Quim. Nova, Vol. 47, No. 7, e-20240024, 1-16, 2024 http://dx.doi.org/10.21577/0100-4042.20240024

APRENDIZADO DE MÁQUINA APLICADO A QSAR

Renata P. B. de Menezes[a], Luciana Scotti[a] e Marcus T. Scotti[a,*]
[a]Departamento de Farmácia, Universidade Federal da Paraíba, 58051-900 João Pessoa – PB, Brasil

Recebido em 13/10/2023; aceito em 22/12/2023; publicado na web 05/03/2024

MACHINE LEARNING APLLIED TO QSAR. Over the years the study of the quantitative structure-activity relationship (QSAR) has transformed from a simple regression analysis to the implementation of machine learning (ML) with multiple statistics. Today ML-based QSAR models are quite important and play a notable role in drug design and screening, property prediction, biological activity, etc. ML methods applied to QSAR build classification or regression models to describe/predict the complex relationships between the chemical structure of molecules and biological activity. Even with the increase in scientific publications addressing this topic written in Portuguese, there is still a shortage of scientific articles explaining ML techniques applied to QSAR, how to build models, the types of models, algorithms, for the Brazilian scientific community. And to fill this need, we intend to approach the subject in a simple and didactic way for students and researchers who are starting in this very promising and important area. We will describe the fully explained theory of machine learning by applying QSAR, abstracting the complexity, and well-illustrated.

Keywords: machine learning; QSAR; predictive models; data mining; virtual screening.

INTRODUÇÃO

A quimioinformática surgiu por volta dos anos 90 e tem se desenvolvido e aperfeiçoado com o passar dos anos e com a evolução da tecnologia. As metodologias desenvolvidas pela quimioinformática têm sido aplicadas cada vez mais no campo da descoberta de novas drogas (fármacos e agroquímicos), inclusive, uma das principais razões para o seu surgimento foi a necessidade de trabalhar com a imensa quantidade de dados gerados por diversas abordagens cada vez mais automatizadas para a descoberta de novos medicamentos, como alta triagem de rendimento (*high-throughput screening* - HTS) e química combinatória.[1,2]

Um dos fundadores da quimioinformática, Frank Brown, a definiu em 1998 como:[3,4]

> "A mistura de recursos de informação para transformar dados em informação e informação em conhecimento, no intuito de tomar decisões melhores e mais rápidas na área de identificação e otimização de compostos líderes."

Outros pesquisadores, tidos como pilares da quimioinformática, também a definiram ao longo dos anos. J. Gasteiger a definiu em 2004 como sendo a aplicação de métodos de informática para resolver problemas químicos. E em 2007, Alexander Varneck, definiu a quimioinformática como um campo que lida com objetos moleculares (gráficos, vetores) no espaço químico multidimensional.[1,5,6]

O aperfeiçoamento da quimioinformática tornou possível a exploração de grandes bases de dados químicos na busca e descoberta de novos compostos com potenciais elevados de atividade biológica ou propriedade química desejadas, ou seja, tem sido implementada para a triagem virtual (*virtual screening* - VS), acelerando a descoberta de medicamentos.[1,3,7,8]

Essa ciência tem papel fundamental no desenvolvimento racional de novas drogas antes de iniciar as análises de fase clínica, ou seja, torna possível a seleção racional de compostos promissores para seguir os estudos biológicos e clínicos poupando tempo de pesquisa e custos.[7,9,10]

*e-mail: mtscotti@gmail.com

Uma das técnicas da quimioinformática mais conhecida, utilizada e estudada é a relação quantitativa estrutura-atividade (*quantitative structure-activity relationship* - QSAR), que é um conjunto de métodos matemáticos e procedimentos computacionais bastante estabelecidos para análise de dados químicos.[11] O objetivo principal do QSAR é estabelecer a relação quantitativa entre a atividade (QSAR)/propriedade (QSPR)/toxicidade molecular (QSTR)/cromatografia (QSCR)/eletroquímica (QSER)/biodegradabilidade (QSBR) entre outros e seus parâmetros estruturais.[12,13]

RELAÇÃO QUANTITATIVA ESTRUTURA-ATIVIDADE (QSAR)

QSAR é constantemente definido como a geração de modelos matemáticos estabelecendo relações empíricas, estatisticamente significativas, lineares ou não lineares entre valores de descritores químicos calculados a partir da estrutura molecular e propriedades ou bioatividade dessas moléculas medidas experimentalmente, seguidas da aplicação desses modelos para prever ou projetar novos produtos químicos com propriedades desejadas.[3,8,11,14-16]

O método de QSAR tem facilitado significativamente a descoberta e desenvolvimento de novas drogas e agroquímicos, demostrando alta eficiência na redução de custos e tempo nesse processo, além de aumentar a taxa de sucesso e o retorno do investimento.[17-21] Isso se deve ao fato de que nenhum composto precisa ser sintetizado ou testado antes da avaliação computacional. Embora o relato mais antigo da origem de QSAR remonte a um estudo sobre a relação estrutura-atividade de compostos orgânicos no século 19,[22,23] o método proposto por Hansch e Fujita no início dos anos 60 marcou o advento do 2D-QSAR. Esse método estabeleceu uma correlação linear entre atividade biológica e parâmetros físico-químicos bidimensionais, como massa molecular, número de átomos, ligações, área superficial, número de anéis e tipos de fragmentos. Essencialmente, utiliza-se a representação da estrutura química da molécula em duas dimensões, seguido pelo estudo de Free-Wilson.[22,24-26] QSAR continuou evoluindo com o tempo, seguindo com o método representativo 3D-QSAR, no qual, passou-se a utilizar propriedades moleculares tridimensionais (3D) estéricos e eletrostáticos, que quando combinada com a regressão por mínimos quadrados parciais (*partial least*

squares - PLS) para estabelecer relação com a atividade biológica, é conhecido como método de análise comparativa de campo molecular (*comparative molecular fields analysis* - CoMFA), proposto e amplamente utilizado no projeto e previsão de novas moléculas.[22,27]

Com a evolução da tecnologia o QSAR de quatro, cinco, seis e sete dimensões (4D, 5D, 6D e 7D) foram desenvolvidos no decorrer dos anos.[28-30] O 4D-QSAR, por exemplo, expandiu as análises para considerar o tempo como um fator adicional, levando em conta a dinâmica das interações moleculares ao longo do tempo. Já o 5D-QSAR acrescentou a dimensão da solvatação, considerando as interações da molécula com o solvente. O 6D-QSAR introduziu a complexidade da cinética química, permitindo uma avaliação mais precisa das taxas de reação. Por fim, o 7D-QSAR foi uma evolução adicional, incorporando informações sobre as propriedades químicas tridimensionais das moléculas, a influência do solvente, o tempo e a cinética, proporcionando uma análise ainda mais abrangente e precisa das relações entre estrutura molecular e atividade biológica. Essas abordagens mais avançadas têm contribuído significativamente para o campo da química medicinal e o desenvolvimento de novos fármacos e agroquímicos.[3,31-33]

Até os dias atuais, os métodos de QSAR tem experimentado um progresso enorme, em uma ampla área de pesquisa se estendendo até fora dos limites tradicionais (desenvolvimento de novas drogas), como planejamento de rotas sintéticas, nanotecnologia, ciência de materiais, biomateriais, na indústria de alimentos, clínica informática, entre outros.[11,22,31]

Podemos afirmar que no decorrer dos anos o QSAR se transformou de uma análise de regressão simples, ou seja, que só podia trabalhar com séries de compostos estruturalmente semelhantes e algumas poucas propriedades, para o implemento da técnica de aprendizado de máquina (*machine learning* - ML) com estatísticas múltiplas, sendo possível analisar grandes conjuntos de moléculas estruturalmente diversas.[20] Na análise de regressão linear simples em QSAR convencionou-se o uso de uma propriedade a cada conjunto de cinco moléculas, isso para diminuir as chances de super ajuste no modelo, dessa forma, um modelo contendo uma série de 15 moléculas só pode acomodar três propriedades, já em uma análise de QSAR com ML é possível usar centenas de propriedades mesmo que para um conjunto de poucas moléculas. Sendo amplamente utilizado em indústrias, universidades e centros de pesquisas em todo o mundo.[14,33]

Hoje, os modelos de QSAR baseados em ML são bastante importantes e tem desempenhado um papel notável no desenho e triagem de drogas, previsão de propriedade, atividade biológica etc. A aplicação de técnicas de ML em QSAR teve início desde a década de 1990, com o uso de algoritmos como máquina de vetor de suporte (*support vector machine* - SVM) e florestas aleatórias (*random forest* - RF) para a descoberta ou projeção de novos medicamentos e também de compostos agroquímicos.[20]

O ML pode ser definido como o uso de algoritmos de inteligência artificial que podem aprender do seu ambiente.[34] Os métodos de ML aplicado a QSAR constroem modelos de classificação ou de regressão para descrever/prever as relações complexas entre a estrutura química das moléculas e a atividade biológica.[35]

O ML tem a capacidade de aprender com os dados quais recursos são mais importantes para determinar a relação estrutura-atividade e realizar a predição. Então, na construção dos modelos preditivos é imprescindível ter um banco de dados com todos os recursos/informações possíveis relacionados a finalidade desejada do QSAR e quando aplicado o algoritmo ele irá aprender e extrair aquelas informações mais importantes.

Independente da natureza dos dados, as abordagens de ML podem ser usadas universalmente para analisar e processar dados em qualquer domínio. É importante lembrar que mesmo a ideia geral do ML aplicado ao QSAR seja a mesma, as características de entrada, ou seja, os dados utilizados podem ser diferentes além de que os alvos também podem ser diferentes. Em suma, quando o ML é aplicado ao QSAR essa ampla aplicabilidade só comprova o QSAR como uma ferramenta de modelagem e análise de dados preditiva e robusta.[11,20]

TRIAGEM VIRTUAL

A triagem virtual (*virtual screening* - VS) foi um termo criado na década de 1990 com o intuito de descrever a utilização de modelos computacionais e algoritmos para analisar grandes bancos de moléculas de forma automatizada. O objetivo da VS é triar e selecionar compostos potencialmente ativos, identificar e remover compostos tóxicos ou que apresentem propriedades farmacodinâmicas e farmacocinéticas desfavoráveis.[3,36-38]

Podemos compreender VS como um método que utiliza uma cascata de filtros sequenciais que irão restringir o espaço químico e selecionar as moléculas mais semelhantes aos compostos líderes e assim apresentarem uma maior probabilidade de potencial ativo.[7,39]

VS teve seu surgimento na mesma década que QSAR e ambos estão intimamente relacionados, pois em comum foram implementados para solucionar alguns problemas do HTS (alta triagem de rendimento, do inglês *high-throughput screening* - HTS) e da química combinatória, como analisar a grande quantidade de dados gerados e diminuir o tempo e os gastos da pesquisa. Enquanto no HTS a busca por compostos potencialmente ativos é praticamente uma busca aleatória, por sua vez, VS é um processo orientado pelo conhecimento utilizando técnicas computacionais também denominadas *in silico*.[7]

A disponibilidade de informação é imprescindível para a realização de VS, e a depender do tipo de informação, VS pode ser realizada através de duas técnicas: triagem virtual baseada na estrutura do receptor (*structure-based virtual screening* - SBVS) e a triagem virtual baseada na estrutura do ligante (*ligand-based virtual screening* - LBVS).[9,40]

Na técnica de SBVS é necessário o conhecimento tridimensional do alvo biológico, ou seja, da proteína alvo. A partir dessa informação inicia-se a busca por ligantes que possam interagir no sítio ativo dessa proteína, sendo, portanto, o *docking* molecular, a principal estratégia desta técnica. Essa abordagem permite que o pesquisador aprofunde seus conhecimentos da estrutura 3D do alvo e seu sítio ativo possibilitando estudos de otimização dos compostos para as interações moleculares com a proteína alvo além de otimização de propriedades farmacocinéticas.[7,9,41]

Na técnica de LBVS não é preciso a informação da estrutura da proteína alvo, o mais importante é ter conhecimento de moléculas que já foram avaliadas experimentalmente quanto a atividade biológica desejada. Explorando o princípio de que moléculas semelhantes apresentam perfil de atividade/propriedade semelhante, a técnica LBVS busca a partir de parâmetros, propriedades físico-químicas e impressões digitais (*fingerprint*) das moléculas, obter compostos com potencial atividade biológica através de uma análise de QSAR.[7,9,39,41] Por não precisar do conhecimento da estrutura 3D da proteína alvo, a abordagem LBVS acaba ficando restrita ao espaço químico dos compostos que já foram avaliados experimentalmente utilizados como referência.[37,42,43]

Neste artigo, focaremos nos modelos computacionais utilizando o ML na técnica de LBVS em QSAR.

Triagem virtual baseada na estrutura do ligante (LBVS)

O processo de criação do modelo computacional com abordagem

LBVS em QSAR é um ciclo que pode ser resumido da seguinte maneira:[35]

1. A coleta de dados;
2. Cálculo de descritores moleculares, descritores *fingerprint*, ou outras informações moleculares;
3. Construção dos modelos;
4. Verificação e validação dos modelos;
5. Testes *in vitro*

Esse ciclo tem início com a coleta de dados a partir de fontes externas, na literatura, em bibliotecas de laboratórios de pesquisa, em bancos de dados, no qual serão coletadas bancos de moléculas que já tiveram sua atividade alvo avaliada, um exemplo de fonte é o ChEMBL,[44] desenvolvido pelo EMBL - Laboratório Europeu de Biologia Molecular, é um banco de dados "quimiogênico" que integra informações químicas, de bioatividade e genômicas. Oferece acesso a uma gama de dados, incluindo informações de ensaios biológicos (*in vitro*, *in vivo*) e resultados de ensaios de HTS. Em seguida, é necessário o cálculo de descritores moleculares, descritores de *fingerprint*, ou obtenção de outras informações (dados de ressonância magnética nuclear - RMN, espectros de massas, entre outras) que descrevam as moléculas que foram coletadas para que os modelos computacionais possam ser gerados. A seguir os modelos desenvolvidos serão validados para serem utilizados na predição de compostos potencialmente ativos. Uma vez identificados esses compostos, eles serão encaminhados para a avaliação *in vitro*, gerando novas informações que podem ser coletadas e dar início novamente ao ciclo. Ou seja, a cada estudo realizado, novas informações são geradas que ficarão disponíveis para novos estudos.

A Figura 1 resume essas etapas. Cada uma dessas etapas engloba vários outros passos que precisam ser seguidos para que a qualidade e sucesso do modelo QSAR seja atingido. A seguir cada um desses passos será abordado e explicado a fim de que o leitor tenha uma boa compreensão sobre a criação de modelos preditivos de QSAR com o ML.

MINERAÇÃO DE DADOS

Há disponível hoje uma grande quantidade de informação de moléculas em vários repositórios de domínio público e privado, e para processar essas informações é necessário ferramentas computacionais. Por exemplo, no PubChem, um repositório público de estruturas químicas e suas propriedades biológicas há mais de 115 milhões de compostos e 1,5 milhão de bioensaios.[45-47] É uma quantidade de dados absurda que são atualizados diariamente, possuindo uma variedade enorme de informação de resposta alvo.

Semelhante ao PubChem, há também o ChEMBL que já foi mencionado acima. Comparado ao PubChem, no ChEMBL há uma grande quantidade de informações de dados selecionados manualmente da literatura. Estima-se que no ChEMBL contenha mais de 2,4 milhões de compostos testados contra mais de quinze mil alvos biológicos.[45,48] Há outras fontes projetadas especificamente para medicamentos e candidatos a fármacos, como o DrugBank que é um banco de dados público que contém todos os medicamentos aprovados e informações como seus mecanismos de ação, interações e alvos relevantes.[45,49]

Há também disponível na internet vários bancos de dados voltados para metabólitos secundários, disponibilizando informações importantes para pesquisadores de produtos naturais, como o Dicionário de Produtos Naturais (DPN), NAPRALERT, Marinelit para produtos naturais marinhos, o NPASS que além de metabolitos secundários fornece valores de atividade experimental e informações de fontes de espécies, propriedades químicas, NuBBEDB a primeira biblioteca de produtos naturais da biodiversidade brasileira, BIOFACQUIM, um banco de dados de produtos naturais isolados e caracterizados no México, há também bancos de produtos naturais provenientes de bactérias como o PAMDB e o StreptomeDB (*Pseudomonas aeruginosa* e gênero *Streptomyces*, respectivamente) e há o SistematX com um diferencial por fornecer bancos estruturais com capacidade de busca por estrutura, código SMILES (sistema simplificado de entrada

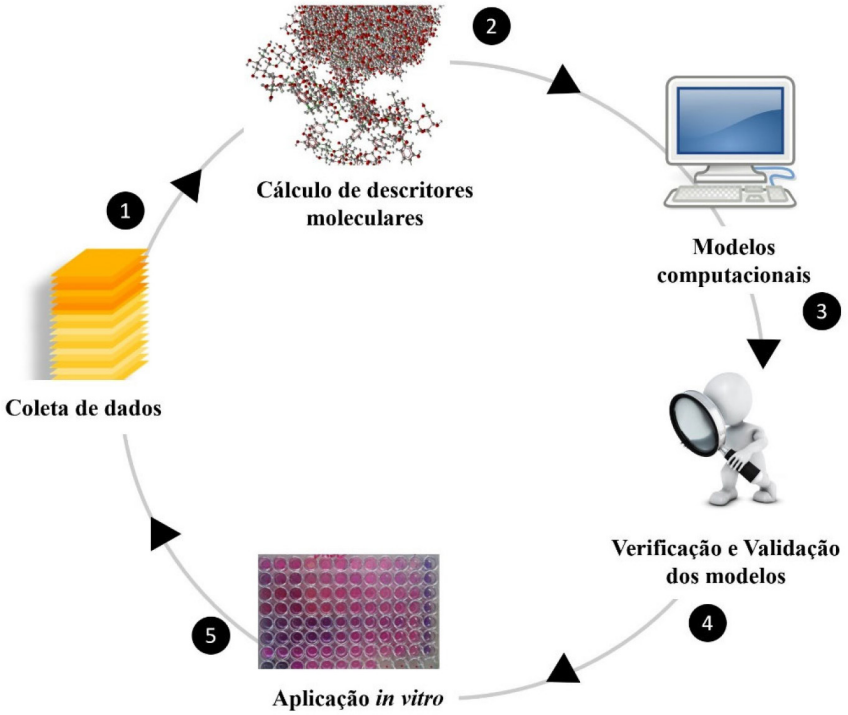

Figura 1. *Ciclo do processo de criação do modelo computacional com abordagem LBVS em QSAR; iniciando pela coleta de dados (1), seguindo para cálculo de descritores moleculares (2), modelos computacionais (3), verificação e validação dos modelos (4) e por fim a aplicação in vitro (5) que seguirá para a etapa 1 em um ciclo contínuo (fonte: elaborado pelo autor)*

de linha de entrada molecular), informações do nome do composto e espécie, possibilidade de salvar estruturas químicas encontradas pela busca, contém informações e características importantes para a química de produtos naturais e o usuário pode encontrar informações específicas sobre classificação taxonômica (da família a espécie) da planta a partir da qual o composto foi isolado, além das referências bibliográficas e do sistema de posicionamento global (GPS) da molécula pesquisada.[50-56]

Esses dados disponíveis podem e devem ser trabalhados para gerarem informações, e no nosso caso, esses dados devem ser trabalhos para obtenção de informação de novos possíveis medicamentos, ou seja, na descoberta de novas drogas. Assim, após a obtenção de dados precisamos minerá-lo para extrair as informações e o conhecimento que buscamos.

A mineração de dados são processos e etapas que serão realizadas para explorar e analisar uma grande quantidade de dados em busca de padrões, previsões, erros, associações, etc.[57] A mineração de dados está geralmente associada ao aprendizado de máquina. Mas para minerar dados é importante que esses dados sejam de qualidade. Dados com problemas, como erros em estruturas químicas e resultados experimentais, dificultam e se tornam um grande obstáculo para a construção de modelos preditivos.[33,58,59]

Por essa razão é preciso avaliar a qualidade dos dados, o que Fourches *et al.*[60] denominaram de "cinco Is", ou seja, avaliar se os dados podem estar incompletos, imprecisos, incorretos, incompatíveis e/ou irreprodutíveis. Os autores ainda reforçam a necessidade de tratar os dados como o primeiro passo crítico em qualquer estudo de análise de dados com o intuito de garantir estabilidade e confiabilidade dos resultados obtidos. Fourches *et al.*[60] propuseram ainda um fluxo geral de tratamento de dados químicos e biológicos, visando estudos de QSAR. O fluxo está representado na Figura 2 o qual apresenta 8 etapas que serão explicadas a seguir:

Etapa 1: Aplicação de algumas abordagens quimioinformáticas para sinalizar e, em alguns casos, corrigir entradas possivelmente errôneas em grandes conjuntos de dados. É nesta etapa que será feita a remoção de misturas, de compostos inorgânicos e organometálicos, verificação da falta de estruturas químicas, conversão estrutural, limpeza/remoção de sal, padronização de quimiotipos específicos como por exemplo a representação canônica personalizada, ou seja, alterar a forma de escrever o anel aromático de modo a incluir todas as formas ressonantes. Tratamento de formas tautoméricas, e esta etapa inclui também uma inspeção manual para verificação das estruturas químicas, se há erros em suas estruturas.[3,60]

Etapa 2: Análise de duplicatas. Pode acontecer do banco de dados apresentar moléculas repetidas, caso positivo é preciso avaliar a atividade experimental dessas moléculas. Se apresentarem valores de atividades contraditórios, essas moléculas devem ser investigadas, e apresentando valores semelhantes, retira-se apenas uma delas. Moléculas repetidas no conjunto de teste geram modelos superestimados. Moléculas repetidas no conjunto de treino podem afetar o modelo de várias formas, como na sua confiabilidade e super ajuste.

Etapas 3 e 4: Análise da variabilidade intra- e inter-laboratorial e exclusão de fontes não confiáveis. Nessas etapas os dados experimentais serão analisados e aqueles de fontes não confiáveis devem ser excluídos. Com esses passos, a qualidade dos dados é aumentada e consequentemente, aumenta-se a confiabilidade do modelo.

Etapa 5: Detecção e análise dos *cliffs* de atividade. Um *cliff* de atividade pode ser explicado como sendo estruturas semelhantes, mas com atividade muito diferentes.[3,60,61] É nesse passo também que os *outliers* serão detectados e analisados, ou seja, aquelas amostras com valores de atividade atípicos. A análise de *cliffs* e *outliers* deve ser rigorosa e a remoção dessas amostras devem ser justificadas. Uma vez que a remoção delas pode levar a melhora das estatísticas dos modelos, assim, se não houver uma justificativa, esse ato pode ser considerado manipulação dos dados.

Etapa 6: Cálculo do índice de modelabilidade do conjunto de dados. Esse índice foi criado por Golbraikh *et al.*[62] que estimam a

Figura 2. *Esquema geral de tratamento de dados (fonte: adaptado de Fourches et al.)*[60]

viabilidade de obter modelos QSAR preditivos de bancos de dados binário de compostos bioativos. Através do cálculo desse índice é possível identificar conjuntos de dados modeláveis ou não modeláveis. Sendo, portanto, um indicador da qualidade dos dados.

Etapa 7: Predição consensual por múltiplos modelos de QSAR. Durante esta etapa, serão gerados diversos modelos de QSAR independentes, variando tipos de descritores moleculares e/ou algoritmos. Em seguida será realizado o consenso dos modelos, ou seja, será calculado a média das predições dos modelos independentes.[3,60]

Etapa 8: Identificação e correção de compostos erroneamente anotados. Através do passo 7 é possível realizar o último passo do fluxo de curadoria de dados, uma vez que, dados biológicos incorretos podem ser identificados investigando-se e analisando aqueles compostos que apresentaram atividade biológica predita muito diferente do valor experimental.[3,60]

Estrutura dos dados

Após obter o banco de dados tratado com o conjunto de moléculas e informações biológicas confiáveis, o próximo passo é obter informações das moléculas. Essas informações são os descritores moleculares.

De acordo com o os pesquisadores Todeschini e Consonni,[63] descritores moleculares são um resultado final de um procedimento lógico e matemático que transforma a informação química codificada dentro de uma representação simbólica de uma molécula em um número útil, ou o resultado de algum experimento padronizado.

Em outras palavras, um descritor molecular é uma representação matemática da informação química de uma molécula, ou seja, de alguma característica da estrutura da molécula, oferecendo uma visualização da molécula com uma perspectiva diferente.[12,37] É através dos descritores moleculares que é possível encontrar a relação entre a estrutura molecular e suas propriedades biológicas.

Os descritores químicos calculados podem ser agrupados nas seguintes categorias de acordo sua natureza:[12,37,64,65]

- Descritores constitucionais: possuem informações como número de ligações, peso molecular etc.
- Descritores topológicos: informações de número de polaridade, índice de Zagreb, etc.
- Descritores quânticos: orbital molecular mais alto ocupado (*highest occupied molecular orbital* - HOMO) e orbital molecular não ocupado mais baixo (*lowest unoccupied molecular orbital* - LUMO)
- Parâmetros estéricos e eletrônicos: constante de Taft, constante de Hammett, por exemplo.[63]
- Descritores geométricos: índice de aromaticidade, excentricidade molecular, etc.

Os descritores moleculares podem ainda ser classificados quanto a sua dimensionalidade. Descritores unidimensionais (1D), os mais simples, descritores bidimensionais (2D), tridimensionais (3D), descritores 4D proposto for Alvez *et al.*[3] e Hopfinger *et al.*[32] que utiliza simulação de dinâmica molecular, descritores 5D propostos por Alvez *et al.*[3] e, Vedani e Dobler[66] que são na verdade uma extensão dos descritores 4D, o qual adiciona liberdade conformacional ao modelo de dinâmica o que permite uma representação múltipla da topologia das moléculas. E posteriormente o mesmo grupo também propôs os descritores 6D que consideram vários modelos de solvatação ao mesmo tempo.[3,67]

Há disponível hoje uma grande diversidade de programas, *softwares* e *web tools* que calculam descritores moleculares. Como por exemplo o Dragon®,[68] CDK® (http://www.rguha.net/code/java/cdkdesc.html), Volsurf®,[69] RDKit (http://www.rdkit.org/), AlvaDesc,[70] entre outros. Uma vez calculado os descritores moleculares, é gerada uma tabela na qual as linhas representam as moléculas e as colunas são os descritores moleculares.

Alguns programas são capazes de calcular milhares de descritores moleculares, como o AlvaDesc que pode calcular mais de quatro mil descritores.[68] E pode nos levar a pensar que quanto mais descritores moleculares melhor será o modelo que será gerado. Mas o efeito pode ser justamente o oposto, ou seja, muitos descritores podem causar super ajuste no modelo, tornando-o ineficaz.

Para evitar esse problema, é preciso realizar a seleção de variáveis/descritores. Selecionamos aqueles descritores mais importantes, ou seja, que dão informações relevantes na relação estrutura atividade. Rotineiramente, são excluídos aqueles descritores que apresentam valores iguais para todas as moléculas, aqueles que o valor muda apenas para uma ou duas moléculas, e aqueles descritores altamente correlacionados ($r > 0.9$) que geram "ruídos" e prejudicam a qualidade do modelo.[12,37,71,72]

Com a tabela já finalizada acrescentamos por fim a coluna com a informação da atividade biológica/ou propriedade que desejamos encontrar a relação com a estrutura química das moléculas. Essa informação é também chamada de variável dependente (Y). A depender de como essa informação for fornecida, um tipo específico de aprendizado de máquina será empregado. Se essa informação for dada por valores numéricos é chamado de dados contínuos e caso seja nominal, é chamado então de dados categóricos.[45,57,73]

APRENDIZADO DE MÁQUINA

O aprendizado de máquina como já mencionado, é a utilização de algoritmos matemáticos de inteligência artificial que podem aprender com os dados de entrada e construir com essas informações modelos computacionais a fim de obter previsões, decisões guiadas, agrupamentos.[34,35]

O ML pode ser classificado quanto a saída desejada em três grandes grupos: classificação, regressão e agrupamentos.[57,74-76] Na classificação as entradas da classe dependente são divididas em duas ou mais classes, e o modelo será treinado a partir dos dados de entrada a classificar as amostras nas classes corretas e assim ficar apto a analisar novas amostras e a partir de suas características (descritores moleculares) atribuir uma classe. A regressão pode ser dita que é um tipo de classificação, enquanto na classificação nós vamos ter a classe dependente nominal ou categórico, na regressão a variável dependente é um valor numérico. A Figura 3 representa o modelo de classificação.[57,74-76]

ML do tipo agrupamentos é uma abordagem na qual não é necessária uma variável dependente predefinida. O objetivo desse tipo de algoritmo é analisar os dados de entrada e organizar os objetos em grupos com base em suas características semelhantes, visando a identificação de estruturas intrínsecas nos dados. Ou seja, o algoritmo irá buscar por semelhanças entre as características/descritores moleculares das moléculas dos dados de entrada e atribuir um grupo a elas. A depender do tipo de agrupamento que for utilizado, uma molécula pode pertencer a mais de um grupo ou mesmo não ser agrupado, sendo considerado um ruído.[57,74-76] A Figura 4 representa esse tipo de ML.

Outro tipo de classificação de ML é com relação a técnica utilizada, podendo ser aprendizado supervisionado, não supervisionado e por reforço. No aprendizado supervisionado é fornecido ao algoritmo informações de entrada e saída desejada, onde o objetivo será encontrar uma regra geral que mapeia as entradas as saídas desejadas, ou seja, a variável dependente esperada. Um exemplo de aprendizado supervisionado são os modelos de classificação.

No aprendizado não supervisionado não são fornecidas as informações de saída desejada, desta forma, o algoritmo precisa

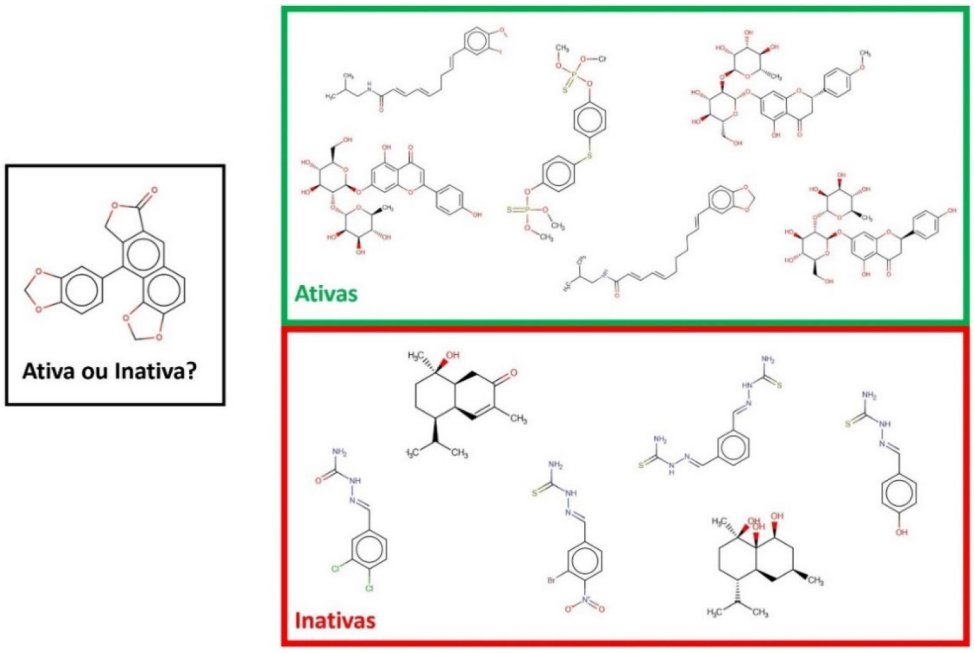

Figura 3. *Representação de um modelo de classificação. No quadrado preto uma molécula será predita entre as classes ativa (retangulo verde) ou inativa (retangulo vermelho) (fonte: elaborado pelo autor)*

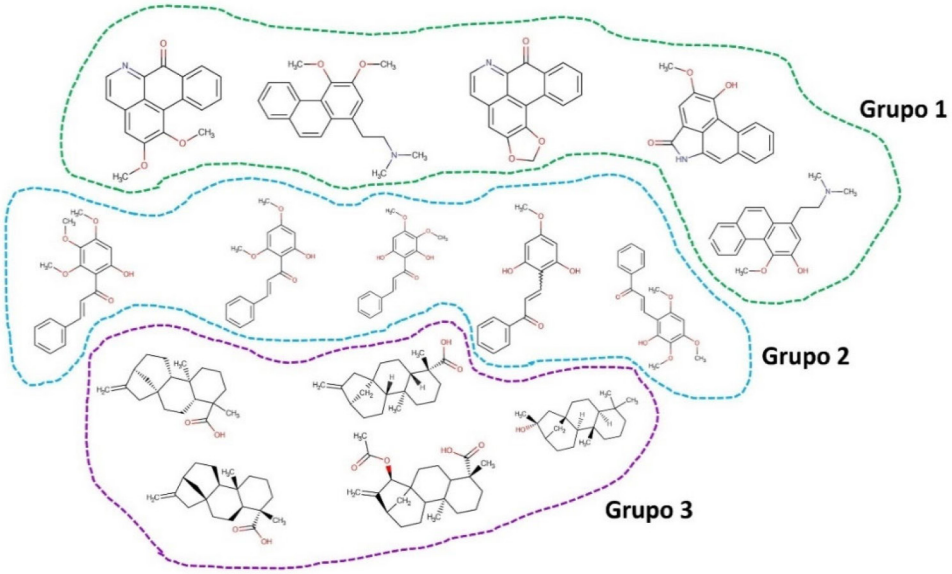

Figura 4. *Representação de um modelo de agrupamento. Moléculas sendo agrupadas em diferentes grupos de acordo com suas similaridade químicas (fonte: elaborado pelo autor)*

encontrar padrões nos arquivos de entrada. Um exemplo de aprendizado não supervisionado são os modelos de agrupamento.

No aprendizado por reforço o objetivo é que o algoritmo aprenda com seus erros a tomar a melhor decisão para se chegar ao objetivo. Ou seja, no apredizado por reforço os modelos são treinados para desenvolver uma sequência de decisões para que se chegue ao objetivo da melhor forma. No aprendizado por reforço, o modelo irá enfrentar um problema, e por meio de tentativa e erro, deverá encontrar a solução, então o algoritmo irá realizar diversas tentativas a fim de alcançar seu objetivo, onde para cada tentativa será atribuído recompensas ou punição dependendo da decisão que foi tomada e assim o algoritmo vai se ajustando até alcançar uma margem satisfatória ou o objetivo proposto. Um exemplo desse aprendizado é um tipo de rede neural artificial.[20,45,74,77] A Figura 5 mostra as classificações do aprendizado de máquina.

Como já foi descrito anteriormente, o ML utiliza algoritmos matemáticos em sua execução. Há uma grande diversidade de algoritmos que são utilizados em ML e alguns foram trazidos para a quimioinformática e consequente para os estudos de QSAR, como árvore de decisão (*decision tree* - DT), florestas aleatórias (*random forest* - RF), máquina de vetor de suporte (*support vector machine* - SVM), k-enésimo vizinho mais próximo (*k-nearest neighbors* - k-NN), *Naive Bayes* (NB), redes neurais artificiais (*artificial neural network* - ANN), que serão brevemente discutidos a seguir.[7,71,74,78]

Árvore de decisão e florestas aleatórias

Esses algoritmos se baseiam na estrutura de uma árvore de decisão hierárquica composta por nós e ramos que se unem para gerar modelos preditivos. São algoritmos de ML supervisionados e

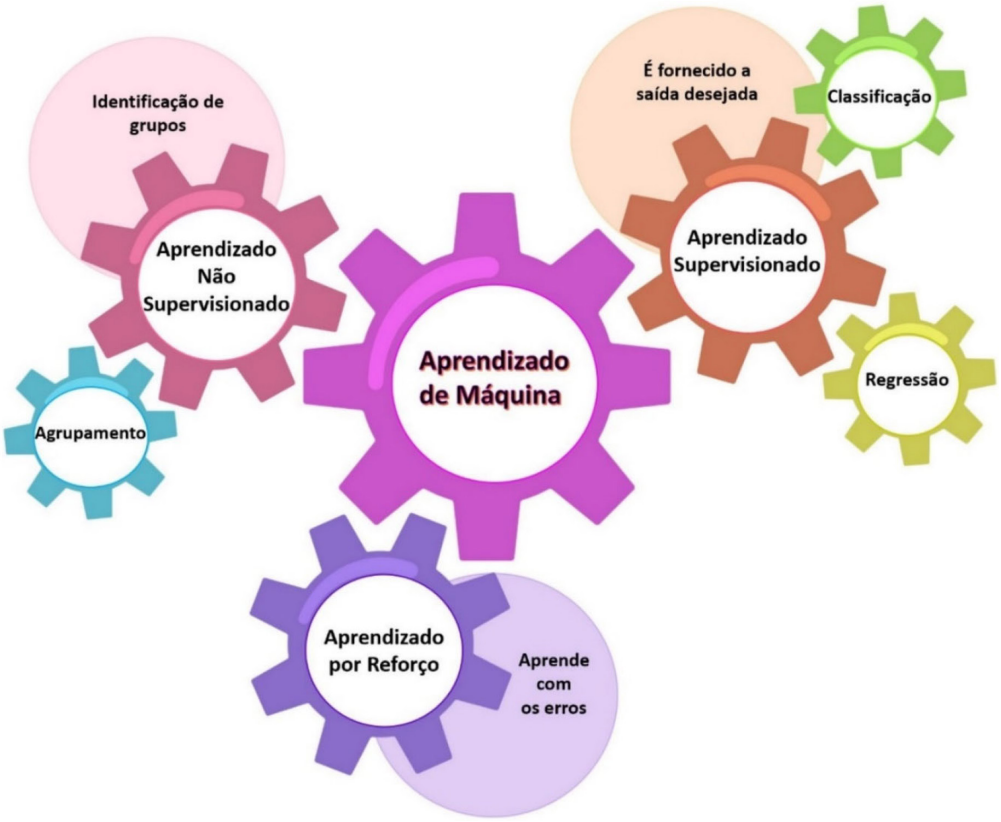

Figura 5. *Classificação do aprendizado de máquina (fonte: elaborado pelo autor)*

podem ser utilizados tanto para modelos de classificação como de regressão.[74,79,80]

A estrutura de uma árvore de decisão é composta por nós, nós raiz, internos e terminais. O nó raiz é normalmente representado no topo da árvore sem galhos chegando até ele, mas com galhos saindo dele. Os nós internos possuem galhos que chegam neles e galhos que saem deles em direção ao próximo nível hierárquico. E a árvore finaliza com os nós terminais que não possuem ramificações que começam a partir deles, uma vez que são o último nível da hierarquia.[79,81]

A cada novo nó que é iniciado e terminado na árvore são tomadas de decisões. Na árvore o resultado de cada caminho é ponderado pela probabilidade associada, ou seja, a probabilidade deste caminho ocorrer, e o resultado será somado e o valor de cada curso é então determinado. Daí, o curso que fornece o maior valor esperado será o curso preferido. De maneira simplificada a estrutura da DT permite a representação e avaliação de problemas envolvendo decisões sequenciais, destacando os riscos e resultados identificados em cada decisão e rumo tomado.[74,80,82]

O algoritmo RF é na verdade um conjunto de DTs que é gerado e, ao final, será realizado uma análise de consenso para o caso de maior probabilidade, que é considerado o melhor ajuste.[74,83,84] Uma das vantagens mais significativas do RF é que produz melhores resultados devido à ausência de incremento no viés, pois reduz a variância no modelo.[79] A Figura 6 representa a estrutura do RF.

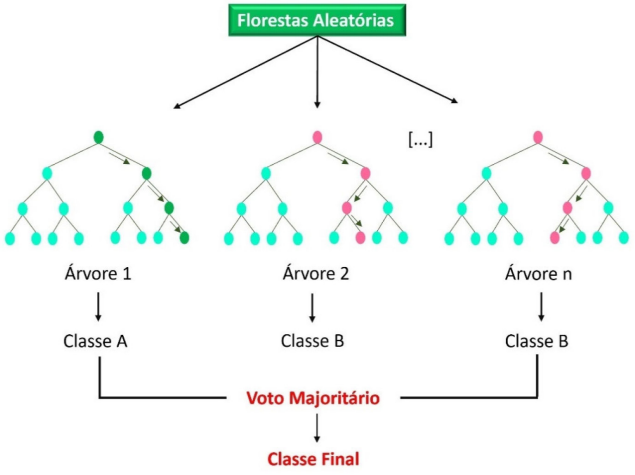

Figura 6. *Representação do algoritmo florestas aleatórias. O modelo cria uma estrutura de decisão em forma de árvore e a cada nova instância ela é percorrida até chegar a um nó terminal onde está a classe (fonte: elaborado pelo autor)*

Máquina de vetor de suporte

As SVMs (máquinas de vetores de suporte, do inglês *support vector machines*) são algoritmos de ML supervisionado versáteis, adequados tanto para tarefas de classificação quanto para regressão.[74,79,81] Sua popularidade é atribuída ao seu desempenho robusto e habilidade de generalização, especialmente em domínios de alta dimensionalidade.[81,85,86] Os SVMs são conhecidos por serem baseados em *kernel*, o que significa que sua eficácia depende da medida de similaridade entre objetos no espaço de características, permitindo que eles capturem relações complexas não lineares nos dados de entrada.[81,85-87] Essa flexibilidade torna os SVMs uma escolha valiosa em uma ampla gama de aplicações, desde reconhecimento de padrões até análise de dados em domínios complexos. [80-82]

Um modelo SVM representa as amostras do banco de dados como em pontos no espaço, de modo que os pontos de diferentes categorias sejam separados por um hiperplano que seja tão amplo quanto possível.[81,85-87] As novas amostras serão mapeadas no mesmo

espaço e serão previstos para qual lado do hiperplano pertencem.[74] As amostras utilizadas para definir as margens do hiperplano são aqueles mais próximas ao hiperplano, mas que maximiza a separação das classes, e são chamados de vetores de suporte. [74,79,81]

Uma observação importante é que o SVM pode ainda realizar o ML não supervisionado, ou seja, uma vez que os dados não estejam rotulados (não possuam informação da saída desejada), o SVM por meio do algoritmo de vetorização de suporte, criado por Hava Siegelman e Vladimir Vapnik, que aplicará estatísticas de vetor suporte, desenvolvidas no SVM, e categorizará os dados não rotulados, criando assim grupos.[74,80,88] A Figura 7 representa o SVM.

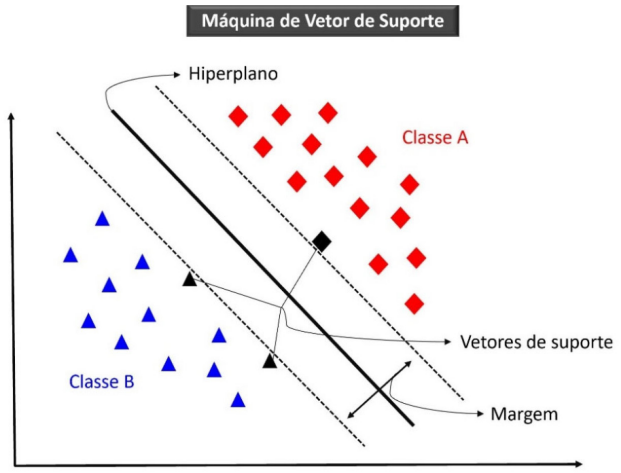

Figura 7. *Representação do algoritmo máquina de vetor de suporte (SVM). O algoritmo separa as classes por meio de um hiperplano ao identificar as amostras mais distantes entre as classes, os vetores de suporte. Os vetores de suporte maximizam a margem de sepraração entre as classes (fonte: elaborado pelo autor)*

K-Vizinho mais próximo

Diferentemente dos outros algoritmos apresentados, o k-NN não gera um modelo. Ele busca em tempo de execução qual nova amostra (instância) se parece mais com os dados históricos e todo o cálculo é adiado até a classificação. Devido a essa característica, é um algoritmo de alto custo computacional uma vez que cada nova instância a ser classificada terá que ser comparada com todos os dados do conjunto de treino e avaliar aqueles com menor distância.[74,89,90]

O k-NN é chamado de algoritmo baseado em instância onde sua mecânica é apenas classificar uma nova instância atribuindo a classe que é mais frequente entre os k vizinhos (amostras) mais próximos do conjunto de treinamento a esse elemento, onde essa distância é determinada através de alguma medida de similaridade (pode ser distancias euclidianas, de manhatham, ponderada etc.). O k é um valor inteiro e será determinado pelo usuário. É um método não paramétrico que pode ser utilizado para classificação, regressão e agrupamentos.[7,74,90] A Figura 8 representa o funcionamento do k-NN.

Naive Bayes

O algoritmo NB é na verdade uma família de algoritmos classificadores de ML supervisionado. Sua grande vantagem está em sua simplicidade e em sua eficiência, com capacidade de gerar modelos de classificação de boa precisão a partir de um pequeno conjunto de dados, às vezes seu desempenho supera classificadores mais complexos.[57,74,80]

O NV se baseia no Teorema de Bayes e se supõe que os atributos irão influenciar a variável dependente de maneira independente.

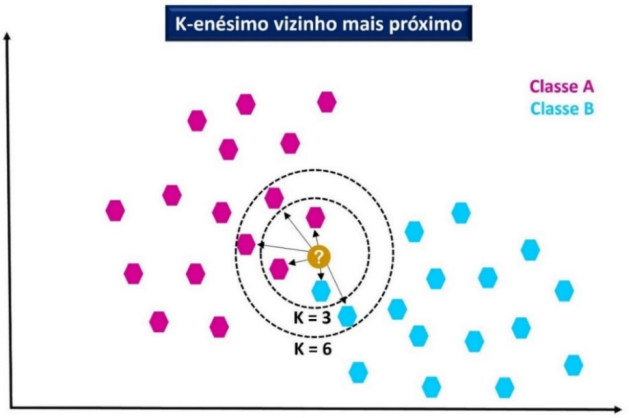

Figura 8. *Representação do algoritmo k-enésimo vizinho mais próximo (k-NN). A instância "?"será classificada de acordo com sua semelhança entre os vizinhos próximos defino pelo K. Quando K = 3 ou 6, a instância será classificada como classe A por apresentar mais semelhança e proximidade com instâncias da classe A (fonte: elaborado pelo autor)*

O teorema de Bayes descreve a probabilidade de um evento, por exemplo a probabilidade de uma molécula ter potencial ativo, com base em um conhecimento prévio das condições que podem estar relacionadas a esse evento, ou seja, com base naqueles atributos (descritores moleculares) que melhor descrevem as moléculas que tem atividade.[74,80,81]

Explicando de maneira simples e objetiva esse algoritmo irá construir uma tabela mostrando a importância de cada atributo para a variável dependente, assim, ao submeter uma nova instância (amostra) o NB irá analisar os pesos de cada atributo comparar com os atributos da nova instância somá-los e ver qual classe teve maior peso, sendo então, classificado com esta classe "vitoriosa".[57,80,81] A Figura 9 traz a representação do teorema de Bayes e do funcionamento do algoritmo NB.

Rede neural artificial

As ANNs são algoritmos que geram modelos computacionais inspirados nos neurônios do sistema nervoso central humano. Para compreender melhor uma rede neural, precisamos começar entendendo o que é um neurônio artificial e seu funcionamento. Um neurônio artificial, assim como neurônios do cérebro, pode se comunicar entre si e as conexões dos neurônios artificiais são representadas por pesos que não passam de um valor específico e esse valor tenta expressar a força sináptica dessa conexão entre neurônios.[22,74,79,81]

Existem diferentes tipos de arquitetura de ANN, elas podem ser supervisionadas, não supervisionadas e por reforço, porém todas possuem em comum alguns elementos em sua estrutura. A estrutura básica de uma ANN possui um conjunto de nós, primeiro são os nós de entrada, são aqueles que obtém informações, esses nós são chamados de camada de entrada. A ANN também precisa de nós de saída que são aqueles que transmitem o resultado da ANN, e por fim, tem os nós da camada oculta, são aqueles que transmitem informações entre os neurônios artificiais.[22,33,74,81] Vale ressaltar que o importante da estrutura de uma ANN não é apenas a topologia das interconexões entre os neurônios e das funções de ativação, mas uma parte fundamental é a relevância de cada uma das conexões.

O funcionamento básico de uma ANN se dá da seguinte maneira: quando a camada de entrada é determinada, os pesos serão atribuídos. Esses pesos são valores numéricos que auxiliam a determinar a importância de cada variável (em QSAR, de cada descritor molecular) fornecida, onde os maiores pesos serão para aquelas variáveis que

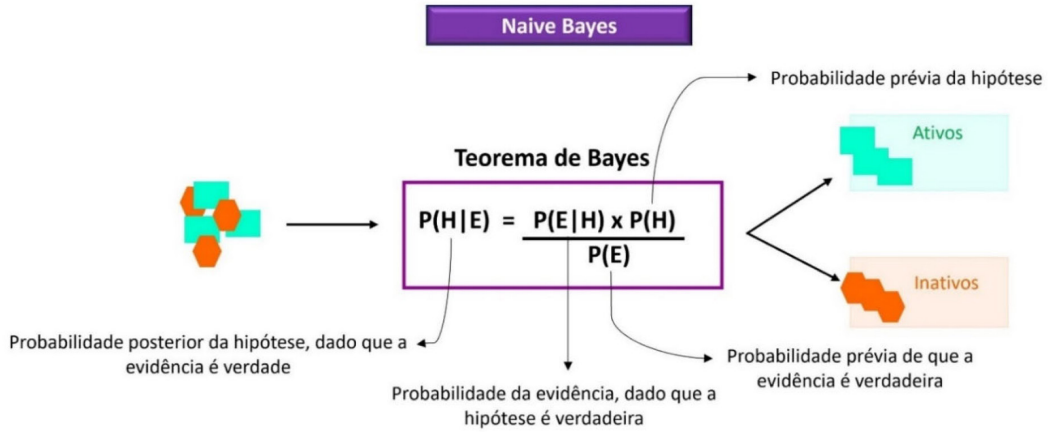

Figura 9. *Representação do algoritmo Naive Bayes, que é baseado na teoria das probabilidades (fonte: elaborado pelo autor)*

contribuem de forma mais significativa para a classe Y, quando comparado as demais variáveis. Assim, todas as entradas serão multiplicadas por seus respectivos pesos, e em seguida, somadas. A próxima etapa é a transmissão da saída através de uma função de ativação, que determina a saída (classe Y). Se a saída exceder um determinado valor, o nó será ativado, que irá transmitir as informações para a camada seguinte da ANN. Dessa forma, a saída de um nó torna-se a entrada do próximo nó. Caso a saída não exceda o valor determinado, não haverá transmissão de informação. Este processo de transmissão de informação de uma camada para a seguinte define o funcionamento de uma ANN.[22,33,74,80,81] A Figura 10 representa a estrutura básica de uma ANN.

PRINCÍPIOS ÉTICOS DA INTELIGÊNCIA ARTIFICIAL

Antes de falarmos sobre construção de modelos e avaliação de seus desempenhos, precisamos conhecer e fazer uso de princípios éticos para assegurar uma aplicação ética e benéfica da inteligência artificial (IA). Nesse sentido, é crucial considerar e seguir questões de transparência, explicabilidade e interpretabilidade nos modelos de aprendizado de máquina em QSAR.[91,92]

A OECD (Organização para a Cooperação e Desenvolvimento Econômico) é uma organização internacional composta por 38 países membros, com o objetivo primordial de promover políticas para melhorar o bem-estar econômico e social globalmente. No campo da inteligência artificial (IA), a OECD está engajada na formulação de diretrizes, políticas e princípios éticos relacionados à IA concentrando-se na promoção da cooperação internacional para harmonizar políticas práticas entre países, visando benefícios mais amplos e uma abordagem mais unificada em relação à IA em escala global [93].

Apesar da OECD não ter emitido diretrizes específicas para modelos de QSAR, muitos dos princípios gerais relacionados à transparência, explicabilidade e responsabilidade em IA e sistemas de aprendizado de máquina são aplicáveis a esses modelos. Instituições regulatórias e científicas geralmente buscam assegurar que os modelos sejam confiáveis, transparentes e interpretáveis de forma significativa para sua utilização adequada e segura.[91,92]

Existem oito princípios éticos fundamentais para modelos de IA:[91-94]

1. Transparência: os modelos de IA devem ser transparentes, explicáveis e compreensíveis para os usuários, permitindo entender como tomam decisões.
2. Justiça e Equidade: devem ser desenvolvidos e utilizados de maneira a garantir a equidade, minimizando vieses indevidos e promovendo igualdade de oportunidades.
3. Privacidade e Segurança: a privacidade dos dados deve ser protegida, garantindo a segurança e confidencialidade das informações pessoais usadas pelos sistemas de IA.

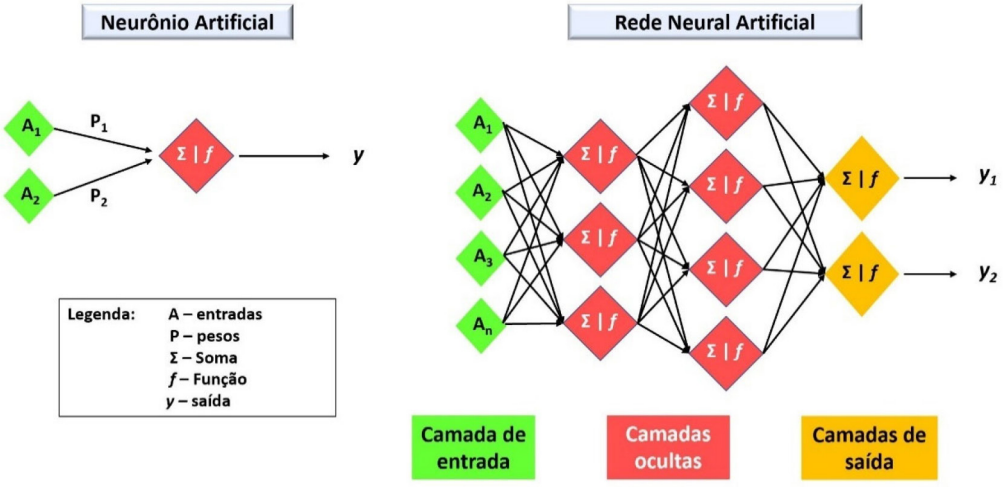

Figura 10. *Representação de um neurônio artificial e de uma rede neural artificial. No neurônio artificial, cada entrada (A_1 e A_2) recebe apenas um tipo de sinal ou informação e possui um peso (P_1 e P_2) que irá determinar a influência do sinal no resultado do processamento. Na rede neural há uma grande quantidade de neurônios, assim, temos uma camada de entrada, camadas ocultas onde ocorre a determinação do peso de cada entrada, e aplicação soma e função, e por fim as camadas de saída onde será definido o resultado final do processamento (fonte: elaborado pelo autor)*

4. Responsabilidade e Prestação de Contas: desenvolvedores, fabricantes e usuários devem assumir a responsabilidade por decisões tomadas por esses sistemas, incluindo consequências negativas.
5. Beneficência e Não Maleficência: a IA deve beneficiar a humanidade, evitando causar danos e garantindo resultados positivos.
6. Transcendência Cultural e Diversidade: deve ser sensível a diferenças culturais, respeitando a diversidade e considerando diferentes perspectivas e valores éticos.
7. Conformidade Legal e Ética: deve estar em conformidade com leis, regulamentos e princípios éticos estabelecidos.
8. Explicabilidade e Interpretabilidade: os sistemas de IA devem ser capazes de explicar suas decisões e funcionamento de forma compreensível, permitindo a interpretação de suas ações.

Na área da descoberta de novas drogas (fármacos e agroquímicos), especialmente no QSAR, os princípios de transparência, explicabilidade e interpretabilidade são cruciais para seguir e obedecer. Garantir a explicabilidade de um modelo de aprendizado de máquina pode ser desafiador, principalmente em modelos complexos como redes neurais profundas. Entretanto, algumas abordagens podem aumentar essa explicabilidade.

Por exemplo, utilizar modelos mais simples e interpretáveis sempre que possível, como modelos lineares ou árvores de decisão, pode facilitar a compreensão e explicação do modelo. Além disso, técnicas específicas para interpretar o modelo, como análise de importância de características, SHAP (*shapley additive explanations*) e LIME (*local interpretable model-agnostic explanations*), podem esclarecer como cada característica influência nas decisões do modelo.[91-94]

A representação visual do comportamento do modelo em relação aos dados, como gráficos de dispersão, mapas de calor, entre outros, também ajuda a explicar o que o modelo está fazendo. Reduzir a complexidade do modelo por meio de técnicas como normalização dos dados, redução de dimensionalidade ou simplificação de arquiteturas complexas é outra estratégia para tornar o modelo mais explicável sem comprometer muito seu desempenho.[91-94]

Manter documentação detalhada sobre o processo de construção do modelo, incluindo informações sobre os dados utilizados, pré-processamento, arquitetura do modelo, hiperparâmetros e métricas de avaliação, é crucial. Validar o modelo é um passo extremamente importante para garantir que as previsões ou decisões sejam confiáveis dentro do contexto do problema. Comunicar-se de maneira transparente sobre o modelo, seus objetivos, limitações e possíveis áreas de viés é essencial para sua interpretação e aplicação adequadas.[91-94]

CONSTRUINDO MODELOS E AVALIANDO SEU DESEMPENHO

A depender do modelo que desejamos construir, seja supervisionado ou não supervisionado, há um caminho básico que devemos seguir. Mas independentemente do tipo de modelo sempre será preciso validá-lo e avaliar o seu desempenho.

Como explicado anteriormente, em um aprendizado de máquina, um algoritmo irá aprender a partir de um conjunto de dados. Nesse aprendizado é construído o modelo, que é simplesmente uma fórmula criada pelo algoritmo para prever, classificar ou agrupar os novos dados que sejam apresentados para o algoritmo, ou seja, dados que o algoritmo não conheça.[57]

A Figura 11 representa o esquema geral da construção de um modelo. Cinco etapas são cruciais para a construção de qualquer modelo utilizando aprendizado de máquina. Inicialmente o banco de dados que será utilizado para construção de modelo, previamente tratado, será particionado de maneira estratificada em dois subconjuntos, como por exemplo, 80% do conjunto na série de treino e 20% na série de teste. Nesta partição os conjuntos são selecionados de maneira aleatória, mas mantêm a mesma proporção de classes, por exemplo, entre moléculas ativas e inativas frente a um alvo biológico, entre os conjuntos de treino e teste para um modelo de classificação. Há um consenso em particionar o banco de dados em 80% para treino e 20% para teste, mas há possibilidade de variar um pouco essa proporção para 70% treino e 30% teste.[57] Na segunda etapa, o conjunto de treino será utilizado para gerar o modelo, ou seja, o conjunto de treino será apresentado ao algoritmo para que ele possa aprender e então gerar o modelo. Na terceira e quarta etapa, o conjunto de teste será utilizado para que possamos compreender o desempenho do modelo, ou seja, esse conjunto terá sua classe ocultada e então os dados descritivos serão apresentados ao modelo, em seguida será realizada a predição das amostras do conjunto de

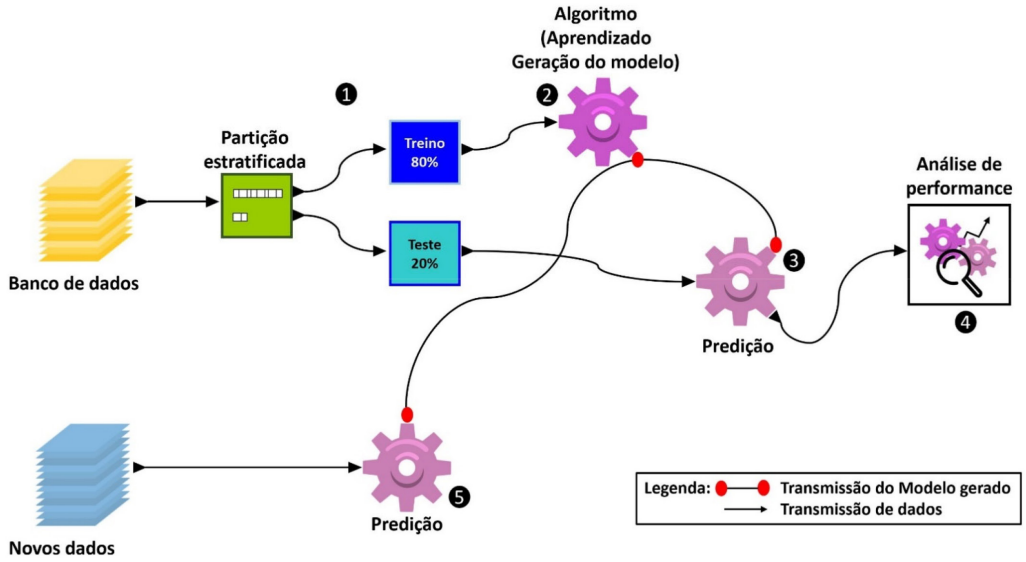

Figura 11. *Representação global de um modelo computacional utilizando aprendizado de máquina. (1) Partição estratificada do conjunto de dados em 80% treino e 20% teste. (2) O conjunto de treino é apresentado ao algoritmo para aprendizado e geração do modelo computacional. (3) O conjunto de teste é utilizado no modelo para obter sua predição. (4) A partir da predição do conjunto de teste é realizada a análise de performance do modelo criado. (5) Predição de novos dados utilizando o modelo criado (fonte: elaborado pelo autor)*

teste. Após a predição, será então realizada a análise de performance do modelo no qual iremos avaliar a taxa de acerto do modelo. Uma vez que temos o conhecimento da classe de cada amostra do conjunto de teste, é realizada a comparação com o que foi predito e assim temos conhecimento da taxa de acerto do modelo e sua performance. Um modelo ideal irá acertar 100% em sua predição, caso ele acerte 50% isso nos informa que sua predição foi um mero palpite, o algoritmo não conseguiu aprender com os dados do conjunto de treino. A última etapa do processo da criação de modelos computacionais utilizando aprendizado de máquina será a predição de um novo conjunto de dados, que é o objetivo final da criação do modelo, realizar a predição de um conjunto para o qual não há conhecimento de sua classe.

Análise de desempenho para modelos de classificação

Em um modelo de classificação existem quatro possibilidades de avaliação de desempenho básicas. Considerando um banco de dados no qual possui duas classes, positivos (moléculas ativas) e negativos (moléculas inativas), vamos ter as seguintes possibilidades: verdadeiros positivos (VP), aqueles que são ativos e foram acertadamente classificados como ativos; falso negativo (FN), aquelas amostras que são verdadeiras positivas, mas foram classificadas como inativas; verdadeiros negativos (VN), amostras inativas sendo classificadas como inativas; e por fim os falsos positivos (FP), as amostras que são inativas mas que foram classificadas como ativas. Assim, os verdadeiros ativos e verdadeiros inativos são os acertos do modelo, ou seja, a taxa de acerto, enquanto os falsos ativos e falsos inativos são os erros do modelo. Esses valores quando colocados em uma tabela, temos então, a matriz de confusão.[57,95,96] A Figura 12 demonstra uma matriz de confusão.

		Informação Real	
		Positivo	Negativo
Predição	Positivo	Verdadeiro Positivo (VP)	Falso Positivo (FP)
	Negativo	Falso Negativo (FN)	Verdadeiro Negativo (VN)

Figura 12. *Matriz de confusão (fonte: elaborado pelo autor)*

A partir da matriz de confusão várias métricas podem ser calculadas para avaliar o modelo fornecendo informações fundamentais para total compreensão de seu desempenho. Essas métricas são: precisão (acurácia), sensibilidade, especificidade, verdadeiro preditivo positivo, verdadeiro preditivo negativo, coeficiente de correlação de Matthews e a curva ROC (características operacionais do receptor, do inglês *receiver operating characteristics*). Cada uma será explicada detalhadamente a seguir.

Precisão (acurácia - ACC): se refere a taxa de acerto global do modelo. O quanto que o modelo está classificando corretamente tanto os positivos como os negativos. É calculada da seguinte maneira:

$$\text{ACC} = \frac{(\text{VP} + \text{VN})}{\text{P} + \text{N}} \quad (1)$$

Em que P é o total de amostras positivas, e N o total de amostras negativas.

Sensibilidade: avalia a taxa de acerto do modelo apenas das amostras positivas, ou seja, a capacidade de aprendizado do modelo em classificar uma amostra verdadeiramente positiva como positiva.

$$\text{Sensibilidade} = \frac{\text{VP}}{(\text{VP} + \text{FN})} \quad (2)$$

Especificidade: temos informação da capacidade de aprendizado do modelo em classificar amostras verdadeiramente negativas como negativas, ou seja, a taxa de acerto das amostras negativas.

$$\text{Especificidade} = \frac{\text{VN}}{(\text{VN} + \text{FP})} \quad (3)$$

Verdadeiro preditivo positivo (VPP): diz respeito a taxa de acerto de amostras verdadeiros positivos em relação a toda predição de positivos.

$$\text{VPP} = \frac{\text{VP}}{(\text{VP} + \text{FN})} \quad (4)$$

Verdadeiro preditivo negativo (VPN): informa a taxa de acerto de amostras verdadeiras negativas em relação a toda predição de negativos.

$$\text{VPN} = \frac{\text{VN}}{(\text{VN} + \text{FP})} \quad (5)$$

Coeficiente de correlação de Matthews (MCC): esse coeficiente nos ajuda a ter informação de performance global do modelo. Seu valor varia entre –1 e 1, onde um coeficiente de valor 1 indica um modelo de predição perfeita, o 0 indica um modelo de predição aleatória, e –1 indica um modelo em total desacordo entre o que previsto e o que é verdadeiro.[97] Dessa forma, buscamos modelos que possuam um MCC o mais próximo de 1 possível, é calculado da seguinte forma:

$$\text{MCC} = \frac{(\text{VP} \times \text{VN}) - (\text{FP} \times \text{FN})}{\sqrt{(\text{VP} + \text{FP})\,(\text{VP} + \text{FN})\,(\text{VN} + \text{FP})\,(\text{VN} + \text{FN})}} \quad (6)$$

Curva ROC: a curva ROC nos mostra de maneira bidimensional o desempenho real do modelo com mais clareza que a precisão, conciliando a especificidade e sensibilidade. A curva ROC é expressa através do cálculo da área abaixo da curva (AUC), que é uma porção de um quadrado unitário, assim seu valor pode variar de 0 a 1, mas na prática costuma variar ente 0.5 e 1, onde 1 é um desempenho perfeito e 0,5 um modelo completamente aleatório. Em geral, não há modelos classificadores piores que os aleatórios, por esta razão, não se considera modelos com valores de curva ROC abaixo de 0,5. O gráfico da curva ROC é plotado pelo valor da sensibilidade (eixo Y), o valor dos verdadeiros positivos, em função do valor de falsos positivos, ou seja, 1 menos especificidade.[96] A Figura 13 representa a plotagem da curva ROC.

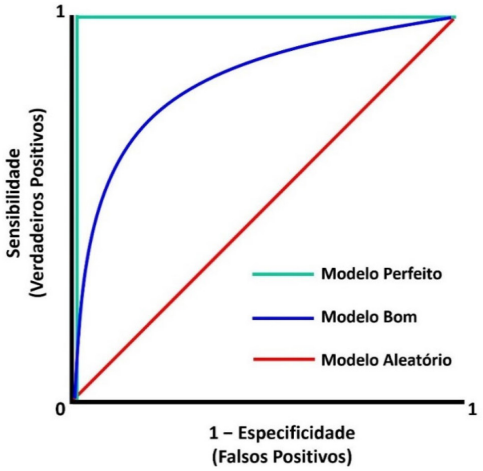

Figura 13. *Representação da curva ROC (fonte: elaborado pelo autor)*

Quanto menos falsos positivos há no modelo, maior o acerto em verdadeiros negativos, e quanto maior a sensibilidade, menor é o erro em falsos negativos. Desta forma, o valor da AUC será maior quanto mais próximo de 1 no eixo Y e mais próximo de 0 no eixo X.

Há alguns problemas que podem acontecer em modelos de classificação e que afetam diretamente as suas métricas de avaliação de desempenho e que devemos sempre estar atentos, são eles: super ajuste e classe rara.

Na construção de modelos buscamos por modelos genéricos, modelos que tenham um bom desempenho e que sejam confiáveis. O contrário desses modelos, são os modelos super ajustados (do inglês *overfitting*). Os modelos super ajustados apresentam ótimo desempenho, alta precisão, funcionam muito bem para o conjunto de treino, mas não generalizam bem para novos dados. É como se o modelo decorasse o conjunto de treino, mas não aprendeu de fato a diferenciar as classes e por isso tem um pobre desempenho para novos dados. A identificação de modelos super ajustados acontece principalmente nas etapas de análise de desempenho e validação do modelo, etapas cruciais que serão detalhadas mais adiante.

São vários os fatores que podem causar um super ajuste, mas o principal é quando os dados de treino são poucos, aplicando para QSAR, possuem pouca variedade de moléculas e pouca quantidade, o que leva o algoritmo a criar um modelo que não representa eficientemente o conjunto de novas moléculas. Classe rara também pode causar modelos super ajustados.

Classe rara acontece quando a quantidade de uma classe é consideravelmente menor que outra classe. Por exemplo, iremos trabalhar com um banco de dados para construção de um modelo preditivo que possui 1000 moléculas. Dessas 1000 moléculas, 900 são da classe de moléculas ativas e 100 são moléculas inativas, veja que a classe de moléculas inativas representa apenas 10% de todo o conjunto de dados. A consequência disso é que o modelo aprenderá muito bem a classificar moléculas ativas, mas terá sérios problemas de aprendizado na classificação de moléculas inativas.

A solução desse problema é equilibrar as classes no conjunto de dados, e é de extrema importância que as classes estejam equilibradas no conjunto de treino. Por esse motivo, quando o conjunto de dados é particionado no conjunto de treino e teste, essa partição aconteça de maneira estratificada, ou seja, mantendo a proporção de moléculas ativas e inativas no conjunto de treino e no conjunto de teste. Desta forma, evita-se que na partição aconteça o problema de classe rara devido a uma partição inadequada, sem equilíbrio entre as classes, pois mesmo que o banco de dados tenha equilíbrio entre as classes, o erro na partição pode gerar um problema de classe rara.

Análise de desempenho para modelos de regressão linear

Como já mencionado, a regressão linear é bastante semelhante a classificação, a diferença está na classe, ao invés de um valor nominal ou categórico, é um valor numérico. É preciso entender alguns conceitos para compreender os resultados e a análise de desempenho de um modelo de regressão.

Em um modelo de regressão linear a variável que desejamos prever não é chamada de classe como nos modelos de classificação, é conhecida por variável dependente e os descritores, as variáveis que utilizaremos para prever, são chamadas de variáveis independentes. Dessa forma, na regressão, irá se utilizar de técnicas estatísticas para modelar correlações e predizer o valor de uma variável dependente (Y) a partir das variáveis independentes (X).[57,95,98] Podemos ter dois tipos de regressão linear, a simples onde temos uma variável independente para prever uma variável dependente, e a regressão linear múltipla, onde utiliza-se duas ou mais variáveis independentes para prever uma variável dependente.

Então, o objetivo do modelo de regressão linear é avaliar o grau de associação entre duas variáveis, X e Y, ou seja, mede a "força" de correlação linear entre as variáveis X e Y. Para investigar essa correlação e analisar a qualidade do modelo, pode-se representar os valores das variáveis num gráfico de dispersão. Desta forma, haverá presença de relação linear entre as variáveis dependente e independente se os valores se aproximarem de uma linha reta, e quanto mais os valores estiverem da reta, mais forte será a correlação. Assim, apenas observando o gráfico de dispersão já teremos uma ideia se o modelo está bom ou ruim.[57,95,98]

A correlação entre as variáveis X e Y pode ser positiva ou negativa, e influencia diretamente na direção da reta. Uma correlação positiva significa que enquanto uma variável cresce, a outra que está correlacionada também cresce. Já na correlação negativa acontece o inverso, enquanto uma variável cresce a outra diminui.[57,95,98] Influenciando, portanto, no direcionamento da reta, como pode ser observado na Figura 14.

Para medir e quantificar a explicação da reta no gráfico de dispersão utiliza-se o cálculo do coeficiente de determinação (R^2). O R^2 também pode ser entendido como a porcentagem da variância que pode ser prevista pelo modelo de regressão e consequentemente, informar o quão próximo a predição gerada está dos valores reais. Desta forma, em uma regressão linear de correlação positiva o modelo perfeito possui R^2 igual a 1, então neste tipo de correlação quanto mais próximo de 1, melhor é o modelo. Já em uma regressão linear de correlação negativa o modelo perfeito possui R^2 igual a –1, então

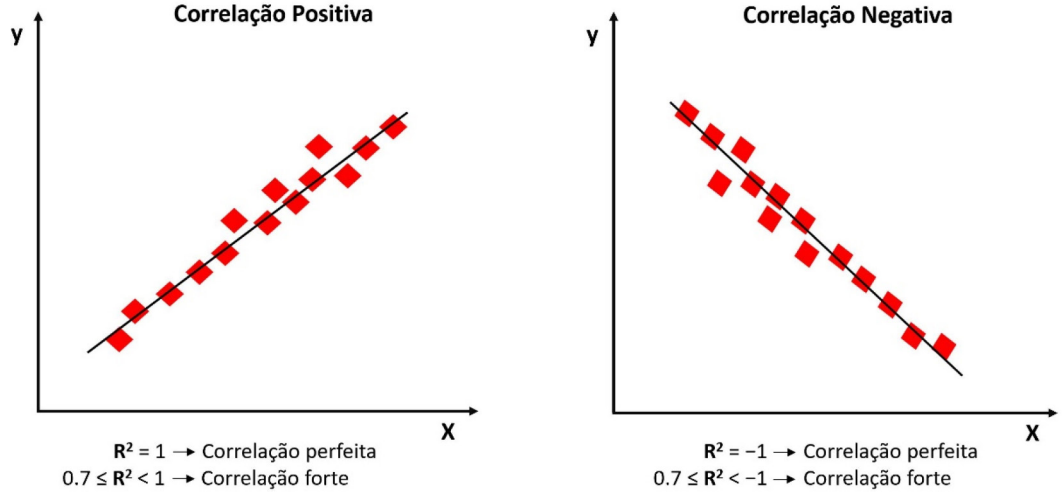

Figura 14. *Representação da correlação através do diagrama de dispersão (fonte: elaborado pelo autor)*

quanto mais próximo de –1 melhor será o modelo. Um modelo de R^2 igual 0 nos informa não há correlação entre as variáveis, então tanto na correlação positiva quanto na negativa, quanto mais o R^2 se aproximar de 0, mais fraca é a correlação entre as variáveis e pior é o modelo.[57,95,98,99] Como demonstrado na Figura 14.

Mas a análise de desempenho de uma regressão linear não se limita apenas ao cálculo de R^2, pois há limitações que podem acabar gerando mal interpretação do modelo. Por exemplo, caso o modelo esteja super ajustado, o valor de R^2 continuará alto. Por isso, outras métricas precisam ser avaliadas para ter total compreensão e uma boa interpretação da qualidade do modelo.[57,95,98] Essas métricas são:

- Erro quadrático médio (MSE): consiste em calcular a média de erro das previsões ao quadrado, ou seja, calcula-se a diferença entre o valor predito e o valor real e eleva-se o resultado ao quadrado. Isso é feito para todas as amostras, depois é só somar e dividir pelo número total de amostras. Quanto maior for o MSE, pior será o modelo.
- Raiz do erro quadrático médio (RMSE): como o próprio nome já diz, essa métrica é a raiz do MSE. Tanto o MSE quanto o RMSE penalizam a presença de outliers nos dados.
- Erro absoluto médio (MAE): essa métrica é a média do cálculo das distâncias entre os valores preditos de todas as amostras com os valores reais. O MAE diferente do MSE e RMSE é sensível a presença de *outliers*, ou seja, ele não detecta sua presença. Quanto maior o MAE, pior é o modelo.
- Erro percentual absoluto médio (MAPE): essa métrica gera uma porcentagem que é obtida por meio da divisão da diferença entre os valores preditos e real pelo valor real, ou seja, o MAE dividido pela soma de todos os valores reais. Quanto maior essa porcentagem, pior é o modelo.

Análise de desempenho para modelos de agrupamento

A avaliação de desempenho de modelos de aprendizado não supervisionado é muito importante, mas devido a sua natureza, a avaliação ou validação como também é conhecida, não é muito bem desenvolvida e não é tão trivial como avaliar modelos de aprendizado supervisionado.[100,101] Aqui farei uma breve explicação visto que os modelos de agrupamento não são muito utilizados para QSAR.

Apesar de ser difícil a análise de desempenho de modelos de agrupamento, tem sido proposto na literatura uma vasta quantidade de métricas para quantificar a qualidade dos resultados dos modelos de agrupamento. Essas métricas se dividem em dois grupos: métricas de validação externa, quando há informações externas para avaliar a qualidade dos resultados de agrupamento, semelhante a análise de modelos de classificação, por exemplo; e métricas de validação interna, quando não há informação externa para auxiliar na análise de desempenho.[95,100,101]

As métricas de validação interna se baseiam na medição da coesão e separação dos *clusters* (grupos), na análise estatística da matriz de proximidade ou no estudo do dendrograma gerado por algoritmos de agrupamento hierárquico. A depender do tipo de algoritmo utilizado, algoritmo de agrupamento hierárquico e algoritmo particional, há métricas mais específicas para analisar o desempenho. Para algoritmos particionais métricas baseadas em coesão e separação, e na matriz de proximidade, como por exemplo o coeficiente de silhueta e o índex de Dunn, são os mais utilizados. Já para algoritmos hierárquicos o coeficiente cofenético é o mais usual.[95,100,101] A Figura 15 traz informações dos métodos de avaliação dos modelos não supervisionados e são listados alguns dos coeficientes utilizados.

As métricas de validação externa avançam incorporando as informações externas no processo de validação dos grupos, ou seja, rotula as amostras de treino com classes. Há uma grande quantidade de métricas de validação externa na literatura, que podem ser divididas em conjuntos de correspondência, correlação ponto a ponto e índices teóricos de informações (Figura 15).[95,100,101]

VALIDAÇÃO DE MODELOS DE APRENDIZADO DE MÁQUINA

Ao particionarmos o banco de dados em treino e teste, o conjunto de teste é utilizado para visualizarmos a qualidade do modelo, sua capacidade preditiva, avaliarmos a sua performance. Esta técnica é conhecida por validação Holdout, mas o problema dessa técnica é que não é suficiente para ser usado como validação do modelo, uma vez que sua amostragem pode não representar todo o conjunto de treino. Assim, é preciso aplicar outras técnicas de validação para compreendermos a real performance do modelo criado.

A validação de um modelo é de extrema importância para avaliar a capacidade preditiva de modelo, bem como a sua confiabilidade

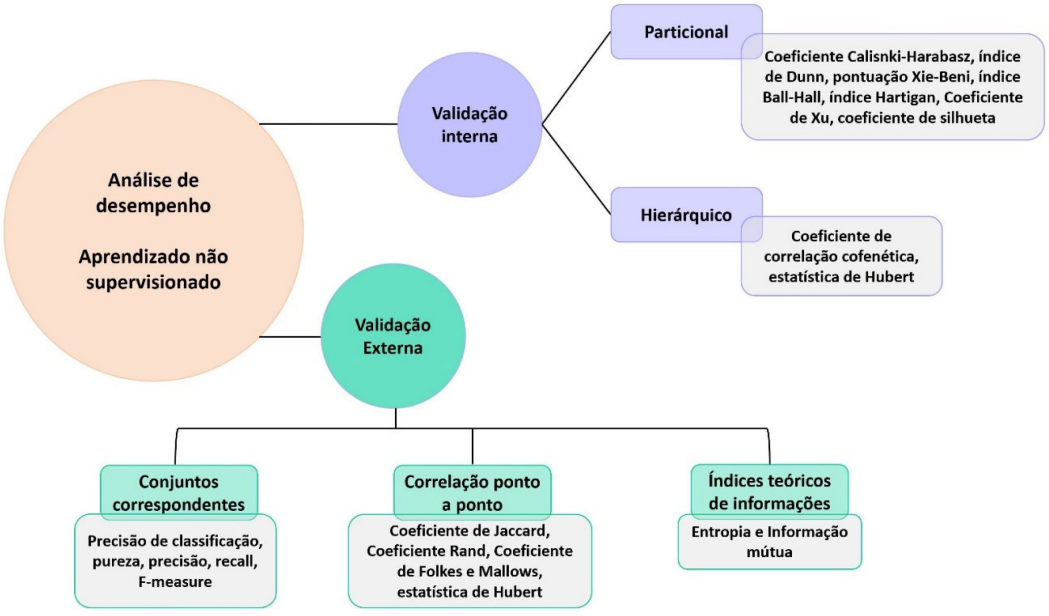

Figura 15. *Métodos de análise de desempenho de aprendizado de máquina não supervisionado (fonte: elaborado pelo autor)*

e reprodutibilidade, podendo ser avaliada por um coeficiente de correlação de validação cruzada, o Q^2.[22,102] A validação cruzada (CV, do inglês *cross-validation*) utilizando as técnicas de *k* partes (do inglês *k-fold*), *leave-one-out* e estratificada, e validação *bootstrap*, são os métodos mais utilizados em validação interna.[22,102] Cada um desses métodos será brevemente explicado a seguir, como também a validação externa.

Primeiro vamos compreender o que é uma validação interna e externa. Na validação interna, o modelo será validado utilizando um conjunto de moléculas que foram utilizadas para criar o modelo, o conjunto de treino. Na validação externa, o modelo é validado com um conjunto de moléculas na qual o modelo não foi apresentado.

A CV é uma técnica fundamental para avaliar a capacidade de generalização de um modelo, bastante importante para modelos de predição. Neste método de validação o modelo é avaliado uma série de vezes e ao final o desempenho é calculado a partir da média aritmética das avaliações.[57,95,102]

Na validação cruzada interna de *k* partes o conjunto de treinamento é dividido em *k* partes semelhantes, de forma aleatória e sem substituição. Então, o processo de validação irá fazer *k* iterações: a cada iteração *k*−1 partes será utilizada para treino e gerar um modelo, e uma parte será utilizada para teste. Na próxima iteração uma nova parte será utilizada para teste e as outras *k*−1 restantes como treino, de forma que ao final, as *k* partes tenham sido utilizadas como teste e assim avaliar o modelo. Ao final, a média de todas as *k* iterações será calculada e teremos a performance final do modelo. Na CV interna estratificada de *k* partes, a única diferença é que a divisão do conjunto de treinamento em *k* partes ao invés de ser aleatória, é realizada de forma estratificada, ou seja, mantendo o equilíbrio entre as classes, e em modelo de regressão o valor médio da resposta é aproximadamente igual em todos as *k* partes.

O método de CV estratificada em *k* partes é a mais utilizada e recomendada para uma validação externa. Nesta técnica o modelo é criado já realizando a validação cruzada, desta forma, o conjunto de treino é particionado em *k* partes de maneira estratificada e cada modelo será analisado por um conjunto de teste que conterá moléculas que o modelo não conhece, por isso validação externa. De modo que ao final do processo teremos cinco modelos que foram validados separadamente, então realiza-se a média aritmética dos modelos e teremos a performance do modelo. O conjunto para o qual desejamos prever a atividade será avaliado pelos *k* modelos criados, e ao final é feita a média aritmética da predição para cada molécula.[14,75] Normalmente na CV de *k* partes, $k = 5$ ou 10 são boas opções, uma vez que há um bom equilíbrio entre complexidade computacional e precisão de validação.[14,75]

A CV *leave-one-out* é uma variação da CV de *k*-partes. Neste método, cada amostra do conjunto de dados de treino é separada como um conjunto de teste, assim o número de combinações possíveis é igual ao número de pontos de dados no conjunto de treino. É um método exaustivo, e a depender do tamanho do conjunto de treino pode se tornar bastante custoso computacionalmente.

O último método que será comentado é o método *bootstrap*. Ao contrário da CV que utiliza amostragem sem repetição, ou seja, mesmas amostras não são incluídas em uma parte, no método *bootstrap* a amostragem é feita com repetição. Dessa forma, pode haver duplicação de algumas amostras. Embora essa técnica pareça suspeita, uma vez que pode estar gerando uma análise super ajustada, já foi comprovado que para esse caso, nesta técnica de validação, essa repetição pode melhorar a estimativa de performance do modelo.[101-103] Este método subdivide conjunto de dados em treino e teste, várias vezes (a ser determinado pelo operador), e a cada nova subdivisão, moléculas estarão repetidas no conjunto de treino.

DOMÍNIO DE APLICABILIDADE

Uma vez que o modelo foi criado e validado, a próxima etapa é a predição. Um conjunto de moléculas que não possui informações de atividades é utilizado para ser avaliado pelo modelo e ter sua predição realizada. Mas para ter confiança no resultado da predição é imprescindível analisar o domínio de aplicabilidade (APD, do inglês *applicability domain*).

O APD corresponde ao espaço químico que envolve os descritores das moléculas utilizadas na construção do modelo. Dessa forma, o APD fornecerá informações sobre a semelhança entre o que está sendo testado e o que foi utilizado para construir o modelo.[104-106] A predição de uma molécula só é utilizável/confiável quando estiver dentro do APD do modelo.[104-106]

O cálculo do APD é feito a partir de uma varredura de similaridade das estruturas presentes no conjunto de teste em relação ao banco de dados do treino, em que o melhor limiar de similaridade é aquele que apresenta o menor erro e o menor número de moléculas abaixo do limiar. As distâncias euclidianas geralmente são as mais utilizadas para calcular a similaridade química.[104-106]

PERSPECTIVAS

A pesquisa no campo da descoberta de medicamentos, com foco no QSAR, está passando por uma transformação notável, impulsionada pela crescente adoção de técnicas computacionais avançadas, como o aprendizado de máquina. Essas abordagens estão se revelando não apenas como ferramentas poderosas para a otimização de candidatos a medicamentos, mas também encontrando aplicações em diversas outras áreas.

Um dos aspectos mais notáveis é o impacto positivo que o aprendizado de máquina está tendo na química medicinal e ambiental. A capacidade de analisar grandes volumes de dados químicos e biológicos de maneira eficiente, identificar padrões complexos e prever interações moleculares com precisão tem o potencial de acelerar significativamente o desenvolvimento de novos medicamentos e a avaliação de compostos químicos em termos de impacto ambiental.

Além disso, a combinação do aprendizado de máquina com outras técnicas de inteligência artificial, como o *docking* molecular e a dinâmica molecular, está abrindo novas perspectivas para o QSAR. Essa integração permite uma análise mais abrangente das interações moleculares, fornecendo *insights* valiosos sobre como as moléculas interagem com alvos biológicos e como podem ser otimizadas para melhorar sua eficácia terapêutica ou reduzir seus efeitos adversos.

Em suma, a incorporação de técnicas de aprendizado de máquina no QSAR representa uma evolução essencial na pesquisa farmacêutica. Neste contexto, é importante que os químicos e farmacêuticos busquem se familiarizar com essas novas abordagens computacionais para que possam liderar avanços significativos na descoberta de medicamentos. Como explicado por Muratov *et al.*[11] os químicos que abraçam a evolução tecnológica estão destinados a substituir aqueles que não o fazem.

AGRADECIMENTOS

Os autores agradecem ao CNPq (grant 302469/2022-2) e FAPESQ (grant 2002/2022) pelo auxílio finaceiro.

REFERÊNCIAS

1. Leach, A. R.; Gillet, V. J.; *An Introduction to Chemoinformatics*, revised ed.; Springer: Dordrecht, 2007.

2. Chen, H.; Kogej, T.; Engkvist, O.; *Mol. Inf.* **2018**, *37*, 1800041. [Crossref]
3. Alves, V.; Braga, R.; Muratov, E.; Andrade, C.; *Quim. Nova* **2017**, *41*, 202. [Crossref]
4. Song, C. M.; Lim, S. J.; Tong, J. C.; *Brief. Bioinform.* **2009**, *10*, 579. [Crossref]
5. Gasteiger, J. Em *Handbook of Chemoinformatics*, vol. 1-4; Gasteiger, J., ed.; Wiley-VCH: Weinheim, 2008. [Crossref]
6. The Royal Society of Chemistry; *Chemoinformatics Approaches to Virtual Screen*; Varnek, A.; Tropsha, A., eds.; RSC Publishing: London, 2008. [Crossref]
7. Pérez-Sianes, J.; Pérez-Sánchez, H.; Díaz, F.; *Curr. Comput.-Aided Drug Des.* **2018**, *15*, 6. [Crossref]
8. Childs, C. M.; Canbek, O.; Kirby, T. M.; Zhang, C.; Zheng, J.; Szeto, C.; Póczos, B.; Kurtis, K. E.; Washburn, N. R.; *Cem. Concr. Res.* **2020**, *136*, 106173. [Crossref]
9. Barreiro, E. J.; Fraga, C. A. M.; *Química Medicinal: As Bases Moleculares da Ação dos Fármacos*, 2ª ed.; Artmed: Porto Alegre, 2000.
10. Bajorath, J.; *Nat. Rev. Drug Discovery* **2002**, *1*, 882. [Crossref]
11. Muratov, E. N.; Bajorath, J.; Sheridan, R. P.; Tetko, I. V.; Filimonov, D.; Poroikov, V.; Oprea, T. I.; Baskin, I. I.; Varnek, A.; Roitberg, A.; Isayev, O.; Curtalolo, S.; Fourches, D.; Cohen, Y.; Aspuru-Guzik, A.; Winkler, D. A.; Agrafiotis, D.; Cherkasov, A.; Tropsha, A.; *Chem. Soc. Rev.* **2020**, *49*, 3525. [Crossref]
12. Quadri, T. W.; Olasunkanmi, L. O.; Fayemi, O. E.; Akpan, E. D.; Verma, C.; Sherif, E. S. M.; Khaled, K. F.; Ebenso, E. E.; *Coord. Chem. Rev.* **2021**, *446*, 214101. [Crossref]
13. Oyekunle, D.; Agboola, O.; Ayeni, A.; Al-Baghdadi, S. B.; Kadhim, A.; Sulaiman, G.; Edan Salman, H.; Balakit, A. A.; Albo Hay Allah, M. A.; Ebrahimi, S.; Kalhor, E. G.; Nabavi, S. R.; Alamiparvin, L.; Pogaku, R.; *IOP Conference Series: Earth and Environmental Science* **2016**, *36*, 012011. [Crossref]
14. Cherkasov, A.; Muratov, E. N.; Fourches, D.; Varnek, A.; Baskin, I. I.; Cronin, M.; Dearden, J. C.; Gramatica, P.; Martin, Y. C.; Todeschini, R.; Consonni, V.; Kuz'min, V. E. V. E.; Cramer, R. D.; Benigni, R.; Yang, C.; Rathman, J. F.; Terfloth, L.; Gasteiger, J.; Richard, A. M.; Tropsha, A.; *J. Med. Chem.* **2014**, *57*, 4977. [Crossref]
15. Neves, B. J.; Braga, R. C.; Melo-Filho, C. C.; Moreira-Filho, J. T.; Muratov, E. N.; Andrade, C. H.; *Front. Pharmacol.* **2018**, *9*, 1275. [Crossref]
16. Sippl, W.; Robaa, D. Em *Applied Chemoinformatics: Achievements and Future Opportunities*; Engel, T.; Gasteiger, J., eds.; Willey: Weinheim, 2018, p. 9. [Crossref]
17. Cheng, T.; Li, Q.; Zhou, Z.; Wang, Y.; Bryant, S. H.; *AAPS J.* **2012**, *14*, 133. [Crossref]
18. Lu, I. L.; Huang, C. F.; Peng, Y. H.; Lin, Y. T.; Hsieh, H. P.; Chen, C. T.; Lien, T. W.; Lee, H. J.; Mahindroo, N.; Prakash, E.; Yueh, A.; Chen, H. Y.; Goparaju, C. M. V.; Chen, X.; Liao, C. C.; Chao, Y. S.; Hsu, J. T. A.; Wu, S. Y.; *J. Med. Chem.* **2006**, *49*, 2703. [Crossref]
19. Chu, W. T.; Yan, Z.; Chu, X.; Jain, A.; *J. Phys.: Conf. Ser.* **2017**, *884*, 012072. [Crossref]
20. Mao, J.; Akhtar, J.; Zhang, X.; Sun, L.; Guan, S.; Li, X.; Chen, G.; Liu, J.; Jeon, H. N.; Kim, M. S.; No, K. T.; Wang, G.; *iScience* **2021**, *24*, 103052. [Crossref]
21. Li, J.; Luo, D.; Wen, T.; Liu, Q.; Mo, Z.; *J. Mol. Struct.* **2021**, *1244*, 131249. [Crossref]
22. Bo, W.; Chen, L.; Qin, D.; Geng, S.; Li, J.; Mei, H.; Li, B.; Liang, G.; *Trends Food Sci. Technol.* **2021**, *114*, 176. [Crossref]
23. Kubinyi, H.; *Quant. Struct.-Act. Relat. Environ. Sci.-VII, Proc. QSAR 96* **2002**, *21*, 348. [Crossref]
24. Christensen, A. S.; Kubař, T.; Cui, Q.; Elstner, M.; *Chem. Rev.* **2016**, *116*, 5301. [Crossref]
25. Hansch, C.; Maloney, P. P.; Fujita, T.; Muir, R. M.; *Nature* **1962**, *194*, 178. [Crossref]
26. Free, S. M.; Wilson, J. W.; *J. Med. Chem.* **1964**, *7*, 395. [Crossref]
27. Cramer, R. D.; Patterson, D. E.; Bunce, J. D.; *J. Am. Chem. Soc.* **1988**, *110*, 5959. [Crossref]
28. Marchand-Geneste, N.; Carpy, A. J. M.; *SAR QSAR Environ. Res.* **2004**, *15*, 43. [Crossref]
29. Karelson, M.; Lobanov, V. S.; Katritzky, A. R.; *Chem. Rev.* **1996**, *96*, 1027. [Crossref]
30. Wang, L.; Ding, J.; Pan, L.; Cao, D.; Jiang, H.; Ding, X.; *Chemom. Intell. Lab. Syst.* **2021**, *217*, 104384. [Crossref]
31. Damale, M.; Harke, S.; Kalam Khan, F.; Shinde, D.; Sangshetti, J.; *Mini-Rev. Med. Chem.* **2014**, *14*, 35. [Crossref]
32. Hopfinger, A. J.; Wang, S.; Tokarski, J. S.; Jin, B.; Albuquerque, M.; Madhav, P. J.; Duraiswami, C.; *J. Am. Chem. Soc.* **1997**, *119*, 10509. [Crossref]
33. Neves, B. J.; Braga, R. C.; Melo-Filho, C. C.; Moreira-Filho, J. T.; Muratov, E. N.; Andrade, C. H.; *Front. Pharmacol.* **2018**, *9*, 945. [Crossref]
34. Dobchev, D.; Pillai, G.; Karelson, M.; *Curr. Top. Med. Chem.* **2014**, *14*, 1913. [Crossref]
35. Zhang, L.; Zhang, H.; Ai, H.; Hu, H.; Li, S.; Zhao, J.; Liu, H.; *Curr. Top. Med. Chem.* **2018**, *18*, 987. [Crossref]
36. Thiry, A.; Ledecq, M.; Cecchi, A.; Frederick, R.; Dogné, J. M.; Supuran, C. T.; Wouters, J.; Masereel, B.; *Bioorg. Med. Chem.* **2009**, *17*, 553. [Crossref]
37. Rocha, S. F. L. S.; Olanda, C. G.; Fokoue, H. H.; Sant'Anna, C. M. R.; *Curr. Top. Med. Chem.* **2019**, *19*, 1751. [Crossref]
38. Schneider, G.; *Nat. Rev. Drug Discovery* **2010**, *9*, 273. [Crossref]
39. Lavecchia, A.; Giovanni, C.; *Curr. Med. Chem.* **2013**, *20*, 2839. [Crossref]
40. Varnek, A.; *Methods Mol. Biol*. **2011**, *672*, 213. [Crossref].
41. Rodrigues, R. P.; Mantoani, S. P.; de Almeida, J. R.; Pinsetta, F. R.; Semighini, E. P.; da Silva, V. B.; da Silva, C. H. T. P.; *Rev. Virtual Quim.* **2012**, *4*, 739. [Crossref]
42. Drwal, M. N.; Griffith, R.; *Drug Discov. Today: Technol.* **2013**, *10*, e395. [Crossref]
43. Fu, Y.; Sun, Y. N.; Yi, K. H.; Li, M. Q.; Cao, H. F.; Li, J. Z.; Ye, F.; *Front. Chem.* **2018**, *6*, 1. [Crossref]
44. Bento, A. P.; Gaulton, A.; Hersey, A.; Bellis, L. J.; Chambers, J.; Davies, M.; Krüger, F. A.; Light, Y.; Mak, L.; McGlinchey, S.; Nowotka, M.; Papadatos, G.; Santos, R.; Overington, J. P.; *Nucleic Acids Res.* **2014**, *42*, 1083. [Crossref]
45. Zhu, H.; *Annu. Rev. Pharmacol. Toxicol.* **2020**, *60*, 573. [Crossref]
46. Sayers, E. W.; Beck, J.; Bolton, E. E.; Bourexis, D.; Brister, J. R.; Canese, K.; Comeau, D. C.; Funk, K.; Kim, S.; Klimke, W.; Marchler-Bauer, A.; Landrum, M.; Lathrop, S.; Lu, Z.; Madden, T. L.; O'Leary, N.; Phan, L.; Rangwala, S. H.; Schneider, V. A.; Skripchenko, Y.; Wang, J.; Ye, J.; Trawick, B. W.; Pruitt, K. D.; Sherry, S. T.; *Nucleic Acids Res.* **2021**, *49*, D10. [Crossref]
47. Sayers, E. W.; Beck, J.; Brister, J. R.; Bolton, E. E.; Canese, K.; Comeau, D. C.; Funk, K.; Ketter, A.; Kim, S.; Kimchi, A.; Kitts, P. A.; Kuznetsov, A.; Lathrop, S.; Lu, Z.; McGarvey, K.; Madden, T. L.; Murphy, T. D.; O'Leary, N.; Phan, L.; Schneider, V. A.; Thibaud-Nissen, F.; Trawick, B. W.; Pruitt, K. D.; Ostell, J.; *Nucleic Acids Res.* **2020**, *48*, D9. [Crossref]
48. Mendez, D.; Gaulton, A.; Bento, A. P.; Chambers, J.; De Veij, M.; Félix, E.; Magariños, M. P.; Mosquera, J. F.; Mutowo, P.; Nowotka, M.; Gordillo-Marañón, M.; Hunter, F.; Junco, L.; Mugumbate, G.; Rodriguez-Lopez, M.; Atkinson, F.; Bosc, N.; Radoux, C. J.; Segura-Cabrera, A.; Hersey, A.; Leach, A. R.; *Nucleic Acids Res.* **2019**, *47*, D930. [Crossref]
49. Wishart, D. S.; Feunang, Y. D.; Guo, A. C.; Lo, E. J.; Marcu, A.; Grant, J. R.; Sajed, T.; Johnson, D.; Li, C.; Sayeeda, Z.; Assempour, N.;

Iynkkaran, I.; Liu, Y.; MacIejewski, A.; Gale, N.; Wilson, A.; Chin, L.; Cummings, R.; Le, Di.; Pon, A.; Knox, C.; Wilson, M.; *Nucleic Acids Res.* **2018**, *46*, D1074. [Crossref]
50. Scotti, M. T.; Herrera-Acevedo, C.; Oliveira, T. B.; Costa, R. P. O.; Santos, S. Y. K. O.; Rodrigues, R. P.; Scotti, L.; da Costa, F. B.; *Molecules* **2018**, *23*, 103. [Crossref]
51. Costa, R. P. O.; Lucena, L. F.; Silva, L. M. A.; Zocolo, G. J.; Herrera-Acevedo, C.; Scotti, L.; da Costa, F. B.; Ionov, N.; Poroikov, V.; Muratov, E. N.; Scotti, M. T.; *J. Chem. Inf. Model.* **2021**, *61*, 2516. [Crossref]
52. Zeng, X.; Zhang, P.; He, W.; Qin, C.; Chen, S.; Tao, L.; Wang, Y.; Tan, Y.; Gao, D.; Wang, B.; Chen, Z.; Chen, W.; Jiang, Y. Y.; Chen, Y. Z.; *Nucleic Acids Res.* **2018**, *46*, D1217. [Crossref]
53. Pilon, A. C.; Valli, M.; Dametto, A. C.; Pinto, M. E. F.; Freire, R. T.; Castro-Gamboa, I.; Andricopulo, A. D.; Bolzani, V. S.; *Sci. Rep.* **2017**, *7*, 7215. [Crossref]
54. Huang, W.; Brewer, L. K.; Jones, J. W.; Nguyen, A. T.; Marcu, A.; Wishart, D. S.; Oglesby-Sherrouse, A. G.; Kane, M. A.; Wilks, A.; *Nucleic Acids Res.* **2018**, *46*, D575. [Crossref]
55. Klementz, D.; Döring, K.; Lucas, X.; Telukunta, K. K.; Erxleben, A.; Deubel, D.; Erber, A.; Santillana, I.; Thomas, O. S.; Bechthold, A.; Günther, S.; *Nucleic Acids Res.* **2016**, *44*, D509. [Crossref]
56. Pilón-Jiménez, B.; Saldívar-González, F.; Díaz-Eufracio, B.; Medina-Franco, J.; *Biomolecules* **2019**, *9*, 31. [Crossref]
57. Amaral, F.; *Aprenda Mineração de Dados*, 1ª ed.; Alta Books: Rio de Janeiro, 2016.
58. Young, D.; Martin, T.; Venkatapathy, R.; Harten, P.; *QSAR Comb. Sci.* **2008**, *27*, 1337. [Crossref]
59. Williams, A. J.; Ekins, S.; *Drug Discov. Today* **2011**, *16*, 747. [Crossref]
60. Fourches, D.; Muratov, E.; Tropsha, A.; *Nat. Chem. Biol.* **2015**, *11*, 535. [Crossref]
61. Maggiora, G. M.; *J. Chem. Inf. Model.* **2006**, *46*, 1535. [Crossref]
62. Golbraikh, A.; Muratov, E.; Fourches, D.; Tropsha, A.; *J. Chem. Inf. Model.* **2014**, *54*, 1. [Crossref]
63. Todeschini, R.; Consonni, V.; *Handbook of Molecular Descriptores*, 1st ed.; Wiley-VCH: Weinheim, 2000. [Crossref]
64. Danishuddin; Khan, A. U.; *Drug Discov. Today* **2016**, *21*, 1291. [Crossref]
65. Hutter, M. C.; *ChemMedChem* **2010**, *5*, 306. [Crossref]
66. Vedani, A.; Dobler, M.; *J. Med. Chem.* **2002**, *45*, 2139. [Crossref]
67. Vedani, A.; Dobler, M.; Lill, M. A.; *J. Med. Chem.* **2005**, *48*, 3700. [Crossref]
68. Talete srl; *Dragon v. 7.0*; Kode Chemoinformatics, Italy, 2006.
69. Cruciani, G.; Pastor, M.; Guba, W.; *Eur. J. Pharm. Sci.* **2000**, *11*, S29. [Crossref]
70. Mauri, A.; Bertola, M.; *Int. J. Mol. Sci.* **2022**, *23*, 12882. [Crossref]
71. Scotti, M.; Speck-Planche, A.; Tavares, J.; da Silva, M.; Cordeiro, M. N. D. S.; Scotti, L.; *Curr. Bioinf.* **2015**, *10*, 509. [Crossref]
72. Todeschini, R.; Consonni, V.; *Molecular Descriptors for Chemoinformatics*, 2nd ed.; Wiley-VCH: Weinheim, 2009.
73. Baskin, I. I. Em *Methods in Molecular Biology*; Wlaker, J. M., ed.; Humana Press: New Jersey, 2018, p. 119.
74. Barros, R. P. C.; Sousa, N. F.; Scotti, L.; Scotti, M. T. Em *Ecotoxicological QSARs*; Roy, K., ed.; Humana Press: New Jersey, 2020, p. 151. [Crossref]
75. Fourches, D.; Pu, D.; Tassa, C.; Weissleder, R.; Shaw, S. Y.; Mumper, R. J.; Tropsha, A.; *ACS Nano* **2010**, *4*, 5703. [Crossref]
76. Salzberg, S.; Quinlan, J. R.; *Machine Learning* **1994**, *16*, 235. [Crossref]
77. Gertrudes, J. C.; Maltarollo, V. G.; Silva, R. A.; Oliveira, P. R.; Honorio, K. M.; da Silva, A. B. F.; *Curr. Med. Chem.* **2012**, *19*, 4289. [Crossref]
78. Yamazaki, K.; Kusunose, N.; Fujita, K.; Sato, H.; Asano, S.; Dan, A.; Kanaoka, M.; *Bioorg. Med. Chem. Lett.* **2006**, *16*, 1371. [Crossref]
79. Dara, S.; Dhamercherla, S.; Jadav, S. S.; Babu, C. M.; Ahsan, M. J.; *Artificial Intelligence Review* **2022**, *55*, 1947. [Crossref]
80. Patel, L.; Shukla, T.; Huang, X.; Ussery, D. W.; Wang, S.; *Molecules* **2020**, *25*, 5277. [Crossref]
81. Carracedo-Reboredo, P.; Liñares-Blanco, J.; Rodríguez-Fernández, N.; Cedrón, F.; Novoa, F. J.; Carballal, A.; Maojo, V.; Pazos, A.; Fernandez-Lozano, C.; *Comput. Struct. Biotechnol. J.* **2021**, *19*, 4538. [Crossref]
82. Breiman, L.; Friedman, J. H.; Olshen, R. A.; Stone, C. J.; *Classification and Regression Trees*, 1st ed.; Routledge: England, 2017.
83. Breiman, L.; *Machine Learning* **2001**, *45*, 5. [Crossref]
84. Barros, R. P. C.: *Triagem Virtual de Metabólitos Secundários com Potencial Atividade Antimicrobiana do Gênero Solanum e Estudo Fitoquimico de Solanum Capsicoides all*; Disertação de Mestrado, Universidade Federal da Paraíba, João Pessoa, 2017. [Link] acessado em Janeiro 2024
85. Fernandez-Lozano, C.; Gestal, M.; Pedreira-Souto, N.; Postelnicu, L.; Dorado, J.; Munteanu, C.; *Curr. Top. Med. Chem.* **2013**, *13*, 1681. [Crossref]
86. Fernandez-Lozano, C.; Gestal, M.; González-Díaz, H.; Dorado, J.; Pazos, A.; Munteanu, C. R.; *J. Theor. Biol.* **2014**, *349*, 12. [Crossref]
87. Campbell, C.; Ying, Y.; *Synthesis Lectures on Artificial Intelligence and Machine Learning* **2011**, *5*, 1. [Crossref]
88. Lorena, A. C.; de Carvalho, A. C. P. L. F.; *Revista de Informática Teórica e Aplicada* **2007**, *14*, 43. [Crossref]
89. Altman, N. S.; Altman, N. S.; *Am. Stat.* **1991**, *46*, 175. [Link] acessado em Janeiro 2024
90. Everitt, B. S.; Landau, S.; Leese, M.; Stahl, D.; *Miscellaneous Clustering Methods, in Cluster Analysis*, 5th ed.; John Wiley & Sons: Chichester, 2011.
91. Zhou, J.; Chen, F.; *AI & Society* **2023**, *38*, 2693. [Crossref]
92. Zhou, J.; Chen, F.; Berry, A.; Reed, M.; Zhang, S.; Savage, S.; *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, Canvberra, Australia, 2020, p. 3010. [Link] acessado em Janeiro 2024
93. The Organization for Economic Cooperation and Development; Artificial Intelligence, https://www.oecd.org/finance/artificial-intelligence-machine-learning-big-data-in-finance.htm, acessado em Janeiro 2024.
94. Siau, K.; Wang, W.; *Journal of Database Management* **2020**, *31*, 74. [Crossref]
95. Livingstone, D.; *A Practical Guide to Scientific Data Analysis*, 1st ed.; Wiley-VCH: Weinheim, 2009.
96. Hand, D. J.; Till, R. J.; *Machine Learning* **2001**, *45*, 171. [Crossref]
97. Matthews, B. W.; *Biochimica et Biophysica Acta (BBA) - Protein Structure* **1975**, *405*, 442. [Crossref]
98. Marôco, J.; *Análise Estatística com o SPSS Statistics*, 8ª ed.; Report Number: Pero Pinheiro, 2021.
99. Santos, C.; *Estastística Descritiva: Manual de Auto-Aprendizagem*, 3ª ed.; Sílabo: Lisboa, 2018.
100. Palacio-Niño, J.-O.; Berzal, F.; arXiv, 2019. [Link] acessado em Março 2024.
101. Tan, P. N.; Steinbach, M.; Kumar, V.; *Introductionto Data Minig*, 1st ed.; Pearson: London, 2005.
102. Tropsha, A.; Gramatica, P.; Gombar, V. K.; *QSAR Comb. Sci.* **2003**, *22*, 69. [Crossref]
103. Bishop, C. M.; *Pattern Recognition and Machine Learning*, corr. 2nd printing 2011 ed.; Springer: Berlin, 2006.
104. Gramatica, P.; *Int. J. Quant. Struct.-Prop. Relat.* **2020**, *5*, 37. [Crossref]
105. Roy, K.; Kar, S.; Das, R. N.; *Understanding the Basics of QSAR for Applications in Pharmaceutical Sciences and Risk Assessment*, 1st ed.; Academic Press: San Diego, 2015.
106. Kar, S.; Roy, K.; Leszczynski, J.; *Methods Mol. Biol.* **2018**, *1800*, 141. [Crossref]