Revista Brasileira de Ensino de Física, vol. 43, suppl. 1, e20200407 (2021)
www.scielo.br/rbef
DOI: https://doi.org/10.1590/1806-9126-RBEF-2020-0407

Artigos Gerais

Machine Learning na Física, Química, e Ciência de Materiais: Descoberta e Design de Materiais

Machine Learning in Physics, Chemistry, and Materials Science: Materials Discovery and Design

Gabriel R. Schleder[1,2], Adalberto Fazzio*[1,2]

[1]Laboratório Nacional de Nanotecnologia, Centro Nacional de Pesquisa em Energia e Materiais, Campinas, SP, Brasil.
[2]Universidade Federal do ABC, Santo André, SP, Brasil.

Recebido em 23 de setembro de 2020. Aceito em 19 de outubro de 2020.

Avanços recentes nas técnicas experimentais e desenvolvimentos teóricos e computacionais resultaram em um aumento crescente na geração de dados. Essa disponibilidade de dados, associada à novas ferramentas e tecnologias capazes de armazenar e processar esses dados, culminaram na chamada ciência de dados. Uma das áreas de maior destaque recente são os algoritmos de aprendizado de máquina (*machine learning*), que têm como objetivo a identificação de correlações e padrões nos conjuntos de dados. Esses algoritmos vêm sendo usados há décadas, por exemplo nas áreas da saúde. Apenas recentemente a comunidade introduziu a sua aplicação para materiais, devido à criação, padronização e consolidação de bancos de dados consistentes. O uso dessas metodologias permite extrair conhecimento e *insights* da enorme quantidade de dados brutos e informações agora disponíveis. A área apresenta diversas oportunidades para a solução de desafios na física, química e ciência de materiais. Especificamente, os métodos de machine learning são uma poderosa ferramenta para a descoberta e design de novos materiais com propriedades e funcionalidades desejadas e otimizadas. Neste artigo apresentamos o contexto do surgimento do machine learning, seus fundamentos e aplicações para a descoberta e design de materiais.
Palavras-chave: Inteligência artificial, ciência de dados, ciência baseada em ferramentas, informática de materiais.

Nowadays, we are witnessing a tremendous increase in data generation enabled by advances in experimental techniques and theoretical and computational developments. This availability of data, associated with new tools and technologies capable of storing and processing that data, culminated in the so-called data science. One of the most prominent areas (*machine learning*), which aims to identify correlations and patterns in the data sets. These algorithms have been used for decades in different areas. Only recently the community introduced its application for materials, due to the creation, standardization, and consolidation of consistent databases. The use of these methodologies allows to extract knowledge and *insights* from the huge amount of raw data and information now available. The area presents several opportunities for solving challenges in physics, chemistry, and materials science. Specifically, machine learning methods are a powerful tool for discovering and designing new materials with desired and optimized properties and functionalities. In this article, we present the context of the emergence of machine learning, its foundations, and applications for the discovery and design of materials.
Keywords: Artificial intelligence, data-driven science, tool-driven science, materials informatics.

1. Introdução

Um dos objetivos da ciência é descrever, explicar e predizer o comportamento de fenômenos físicos, químicos e biológicos. A natureza é complexa, e descrevê-la exige a busca de aproximações ou simplificações. Ao longo da história, as ciências fizeram uso de aproximações para separar e extrair o que é fundamental do que é supérfluo, para a compreensão das diversas manifestações da natureza. Com isso, a tarefa complexa se torna um problema de algumas variáveis tangíveis, e assim se estuda o problema de interesse. Isso levou a física e a química a um desenvolvimento mais rápido se comparado com o campo da biologia.

Filósofos argumentam que historicamente a ciência evolui pela chamada "concept-driven science", com estabelecimento e quebra de paradigmas, que são pressupostos fundamentais compartilhados pela comunidade de cada área científica; visão expressa por Thomas Kuhn [1]. Um contraponto complementar à essa visão, especialmente aplicado às ciências naturais, é que as ciências sejam "tool-driven" [2], ou seja, que novas ferramentas (de qualquer natureza) avançam as ciências tanto quanto novos conceitos [3]. Essa visão defendida por Peter Galison tem vários exemplos mais recentes, tal como o desenvolvimento de aceleradores de

*Endereço de correspondência: adalberto.fazzio@lnnano.cnpem.br

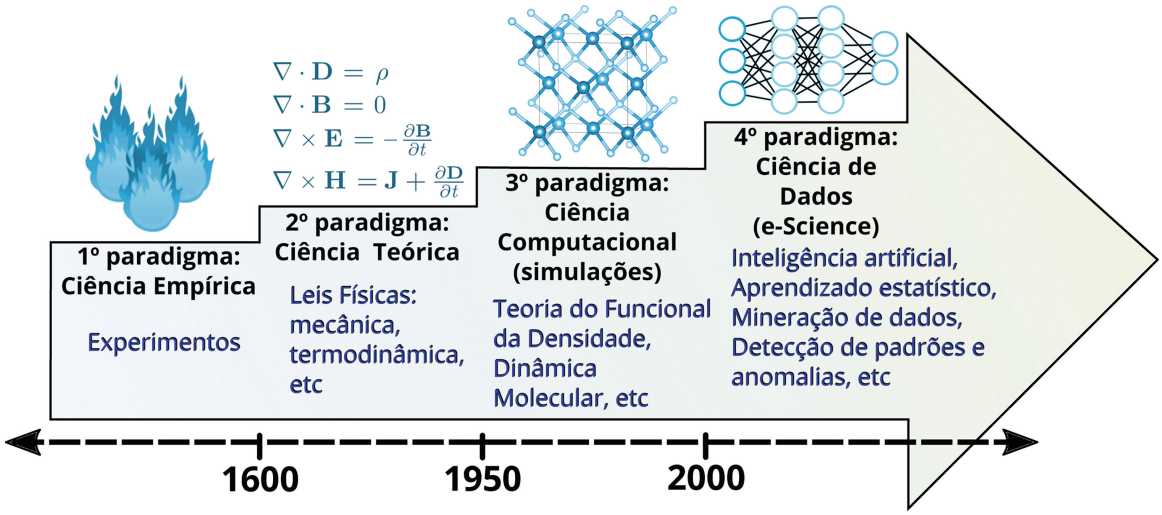

Figura 1: Os quatro paradigmas da ciência: experimental, teórica, computacional e baseada em dados. Cada paradigma se beneficia e contribui para os outros, sendo complementares. Adaptado de [6, 7] CC BY 3.0.

partículas, microscópios, telescópios e técnicas computacionais. O físico Freeman Dyson expressa essa visão[1]:

"Novas direções na ciência são criadas por novas ferramentas com muito mais frequência do que por novos conceitos. O efeito de uma revolução impulsionada por conceitos é explicar coisas antigas de novas maneiras. O efeito de uma revolução impulsionada por ferramentas é descobrir coisas novas que precisam ser explicadas" [4].

O aprendizado de máquina (*machine learning, ML*) surge nesse contexto como uma nova classe de ferramentas estatísticas para identificar padrões de forma automatizada em diferentes conjuntos de dados, e assim auxiliar na construção de novo conhecimento científico. Com o crescimento cada vez maior da geração e diversidade dos dados experimentais, teóricos e de simulação, a forma e técnicas de análise e processamento da informação está mudando de tal forma que distingue-se a ciência baseada em dados (do inglês *data-driven science*) como um novo "paradigma" ferramental da ciência [5]. Esse paradigma naturalmente se utiliza e complementa os paradigmas anteriores de experimentos, teoria e computação/simulação, conforme mostrado na Figura 1. Embora essas técnicas sejam amplamente utilizadas há tempos em áreas de grande complexidade como biologia, medicina e astronomia, apenas recentemente elas vêm se consolidando na física, química e ciência de materiais. Os avanços obtidos nessas áreas se mostram disruptivos na complexidade de descobrir novos materiais com propriedades desejáveis e otimizadas.

[1] Tradução livre do original: *"New directions in science are launched by new tools much more often than by new concepts. The effect of a concept-driven revolution is to explain old things in new ways. The effect of a tool-driven revolution is to discover new things that have to be explained."*[4].

2. Machine Learning

Esta seção é majoritariamente adaptada de [6].

2.1. Contexto: quando e por quê?

A descoberta de novos materiais e/ou sua funcionalização possibilita a criação de aplicações tecnológicas fundamentais para superar muitos dos desafios da sociedade moderna. O impacto do uso de materiais ao longo da história é difícil de quantificar, desde a idade da pedra, passando pela idade do bronze e ferro [8]. Entretanto, o impacto baseado na tecnologia do silício e a revolução dos plásticos são bem palpáveis [9]. Estima-se que o desenvolvimento de materiais permitiu dois terços dos avanços na área da computação e transformou também outras indústrias como armazenamento de energia [10].

Dada a demanda crescente por novos materiais e o desenvolvimento relativamente lento deles, ao mesmo tempo em que os recursos computacionais e algoritmos enfrentam grandes avanços, é natural perguntar: como a ciência computacional pode melhorar a eficiência da descoberta de materiais? Outras áreas, como a indústria farmacêutica e de biotecnologia, sugerem caminhos possíveis [11, 12]. No entanto, dentro do quarto paradigma da ciência baseada em dados, a comunidade de materiais está aparentemente atrasada em comparação com esses campos. Essa chegada tardia está relacionada a gargalos na capacidade computacional e de geração e armazenamento de dados, mas desde que as primeiras simulações computacionais de materiais foram realizadas, uma quantidade cada vez maior de estudos faz uso deste paradigma [6].

A teoria do funcional da densidade (*density functional theory*, DFT) se estabeleceu como a ferramenta padrão para simulação de materiais após seu sucesso na descrição de muitas propriedades físicas

importantes, como geometrias nos estados fundamentais, energias totais e relativas e estruturas eletrônicas. Posteriormente, diversos marcos foram alcançados, como a descrição de propriedades estruturais, eletrônicas, ópticas, magnéticas, catalíticas e quânticas, tanto para materiais bulk quanto na nanoescala [13].

Conforme os desenvolvimentos computacionais aumentaram seu desempenho, o armazenamento de dados tornou-se mais barato e novos algoritmos foram desenvolvidos, uma mudança gradual na forma usual de trabalho de cientistas de materiais, especialmente computacionais, ocorreu na última década. Surgiu uma nova forma de estudar os materiais: antes, a ideia era escolher um ou alguns candidatos e investigá-los minuciosamente para obter um conhecimento mais profundo sobre suas propriedades e possíveis aplicações. Agora é possível simular facilmente um grande número de compostos – técnica conhecida como *high-throughput (HT)* – e buscar uma propriedade particular em um catálogo de candidatos [14]. Isso marcou o nascimento dessa nova filosofia na área de materiais, dentro do contexto da ciência baseada em dados e descrita na Figura 2.

A simples geração massiva de dados não é garantia de convertê-los em informação e, posteriormente, em conhecimento. Converter o conhecimento em avanços para a sociedade é um desafio ainda maior. Existem lacunas entre a criação, o armazenamento de dados e a capacidade de obter conhecimento e tecnologias utilizáveis a partir deles. A tendência dessa lacuna é aumentar com o tempo [15]. Portanto, dado esse cenário, o uso de abordagens orientadas a dados é fundamental para reduzir essa lacuna e avançar nas pesquisas. Avanços recentes em técnicas experimentais e computacionais resultaram em um aumento exponencial nas quantidades de dados gerados, apresentando também complexidade crescente, levando ao conceito de *big-data*. Nesse sentido, as técnicas de machine learning visam extrair conhecimento e *insights* desses dados, identificando suas correlações e padrões [6].

2.2. O que é e quando utilizar?

Explorando a evolução do quarto paradigma da ciência, um paralelo pode ser feito entre o artigo de Eugene Wigner de 1960 "A eficácia irracional da matemática nas ciências naturais" [16] para o caso atual da "A eficácia irracional dos dados" [17]. O que leva à essa eficácia irracional dos dados recentemente? Principalmente a extração de conhecimento dessa grande quantidade de dados acumulados. Isso é feito por meio de técnicas de aprendizado de máquina que podem identificar padrões e relações nos dados, por mais complexas que sejam, mesmo para espaços de dimensionalidade arbitrariamente altas, inacessíveis à compreensão humana [6].

O machine learning (ML) pode ser definido como uma classe de métodos para análise automatizada de dados, que são capazes de detectar padrões nos dados. Esses padrões extraídos podem ser usados para prever informações desconhecidas ou para auxiliar nos processos de tomada de decisão sob incerteza [18]. A definição tradicional diz que o aprendizado da máquina melhora progressivamente com a experiência (dados) em tarefas determinadas, de acordo com uma métrica de sucesso definida, mas sem ser explicitamente programada para isso [19, 20]. Este campo de pesquisa evoluiu a partir da área mais geral de inteligência artificial (IA), inspirada pelos avanços na década de 1950 em estatística, ciência e tecnologia da computação e neurociência. A Figura 3b mostra a relação entre a área de IA e o ML. Em contraste, uma definição menos rigorosa de IA é qualquer técnica que permite aos computadores imitar a inteligência humana. Isso pode ser alcançado não apenas por ML, mas também por estratégias programadas "menos inteligentes", como árvores de decisão, regras

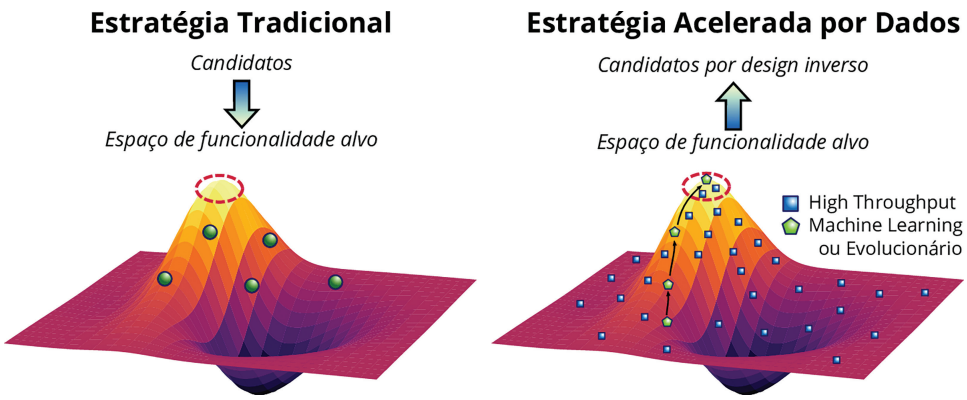

Figura 2: Diferenças das abordagens tradicional e baseadas em dados para a descoberta e design de moléculas e materiais. Na abordagem tradicional um candidato é avaliado por estratégias de tentativa e erro (esquerda), já na abordagem de dados, é substituída por estratégias de design inverso (direita), capaz de buscar materiais que maximizam as funcionalidades-alvo por meio de high-throughput, aprendizado de máquina ou técnicas evolutivas. Adaptado com permissão de [14]. Copyright 2019 American Chemical Society.

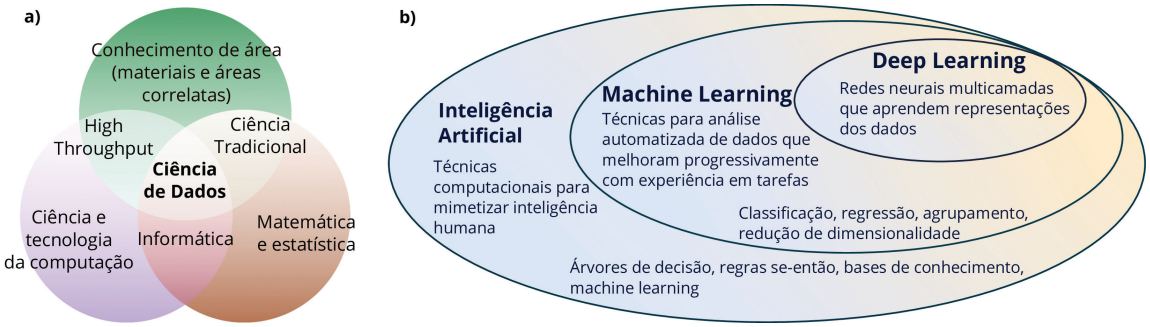

Figura 3: (a) Ciência de dados como disciplina integrativa, pela convergência da matemática e estatística, ciência e tecnologia da computação e conhecimento específico de área [22]. (b) Descrição hierárquica e exemplos de técnicas de inteligência artificial e suas subáreas aprendizado de máquina (*machine learning*) e aprendizado profundo (*deep learning*). Adaptado de [6] CC BY 3.0 e [22], com permissão da *The Royal Society of Chemistry*.

"se-então", bases de conhecimento e lógica computacional. Recentemente, um subcampo do ML que está ganhando cada vez mais atenção devido ao seu sucesso em várias áreas é o aprendizado profundo (*deep learning*) [21]. É um tipo de aprendizagem de representações vagamente inspirado por redes neurais biológicas, tendo um número grande de camadas entre suas entradas (*inputs*) e saídas (*outputs*). Um campo intimamente relacionado e componente muito importante do ML é a fonte de dados que permitirá aos algoritmos aprender. Este é o campo da ciência de dados, apresentado na Figura 3a.

Finalmente, quando o ML deve ou não ser empregado. De maneira geral, podemos usar o ML para 2 tipos de problemas: *(i)* para tratar problemas que métodos tradicionais não conseguem, de maneira aproximada; e *(ii)* para otimizar a solução de problemas já tratáveis, porém ou de maneira melhor e mais robusta, ou mais rápida, ou mais econômica, de preferência os três simultaneamente. O pré-requisito crucial é a disponibilidade de dados, que devem ser consistentes, suficientes, validados e representativos do comportamento de interesse a ser descrito. Além disso, é preciso considerar os pontos fortes dos métodos de aprendizado de máquina, que podem lidar com espaços de alta dimensão na busca de padrões nos dados. Os padrões descobertos são então codificados explicitamente, levando à modelos computacionais que podem ser manipulados. Os métodos de ML são mais adequados para problemas em que as abordagens tradicionais apresentam dificuldades. Embora nem sempre seja claro especificar, se um problema pode ser identificado em um dos tipos gerais de problemas de ML descritos na seção 2.3.*i*, o ML pode ser uma ferramenta útil.

Em ordem crescente de valor agregado e dificuldade, os problemas gerais enfrentados são: a substituição da coleta de propriedades/dados difíceis, complexos ou custosos; generalizar um padrão presente em um conjunto de dados para uma classe de dados semelhante; obter uma relação entre variáveis correlacionadas, mas com ligações desconhecidas ou indiretas, que está além da intuição ou do conhecimento de área; obtenção de um modelo geral aproximado para uma propriedade desconhecida complexa, ou fenômenos que não possuem teoria ou equações fundamentais [23]. Historicamente, áreas que apresentam questões com essas características foram bem-sucedidas na aplicação dos métodos de ML, tal como nas áreas de automação, processamento de imagens e linguagem, ciências sociais, químicas e biológicas, e recentemente cada vez mais exemplos estão surgindo.

Com base nessas características, especificamente aplicados à ciência de materiais, vislumbramos os tipos comuns de problemas que fazem uso de estratégias orientadas a dados, e que são exemplificados na Seção 3. O primeiro é a obtenção de modelos para fenômenos que possuem relações ou mecanismos desconhecidos. Uma estratégia relacionada é substituir a descrição de uma propriedade muito complexa ou onerosa de ser obtida (mas que é parcialmente conhecida pelo menos para uma pequena classe de materiais) por um modelo de ML mais simples, tornando seu cálculo menos custoso. Se devidamente validado, este modelo pode então prever a propriedade para exemplos desconhecidos, expandindo o conjunto de dados [24]. No contexto da descoberta e design de materiais, esta estratégia pode ser empregada como uma forma de estender o conjunto de dados antes da seleção e triagem, onde os dados custosos iniciais levam a mais dados por meio do modelo de ML, que podem então ser filtrados para encontrar novos candidatos promissores. Outros problemas usam técnicas de seleção de características para descobrir modelos e descritores aproximados (uma forma de impressão digital do sistema, ver seção 2.3. *iii*), que auxiliam na compreensão fenomenológica do problema. O mais abundante são os problemas claramente vantajosos em que cálculos dispendiosos computacionalmente podem ser substituídos por um modelo muito mais eficiente, como a substituição de cálculos *ab initio* por modelos ML, como na obtenção direta de propriedades ou de potenciais atomísticos para simulações de dinâmica molecular, que então prevêem o valor de diferentes propriedades tais como o gap eletrônico, energias livres, de formação, total,

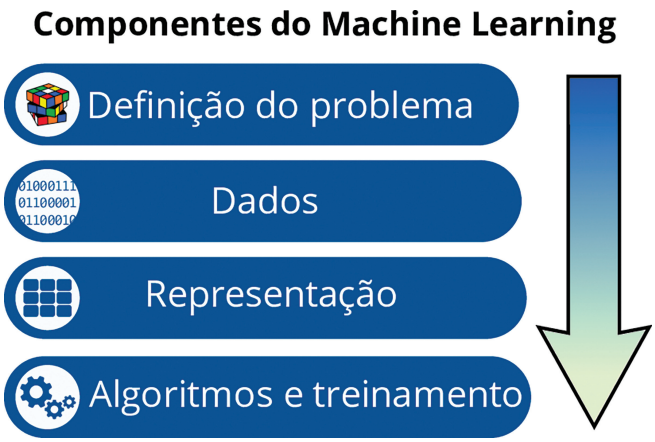

Figura 4: Os quatro componentes básicos do processo de machine learning. Adaptado de [13] CC BY 3.0.

de defeitos, condutividades, difusividade, propriedades térmicas, ópticas, magnéticas, entre muitas outras.

2.3. Como usar?

De maneira geral, podemos organizar o problema de machine learning em 4 passos fundamentais como ilustrado na Fig. 4 que são detalhados a seguir: *i)* definição do problema; *ii)* dados; *iii)* representação; e *iv)* algoritmos, validação, e aplicação.

i) Definição do problema: primeiramente vamos discutir quais os tipos de problemas mais comuns a serem tratados com machine learning. Formalmente, o problema de aprendizagem pode ser descrito [25] por: dado um conjunto de dados conhecido $\mathbf{X}$ – onde a notação $\mathbf{X}$ indica que é um vetor com uma ou várias variáveis –, prever ou aproximar a função de interesse desconhecida $y = f(\mathbf{X})$, em função desses dados conhecidos. O conjunto $\mathbf{X}$ é denominado *espaço de features* de input (também conhecido como atributos ou características) e um elemento $\mathbf{x}$ dele é chamado de *vetor de features*, ou simplesmente uma entrada. Com a função aproximada aprendida $\widehat{y} = \widehat{f}(\mathbf{X})$, o modelo pode então prever a saída para exemplos desconhecidos fora dos dados usados para o treinamento, e sua capacidade de fazer isso é chamada de *generalização* do modelo. Existem algumas categorias de problemas de ML de acordo com os tipos de entradas e saídas tratadas, sendo as duas principais as aprendizagens *supervisionada* e *não supervisionada*.

Na *aprendizagem não supervisionada*, também conhecida como descritiva, o objetivo é encontrar estruturas nos dados brutos $\mathbf{x}_i \in \mathbf{X}$ fornecidos sem rótulos, ou seja, não se usa ou não existem dados de saída y conhecidos. Se $f(\mathbf{X})$ é finito, o aprendizado é denominado *agrupamento (clustering)*, que agrupa dados em um número (conhecido ou desconhecido) de grupos pela similaridade em suas características. Por outro lado, se $f(\mathbf{X})$ está em uma distribuição $[0, \infty)$, a aprendizagem é chamada de *estimativa de densidade*, que aprende a distribuição marginal das características. Outro tipo importante de aprendizagem não supervisionada é a *redução de dimensionalidade*, que comprime o número de variáveis de entrada para representar os dados, útil quando $f(\mathbf{X})$ tem alta dimensionalidade e, portanto, uma estrutura de dados complexa para ser visualizada e usada na detecção de padrões.

Por outro lado, na *aprendizagem supervisionada* ou preditiva, o objetivo é aprender a função que leva as entradas às saídas alvo (*target*), tendo um conjunto de dados rotulados $(x_i, y_i) \in (\mathbf{X}, f(\mathbf{X}))$, conhecido como *conjunto de treinamento* (ao contrário do *conjunto de teste* desconhecido), com $i = N$ número de exemplos. Se a saída y_i é um conjunto finito categórico ou nominal (por exemplo, se um material é um metal ou isolante), o problema é chamado de *classificação*, que prevê o rótulo de classe para amostras desconhecidas. Caso contrário, se as saídas são escalares contínuos de valor real $y_i \in \mathbb{R}$, o problema é então chamado de *regressão*, que irá prever os valores de saída para os exemplos desconhecidos. Veremos os algoritmos relacionados na seção *iv)* adiante.

Outros tipos de problemas de ML são a aprendizagem *semi-supervisionada*, em que um grande número de dados não rotulados é combinado com um pequeno número de dados rotulados; a aprendizagem *multi-tarefa*; a *transferência de aprendizagem*, onde informações de problemas relacionados são exploradas para melhorar a tarefa de aprendizagem (geralmente uma com poucos dados disponíveis [26]); e o chamado *aprendizado por reforço*, no qual nenhuma entrada/saída é fornecida, mas sim feedbacks sobre as decisões como um meio de maximizar um sinal de recompensa, levando ao aprendizado de ações desejadas em determinados ambientes.

ii) Dados (inputs): a disponibilidade de dados é componente fundamental para qualquer processo de machine learning. Os resultados a serem obtidos no processo serão tão bons quanto a quantidade e qualidade dos dados que serão utilizados. Qualidade nesse contexto se refere que os dados sejam representativos do problema a ser estudado, consistentes, e que possuam informação relacionada à tarefa ser realizada [14]. Portanto, o processo para chegar num conjunto de dados de qualidade leva algumas etapas. Inicialmente, a etapa de coleta e/ou curadoria dos dados para geração e seleção de um subconjunto relevante e útil de dados disponíveis para a resolução dos problemas. Posteriormente, o pré-processamento de

dados, que busca uma formatação adequada dos dados, limpeza de dados corrompidos e ausentes, transformação dos dados conforme necessário por operações como normalização, discretização, cálculo da média, suavizar ou diferenciar, conversão uniforme para inteiros, *double* ou *strings* e amostragem adequada para otimizar a representatividade do conjunto [6]. Tendo os dados brutos tecnicamente corretos, é possível a próxima etapa de escolha da representação adequada ao problema.

iii) Representação: também chamada de impressão digital (*fingerprint*) ou descritor [27], a representação determinará a capacidade e o desempenho do processo de machine learning. Somente se as variáveis necessárias forem representadas que o algoritmo será capaz de aprender a relação desejada. Essa etapa mapeia em um vetor as diferentes variáveis de entrada (*features* de input) disponíveis que descrevem e identificam as amostras (no presente contexto, os materiais). Alguns requisitos desejáveis universais são propostos [6, 14, 28], tais como: a representação deve ser *a*) completa (suficiente para diferenciar os exemplos), *b*) única (dois exemplos terão a mesma representação apenas se forem de fato iguais), *c*) discriminativos (sistemas similares ou diferentes serão caracterizados por representações similares ou diferentes), e *d*) eficiente e simples de ser obtido (o cálculo da representação em si deve ser rápido). Esses requisitos apresentados servem para garantir que os modelos sejam eficientes usando apenas informações essenciais. Para qualquer novo problema de machine learning, o processo de engenharia de features, que engloba a seleção, combinação, e transformação destas, é responsável pela maior parte dos esforços e do tempo usado no projeto [29].

iv) Algoritmos, validação, e aplicação: A tarefa de construir e utilizar algoritmos é um estudo caso a caso. Nenhum algoritmo de ML é universalmente superior [30, 31]. Em particular, a escolha do algoritmo de aprendizagem é uma etapa fundamental na construção de um pipeline de ML, e muitas opções estão disponíveis, cada uma adequada para um determinado problema e/ou conjunto de dados. Esse conjunto de dados pode ser de dois tipos: rotulado ou não rotulado. Como vimos, no primeiro caso, a tarefa é encontrar o mapeamento entre os pontos dos dados e os rótulos correspondentes $\{\mathbf{x}^{(i)}\} \rightarrow \{y^{(i)}\}$ por meio de um algoritmo de aprendizagem supervisionada. Por outro lado, se não há rótulos no conjunto de dados, a tarefa é encontrar uma estrutura dentro dos dados, utilizando o aprendizado não supervisionado. A seguir apresentamos de maneira breve um exemplo simples e os principais algoritmos para cada um dos tipos de problemas de ML que apresentamos na seção 2.3. *i.* Esses tipos de problemas e os algoritmos relacionados são resumidos na Figura 5.

Redução de dimensionalidade. Devido à grande abundância de dados, pode-se facilmente obter vetores de features de tamanho incrivelmente grandes, levando ao que é conhecido como

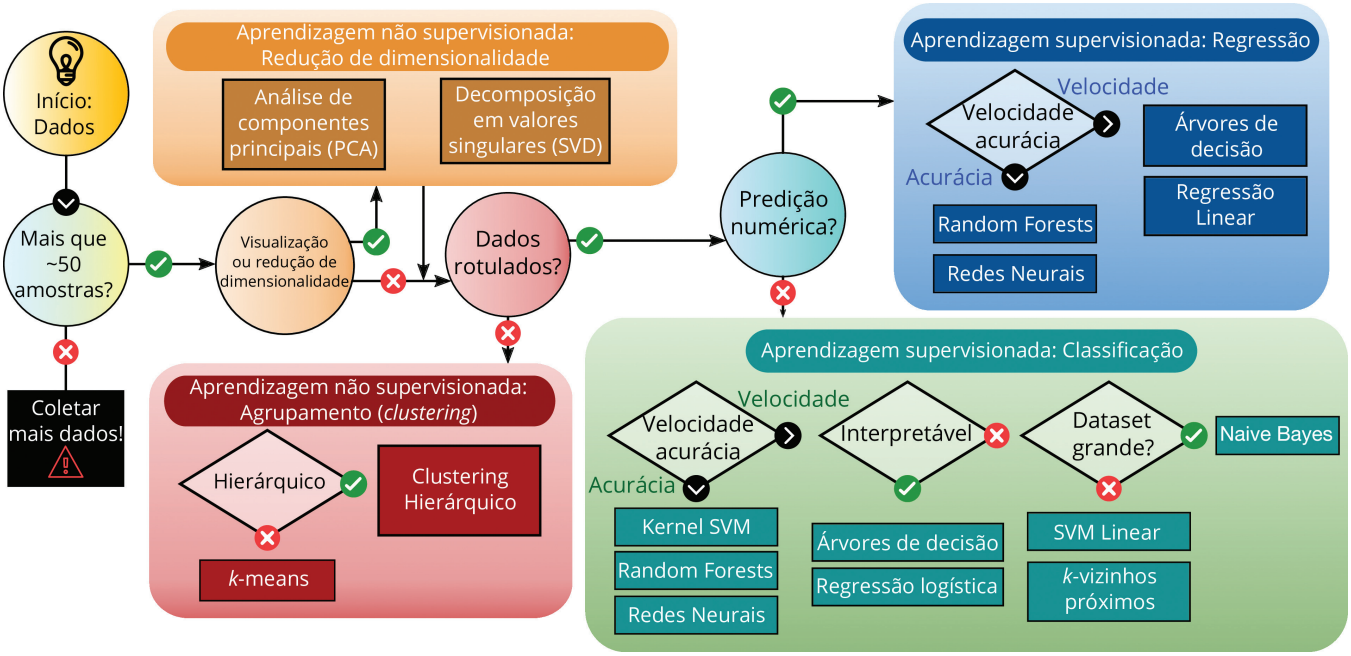

Figura 5: Algoritmos de machine learning e diagrama de uso, divididos nos principais tipos de problemas: aprendizado não supervisionado (redução de dimensionalidade e clustering) e supervisionado (classificação e regressão). Adaptado de [6] CC BY 3.0.

 DOI: https://doi.org/10.1590/1806-9126-RBEF-2020-0407

"maldição da dimensionalidade". Por exemplo, imagine um algoritmo de ML que recebe como entrada imagens de $n \times n$ pixels em escala de cinza, cada um representado como um valor numérico. Nesse caso, a matriz que contém esses números é achatada em um vetor de comprimento n^2, o vetor de características, descrevendo esse ponto (amostra) em um espaço de alta dimensionalidade. Devido à dependência exponencial, um número grande de dimensões é facilmente atingido para imagens de tamanho médio. A memória ou o poder de processamento computacional tornam-se fatores limitantes neste caso. Um ponto chave é que dentro da nuvem de dados de alta dimensão abrangida pelo conjunto de dados, pode-se encontrar uma estrutura de dimensão inferior. O conjunto de pontos pode ser projetado em um hiperplano ou variedade, reduzindo sua dimensionalidade enquanto preserva a maior parte das informações contidas na nuvem de dados original. Uma série de procedimentos com esse objetivo, como **análise de componentes principais** (PCA) são rotineiramente empregados em algoritmos de ML [32]. Em poucas palavras, a PCA é uma rotação de cada eixo do sistema de coordenadas do espaço onde residem os pontos de dados, levando à maximização da variância ao longo desses eixos. A maneira de descobrir para onde o novo eixo deve apontar é obtendo o autovetor correspondente ao maior autovalor de $\mathbf{X}^T\mathbf{X}$, onde $\mathbf{X}$ é a matriz de dados. Uma vez que o maior autovetor de variância, também conhecido como o componente principal, é encontrado, os pontos são projetados nele, resultando em uma compressão dos dados. Usualmente escolhe-se um número de componentes principais que irão descrever a maior parte da variância do conjunto de dados. A generalização dos algoritmos de redução de dimensionalidade para estruturas não-lineares é chamada de *manifold learning*, dos quais exemplos conhecidos são o *multi-dimensional scaling* (**MDS**), *isometric mapping* (**Isomap**) e *t-distributed stochastic neighbor embedding* (**t-SNE**).

Clustering. O **clustering hierárquico** é um método empregado na aprendizagem não supervisionada, podendo ser de dois tipos, aglomerativo ou divisivo. O primeiro pode ser descrito por um algoritmo simples: começando com n classes, ou clusters, cada um deles contendo um único exemplo $\mathbf{x}^{(i)}$ do conjunto de treinamento, e então é medida a dissimilaridade $d(A, B)$ entre pares de clusters rotulados A e B. Os dois clusters com a menor dissimilaridade, ou seja, mais semelhantes, são mesclados em um novo cluster. O processo é então repetido recursivamente até que apenas um cluster, contendo todos os elementos do conjunto de treinamento, permaneça. O processo pode ser melhor visualizado traçando um dendrograma, tal como mostrado na Figura 6. Para agrupar os dados em k clusters, $1 < k < n$, o usuário deve cortar a estrutura hierárquica obtida em alguma etapa intermediária do agrupamento. Há certa liberdade na escolha da medida de dissimilaridade $d(A, B)$, e três medidas principais são populares. Primeiro, a ligação única leva em consideração o par mais próximo de membros do cluster,

$$d_{SL}(A, B) = \min_{i \in A, j \in B} d_{ij} \tag{1}$$

onde d_{ij} é uma medida de dissimilaridade de membros do par. Em segundo lugar, a ligação completa considera o par mais distante ou mais diferente de cada cluster,

$$d_{CL}(A, B) = \max_{i \in A, j \in B} d_{ij} \tag{2}$$

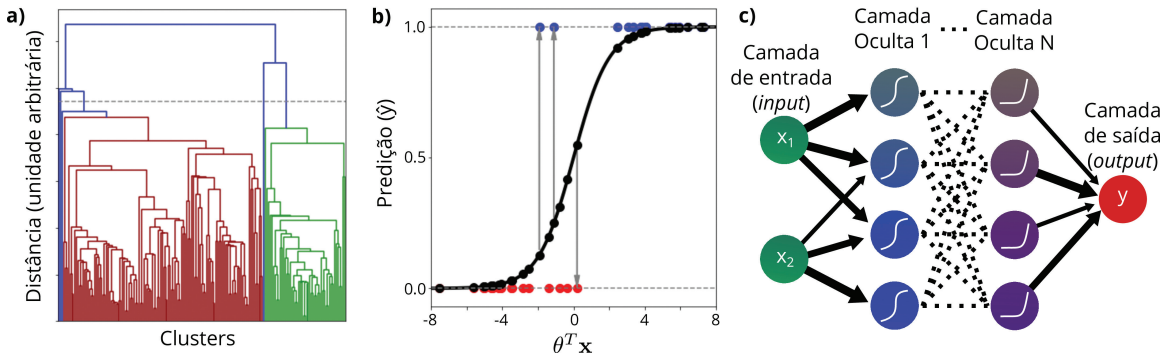

Figura 6: (a) Dendrograma demonstrando o clustering hierárquico. O código de cores é um guia para visualizar os clusters, representados pelas linhas verticais. As linhas horizontais indicam a fusão de dois clusters. O número de cruzamentos entre a linha horizontal e as linhas dos clusters corresponde ao número de clusters em uma determinada altura selecionada, no exemplo da linha tracejada cinza, são cinco clusters. (b) Exemplo da função sigmóide e a classificação de exemplos negativos em vermelho e positivos em azul na regressão logística. A seta cinza aponta para os dados classificados incorretamente no conjunto de dados. Adaptado de [6] CC BY 3.0. (c) Exemplo de uma rede neural com N camadas ocultas e um único neurônio na camada de output.

e, finalmente, o agrupamento da média do grupo considera a dissimilaridade média, representando um balanço entre as duas medidas anteriores,

$$d_{GA}(A,B) = \frac{1}{|A||B|}\sum_{i\in A}\sum_{j\in B} d_{ij}. \qquad (3)$$

A forma particular de d_{ij} também pode ser escolhida, geralmente sendo considerada a distância euclidiana para dados numéricos. A menos que os dados disponíveis sejam altamente agrupados, a escolha da medida de dissimilaridade pode resultar em dendrogramas distintos e, portanto, clusters distintos. Como o nome sugere, o clustering divisivo executa a operação oposta, começando com um único cluster contendo todos os exemplos do conjunto de dados e o divide recursivamente de forma que a dissimilaridade do cluster seja maximizada. O processo termina quando cada cluster possuir uma entrada. Da mesma forma, requer que o usuário determine a linha de corte para agrupar os dados. Outros algoritmos de clustering bastante utilizados são o K-médias (***K-means***) e o ***DBSCAN***.

Supervisionado: Regressão e Classificação. No caso dos algoritmos supervisionados, a ideia geral é aprender a função que aproxime da melhor forma possível a distribuição dos dados disponíveis para treinamento do modelo. No caso da regressão, o modelo retorna um valor contínuo, e no caso da classificação, um valor (rótulo) discreto. Seja qual for o algoritmo, o objetivo é minimizar o erro entre o valor predito pelo modelo aproximado e os valores de referência usados para o treinamento. Isso é feito ao se definir uma função de custo que será minimizada. Portanto, a infinidade de diferentes modelos possíveis pode ser resumida na escolha desses 2 componentes: qual a forma da função usada para a aproximação, e qual a função de custo para a minimização. Diferentes algoritmos empregam diferentes estratégias na solução desse objetivo. Vamos apresentar os exemplos mais simples pra cada uma dessas tarefas, como forma de exemplificar essa ideia geral.

No caso da regressão, o algoritmo mais simples e usado é conhecido como **regressão Linear**. Sua suposição básica é que os dados são normalmente distribuídos em relação a uma expressão ajustada,

$$\hat{y}^{(i)} = \theta^T \mathbf{x}^{(i)} \qquad (4)$$

onde o sobrescrito T denota o vetor transposto, $\hat{y}^{(i)}$ é o valor previsto e θ é o vetor de parâmetros (coeficientes) a serem aprendidos. A fim de obter os parâmetros θ, insere-se uma função de custo no modelo, que é dada por uma soma dos termos de erro usando mínimos quadrados,

$$\begin{aligned} J(\theta) &= \sum_{i=1}^{n} L[\hat{y}^{(i)}(\mathbf{x}^{(i)},\theta), y^{(i)}] \\ &= \frac{1}{2}\sum_{i=1}^{n}(\theta^T\mathbf{x}^{(i)} - y^{(i)})^2 + \lambda\|\theta\|_p. \end{aligned} \qquad (5)$$

Ao minimizar a função acima com relação a seus parâmetros, encontra-se o melhor conjunto de θ para o problema em questão, levando assim a um modelo de ML treinado. O último termo inserido na função de custo é opcional, conhecido como parâmetro de regularização λ, sendo diferentes extensões da regressão linear, tal como a regressão **ridge** ou **LASSO**. O valor de p denota a métrica, $p = 0$ é simplesmente o número de coeficientes diferentes de zero (normalmente não considerados uma métrica formalmente) em θ enquanto $p = 1$ é referido como a métrica de Manhattan ou táxi, e $p = 2$ é a métrica euclidiana usual. Quando se usa $p = 1$, o modelo de regressão é LASSO (Least Absolute Shrinkage and Selection Operator), onde devido à restrição imposta ao problema de minimização, nem todas as features presentes nos descritores são consideradas para o ajuste. Por outro lado, a regressão ridge corresponde a $p = 2$, e o resultado neste caso é apenas a redução dos valores absolutos das features, ou seja, features com valores muito grandes são penalizados, somando à função de custo. Tanto no LASSO quanto na regressão ridge, o parâmetro λ controla a complexidade do modelo, diminuindo e/ou selecionando as features. Assim, em ambos os casos, é recomendável começar com um modelo mais complexo e usar λ para diminuir sua complexidade. O parâmetro λ, entretanto, não pode ser aprendido da mesma maneira que θ, sendo referido como um hiperparâmetro que deve ser ajustado por, por exemplo, uma busca em grid para encontrar aquele que maximiza o poder de previsão sem introduzir muito viés (*bias*).

A classificação é usada para prever rótulos discretos. Um algoritmo de classificação muito popular é a **regressão logística** [20], que pode ser interpretado como um mapeamento das previsões feitas por regressão linear no intervalo [0, 1]. Vamos supor que a tarefa de classificação em questão é decidir se um dado ponto $\mathbf{x}^{(i)}$ pertence a uma classe particular ($y^{(i)} = 1$) ou não ($y^{(i)} = 0$). A previsão binária desejada pode ser obtida a partir de

$$\hat{y} = \sigma(\theta^T\mathbf{x}) = \frac{1}{1+e^{-\theta^T\mathbf{x}}} \qquad (6)$$

onde θ é novamente um vetor de parâmetros e σ é referido como a função logística ou sigmóide. Como

 DOI: https://doi.org/10.1590/1806-9126-RBEF-2020-0407

exemplo, a função sigmóide junto com uma previsão de um conjunto de dados fictício é apresentada na Figura 6. Normalmente considera-se que a amostra $\mathbf{x}^{(i)}$ pertence à classe rotulada por $y^{(i)}$ se $\hat{y}^{(i)} \geq 0.5$, mesmo que o rótulo previsto possa ser interpretado como uma probabilidade $\hat{y} = P(y = 1|\mathbf{x}, \theta)$. No caso da classificação, a função de custo é obtida a partir da log-probabilidade negativa. Assim, a obtenção dos melhores parâmetros θ requer a minimização dessa quantidade, dada por

$$J(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\Big[y^{(i)}\log(\hat{y}^{(i)}) + (1-y^{(i)})\log(1-\hat{y}^{(i)})\Big] \quad (7)$$

onde $y^{(i)}$ e $\hat{y}^{(i)} = \sigma(\theta^T\mathbf{x}^{(i)})$ são os rótulos binários verdadeiros (de referência) e previstos. Um parâmetro de regularização λ pode ser inserido na equação (7) com a mesma intenção de selecionar as features, como visto na regressão linear anteriormente. Observe que a regressão logística também pode ser usada quando os dados apresentam várias classes. Neste caso, deve-se empregar a estratégia um-contra-todos, que consiste em treinar n modelos de regressão logística, um para cada classe, e prever os rótulos usando o classificador que apresentar maior probabilidade.

Entre os métodos supervisionados, algoritmos populares de classificação são classificadores usando máquinas de vetores de suporte (***support vector machines**, SVM*) [33] com uma variedade de funções kernel (linear, polinomial, Gaussiana, entre outras); o algoritmo não paramétrico *k-nearest neighbors* (***k*-NN**); algoritmos probabilísticos como ***Naive Bayes***; **árvores de decisão** [34, 35] e métodos relacionados de *ensemble*, como **Random Forests** [20] e ***gradient boosting***. No caso de tarefas de regressão, regressão linear, Ridge e LASSO, bem como árvores de decisão e regressores de Random Forests são algoritmos simples e populares. Uma revisão recente sobre métodos de ML para ciência de materiais apresenta uma explicação abrangente do funcionamento interno de muitos desses algoritmos [6], e um vasto material também pode ser encontrado na literatura [18, 20, 21].

Vale destacar separadamente, as **redes neurais** (*neural networks, NN*), particularmente profundas (***deep learning***) [21], que estão se tornando muito populares devido aos seus avanços recentes tal como para reconhecimento de imagens, bem como em tarefas de processamento de linguagem natural. Elas correspondem a uma classe de algoritmos que foram, pelo menos em seu início, inspiradas pela estrutura do cérebro. Uma NN pode ser descrita como um grafo direcionado ponderado, ou seja, uma estrutura composta por camadas contendo unidades de processamento chamadas neurônios, que por sua vez são conectadas a outras camadas, conforme ilustrado na Figura 6. Muitos tipos de NNs são usados para uma variedade de tarefas como regressão e classificação, e algumas das arquiteturas mais populares para tais redes são NNs *feed-forward*, recorrentes e convolucionais. As principais diferenças entre essas arquiteturas são basicamente os padrões de conexão e as operações que seus neurônios executam nos dados. Normalmente em uma NN, uma camada de entrada recebe os vetores de features do conjunto de treinamento, e uma série de operações não lineares é realizada enquanto os dados propagam através das subsequentes chamadas *camadas ocultas*. Finalmente, o resultado do processamento é coletado na camada de saída, que pode ser uma classificação binária ou multinária (probabilística), ou mesmo um mapeamento contínuo como em um modelo de regressão linear. Um ponto muito interessante nesse processo, é que a própria rede aprende representações otimizadas do conjunto de dados, normalmente ao custo de cojuntos de dados maiores, visto o número exponencialmente grande de coeficientes a serem determinados. Em uma NN, a entrada $z_i^{(k)}$ do i-ésimo neurônio na k-ésima camada é uma função das saídas $y_j^{(k-1)}$ da camada anterior

$$z_i^{(k)} = \omega_{i0}^{(k)} + \sum_j y_j^{(k-1)}\omega_{ij}^{(k)} \quad (8)$$

onde $\omega_{ij}^{(k)}$ é o elemento da matriz que conecta as camadas adjacentes. O elemento $w_{i0}^{(k)}$ é conhecido como viés (*bias*), porque não faz parte da combinação linear de entradas. A entrada é então transformada por meio de uma função não-linear ou de ativação, tal como a tangente hiperbólica,

$$y_i^{(k)} = \frac{e^{z_i^{(k)}} - e^{-z_i^{(k)}}}{e^{z_i^{(k)}} + e^{-z_i^{(k)}}}, \quad (9)$$

que resulta no mapeamento do vetor de entrada da camada anterior em um novo espaço vetorial, permitindo que a rede forneça predições para problemas altamente complexos.

Por fim, vamos discutir como avaliar a qualidade dos modelos, métricas de desempenho e precauções a serem tomadas para gerar modelos coerentes. Um algoritmo de aprendizado supervisionado de ML é considerado treinado quando seus parâmetros ótimos dados as amostras de treinamento são encontrados, minimizando a função de custo. No entanto, os hiperparâmetros geralmente não podem ser aprendidos dessa maneira, e o estudo do desempenho do modelo em um conjunto separado, denominado conjunto de validação, em função de tais parâmetros é necessário.

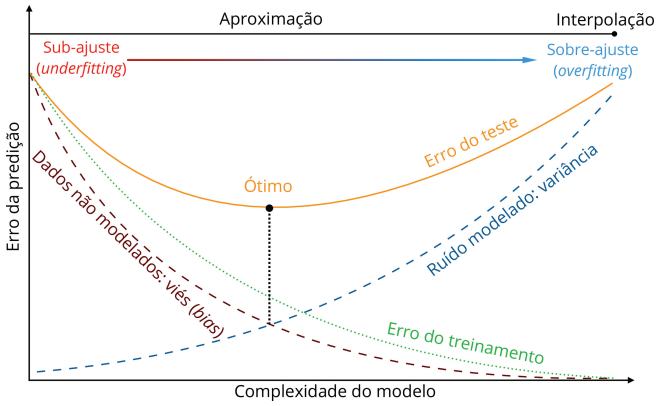

Figura 7: Balanço entre viés (*bias*) × variância. A complexidade ideal do modelo é avaliada em relação ao erro das predições avaliadas no conjunto de teste. Adaptado de [6] CC BY 3.0.

Este processo é conhecido como **validação**. A maneira usual de fazer isso é separar o conjunto de dados em 3 conjuntos separados: os conjuntos de treinamento, validação e teste. Espera-se que seus elementos sejam da mesma natureza, ou seja, venham da mesma distribuição estatística. O processo de aprendizagem é então realizado várias vezes para otimizar o modelo. Finalmente, usando o conjunto de teste, pode-se confrontar as previsões com os rótulos de referência e medir o quão bem o modelo está desempenhando. Particularmente em métodos supervisionados, dois problemas principais podem surgir então: *(i)* se os vetores de descritores apresentam um número insuficiente de features, ou seja, não é geral o suficiente para capturar as tendências nos dados e o modelo de regressão é considerado enviesado, e *(ii)* se o descritor apresenta muitas informações, o que faz com que o modelo de regressão se ajuste aos dados de treinamento excessivamente bem, mas sofre para generalizar para novos dados, então é dito que o modelo sofre sobre-ajuste (*overfitting*) ou variância. Esses são dois extremos da complexidade dos modelos, diretamente relacionado ao número de parâmetros, onde o equilíbrio ideal é representado na Figura 7. Normalmente se utiliza o parâmetro de regularização λ a fim de diminuir de forma sistemática a complexidade do modelo e encontrar o ponto ótimo.

Quando uma quantidade limitada de dados está disponível para treinamento, remover uma fração desse conjunto para criar o conjunto de teste pode impactar negativamente o processo de treinamento, e formas alternativas devem ser empregadas. Um dos métodos mais populares neste cenário é a **validação cruzada**, que consiste em particionar o conjunto de treinamento em k subconjuntos, treinar o modelo usando $k-1$ subconjuntos e validar o modelo treinado usando o conjunto que não foi usado para o treinamento. Este processo é executado k vezes e a média de cada etapa de validação é usada para calcular a média do desempenho,

$$E_{cv}^{K} = \frac{1}{K}\sum_{k=1}^{K}\sum_{i=1}^{n_k} L(\hat{y}_k^{(i)}, y^{(i)}) \qquad (10)$$

onde L é a função de perda e $\hat{y}_k^{(i)}$ é o rótulo previsto do i-ésimo exemplo de treinamento do modelo treinado usando o subconjunto dos dados de treinamento, excluindo o subconjunto k, que tem o tamanho n_k.

Existem muitas maneiras de avaliar o desempenho, sendo de suma importância particularmente para modelos supervisionados. Em tarefas de classificação binária ou multinária, é muito comum o uso de matrizes de confusão, onde o número de elementos preditos corretamente são apresentados nas entradas diagonais enquanto os elementos que foram preditos incorretamente são contados nas entradas fora da diagonal. Pode-se pensar no índice vertical como os rótulos reais e no índice horizontal como as previsões, e falsos (F) positivos (P) ou negativos (N) são previsões positivas para casos negativos e vice-versa, respectivamente. A curva de característica de operação do receptor (*ROC curve*) também é usada rotineiramente, sendo o gráfico da taxa de verdadeiros (V) positivos $TVP = \frac{VP}{VP+FN}$ versus a taxa de falsos positivos $TFP = \frac{FP}{FP+VN}$ variando o limiar interclasses. Um exemplo é mostrado na Fig. 11a.

No caso de tarefas de regressão, existem várias métricas da performance do ajuste. O erro médio absoluto $MAE = \frac{1}{n}\sum_i^n |y_i - \hat{y}_i|$, mede desvios na mesma unidade da variável e também não é sensível a *outliers*. Existe a versão normalizada expressa em porcentagem $MAPE = \frac{100\%}{n}\sum_i^n \frac{y_i-\hat{y}_i}{y_i}$. O erro quadrático médio $MSE = \frac{1}{n}\sum_i^n (y_i - \hat{y}_i)^2$ combina medições de *bias* e variância da previsão. Do ponto de vista frequentista, a estimativa $\hat{\theta}_m$ de um parâmetro de distribuição θ está intimamente relacionada com o MSE, através da fórmula $MSE = \mathbb{E}[(\hat{\theta}_m - \theta)^2] = Bias(\hat{\theta}_m)^2 + Var(\hat{\theta}_m)$. O MSE, ou seja, a função de custo dada na equação (5) (quando se introduz ou não um parâmetro de regularização λ), idealmente seria zero para pontos de dados exatamente em cima da função obtida por meio da regressão. O MSE costuma ser utilizado tomando sua raiz (RMSE), que recupera a unidade original, facilitando a interpretação da precisão do modelo. Finalmente, também é utilizado o coeficiente de determinação estatístico R^2, definido como $R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$, onde a soma total dos quadrados é $SS_{tot} = \sum_i (y_i - \bar{y})^2$ e a soma residual

 DOI: https://doi.org/10.1590/1806-9126-RBEF-2020-0407

dos quadrados é $SS_{res} = \sum_i (y_i - \hat{y}_i)^2$. O R^2 é usualmente visualizado em gráficos de paridade, onde compara-se os valores preditos pelo modelo de ML com os valores de referência. Um exemplo é mostrado na Fig. 11c.

Na prática, existem diversos *softwares* e programas computacionais que implementam os diferentes algoritmos de machine learning. Um dos mais acessíveis, completos e utilizados é o *scikit-learn* [36], implementado como uma biblioteca escrita em *python*. Uma implementação também acessível e com interface gráfica é o software *Weka* [37]. Duas das implementações mais utilizadas de *deep learning* são os códigos *tensorflow* [38] e *pytorch* [39]. Um tutorial prático introdutório de uso do machine learning com aplicações para materiais pode ser encontrado em [40].

Finalmente, vale destacar que existe um balanço entre os diferentes componentes para cada determinado problema de ML. O tamanho do conjunto de dados, a representação usada e o algoritmo a ser empregado estão intimamente relacionados à construção de cada modelo e devem ser balanceados com cuidado, conforme discutido na Fig. 8. Em relação aos dados, na física, química e ciência de materiais, a natureza dos conjuntos de dados envolvidos é muito diferente daqueles com os quais o aprendizado de máquina foi historicamente projetado para trabalhar, que são, tamanhos grandes, espaços com poucas features de dimensionalidade fixa e baixa variância; sendo então caracterizado como "*little-data*" [46]. Para uma complexidade de descritor fixa, o número de pontos de dados necessários para o treinamento do modelo é uma quantidade chave. Poucos exemplos podem levar ao sobreajuste, isto é, o modelo se ajusta excessivamente bem aos dados, incluindo ruídos indesejáveis. Aumentando o número de pontos de treinamento, tal problema é minimizado ao custo de diminuir ligeiramente a precisão do conjunto de treinamento (curva azul Fig. 8a). Por outro lado, isso leva a uma melhor generalização do modelo, o que pode ser representado pelo aumento da precisão dos dados que não estavam presentes na etapa de treinamento (dados de validação, curva vermelha crescente na Fig. 8a). Para cada modelo, o tamanho do conjunto de dados necessário para convergência será diferente, e não pode ser determinado*a priori*, mas avaliado por meio de curvas de aprendizado (Fig. 8a). Podemos pensar em uma escada de crescente complexidade na representação de materiais e moléculas (Fig. 8b), cada degrau fornecendo informações adicionais sobre os sistemas. No nível mais baixo, a informação depende apenas da fórmula química, ou seja, da composição elementar e estequiometria (escalares). No segundo nível, informações estruturais podem ser incluídas, como posições atômicas, conectividade e propriedades da rede. Nos níveis superiores, informações mais complexas, como estrutura eletrônica ou densidade local, podem ser introduzidas (vetores, tensores e outros). Para cada aplicação, é necessário incluir diferentes informações no descritor, de acordo com a natureza do problema. Portanto, para alguns problemas, features simples podem atingir a acurácia adequada, enquanto em outros problemas a acurácia é limitada. A tarefa da representação é otimizar o descritor, maximizando sua precisão ao mesmo tempo que mantém a maior simplicidade possível. Sempre que o grau de complexidade é aumentado, seja pela expansão do espaço de features ou pelo número de parâmetros a serem aprendidos no modelo de ML (Fig. 8c), a quantidade de dados de treinamento disponíveis deve aumentar de acordo. Finalmente, para qualquer algoritmo de ML,

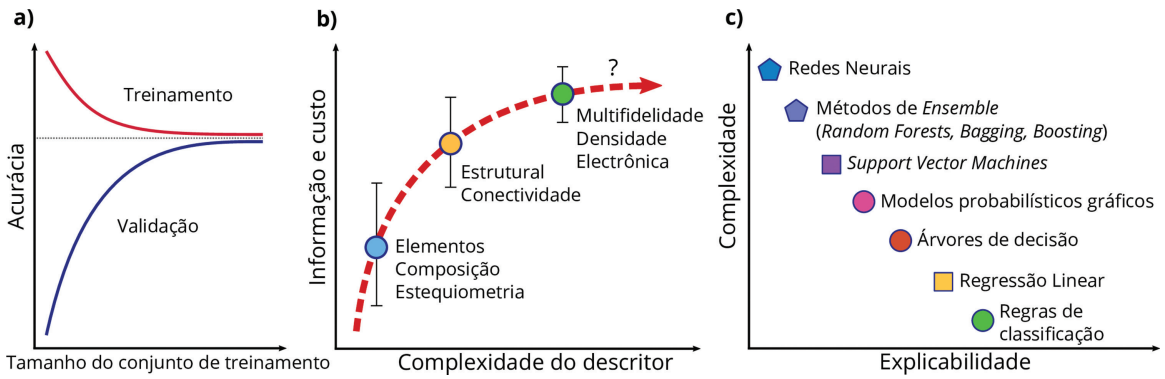

Figura 8: Detalhes dos componentes de Machine Learning: dados **a.**, Descritores **b.** e algoritmos **c.**. **a.** Curva de aprendizado para um determinado modelo de ML, onde a precisão aumenta com o número de dados de treinamento até atingir o platô da capacidade (linha horizontal pontilhada). Por outro lado, a precisão do treinamento diminui como resultado da redução do overfitting inicial. **b.** Representações de ML para moléculas e de materiais. Em cada degrau, mais informações são adicionadas ao descritor, tornando o treinamento e a previsão mais custosos. O desafio é maximizar a acurácia ao mesmo tempo usando o descritor mais simples possível. **c.** Balanço entre de precisão e interpretabilidade dos algoritmos de aprendizado de máquina. Algoritmos complexos como Redes Neurais tendem a ser considerados *caixas pretas* no sentido que entender a importância de cada feature não é uma tarefa simples. A quantidade de dados de treinamento disponíveis deve ser compatível a complexidade do modelo. Adaptado com permissão de [14]. Copyright 2019 American Chemical Society.

seu sucesso preditivo é determinado por um equilíbrio entre o conjunto de features disponíveis, a qualidade dos descritores, a otimização do algoritmo e, mais importante, a precisão dos dados usados para o treinamento [14]. Um primeiro passo para avaliação do sucesso do modelo de ML é verificar se o conjunto de features incluídas com o algoritmo utilizado é adequado para a descrição da propriedade-alvo de interesse. Isso pode ser verificado em relação às características estatísticas do próprio conjunto de dados, como o valor médio e desvio padrão da propriedade avaliada. Por exemplo, num problema de regressão, é possível comparar o erro médio obtido com um simples modelo de regressão linear usando as features iniciais com relação ao desvio padrão da propriedade. Se o resultado não for significativamente superior, pode ser um indicativo que um modelo linear não descreve bem a propriedade e/ou que as features incluídas não são suficientes para descrever o problema.

3. Aplicações em Materiais: Descoberta e Design

Como mostrado na seção anterior, o ML é uma ferramenta matemática e estatística, podendo então ser aplicada aos mais diferentes tipos de problemas. Nesse sentido, não apenas os problemas científicos em si podem ser estudados, como também tarefas que usualmente não são estudadas mas que existem dados disponíveis, tal como estimar o tempo que um processo computacional vai levar, permitindo sua otimização [47].

No contexto específico de materais, o ML pode ser usado para a descoberta, design e otimização de propriedades tanto partindo de dados experimentais [48–50] como de simulação, e esta pode ser atomística (clássica) ou *ab initio* (quântica). Diferentes estratégias podem ser utilizadas, podendo ser amplamente classificadas [13] no aprendizado de propriedades (Fig. 9a): *i)* pré-equação de Schrödinger, como aprender a densidade eletrônica [41], para ser então usada como input para cálculos DFT ou próprio ML; *ii)* acelerar ou substituir a resolução da equação de Schrödinger, criando assim aproximações para resolver o problema quântico [42]; e *iii)* pós-equação de Schrödinger, aprendendo diretamente as saídas do problema, tal como as propriedades observáveis dos materiais. Especificamente na área de simulação de materiais, onde é possível gerar uma grande quantidade de dados consistentes para serem usados em tarefas de ML, atualmente existem três grandes áreas de pesquisas de acordo com o grau de complexidade e aproximações utilizadas (Fig. 9b): ML de hamiltonianos de muitos corpos; ML atomístico, gerando potenciais interatômicos e propriedades relacionadas, e ML de materiais a partir de resultados de simulações de DFT.

Vamos apresentar dois exemplos representativos com diferentes níveis de aprendizado: *1)* exploração do espaço de fase de materiais, tanto atômico como configuracional, na descoberta e design de novos materiais bidimensionais (2D); *2)* descoberta de novas estruturas hipotéticas e/ou desconhecidas por otimização global no espaço configuracional, na busca e predição de estruturas de materiais usando potenciais atomísticos aprendidos por ML.

3.1. Explorando o espaço atômico, composicional e configuracional

Um dos grandes desafios da descoberta de materiais é descobrir o maior número possível de novos

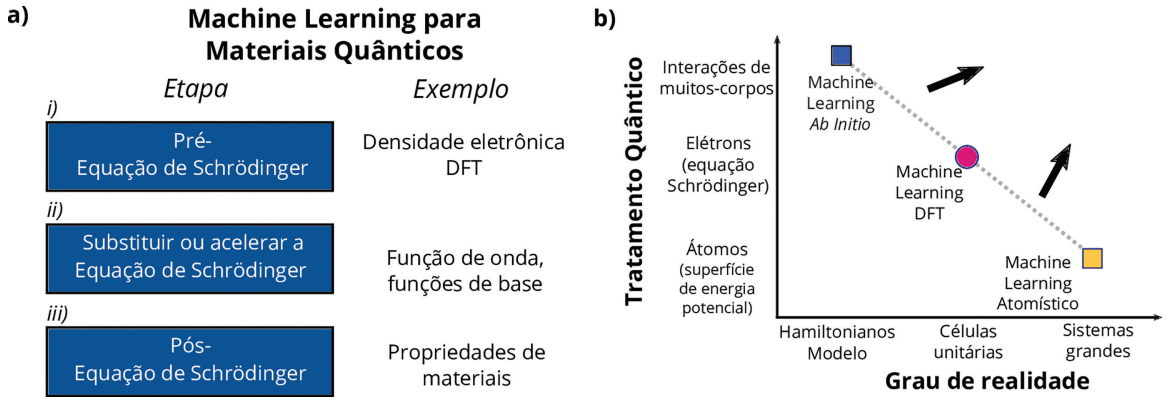

Figura 9: (a) Uso do machine learning em diferentes etapas de simulações de materiais: *i)* pré-equação de Schrödinger, por exemplo, aprendizado da densidade eletrônica obtida por DFT [41]; *ii)* substituindo [42] ou acelerando a equação de Schrödinger, tal como otimização bayesiana para relaxação da geometria, cálculos de barreiras energéticas [43] ou otimização global [44]; ou iii) pós-equação de Schrödinger, como o aprendizado direto das propriedades dos materiais [24]. (b) Áreas atuais de ML para ciência de materiais. O ML atomístico permite a exploração estrutural (otimização global) e propriedades, enquanto desconsidera os efeitos quânticos. O ML de hamiltonianos modelo permite a exploração de efeitos quânticos e de muitos corpos, embora é dificil a aplicação a sistemas de materiais reais. O ML de materiais ou DFT preenche a lacuna entre essas áreas [13], potenciais avanços são por exemplo a extração de hamiltonianos realistas a partir de cálculos DFT [45] e gerar funções de onda por ML de sistemas obtidos por simulações atomísticas [42]. Adaptado de [13] CC BY 3.0.

 DOI: https://doi.org/10.1590/1806-9126-RBEF-2020-0407

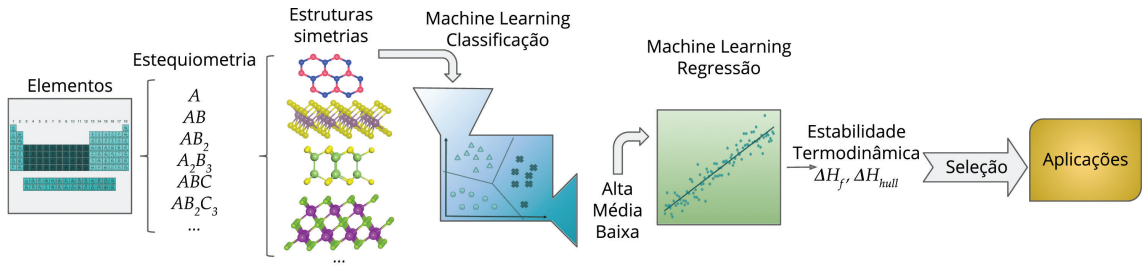

Figura 10: Abordagem baseada em dados para descoberta, design e seleção de materiais. Do grande número de combinações possíveis entre elementos, estequiometrias e simetrias, o espaço de materiais é gerado. Usando dados termodinâmicos disponíveis em bancos de dados, com técnicas de aprendizado de máquina, é possível classificar os materiais quanto à sua estabilidade termodinâmica. Se necessário, um modelo de regressão pode prever as energias de formação em relação à fases rivais. Finalmente, os materiais estáveis podem ser selecionados para potenciais aplicações. Adaptado com permissão de [24]. Copyright 2019 American Chemical Society.

materiais, explorando simultaneamente os espaços atômico (elementos), composicional (estequiometrias) e configuracional (geometrias/estruturas) [13]. Portanto, o número de graus de liberdade é imenso, e a estratégia utilizada na busca (amostragem) neste espaço é de extrema importância. Quando se pensa em design inverso (Fig. 2), ou seja, a busca por materiais que apresentem certas funcionalidades desejadas, a exploração desse grande espaço de materiais é fundamental [51]. Portanto, técnicas de alta eficiência, evolutivas e de machine learning são cada vez mais importantes para amostrar com eficiência o espaço dos materiais, resultando em casos selecionados interessantes para investigação aprofundada.

Na referência [24], os autores demonstraram como essa estratégia pode ser utilizada na descoberta de novos materiais 2D visando um posterior processo de filtragem e seleção de materiais para diversas aplicações. Um ponto fundamental que diferencia a aplicação de ML para a descoberta de materiais é que não há a informação *a priori* de qual será a geometria relaxada da estrutura a ser investigada, pois isso implicaria na necessidade de realização de simulações computacionais, justamente que espera-se que sejam substituídas pelo modelo preditivo. Dessa forma, a Fig. 10 apresenta a estratégia utilizada pelos autores. Tendo como ponto de partida apenas as propriedades atômicas e estequiométricas dos diferentes materiais, é possível prever em diferentes estruturas a estabilidade termodinâmica dos mesmos, que vai indicar a possibilidade de existência desse material de acordo com esse critério. Essa informação é o primeiro critério essencial num processo de filtragem e seleção de materiais. Tendo essa informação, é possível então investigar as diferentes propriedades dos materiais visando aplicações específicas.

Vamos descrever brevemente cada um dos quatro componentes do processo de machine learning (Fig. 4) para esse exemplo.

i) Definição do problema: o objetivo do estudo é a predição e compreensão da estabilidade termodinâmica dos diferentes materiais bidimensionais. Portanto, os autores trataram como um problema de apredizado supervisionado, em dois graus de detalhamento. Um modelo de classificação inicial para determinar se um material tem baixa, média ou alta estabilidade, baseado nos critérios de energia de formação (em relação às fases elementais) e em relação às fases rivais (chamada de *convex hull*). Posteriormente, um segundo modelo de regressão pode obter os valores numéricos dessas quantidades, se desejado.

ii) Dados: Foi utilizado o banco de dados computacional de materiais 2D *C2DB* [52], que possui cálculos baseados na teoria do funcional da densidade (DFT) de materiais 2D distribuídos entre diferentes protótipos comuns. Essas diferentes estruturas prototípicas são decoradas de maneira combinatória com diferentes átomos de toda a tabela periódica, resultando em um total de 3712 materiais 2D na versão utilizada. Para cada um dos materiais calculados, são armazenados os dados da geometria de menor energia após relaxação estrutural, energias total, de formação, em relação à outras fases, e diversas propriedades tal como eletrônicas, mecânicas, magnéticas, entre outras. Os autores utilizaram as energia de formação e de *convex hull* como o alvo dos modelos de machine learning, definindo três classes para a classificação: baixa, para materiais com energia de formação positiva (se decompõe em seus elementos); média, para materiais com energia de formação negativa, porém metaestável em relação à outras fases por uma diferença maior que 0.2 eV/átomo (dificilmente podendo ser estabilizada por diferentes estratégias); e alta estabilidade, para materiais com energia de formação negativa, e estáveis ou metaestáveis em relação à outras fases por uma diferença de até 0.2 eV/átomo (valores de energia compatíveis com materiais já sintetizados em substratos, por exemplo) [24].

iii) Representação: como dito acima, um ponto crucial é evitar a utilização de dados das posições

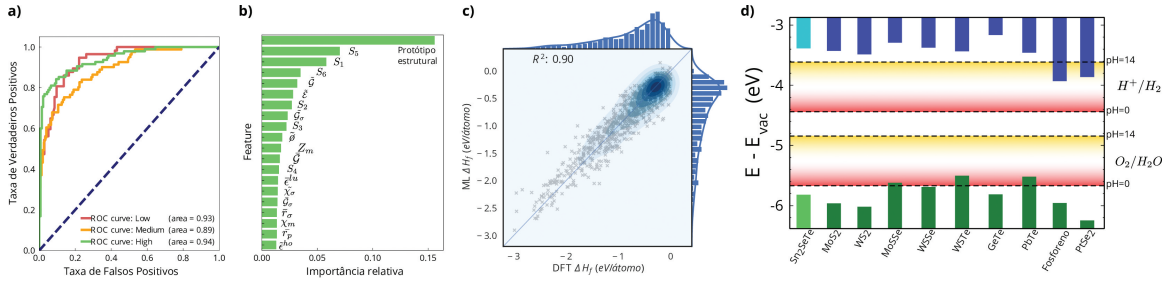

Figura 11: Resultados do modelo de ML e aplicação [24]. (a) Curva ROC e área sob a curva como métrica do modelo de classificação para cada classe: baixa (vermelho), média (laranja) e alta estabilidade (verde). (b) Importância das 20 features mais importantes do modelo de classificação. (c) Gráfico de paridade das energias de formação para a classe de alta estabilidade comparando os valores calculados por DFT com o modelo de regressão obtido via ML. O erro RMSE da validação cruzada é 0.205 eV/átomo. (d) Uso do modelo ML como critério de seleção de materiais para fotoeletrocatálise de água. Alinhamento de banda dos candidatos selecionados em relação ao vácuo. Os potenciais para as reações H^+/H_2 e O_2/H_2O estão destacados entre as linhas tracejadas. O candidato Sn_2SeTe foi previsto pelo modelo de classificação. Adaptado com permissão de [24]. Copyright 2019 American Chemical Society.

atômicas, que requereriam a execução de simulações computacionais custosas a serem feitas em larga escala. Dessa forma, para distinguir as estruturas, foi usado um rótulo categórico para denotar cada um dos prótótipos estruturais (por exemplo, estruturas idealizadas "tipo" grafeno, MoS_2, também conhecida como fase 2H, e assim sucessivamente). Adicionalmente, os autores utilizaram como features apenas dados de estequiometria e propriedades atômicas dos elementos que constituem os materiais, como número atômico, eletronegatividade, valência, autovalores de Kohn-Sham, polarizabilidade e raios atômicos. Dessa forma, nenhuma informação adicional é necessária para predição de novos materiais. Para tratar materiais com diferentes estequiometrias e número de espécies atômicas, por exemplo, A, AB_2, ABC_3, etc, em uma representação de mesma dimensionalidade (ou seja, número de colunas de seu vetor de features), os autores utilizaram diferentes valores estatísticos das propriedades atômicas, tal como o valor mínimo, máximo, médio e desvio padrão, como sugerido na referência [53].

iv) Algoritmos, validação, e aplicação: Tal como discutido no final de seção 2, a combinação de features incluídas com o algoritmo escolhido deve ser capaz de descrever satisfatoriamente a propriedade-alvo de interesse. Para esse exemplo, foi avaliado que o conjunto inicial de features não garante bons resultados independentemente da classe de algoritmo utilizado (linear ou não, paramétrico ou não). Portanto, um primeiro passo foi utilizar o algoritmo SISSO [54] (regressão linear com regularização de norma ℓ_0) para fazer um processo de engenharia de features de forma automatizada. Assumindo que a propriedade de interesse pode ser descrita por uma combinação não-linear das features iniciais, o algoritmo realiza a geração de novas features por meio de um processo combinatório das features inicias, ao realizar operações selecionadas entre elas, tal como soma/subtração, multiplicação/divisão, exponenciais/logaritmo, potências, etc. Associado com um algoritmo baseado em um ensemble de árvores de decisão, *gradient boosting*, os autores atingiram boa performance na classificação (em torno de 90%) tanto na validação do modelo realizada com validação cruzada, como no teste para dados inéditos, visualizada com a métrica da área sobre a curva ROC, mostrada na Fig. 11a. Uma informação interessante é a importância de cada feature para a capacidade preditiva do modelo (Fig. 11b), podendo ser usada para compreensão do problema. Adicionalmente, para cada uma das classe de estabilidade, os autores treinaram modelos de regressão usando o algoritmo SISSO para obtenção de valores aproximados da energia de formação, necessária para o cálculo da estabilidade em relação às fases rivais para decomposição. Na Figura 11c é mostrado um gráfico de paridade entre os valores previstos e os valores de referência para a classe dos materiais altamente estáveis, onde modelos com melhor performance se aproximam mais da diagonal ($R^2 = 1$). Esse modelo apresentou $R^2 = 0.9$, com um erro RMSE de 0.205 eV/átomo, que é em torno de três vezes inferior ao desvio padrão dos dados em si.

Tendo o modelo treinado, é então possível realizar a predição e descoberta de novos materiais até então não investigados. Os autores demonstraram essa capacidade ao gerar milhares de novas combinações inéditas em diferentes protótipos estruturais, satisfazendo regras de valência e neutralidade de carga, e então gerar a predição da estabilidade de cada um desses materiais. Para verificar se a predição de fato se mostrava acurada, realizaram cálculos DFT dessa propriedade para alguns materiais selecionados, e verificaram que em todos os

 DOI: https://doi.org/10.1590/1806-9126-RBEF-2020-0407

casos a predição se mostrou correta. Finalmente, demonstrando a aplicabilidade do modelo de estabilidade, os autores executaram um processo de filtragem e seleção de materiais para geração de hidrogênio a partir da quebra fotoeletrocatalítica de moléculas de água, como prova de conceito. Num processo de seleção de materiais, a estabilidade termodinâmica é o primeiro critério essencial para escolha de um material para qualquer aplicação. Aplicando o modelo treinado, e adicionalmente os filtros de gap de energia eletrônico entre 1.23 e 3 eV e alinhamento de banda de energia adequado para os processos de oxidação e redução da água, encontraram materiais 2D promissores para essa aplicação (Fig. 11d), do qual o material PbTe ainda não havia sido sugerido para essa aplicação na literatura. Adicionalmente, baseado na similaridade estrutural e química dos candidatos promissores, previram a estabilidade e calcularam a viabilidade do novo material Sn_2SeTe para essa aplicação [24].

3.2. Explorando o espaço configuracional

Similarmente ao processo de descoberta de materiais no espaço atômico, composicional e configuracional, há situações em que se é desejado manter uma dessas variáveis fixa, reduzindo assim a complexidade da busca. Por exemplo, pode-se desejar realizar a busca apenas em uma estrutura específica, eliminando assim a variável configuracional, ou buscar apenas materiais formados por determinados elementos, eliminando a variável atômica e realizando uma busca no espaço estrutural e composicional. Vamos apresentar um exemplo desse último caso.

A predição estrutural é um dos principais sucessos obtidos com métodos quânticos de primeiros princípios, visto que as propriedades dos materiais dependem sensivelmente de sua estrutura. As ferramentas de otimização global têm a capacidade de descobrir materiais para os quais existe pouca ou nenhuma informação empírica, sejam estruturas cristalinas, moleculares, defeitos, superfícies e interfaces [56]. O número possível de estruturas distintas é imenso mesmo para sistemas relativamente simples, da ordem de 10^N, onde N é o número de átomos na célula unitária [56]. Nesse contexto, é impossível investigar todas as combinações possíveis, e estratégias para amostrar eficientemente esse espaço de materiais é fundamental. O uso do ML nesse contexto é ideal. Em [55], os autores propõe o treinamento do zero e de forma automatizada de potenciais atomísticos de ML, sem nenhuma informação de quais estruturas são relevantes ou não. Essa forma de aprendizagem ativa converge e permite então a amostragem de grande parte do espaço configuracional de interesse, permitindo a descoberta de materiais de relevância real. A Fig. 12 apresenta a estratégia utilizada pelos autores.

Vamos descrever brevemente cada um dos quatro componentes do processo de machine learning (Fig. 4) para esse exemplo.

i) Definição do problema: o objetivo do estudo é a descoberta de materiais do zero e automatizada, explorando o espaço estrutural de forma iterativa ao treinar potenciais atomísticos baseados em ML. Portanto, nesse caso a tarefa de ML é um problema de apredizado supervisionado, onde é feita a regressão dos parâmetros otimizados de potenciais interatômicos que irão descrever as interações dos materiais.

ii) Dados: a proposta é a geração ativa e iterativa dos dados durante o treinamento do modelo de ML, como mostrado na Fig. 12. Os dados são o conjunto de estruturas e energias e forças atômicas associadas à cada uma delas, obtidas com cálculos de DFT. Inicialmente é gerado um conjunto de estruturas aleatórias, e selecionado um subconjunto de acordo com a maior diversidade estrutural para serem calculadas. Com essas informações, um potencial de ML é treinado, e o processo se repete, porém agora as estruturas aleatórias são relaxadas com o potencial obtido antes da seleção do subconjunto a ser calculado. O processo iterativo é executado até obter resultados de energias e forças satisfatórios, no estudo em torno de 2500 cálculos fixos foram necessários [55].

ii) Representação: o treinamento de um potencial interatômico de ML busca a descrição da superfície de energia potencial (PES) dos sistemas em função das posições dos átomos. Dessa forma, a representação detalhada dos ambientes atômicos é de extrema importância. Os autores utilizam a representação chamada de sobreposição suave de posições atômicas (SOAP) [57], em que a densidade de vizinhos de um átomo i é expandida em uma base local de funções radiais g_n e harmônicos esféricos Y_{lm}:

$$\rho_i(\mathbf{R}) = \sum_j exp(-|r - r_{ij}|^2/2\sigma_{at}^2) \quad (11)$$

$$= \sum_{nlm} c_{nlm}^{(i)} g_n(r) Y_{lm}(\hat{r}), \quad (12)$$

onde j são os vizinhos do átomo dentro de um raio de corte especificado.

iii) Algoritmos, validação, e aplicação: os autores treinam potenciais de aproximações Gaussianas (GAP) [58], que combinam o descritor SOAP apresentado acima com a regressão kernel ridge, obtendo as contribuições atômicas da energia total: $\varepsilon(\mathbf{q}) = \sum_{k=1}^{N} \alpha_k K(\mathbf{q}, \mathbf{q}^k)$, onde $\mathbf{q}$ e $\mathbf{q}^k$ são o descritor do ambiente local do átomo de referência e de outro átomo, N é o número de diferentes configurações de treinamento, indexadas por k. A tarefa de ML que vai determinar o vetor de

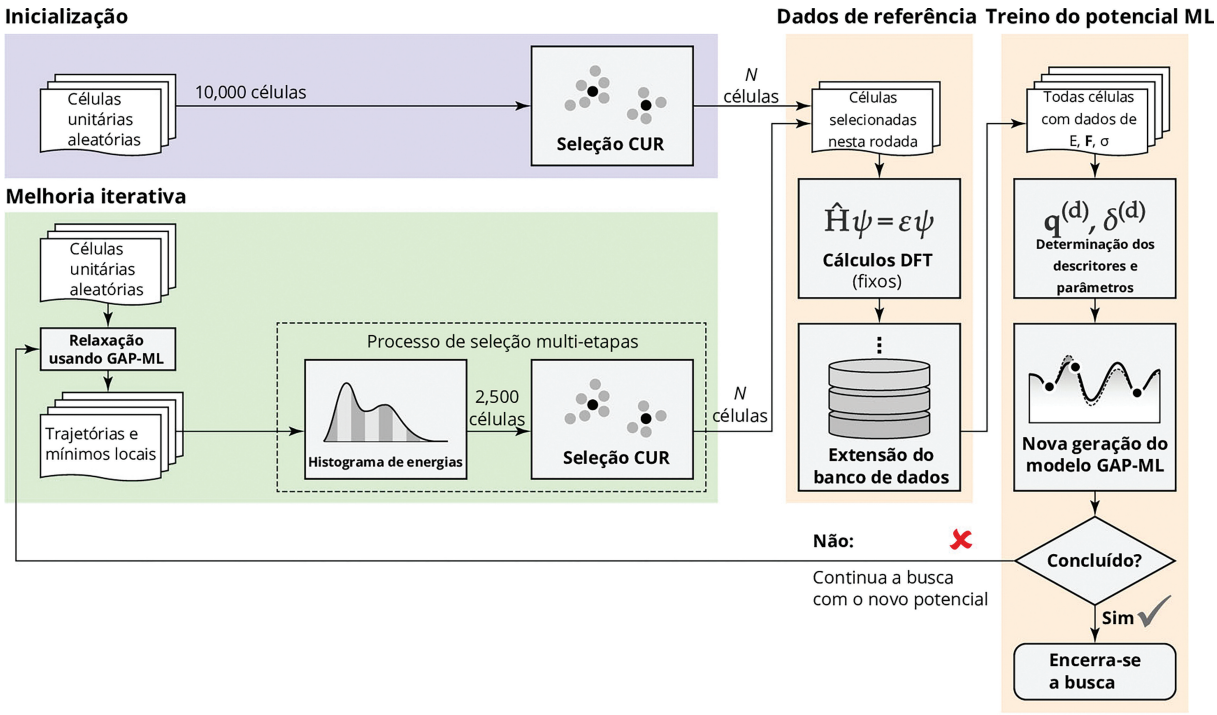

Figura 12: Protocolo automatizado que explora iterativamente o espaço estrutural e treina potenciais interatômicos baseados em ML [55]. De um conjunto aleatório inicial de células unitárias (azul), são selecionadas as geometricamente mais diversas usando o algoritmo CUR. As estruturas selecionadas são calculadas com DFT e usadas para treinar um potencial de ML (GAP) (laranja). Esse potencial é usado para relaxar um novo conjunto de estruturas aleatórias (verde), selecionando novamente os casos mais relevantes e repetindo o ciclo até a convergência, encerrando assim a busca. Adaptado de [55] CC BY 4.0.

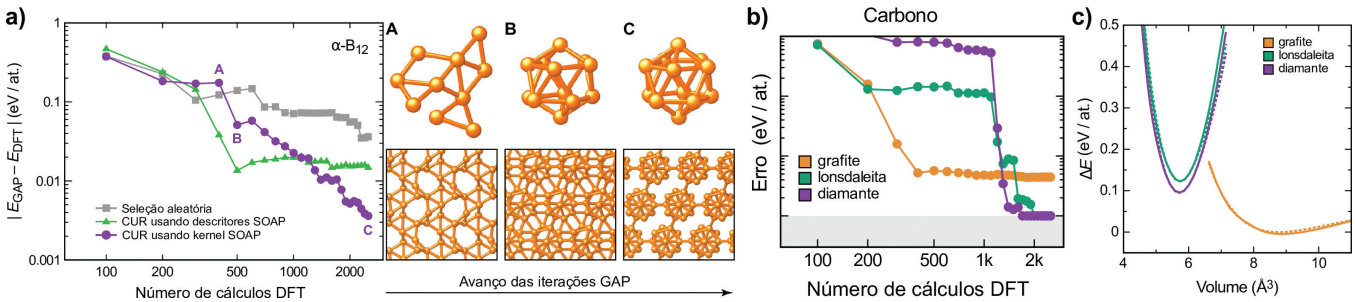

Figura 13: Resultados do modelo de ML e aplicação [55]. (a) Aprendizagem da estrutura cristalina do boro α-romboédrico. Esquerda: erro do modelo GAP gerado iterativamente, para a estrutura otimizada de menor energia do $\alpha - B_{12}$, em relação ao DFT. São comparadas a seleção aleatória de pontos (cinza), o procedimento de seleção usando CUR com descritores SOAP (verde), e o mesmo mas usando CUR com kernels SOAP (roxo). Direita: Evolução do icosaedro B_{12} como o principal fragmento estrutural, para três pontos nos N ciclos, 400 (A), 500 (B) e 2500 (C) cálculos DFT no total. A respectiva estrutura de menor energia da iteração é mostrada. (b) Aprendizagem de diversas estruturas cristalinas de carbono sem conhecimento prévio. Erro entre as energias calculadas por DFT e GAP para as estruturas otimizadas com cada método. (c) Curvas de energia-volume calculadas com o modelo GAP final (linhas sólidas) e a referência DFT (linhas tracejadas). Energias em relação à menor energia obtida com DFT. Adaptado de [55] CC BY 4.0.

coeficientes α, e K é uma função não-linear fixa, chamada de *kernel*, que tem como objetivo medir a similaridade entre os ambientes atômicos de seus dois argumentos. O kernel SOAP é mostrado na equação (11).

Os autores aplicam a estratégia descrita para descoberta de alótropos em diferentes sistemas. Um caso desafiador de ser capturado é o boro, um dos elementos estruturalmente mais complexos. O protocolo é capaz de descobrir a estrutura do $\alpha - B_{12}$ de forma autônoma, como mostrado na Fig. 13a. Percebe-se que o protocolo atinge alta acurácia permitindo a descoberta de materiais sem intervenção do cientista. O método não se limita a um sistema específico, por exemplo, consegue

 DOI: https://doi.org/10.1590/1806-9126-RBEF-2020-0407

aprender corretamente a estrutura e energias relativas de diferentes alótropos de carbono com alta precisão (Fig. 13b-c).

Resultados similares são obtidos para diferentes sistemas isolantes, semicondutores e metálicos [55], permitindo a obtenção de importante informação estrutural na descoberta de materiais.

4. Conclusão: Futuro e Perspectivas

O uso de estratégias baseadas em dados na ciência de materiais e áreas correlatas ainda é muito recente, porém nesse curto período já demonstrou ser capaz de solucionar grandes desafios. Cada vez mais as técnicas de machine learning serão utilizadas. Vislumbramos que estas farão parte do cotidiano das novas gerações de pesquisadores. Como assinalado por Meredig [59], "*o machine learning não vai substituir os cientistas, mas os cientistas que usam machine learning vão substituir os que não usam*". Sendo ferramentas estatísticas, a compreensão das condições necessárias e favoráveis para sua aplicação são fundamentais, assim como as limitações técnicas que prejudicam e eventualmente impedem seu uso. O crescente potencial e resultados obtidos nessa área favorecem cada vez mais o compartilhamento e disseminação dos dados e sua procedência, em direção à uma ciência mais verificável, reprodutível e robusta. Os desafios e possibilidades que estão por vir serão determinados pela criatividade dos cientistas, que ao contrário do que possa parecer, não serão substituídos pelas máquinas. É um período animador para usar, desenvolver e compartilhar as ferramentas da ciência de dados e inteligência artificial na interseção com os conhecimentos específicos das á reas como física, química e ciência de materiais.

Agradecimentos

Os autores têm apoio da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), projetos 17/18139-6 e 17/02317-2.

Referências

[1] T. Kuhn, *The Structure of Scientific Revolutions* (University of Chicago Press, Chicago, 1962).

[2] F.J. Dyson, Science **338**, 1426 (2012).

[3] P. Galison, *Image and logic: A material culture of microphysics* (University of Chicago Press, Chicago, 1997).

[4] F.J. Dyson, *Imagined worlds* (Harvard University Press, Massachusetts, 1997).

[5] T. Hey, S. Tansley e K. Tolle, em: *The Fourth Paradigm: Data-Intensive Scientific Discovery*, editado por T. Hey (Microsoft Research, Redmond, 2009).

[6] G.R. Schleder, A.C.M. Padilha, C.M. Acosta, M. Costa e A. Fazzio, Journal of Physics: Materials **2**, 032001 (2019).

[7] A. Agrawal e A. Choudhary, APL Materials **4**, 053208 (2016).

[8] S. Curtarolo, G.L.W. Hart, M.B. Nardelli, N. Mingo, S. Sanvito e O. Levy, Nature Materials **12**, 191 (2013).

[9] A. Jain, K.A. Persson e G. Ceder, APL Materials **4**, 053102 (2016).

[10] C.L. Magee, Complexity **18**, 10 (2012).

[11] P. Gribbon e S. Andreas, Drug Discovery Today **10**, 17 (2005).

[12] D.A. Pereira e J.A. Williams, British Journal of Pharmacology **152**, 53 (2009).

[13] F. Giustino, M. Bibes, J.H. Lee, F. Trier, R. Valentí, S.M. Winter, Y.W. Son, L. Taillefer, C. Heil, A.I. Figueroa et al., Journal of Physics: Materials, https://doi.org/10.1088/2515-7639/abb74e (2020).

[14] G.R. Schleder, A.C.M. Padilha, A.R. Rocha, G.M. Dalpian e A. Fazzio, Journal of Chemical Information and Modeling **60**, 452 (2020).

[15] J. Glick, em: *Ontologies and Databases—Knowledge Engineering for Materials Informatics*, editado por K. Rajan (Elsevier, Amsterdã, 2013).

[16] E.P. Wigner, Communications on Pure and Applied Mathematics **13**, 1 (1960).

[17] A. Halevy, P. Norvig e F. Pereira, IEEE Intelligent Systems **24**, 8 (2009).

[18] K.P. Murphy, *Machine Learning: A Probabilistic Perspective* (MIT Press, Cambridge, 2012).

[19] A.L. Samuel, IBM Journal of Research and Development **3**, 210 (1959).

[20] T. Hastie, R. Tibshirani e J. Friedman, *The Elements of Statistical Learning* (Springer, New York, 2001).

[21] I. Goodfellow, Y. Bengio e A. Courville, *Deep Learning*, disponível em: http://www.deeplearningbook.org.

[22] B. Sun, M. Fernandez e A.S. Barnard, Nanoscale Horizons **1**, 89 (2016).

[23] R. Ramprasad, R. Batra, G. Pilania, A. Mannodi-Kanakkithodi e C. Kim, npj Computational Materials **3**, 54 (2017).

[24] G.R. Schleder, C.M. Acosta e A. Fazzio, ACS Applied Materials & Interfaces **12**, 20149 (2020).

[25] S.W. Knox, *Machine Learning* (John Wiley & Sons, Hoboken, 2018).

[26] M.L. Hutchinson, E. Antono, B.M. Gibbons, S. Paradiso, J. Ling e B. Meredig, arXiv:1711.05099 (2017).

[27] L.M. Ghiringhelli, J. Vybiral, S.V. Levchenko, C. Draxl e M. Scheffler, Physical Review Letters **114**, 105503 (2015).

[28] L. Ward e C. Wolverton, Current Opinion in Solid State and Materials Science **21**, 167 (2016).

[29] P. Domingos, Commun. ACM **55**, 78 (2012).

[30] D.H. Wolpert e W.G. Macready, Mach. Learn. **20**, 273 (1995).

[31] D.H. Wolpert, Neural Comput. **8**, 1341 (1996).

[32] M. van Heel, R.V. Portugal e M. Schatz, Open J. Stat. **6**, 701 (2016).

[33] C. Cortes e V. Vapnik, Mach. Learn. **20**, 273 (1995).

[34] R. Kohavi e R. Quinlan, *Decision Tree Discovery*, disponível em: http://ai.stanford.edu/∼ronnyk/treesHB.pdf.

[35] J.R. Quinlan, *C4.5: Programs for Machine Learning* (Morgan Kaufmann Publishers Inc., San Francisco, 1993).

[36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg et al., *Journal of Machine Learning Research* **12**, 2825 (2011).

[37] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann e I.H. Witten, ACM SIGKDD explorations newsletter **11**, 10 (2009).

[38] tensorflow.org.

[39] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga et al., arXiv:1912.01703 (2019).

[40] G.R. Schleder, gschleder/MLtutorial-v1.0.0.zip (2019), v. 1, disponível em: https://doi.org/10.5281/zenodo.4041648.

[41] A. Chandrasekaran, D. Kamal, R. Batra, C. Kim, L. Chen e R. Ramprasad, npj Computational Materials **5**, 22 (2019).

[42] K.T. Schütt, M. Gastegger, A. Tkatchenko, K.R. Müller e R.J. Maurer, Nature Communications **10**, 5024 (2019).

[43] J.A. Garrido Torres, P.C. Jennings, M.H. Hansen, J.R. Boes e T. Bligaard, Physical Review Letters **122**, 156001 (2019).

[44] M.H. Hansen, J.A.G. Torres, P.C. Jennings, Z. Wang, J.R. Boes, O.G. Mamun e T. Bligaard, arXiv:1904.00904 (2019).

[45] M. Costa, G.R. Schleder, M.B. Nardelli, C. Lewenkopf e A. Fazzio, Nano Letters **19**, 8941 (2019).

[46] A.S. Barnard, B. Motevalli, A.J. Parker, J.M. Fischer, C.A. Feigl e G. Opletal, Nanoscale, **11**, 19190 (2019).

[47] S. Heinen, M. Schwilk, G.F. von Rudorff e O. Anatole von Lilienfeld, Machine Learning: Science and Technology **1**, 025002, 2020.

[48] G.S. Silva, L.P. Oliveira, G.F. Costa, G.F. Giordano, C.Y.N. Nicoliche, A.A. Silva, L.U. Khan, G.H. Silva, A.L. Gobbi, J.V. Silveira et al., Sensors and Actuators B: Chemical **305**, 127482 (2020).

[49] C.Y.N. Nicoliche, R.A.G. Oliveira, G.S. Silva, L.F. Ferreira, I.L. Rodrigues, R.C. Faria, A. Fazzio, E. Carrilho, L.G. Pontes, G.R. Schleder et al., ACS Sensors **5**, 1864 (2020).

[50] G.F. Giordano, L.C.S. Vieira, A.O. Gomes, R.M. Carvalho, L.T. Kubota, A. Fazzio, G.R. Schleder, A.L. Gobbi e R.S. Lima, Fuel **285**, 119072 (2021).

[51] C. Mera Acosta, E. Ogoshi, A. Fazzio, G.M. Dalpian e A. Zunger, Matter **3**, 145 (2020).

[52] S. Haastrup, M. Strange, M. Pandey, T. Deilmann, P.S. Schmidt, N.F. Hinsche, M.N. Gjerding, D. Torelli, P.M. Larsen, A.C. Riis-Jensen et al., 2D Materials **5**, 042002 (2018).

[53] L. Ward, A. Agrawal, A. Choudhary e Christopher Wolverton, npj Computational Materials **2**, 16028 (2016).

[54] R. Ouyang, S. Curtarolo, E. Ahmetcik, M. Scheffler e L.M. Ghiringhelli, Physical Review Materials **2**, 083802 (2018).

[55] N. Bernstein, G. Csányi e V.L. Deringer, npj Computational Materials **5**, 99 (2019).

[56] A.R. Oganov, C.J. Pickard, Q. Zhu e R.J. Needs, Nature Reviews Materials **4**, 331 (2019).

[57] A.P. Bartók, R. Kondor e G. Csányi, Physical Review B **87**, 184115 (2013).

[58] A.P. Bartók, M.C. Payne, R. Kondor e G. Csányi, Physical Review Letters **104**, 136403 (2010).

[59] B. Meredig, Chemistry of Materials **31**, 9579 (2019).