# Relatório – Projeto previsão Líquidos Iônicos

*Katarina da Silva Vilarins, Marco Túlio Lima Rodrigues, Pedro Henrique Medeiros Bramante*
*Ilum, Escola de Ciência, Centro Nacional de Pesquisa em Energia e Materiais (CNPEM), Brasil.*

**RESUMO**

O documento apresenta os procedimentos adotados na elaboração de um projeto focado nas características de líquidos iônicos, utilizando como referência um banco de dados específico. Além disso, tem como objetivo documentar os processos de desenvolvimento de códigos e o uso de bibliotecas que facilitam a visualização dos resultados. Para isso, serão empregados o Python Notebook e o banco de dados da UNT Digital Library. Utilizou-se o modelo de regressão de floresta de decisões e obteve-se uma previsão satisfatória de um target selecionado acerca dos líquidos iônicos.

## INTRODUÇÃO

A atividade foi desenvolvida na disciplina de Aprendizado de Máquina na instituição Ilum escola de ciências com a orientação do docente Daniel Roberto Cassar e com o banco de dados indicado pelo docente Leandro das Merces Silva. Esse relatório descreve a atividade de leitura de uma dataset que contém medidas de características de líquidos iônicos anteriormente catalogados. Na prática foi feito o uso da linguagem em Python, com o intuito de, tendo alguns atributos informados, possibilitar a previsão de um target. Esse resultado é possibilitado pela interpretação das informações fornecidas por meio de um modelo utilizado que consiga traçar e entender os padrões entre as informações do material fornecidas.

## METODOLOGIA

Primeiramente, foi necessário identificar bancos de dados que atendessem aos pré-requisitos iniciais do projeto, incluindo a presença de pelo menos quatro atributos e a previsão de pelo menos um target. Após selecionar um banco de dados adequado, o próximo passo foi determinar quais seriam os atributos e o target. Para isso, realizamos uma pesquisa para identificar as características que apresentavam maior dificuldade de obtenção, optando por utilizá-las como target. Essa escolha visava garantir que o algoritmo desenvolvido pelo grupo tivesse uma aplicação lógica e vantajosa para o usuário. As informações disponíveis sobre os materiais eram organizadas em colunas com os seguintes nomes: E, S, A, B e V.

- **E:** É a medição que representa a capacidade do líquido iônico de interagir com solutos através de forças de dispersão. Em termos práticos, um valor mais alto de E indica que o líquido iônico é mais propenso a dissolver ou interagir com moléculas não polares, como hidrocarbonetos. Isso é importante em aplicações como extração de compostos orgânicos de misturas ou na dissolução de catalisadores em reações químicas.

- **S:** Essa medição reflete a dipolaridade e a polarizabilidade do líquido iônico. Em outras palavras, indica a capacidade do líquido iônico de interagir com moléculas polares através de forças dipolo-dipolo ou dipolo-induzido. Um valor alto de S sugere que o líquido iônico é bom em dissolver sais, açúcares e outras moléculas polares. Isso é relevante em

processos como a dissolução de biomassa para produção de biocombustíveis ou na síntese de produtos farmacêuticos.

- **A:** Essa medição quantifica a acidez do líquido iônico, ou seja, sua capacidade de doar prótons (íons H+). Um valor alto de A indica um líquido iônico mais ácido, o que pode ser útil em catálise ácida, como na conversão de biomassa em produtos químicos de plataforma. No entanto, alta acidez também pode levar à corrosão de alguns materiais, o que deve ser considerado na escolha do líquido iônico para uma aplicação específica.

- **B:** Essa medição está relacionada à energia necessária para criar uma cavidade no líquido iônico para acomodar uma molécula de gás. Um valor alto de B significa que é preciso mais energia para criar essa cavidade, o que pode influenciar a solubilidade de gases no líquido iônico. Isso é importante em aplicações como a captura de CO2 ou na separação de gases.

- **V:** Essa medição é semelhante a B, mas leva em consideração tanto a energia para criar a cavidade quanto as interações de dispersão entre o gás e o líquido iônico. Um valor alto de V indica que o líquido iônico tem maior capacidade de dissolver gases, o que é crucial em processos como a absorção de gases ácidos ou na utilização de gases como reagentes em reações químicas.

Selecionamos a coluna E como target pois é uma das medidas mais difíceis de ser obtida, consequentemente faz sentido que um futuro usuário queira fazer uso do algoritmo elaborado para conseguir obter um valor próximo dessa medição sem necessidade de realmente medir essa propriedade. As demais medições servem como os atributos necessários para prever o target.

Então, deu-se início a elaboração do algoritmo que desempenharia a tarefa pensada pelo grupo. Trataram-se os dados e retirou-se os "Nans" para ser possível então utilizá-los para os passos seguintes. Para uma melhor compreensão, Nans são lacunas que não foram preenchidas por dados no dataset, esse procedimento de retirada dessas lacunas reduziu o número de linhas que eram levadas em questão para o cálculo do target, contudo essa redução não foi significativa de maneira a não implicar em grandes perdas de informação.

Houve então a etapa de divisão dos dados em dois conjuntos, os chamados treino e teste. A diferença entre eles é a seguinte, o de treino será utilizado para ajustar o modelo, isto é, ele aprende os padrões e as relações entre as variáveis a partir desse conjunto e então realiza o ajuste dos parâmetros do modelo visando uma minimização do erro durante a previsão. Enquanto o conjunto de teste é reservado para avaliar o desempenho do modelo em dados que ele não viu durante o treinamento, dessa forma ele consegue medir a capacidade do modelo de generalizar para novos dados. A divisão é controlada pelo tamanho do teste (10%) e pela semente de aleatoriedade.

No caso desse projeto, não foi necessária a etapa de normalização já que o modelo de Floresta Aleatória não é muito sensível a outliers. Apesar de saber dessa vantagem do modelo, no procedimento seguinte no processo de elaboração do código foi analisada a distribuição dos dados numéricos a serem utilizados para observar se havia um número elevado de outliers, concluiu-se que na verdade a distribuição seguia um padrão muito próximo ao normal, reforçando a inexistência da necessidade de uma normalização desses dados.

O próximo passo é definir uma função que cria um modelo de Random Forest Regressor com hiperparâmetros ajustáveis, usando o Optuna, uma biblioteca de otimização. Essa função sugere diferentes valores para os hiperparâmetros e, em seguida, cria uma instância do RandomForestRegressor com esses parâmetros. Por fim, a função retorna o modelo configurado, que será utilizado na otimização.

Após essa etapa, inicia-se uma sequência de comandos relacionados ao funcionamento do Optuna, essencial para otimizar os hiperparâmetros. Começamos com a elaboração de uma função que cria um objetivo de avaliação para o Optuna, focando na redução da média do RMSE nas divisões da validação cruzada para avaliar o desempenho. O estudo utiliza um banco de dados SQLite para armazenar e carregar os resultados das tentativas de otimização, permitindo que o processo seja salvo e retomado posteriormente. Para visualizar esse processo, é possível conferir o código no notebook disponível na área de arquivos do GitHub.

Em seguida, precisamos definir o número de trials (tentativas) que serão realizadas durante o processo de otimização com o Optuna. Neste contexto, será realizado um total de 100 tentativas para otimizar os hiperparâmetros, testando várias combinações para encontrar a configuração que minimiza a métrica de erro, como o RMSE. A função denominada optimize() é responsável por executar o processo de otimização, chamando repetidamente a função objetivo (funcao_objetivo_parcial) definida anteriormente para cada tentativa.

Após isso, ainda foram testadas mais dusa formas de otimização para verificar qual seria mais vantajosa a ser utilizada, nos levando a concluir que o optuna se destaca com seus resultados e por isso foi favorecido como o otimizador de hiperparâmetros de melhor desenvolvimento e por isso é o que é adotado definitivamente para o desempenho do algoritmo.

Então, treinou-se o modelo com os melhores parâmetros e fez-se as previsões obtendo os resultados disponíveis no notebook e discutidos na área de discussão do relatório presente. No notebook ainda foi realizada a plotagem de gráficos que permitem observar mais facilmente a proximidade do que foi previsto do que é real, demonstrando uma aproximação visivelmente boa, porém conta com alguns itens distantes levando-nos a uma especulação de possíveis motivos para isso ter ocorrido. Essas especulações são exploradas na área de discussões do resultado também presentes nesse relatório.

Por fim, foi estabelecida a influência que cada um dos features(atributos) tem na previsão do target. Essa última informação coletada reforça a complexidade de cálculo do target selecionado.

## DISCUSSÕES

## DISCUSSÃO ACERCA DO MODELO SELECIONADO

### Escolha do modelo de previsão pelo grupo

No projeto optamos pelo modelo de previsão a floresta de decisões. Realizamos essa escolha, pois ao estudar de maneira mais aprofundada esse formato observamos algumas vantagens de aplicação sendo elas a sua robustez, que diz respeito ao bom desempenho que ele possui em uma

variedade de tarefas e conjuntos de dados com pouca necessidade de ajuste fino, a sua forma de lidar com outliers que garante uma menor sensibilidade a eles se comparado a outros modelos como o de regressão linear, a sua capacidade de evitar overfitting (ou sobreajuste), que é possibilitada pela combinação de várias árvores de decisão e resulta em uma melhora na generalização do modelo.

Ademais, como citado anteriormente na parte de metodologia, esse modelo não requer normalização porque pode lidar com dados em diferentes escalar e com diferentes distribuições, sem perder sua qualidade de desempenho.

Contudo, ao selecionar um modelo para ser o utilizado em um projeto com convicção é importante levar em consideração as limitações e desvantagens dele também, por isso fomos atrás desses aspectos. As principais desvantagens são a complexidade, que acaba por torná-lo mais difícil de interpretar do que modelos mais simples, como árvores de decisão individuais; o custo computacional, o qual fica claro devido a necessidade de treinamento de diversas árvores que pode acabar sendo mais caro do que o treinamento em um único modelo.

Além destes, uma das maiores desvantagens que ele apresenta diante do seu uso em um trabalho didático de uma disciplina de aprendizado de máquina é a sua dificuldade de interpretação. Seria de grande vantagem conseguir visualizar com facilidade, para garantir maior fluidez na explicação de seu funcionamento, sua forma de operação, que embora seja possível obter a importância das variáveis, a compreensão do modelo pode acabar sendo um tanto complexa.

Levando em consideração tanto os fatores favoráveis quanto os desfavoráveis concluímos que a escolha da Floresta Aleatória parece adequada ao caso do trabalho desenvolvido pois, o conjunto de dados possui outlier, as variáveis têm escalas diferentes e o modelo que buscávamos deveria apresentar bom desempenho e robustez além de ser pouco influenciado pelas condições do conjunto de dados. Sendo que a principal desvantagem no nosso cenário é a complexidade do modelo que dificulta a interpretação dos resultados.

**Outros modelos que poderiam ter sido escolhidos e por que não foram**

A escolha do modelo ideal depende dos objetivos e das características do conjunto de dados. É sempre recomendável testar diferentes modelos e comparar seus desempenhos para escolher o mais adequado. Por isso, colocamos em teste os demais modelos expostos em sala de aula e compreendemos por que não seriam adequados no nosso caso.

Dentre as opções de modelos lineares havia inicialmente a **regressão linear**. Apesar de possuir como características ser um modelo simples e interpretável, não aparentou adequada ao nosso objetivo nem ao nosso conjunto de dados pois assume uma relação linear entre os atributos e o target. Devido ao nosso dataset haver outliers e a possibilidade de relações não lineares, era esperado que o

desempenho desse formato seria comprometido por causa de sua sensibilidade aos outliers e por não ser capaz de capturar as relações não lineares dos dados.

Outra opção que foi explorada pelo grupo foi a Árvore de decisão. Seu funcionamento consiste em uma divisão dos dados em subconjuntos com base em regaras. Essa escolha possuía grande vantagem pois é fácil de interpretar e de visualizar, sendo uma escolha muito favorável para conseguir aprender melhor e apresentar melhor sobre o passo a passo do algoritmo elaborado e de seu funcionamento. Porém, ele é um modelo mais propenso a realizar sobreajustes e pode não ter o mesmo desempenho de modelos mais complexos. Como há a Floresta aleatória que conta com a combinação de várias árvores, julgamos que seria mais vantajoso utilizá-la.

**Dúvida entre usar k-NN ou Floresta Aleatória**

O método k-NN ficou dentre as opções que consideramos fortemente utilizar, devido a sua aplicação em outros exemplos em sala de aula deixando claro cenários de grande funcionalidade e de fato era um modelo válido para o problema a que o grupo se propôs a solucionar, mas não julgamos que seria a mais adequada ao considerar as características do conjunto de dados relativos aos líquidos iônicos e o objetivo que possuíamos. Antes vale destacar o por quê de levarmos o k-NN em consideração.

Primeiramente, ele pode ser aplicado tanto em casos de classificação quanto regressão e no caso que enfrentamos estamos optando por prever um valor numérico (a medição ‘e’). Pode também se destacar por sua simplicidade, e traria benefícios por ser relativamente de fácil compreensão e implementação.

Por último, seu aspecto de flexibilidade também chamou atenção já que pode capturar relações não lineares nos dados. Até testamos a aplicação do k-NN em nosso projeto, contudo achamos que poderia poluir nosso notebook e por isso não disponibilizamos esse teste para visualização. Após isso conseguimos compreender alguns motivos pelos quais o k-NN talvez não seria a melhor escolha para o nosso caso.

Concluímos que ele apresentava algumas limitações como a sensibilidade à escala dos dados e exigiria a normalização dos dados, que adiciona complexidade ao processo (anteriormente no relatório na área da metodologia foi documentado que não houve normalização para a Floresta Aleatória, mas na área do código relativa à aplicação do k-NN foi necessário aderir a esse recurso para testar o desempenho do modelo). Outro aspecto limitante observado foi o custo computacional, que para grandes conjuntos de dados fica mais significativo, pois precisa calcular a distância entre todas as

amostras. Por fim, demosntra também grande sensibilidade à escolha do k, que é o número de vizinhos e por isso precisa ser otimizado adicionando assim maior complexidade ao processo.

A conclusão foi portanto que, embora o k-NN possa ser aplicado ao problema, a Floresta Aleatória aparenta ser uma escolha mais robusta e eficiente, considerando as características do banco de dados e a busca por um modelo com bom desempenho e menos sensível à escala dos dados.

**DISCUSSÃO DOS RESULTADOS OBTIDOS**

De início é importante indicar ao leitor que os resultados de maneira mais visual estão disponíveis no notebook disponibilizado no repositório do Github, essa área do relatório serve como uma discussão mais aprofundada para compensar qualquer lacuna que possa ficar na breve discussão desenvolvida no notebook. Tendo isso esclarecido, segue a discussão.

Chegamos a conclusão que obtivemos um resultado satisfatório! O modelo de Floresta Aleatória, após a otimização dos hiperparâmetros com o Optuna, apresentou um bom desempenho na previsão da variável alvo 'e' (interação da fase do processo com o soluto por meio de interações de dispersão).

Para validar essa conclusão é necessário apresentar evidencias que validam o bom resultado e por isso calculamos o Raiz do Erro Quadrático Médio (Root Mean Square Error/ RMSE). Obtivemos que o RMSE do projeto ficou igual a 0.19 esse valor indica que, em média, as previsões do modelo se desviam 0.19 unidades do valor real da variável ‘e’. Levando em consideração que a escala da variável ‘e’ no conjunto de dados varia aproximadamente entre -1 e 1, esse erro pode ser interpretado como relativamente baixo, evidenciando assim o bom desempenho do algoritmo desenvolvido.

Além disso foi necessário avaliar qual seria uma boa técnica de otimização a ser aplicada, assim observou-se que o Optuna se destacou como a melhor técnica de otimização de hiperparâmetros, superando a Busca Aleatória e a Busca em Grade em termos de RMSE, mesmo que por uma pequena diferença. Isso mostra que o Optuna foi mais eficiente em encontrar a combinação ideal de hiperparâmetros para o modelo.

Realizou-se também a análise dos erros, por meio do gráfico de dispersão e do histograma dos resíduos, se pode fornecer mais informações sobre o desempenho do modelo e indicar possíveis áreas de aprimoramento. Esses gráficos estão disponíveis no notebook presente no repositório do Github.

Obtivemos também um valor de um $R^2$ de 0.64. Para compreender melhor o que isso significa é importante esclarecer o que o $R^2$ indica. Esse parâmetro mede a proporção da variabilidade bos dados reais que o modelo consegue explicar e varia de 0 a 1. Um $R^2$ de 0.64 ainda é um valor razoável, mas indica que o modelo não está capturando toda a variância na variável alvo 'e'. Isso

significa que existem outros fatores, além das variáveis preditoras utilizadas, que influenciam o valor de 'e'. Esse resultado demonstra na prática a complexidade para a obtenção do valor de ‘e’.

Elaborou-se alguns motivos possíveis para o $R^2$ não ser mais alto, dentre eles o fato de haver a possibilidade de relações não lineares complexas entre as variáveis que o modelo de Floresta Aleatória não foi capaz de compreender completamente, o fato de poder haver outras variáveis relevantes para prever ‘e’ que não estão documentadas no banco de dados utilizado como referência. Pensou-se também na possibilidade de ruído nos dados, isto é, os dados podem conter erros de medição que dificultam a previsão precisa de ‘e’. Por fim, levou-se em consideração a limitação do modelo que mesmo com a otimização de hiperparâmetros, o modelo da Floresta Aleatória pode ter uma capacidade limitada de capturar a complexidade do problema.

## CONCLUSÃO

Por fim, conclui-se que utilizando as ferramentas lecionadas pelo professor Daniel Roberto Cassar na disciplina de aprendizado de máquina (Machine Learning) foi possível desenvolver um algoritmo seguindo o modelo de regressão Floresta Aleatória com aplicações possíveis no uso e pesquisa de líquidos iônicos. Obtivemos ao final bons resultados e uma melhor compreensão dos objetos utilizados no processo de elaboração do trabalho, servindo de grande oportunidade para os alunos realizarem pesquisas alvejando uma melhora do desempenho do projeto em desenvolvimento.

## REFERÊNCIAS

As referências utilizadas já se encontram disponíveis no documento Readme no repositório do Github, mas será repetido no seguinte espaço para evitar que qualquer fonte passe despercebida.

[1].Banco de dados sobre Líquidos Iônnicos: (link: https://digital.library.unt.edu/ark:/67531/metadc307526/)

[2].SCIKIT-LEARN. scikit-learn: machine learning in Python. Disponível em: https://scikit-learn.org/stable/.

[3].Líquidos Iônicos - Alguns aspectos sobre as propriedades, preparação e aplicações. Disponível em: https://wp.ufpel.edu.br/wwverde/files/2014/12/L%C3%ADquidos-I%C3%B4nicos.pdf. Acesso em: 2 set.2024

[4].Halper, Marin S; Ellenbogen, James C. Supercapacitors: A brief overview. Disponível em: https://www.mitre.org/sites/default/files/pdf/06_0667.pdf

[5].Liu, Huan; Yu, Haijun. Ionic liquids for eletrochemical energy storage devices aplication. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S1005030218302640