). Cyanobacteria as a source of biofertilizers for sustainable agriculture. *Biochemistry and Biophysics Reports, 22*, 100737. doi:https://doi.org/10.1016/j.bbrep.2020.100737

Cutler, A., Cutler, D., & Stevens, J. (2011). Random Forests. In (Vol. 45, pp. 157-176).

D'Souza, S., Prema, K. V., & Balaji, S. (2020). Machine learning models for drug-target interactions: current knowledge and future directions. *Drug Discov Today, 25*(4), 748-756. doi:10.1016/j.drudis.2020.03.003

Dahms, H. U., Ying, X., & Pfeiffer, C. (2006). Antifouling potential of cyanobacteria: a mini-review. *Biofouling, 22*(5-6), 317-327. doi:10.1080/08927010600967261

Elemam, N. M., Al-Jaderi, Z., Hachim, M. Y., & Maghazachi, A. A. (2019). HCT-116 colorectal cancer cells secrete chemokines which induce chemoattraction and

intracellular calcium mobilization in NK92 cells. *Cancer Immunol Immunother, 68*(6), 883-895. doi:10.1007/s00262-019-02319-7

Ersmark, K., Del Valle, J. R., & Hanessian, S. (2008). Chemistry and biology of the aeruginosin family of serine protease inhibitors. *Angew Chem Int Ed Engl, 47*(7), 1202-1223. doi:10.1002/anie.200605219

Erwin, P. M., López-Legentil, S., & Schuhmann, P. W. (2010). The pharmaceutical value of marine biodiversity for anticancer drug discovery. *Ecological Economics, 70*(2), 445-451. doi:https://doi.org/10.1016/j.ecolecon.2010.09.030

Fiore, M. F., Alvarenga, D. O., Varani, A. M., Hoff-Risseti, C., Crespim, E., Ramos, R. T. J., . . . Schneider, M. P. C. (2013). Draft Genome Sequence of the Brazilian Toxic Bloom-Forming Cyanobacterium Microcystis aeruginosa Strain SPC777. *Genome announcements, 1*(4), e00547-00513. doi:10.1128/genomea.00547-13. (Accession No. 23908289)

Gaulton, A., Bellis, L. J., Bento, A. P., Chambers, J., Davies, M., Hersey, A., . . . Overington, J. P. (2012). ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Res, 40*(Database issue), D1100-1107. doi:10.1093/nar/gkr777

Gaulton, A., Hersey, A., Nowotka, M., Bento, A. P., Chambers, J., Mendez, D., . . . Leach, A. R. (2017). The ChEMBL database in 2017. *Nucleic Acids Res, 45*(D1), D945-d954. doi:10.1093/nar/gkw1074

Gu, W., Xie, X., He, Y., & Zhang, Z. (2018). [Drug-target protein interaction prediction based on AdaBoost algorithm]. *Sheng Wu Yi Xue Gong Cheng Xue Za Zhi, 35*(6), 935-942. doi:10.7507/1001-5515.201802026

Guo, Y., Song, G., Sun, M., Wang, J., & Wang, Y. (2020). Prevalence and Therapies of Antibiotic-Resistance in Staphylococcus aureus. *Frontiers in Cellular and Infection Microbiology, 10*. doi:10.3389/fcimb.2020.00107

Haapasalo, J., Nordfors, K., Järvelä, S., Bragge, H., Rantala, I., Parkkila, A. K., . . . Parkkila, S. (2007). Carbonic anhydrase II in the endothelium of glial tumors: a potential target for therapy. *Neuro Oncol, 9*(3), 308-313. doi:10.1215/15228517-2007-001

Hähnke, V. D., Kim, S., & Bolton, E. E. (2018). PubChem chemical structure standardization. *Journal of Cheminformatics, 10*(1), 36. doi:10.1186/s13321-018-0293-8

Halland, N., Brönstrup, M., Czech, J., Czechtizky, W., Evers, A., Follmann, M., . . . Kallus, C. (2015). Novel Small Molecule Inhibitors of Activated Thrombin Activatable Fibrinolysis Inhibitor (TAFIa) from Natural Product Anabaenopeptin. *J Med Chem, 58*(11), 4839-4844. doi:10.1021/jm501840b

Heller, S., Darpö, B., Mitchell, M. I., Linnebjerg, H., Leishman, D. J., Mehrotra, N., . . . Sager, P. (2015). Considerations for assessing the potential effects of antidiabetes drugs on cardiac ventricular repolarization: A report from the Cardiac Safety Research Consortium. *American Heart Journal, 170*(1), 23-35. doi:https://doi.org/10.1016/j.ahj.2015.03.007

Jones, M. R., Pinto, E., Torres, M. A., Dörr, F., Mazur-Marzec, H., Szubert, K., . . . Janssen, E. M. L. (2021). CyanoMetDB, a comprehensive public database of secondary metabolites from cyanobacteria. *Water Research, 196*, 117017. doi:https://doi.org/10.1016/j.watres.2021.117017

Joste, V., Maurice, L., Bertin, G. I., Aubouy, A., Boumédiène, F., Houzé, S., . . . Faucher, J. F. (2019). Identification of Plasmodium falciparum and host factors associated with cerebral malaria: description of the protocol for a prospective, case–control study in Benin (NeuroCM). *BMJ Open, 9*(5), e027378. doi:10.1136/bmjopen-2018-027378

Keith, J. A., Vassilev-Galindo, V., Cheng, B., Chmiela, S., Gastegger, M., Müller, K.-R., & Tkatchenko, A. (2021). Combining Machine Learning and Computational Chemistry for Predictive Insights Into Chemical Systems. *Chemical Reviews, 121*(16), 9816-9872. doi:10.1021/acs.chemrev.1c00107

Khalifa, S. A. M., Shedid, E. S., Saied, E. M., Jassbi, A. R., Jamebozorgi, F. H., Rateb, M. E., . . . El-Seedi, H. R. (2021). Cyanobacteria—From the Oceans to the Potential Biotechnological and Biomedical Applications. *Marine Drugs, 19*(5), 241. Retrieved from https://www.mdpi.com/1660-3397/19/5/241

Khalifa, S. A. M., Shedid, E. S., Saied, E. M., Jassbi, A. R., Jamebozorgi, F. H., Rateb, M. E., . . . El-Seedi, H. R. (2021). Cyanobacteria-From the Oceans to the Potential Biotechnological and Biomedical Applications. *Mar Drugs, 19*(5). doi:10.3390/md19050241

Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., . . . Bolton, E. E. (2020). PubChem in 2021: new data content and improved web interfaces. *Nucleic Acids Research, 49*(D1), D1388-D1395. doi:10.1093/nar/gkaa971

Kumar, K., Mella-Herrera, R. A., & Golden, J. W. (2010). Cyanobacterial heterocysts. *Cold Spring Harb Perspect Biol, 2*(4), a000315. doi:10.1101/cshperspect.a000315

Le Manach, S., Duval, C., Marie, A., Djediat, C., Catherine, A., Edery, M., . . . Marie, B. (2019). Global Metabolomic Characterizations of Microcystis spp. Highlights Clonal Diversity in Natural Bloom-Forming Populations and Expands Metabolite Structural Diversity. *Front Microbiol, 10*, 791. doi:10.3389/fmicb.2019.00791

Lee, A. V., Oesterreich, S., & Davidson, N. E. (2015). MCF-7 Cells—Changing the Course of Breast Cancer Research and Care for 45 Years. *JNCI: Journal of the National Cancer Institute, 107*(7). doi:10.1093/jnci/djv073

Lopes, G., Silva, M., & Vasconcelos, V. (2022). *The Pharmacological Potential of Cyanobacteria*.

Marahiel, M. A. (2009). Working outside the protein-synthesis rules: insights into nonribosomal peptide synthesis. *Journal of Peptide Science, 15*(12), 799-807. doi:https://doi.org/10.1002/psc.1183

Martin, C. E., & List, K. (2019). Cell surface-anchored serine proteases in cancer progression and metastasis. *Cancer Metastasis Rev, 38*(3), 357-387. doi:10.1007/s10555-019-09811-7

Martínez-Maqueda, D., Miralles, B., & Recio, I. (2015). HT29 Cell Line. In K. Verhoeckx, P. Cotter, I. López-Expósito, C. Kleiveland, T. Lea, A. Mackie, T. Requena, D. Swiatecka, & H. Wichers (Eds.), *The Impact of Food Bioactives on Health: in vitro and ex vivo models* (pp. 113-124). Cham: Springer International Publishing.

Mazur-Marzec, H., Fidor, A., Cegłowska, M., Wieczerzak, E., Kropidłowska, M., Goua, M., . . . Edwards, C. (2018). Cyanopeptolins with Trypsin and Chymotrypsin Inhibitory Activity from the Cyanobacterium Nostoc edaphicum CCNP1411. *Mar Drugs, 16*(7). doi:10.3390/md16070220

Melo, F. (2013). Area under the ROC Curve. In W. Dubitzky, O. Wolkenhauer, K.-H. Cho, & H. Yokota (Eds.), *Encyclopedia of Systems Biology* (pp. 38-39). New York, NY: Springer New York.

Mendez, D., Gaulton, A., Bento, A. P., Chambers, J., De Veij, M., Félix, E., . . . Leach, Andrew R. (2018). ChEMBL: towards direct deposition of bioassay data. *Nucleic Acids Research, 47*(D1), D930-D940. doi:10.1093/nar/gky1075

Micallef, M. L., D'Agostino, P. M., Al-Sinawi, B., Neilan, B. A., & Moffitt, M. C. (2015). Exploring cyanobacterial genomes for natural product biosynthesis pathways. *Mar Genomics, 21*, 1-12. doi:10.1016/j.margen.2014.11.009

Micallef, M. L., D'Agostino, P. M., Sharma, D., Viswanathan, R., & Moffitt, M. C. (2015). Genome mining for natural product biosynthesis-related gene clusters in the Subsection V cyanobacteria. *BMC Genomics, 16*(1), 669. doi:10.1186/s12864-015-1855-z

Monteiro, P. R., do Amaral, S. C., Siqueira, A. S., Xavier, L. P., & Santos, A. V. (2021). Anabaenopeptins: What We Know So Far. *Toxins (Basel), 13*(8). doi:10.3390/toxins13080522

Moore, D. E., Weise, K., Zawydiwski, R., & Thompson, E. B. (1985). The karyotype of the glucocorticoid-sensitive, lymphoblastic human T-cell line CCRF-CEM shows

a unique deleted and inverted chromosome 9. *Cancer Genetics and Cytogenetics, 14*(1), 89-94. doi:https://doi.org/10.1016/0165-4608(85)90219-5

Moriwaki, H., Tian, Y.-S., Kawashita, N., & Takagi, T. (2018). Mordred: a molecular descriptor calculator. *Journal of Cheminformatics, 10*(1), 4. doi:10.1186/s13321-018-0258-y

Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics, 7*. doi:10.3389/fnbot.2013.00021

O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., & Hutchison, G. R. (2011). Open Babel: An open chemical toolbox. *Journal of Cheminformatics, 3*(1), 33. doi:10.1186/1758-2946-3-33

Pastorekova, S., & Gillies, R. J. (2019). The role of carbonic anhydrase IX in cancer development: links to hypoxia, acidosis, and beyond. *Cancer Metastasis Rev, 38*(1-2), 65-77. doi:10.1007/s10555-019-09799-0

Patel, V., Berthold, D., Puranik, P., & Gantar, M. (2015). Screening of cyanobacteria and microalgae for their ability to synthesize silver nanoparticles with antibacterial activity. *Biotechnol Rep (Amst), 5*, 112-119. doi:10.1016/j.btre.2014.12.001

Pathak, J., Rajneesh, Maurya, P., Singh, s. p., Häder, D., & Sinha, R. (2018). Cyanobacterial Farming for Environment Friendly Sustainable Agriculture Practices: Innovations and Perspectives. *Frontiers in Environmental Science, 6*. doi:10.3389/fenvs.2018.00007

Pattanaik, B., & Lindberg, P. (2015). Terpenoids and their biosynthesis in cyanobacteria. *Life (Basel), 5*(1), 269-293. doi:10.3390/life5010269

Pence, H. E., & Williams, A. (2010). ChemSpider: An Online Chemical Information Resource. *Journal of Chemical Education, 87*(11), 1123-1124. doi:10.1021/ed100697w

Pham, T.-L., & Utsumi, M. (2018). An overview of the accumulation of microcystins in aquatic ecosystems. *Journal of Environmental Management, 213*, 520-529. doi:https://doi.org/10.1016/j.jenvman.2018.01.077

Philmus, B., Christiansen, G., Yoshida, W. Y., & Hemscheidt, T. K. (2008). Posttranslational modification in microviridin biosynthesis. *Chembiochem, 9*(18), 3066-3073. doi:10.1002/cbic.200800560

Prasanna, R., Sood, A., Jaiswal, P., Nayak, S., Gupta, V., Chaudhary, V., . . . Natarajan, C. (2010). Rediscovering cyanobacteria as valuable sources of bioactive compounds (Review). *Applied Biochemistry and Microbiology, 46*(2), 119-134. doi:10.1134/S0003683810020018

Qamar, H., Hussain, K., Soni, A., Khan, A., Hussain, T., & Chénais, B. (2021). Cyanobacteria as Natural Therapeutics and Pharmaceutical Potential: Role in Antitumour Activity and as Nanovectors. *Molecules, 26*(1). doi:10.3390/molecules26010247

Rajeshkumar, S., Malarkodi, C., Paulkumar, K., Vanaja, M., Gnanajobitha, G., & Annadurai, G. (2013). Intracellular and extracellular biosynthesis of silver nanoparticles by using marine bacteria Vibrio alginolyticus. *Nanosci Nanotechnol, 3*(1), 21-25.

Ramos, V., Morais, J., Castelo-Branco, R., Pinheiro, Â., Martins, J., Regueiras, A., . . . Vasconcelos, V. M. (2018). Cyanobacterial diversity held in microbial biological resource centers as a biotechnological asset: the case study of the newly established LEGE culture collection. *J Appl Phycol, 30*(3), 1437-1451. doi:10.1007/s10811-017-1369-y

Sahu, N., Mishra, S., Kesheri, M., Kanchan, S., & Sinha, R. P. (2023). Identification of Cyanobacteria-Based Natural Inhibitors Against SARS-CoV-2 Druggable Target ACE2 Using Molecular Docking Study, ADME and Toxicity Analysis. *Indian J Clin Biochem, 38*(3), 361-373. doi:10.1007/s12291-022-01056-6

Sander, T., Freyss, J., von Korff, M., & Rufener, C. (2015). DataWarrior: An Open-Source Program For Chemistry Aware Data Visualization And Analysis. *Journal of Chemical Information and Modelling, 55*(2), 460-473. doi:10.1021/ci500588j

Schapire, R. E. (2013). Explaining AdaBoost. In B. Schölkopf, Z. Luo, & V. Vovk (Eds.), *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik* (pp. 37-52). Berlin, Heidelberg: Springer Berlin Heidelberg.

Seckbach, J. (2007). *Algae and cyanobacteria in extreme environments* (Vol. 11): Springer Science & Business Media.

Seko, A., Togo, A., & Tanaka, I. (2018). Descriptors for Machine Learning of Materials Data. In I. Tanaka (Ed.), *Nanoinformatics* (pp. 3-23). Singapore: Springer Singapore.

Singh, A. K., Sharma, L., Mallick, N., & Mala, J. (2017). Progress and challenges in producing polyhydroxyalkanoate biopolymers from cyanobacteria. *Journal of Applied Phycology, 29*(3), 1213-1232. doi:10.1007/s10811-016-1006-1

Singh, J. S., Kumar, A., Rai, A. N., & Singh, D. P. (2016). Cyanobacteria: A Precious Bioresource in Agriculture, Ecosystem, and Environmental Sustainability. *Front Microbiol, 7*, 529. doi:10.3389/fmicb.2016.00529

Singh, R., Parihar, P., Singh, M., Bajguz, A., Kumar, J., Singh, S., . . . Prasad, S. M. (2017). Uncovering Potential Applications of Cyanobacteria and Algal Metabolites in Biology, Agriculture and Medicine: Current Status and Future Prospects. *Frontiers in Microbiology, 8*. doi:10.3389/fmicb.2017.00515

Singh, R. K., Tiwari, S. P., Rai, A. K., & Mohapatra, T. M. (2011). Cyanobacteria: an emerging source for drug discovery. *The Journal of Antibiotics, 64*(6), 401-412. doi:10.1038/ja.2011.21

Strunecký, O., Ivanova, A. P., & Mareš, J. (2023). An updated classification of cyanobacterial orders and families based on phylogenomic and polyphasic analysis. *Journal of Phycology, 59*(1), 12-51. doi:https://doi.org/10.1111/jpy.13304

Subramaniyan, V. (2012). Potential applications of cyanobacteria in industrial effluents - a review. *Journal of Bioremediation and Biodegradation, 3*(6), 1000154.

Svirčev, Z., Drobac, D., Tokodi, N., Mijović, B., Codd, G. A., & Meriluoto, J. (2017). Toxicology of microcystins with reference to cases of human intoxications and epidemiological investigations of exposures to cyanobacteria and cyanotoxins. *Archives of Toxicology, 91*(2), 621-650. doi:10.1007/s00204-016-1921-6

Swain, S. S., Paidesetty, S. K., & Padhy, R. N. (2017). Antibacterial, antifungal and antimycobacterial compounds from cyanobacteria. *Biomedicine & Pharmacotherapy, 90*, 760-776. doi:https://doi.org/10.1016/j.biopha.2017.04.030

Tagirasa, R., & Yoo, E. (2022). Role of Serine Proteases at the Tumor-Stroma Interface. *Frontiers in Immunology, 13*. doi:10.3389/fimmu.2022.832418

Tanabe, Y., Hodoki, Y., Sano, T., Tada, K., & Watanabe, M. M. (2018). Adaptation of the Freshwater Bloom-Forming Cyanobacterium Microcystis aeruginosa to Brackish Water Is Driven by Recent Horizontal Transfer of Sucrose Genes. *Frontiers in microbiology, 9*, 1150. doi:10.3389/fmicb.2018.01150. (Accession No. 29922255)

Taunk, K., De, S., Verma, S., & Swetapadma, A. (2019, 15-17 May 2019). *A Brief Review of Nearest Neighbor Algorithm for Learning and Classification.* Paper presented at the 2019 International Conference on Intelligent Computing and Control Systems (ICCS).

Tiwari, A., & Pandey, A. (2012). Cyanobacterial hydrogen production – A step towards clean environment. *International Journal of Hydrogen Energy, 37*(1), 139-150. doi:https://doi.org/10.1016/j.ijhydene.2011.09.100

Tomitani, A., Knoll, A. H., Cavanaugh, C. M., & Ohno, T. (2006). The evolutionary diversification of cyanobacteria: Molecular–phylogenetic and palaeontological perspectives. *Proceedings of the National Academy of Sciences, 103*(14), 5442-5447. doi:doi:10.1073/pnas.0600999103

Townsend, M. H., Anderson, M. D., Weagel, E. G., Velazquez, E. J., Weber, K. S., Robison, R. A., & O'Neill, K. L. (2017). Non-small cell lung cancer cell lines A549

and NCI-H460 express hypoxanthine guanine phosphoribosyltransferase on the plasma membrane. *Onco Targets Ther, 10*, 1921-1932. doi:10.2147/ott.S128416

van Santen, J. A., Jacob, G., Singh, A. L., Aniebok, V., Balunas, M. J., Bunsko, D., . . . Linington, R. G. (2019). The Natural Products Atlas: An Open Access Knowledge Base for Microbial Natural Products Discovery. *ACS Central Science, 5*(11), 1824-1833. doi:10.1021/acscentsci.9b00806

van Santen, J. A., Poynton, E. F., Iskakova, D., McMann, E., Alsup, Tyler A., Clark, T. N., . . . Linington, R. G. (2021). The Natural Products Atlas 2.0: a database of microbially derived natural products. *Nucleic Acids Research, 50*(D1), D1317-D1323. doi:10.1093/nar/gkab941

Vaughan, L., Glänzel, W., Korch, C., & Capes-Davis, A. (2017). Widespread Use of Misidentified Cell Line KB (HeLa): Incorrect Attribution and Its Impact Revealed through Mining the Scientific Literature. *Cancer Research, 77*(11), 2784-2788. doi:10.1158/0008-5472.Can-16-2258

Vercauteren, E., Gils, A., & Declerck, P. J. (2013). Thrombin activatable fibrinolysis inhibitor: a putative target to enhance fibrinolysis. *Semin Thromb Hemost, 39*(4), 365-372. doi:10.1055/s-0033-1334488

Xuan, P., Sun, C., Zhang, T., Ye, Y., Shen, T., & Dong, Y. (2019). Gradient Boosting Decision Tree-Based Method for Predicting Interactions Between Target Genes and Drugs. *Frontiers in Genetics, 10*. doi:10.3389/fgene.2019.00459

Yap, C. W. (2011). PaDEL-descriptor: An open source software to calculate molecular descriptors and fingerprints. *Journal of Computational Chemistry, 32*(7), 1466-1474. doi:https://doi.org/10.1002/jcc.21707

Zhang, Z. (2016). Introduction to machine learning: k-nearest neighbors. *Ann Transl Med, 4*(11), 218. doi:10.21037/atm.2016.03.37