GABRIEL ALENCAR SILVA ALMEIDA DANTAS

SECAGEM DE AREIA EM LEITO FLUIDIZADO: ANÁLISE DE DADOS E MODELAGEM PRESCRITIVA POR MEIO DE ALGORITMOS DE APRENDIZADO DE MÁQUINA

São Paulo

2021

GABRIEL ALENCAR SILVA ALMEIDA DANTAS

SECAGEM DE AREIA EM LEITO FLUIDIZADO: ANÁLISE DE DADOS E MODELAGEM PRESCRITIVA POR MEIO DE ALGORITMOS DE APRENDIZADO DE MÁQUINA

Trabalho de Formatura em Engenharia de Minas do curso de graduação do Departamento de Engenharia de Minas e de Petróleo da Escola Politécnica da Universidade de São Paulo

Orientadora: Prof.ª. Dra. Ana Carolina Chieregati

São Paulo

2021

Dantas, Gabriel Alencar Silva Alemida
Elaboração de um modelo preditivo do processo de secagem em leito fluidizado por meio de algoritmos de aprendizado de máquina / G.A.S.A.DANTAS. São Paulo, 2021. 42 p.

Trabalho de Formatura - Escola Politécnica da Universidade Universidade de São Paulo. Departamento de Engenharia de Minas e de Petróleo.

1.Aprendizado de máquina 2.Secagem de areia 3.Secador de leito fluidizado I. Universidade de São Paulo.Escola Politécnica. Departamento de Engenharia de Minas e de Petróleo
II. t.

Dedico este trabalho ao Lucas Yuji Ono, um grande amigo do Gabriel Alencar. Dedico também ao Playstation, um grande amigo do Fanta.

““*Knowledges are a deadly friend*
If no one sets the rules
The fate of all mankind I see
Is in the hands of fools.”

Peter Sinfield

RESUMO

A constante busca por garantia de qualidade dos processos, associada à redução de custos, prevenção de defeitos, redução de falhas, redução do tempo de resposta às variações do processo, aumentou o interesse das empresas para os adventos da indústria 4.0. A quarta revolução industrial é caracterizada pelo uso de sistemas computacionais para atender as demandas da indústria de tornar a produção cada vez mais ágil e dinâmica e melhorar a eficiência dos processos. A inteligência artificial tem realizado um papel importante nesse cenário, fornecendo ferramentas para ajudar nas tomadas de decisões, diminuindo a intervenção humana dos processos e os tornando mais produtivos. A secagem de areia é um processo custoso e com grande contribuição nas emissões na produção da areia industrial. Um controle eficiente nesse processo implica em redução de consumo de combustível por tonelada alimentada e evita a saída de areia úmida do processo. Este trabalho, diante desse contexto, visa criar um modelo prescritivo baseado no aprendizado de máquina que possa sugerir a carga de gás do queimador de modo que reduza o consumo de gás por tonelada alimentada sem perda de qualidade. Para atingir este objetivo foi utilizado técnicas de tratamento de dados, tais como detecção de outliers, substituição dos valores faltantes por interpolação e normalização dos dados. Os modelos de Random Florest, Multi-Layer Perception e Gradient Boosting foram implementados em Python para avaliar a assertividade de cada modelo frente ao conjunto de dados. Como resultado, os modelos apresentaram boa assertividade de carga do queimador, com R^2 entre 0,88 e 0,92, e a aplicação dos modelos de inteligência artificial se mostrou viável para aplicação nesse processo.

Palavras-chave: Aprendizado de máquina. Secagem por Leito Fluidizado. Inteligência artificial. Areia Industrial. Industria 4.0

ABSTRACT

The constant search for process quality assurance, associated with cost reduction, defect prevention, failure reduction, reduced response time to process variations, has increased companies' interest in the advent of Industry 4.0. The fourth industrial revolution is characterized by the use of computer systems to meet the industry's demands to make production more agile and dynamic and improve process efficiency. Artificial intelligence has played an important role in this scenario, providing tools to help in decision making, reducing human intervention in processes and making them more productive. Sand drying is a costly process with a large contribution to emissions in industrial sand production. An efficient control in this process implies a reduction in fuel consumption per ton fed and prevents the exit of wet sand from the process. This work aims to create a prescriptive model based on machine learning that can suggest the burner gas charge in a way that reduces gas consumption per ton fed without loss of quality. To achieve this objective, data processing techniques were used, such as detection of outliers, replacement of missing values by data interpolation and data normalization. Random Forest, Multi-Layer Perception and Gradient Boosting models were implemented in Python to assess the assertiveness of each model against the dataset. As a result, the models showed good assertiveness of burner charge, with R^2 between 0.88 and 0.92, and the application of artificial intelligence models proved to be viable for application in this process.

Keywords: Machine learning. Fluidized Bed Drying. Artificial intelligence. Industrial sand. Industry 4.0

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

A quarta revolução industrial, também conhecida como Industria 4.0, tornou os ambientes de manufatura cada vez mais dinâmicos, conectados, mas também intrinsicamente mais complexos, com interdependências adicionais, incertezas e grandes volumes de dados sendo gerados. Avanços da indústria 4.0 geram uma grande vantagem em acelerar a fabricação com custo mais baixo e aumentar a produção com redução a desperdícios e aumento de eficiência, além de contribuir com objetivos de redução emissões de carbono (JANA, 2019).

Existe um movimento global das empresas que visa reduzir as emissões de carbono em 50% até 2050. Para atingir esse objetivo as empresas estão trocando as matrizes energéticas por energias consideradas limpas, como biocombustíveis, energia elétrica verde, como energia elétrica oriunda de hidrelétricas e usinas eólicas e também a redução de resíduos e aproveitamento dos mesmos. Além dessas ações, há investimentos no desenvolvimento de processos mais inteligentes suportados por tecnologias da indústria 4.0, que tornam os processos mais inteligentes, eficientes e produtivos o que tem contribuído com as indústrias para reduzir as emissões (ALLWOOD, 2010).

Uma vez que o queimador é um grande emissor de CO_2 devido à grande quantidade de combustível utilizada na queima para se realizar a secagem da areia e sendo o principal componente do custo processo, este trabalho visa criar um modelo prescritivo baseado no aprendizado de máquina para prescrever a carga de combustível do queimador em um sistema de secagem por leito fluidizado buscando atingir os parâmetros de qualidade do processo para assim reduzir o consumo específico de combustível, diminuindo o tempo de resposta do operador em relação às variações do processo.

1.1 OBJETIVOS

1.1.1 *OBJETIVO GERAL*

O objetivo desse trabalho é desenvolver um modelo de aprendizado de máquina para prescrever sugestões de carga de um queimador industrial de um sistema de secagem de areia por leito fluidizado a fim de reduzir o consumo de combustível do sistema e reduzir o tempo de resposta do operador diante das variações do processo.

1.1.2 *OBJETIVOS ESPECÍFICOS*

Para atingir o objetivo geral proposto, estabeleceu-se os seguintes objetivos secundários:

1. Criar modelos prescritivos por meio dos métodos de *random forest*, *multi layer perception* e *gradient boosting*, a fim de gerar prescrições da carga do queimador visando de reduzir o consumo de combustível e reduzir o tempo de resposta dos ajustes do processo em relação as variações do processo.
2. Analisar a performance e assertividade dos modelos utilizando métricas de avaliação de desempenho dos modelos para validar a possibilidade de aplicação no processo.
3. Gerar visualizações dos dados de modo a gerar *insights* das melhores condições do processo, tendo em vista a redução do consumo de combustível por tonelada alimentada.

2 REVISÃO BIBLIOGRÁFICA

2.1 AREIA

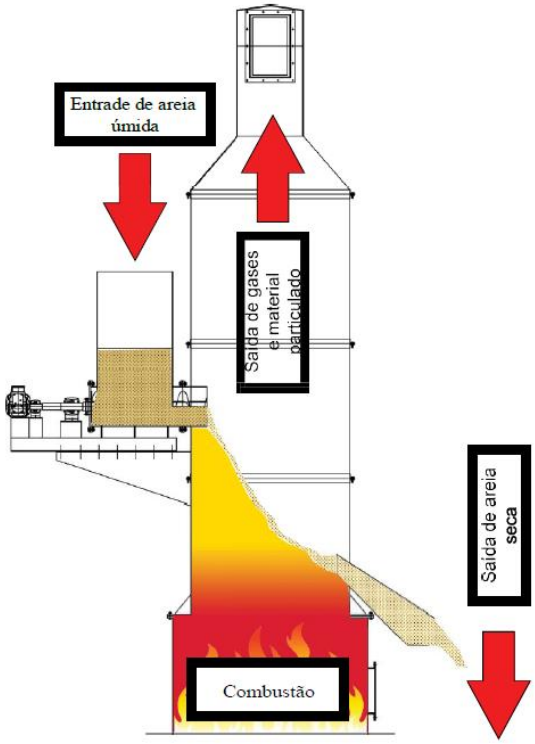
Agregado é definido como o material natural de propriedades adequadas, ou obtido artificialmente pela cominação de rocha, com o diâmetro das partículas entre 100 mm, passante em aberturas de 4”, e 0,075 mm, retido em malha 200#. Dentro do grupo dos agregados, a areia é definida como o material natural, com propriedade adequada e com granulometria passante em aberturas de 2 mm (ALMEIDA, 2007).

A areia pode ser classificada quanto ao seu uso, quando a areia é destinada para processos industriais, como na fabricação de vidros, na indústria de fundição, na indústria cerâmica, na fabricação de refratários e de cimento; na indústria química para fabricação de ácidos e de fertilizantes; no fraturamento hidráulico, e também para usos como horticultura e locais de lazer, esta é considerada areia industrial. As areias industriais geralmente apresentam granulometria entre 0,5 e 0,1 mm, com teores mais elevados de sílicas, SiO2, na forma de quartzo, o teor de sílica e a quantidade de impurezas é determinado a partir do uso que será destinado a areia (DAVIS, 1985).

2.2 SECAGEM EM LEITO FLUIDIZADO

Os secadores de leito fluidizados são comumente utilizados na secagem de areia, pelotas, pós e outros produtos granulados em vários setores, como no setor agrícola, farmacêutico e no beneficiamento mineral. O processo de secagem neste equipamento consiste na injeção de ar quente, oriundo da queima de combustível, fluidizando o leito de sólidos. No processo de secagem por leito fluidizado, o ar de um soprador é aquecido por um queimador e é distribuído para o leito de sólidos por meio de uma placa perfurada, fazendo com que o material de alimentação úmido seja fluidizado. Conforme o ar aquecido passa pelo material úmido, o ar carrega a umidade e é exaurido por uma saída de gases. A Figura 1 mostra o processo de secagem das partículas de areia. As vantagens dos secadores de leito fluidizado sobre outros métodos de secagem são a alta taxa de secagem devido à boa interação entre as partículas e o ar, melhor controle de temperatura e operação ao longo do processo e menor custo de manutenção devido à ausência de movimento componentes dentro do secador. (JIBIN, 2016)

Figura 1 - Representação do secador de leito fluidizado.



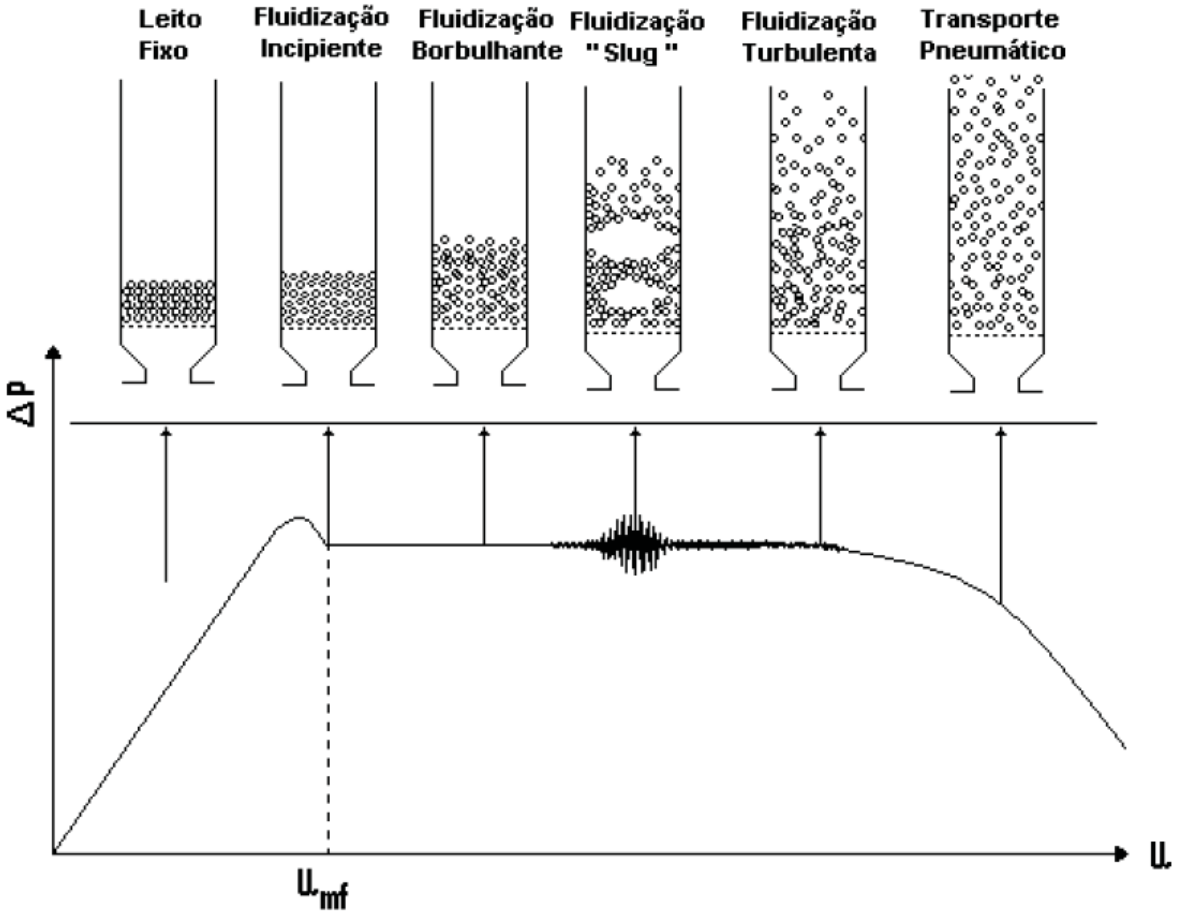
Fonte: GATTI & RESCHINI LTDA, 2019

2.2.1 FATORES QUE AFETAM O DESEMPENHO DO PROCESSO

O desempenho do secador de leito fluidizado está intimamente ligado a qualidade da fluidização do leito da areia. Os modelos hidrodinâmicos são obtidos pelo balanço de forças entre as partículas e a velocidade do gás, portanto a velocidade do gás que passa pelo leito de sólidos determina a fluidização do leito, quanto menor for a velocidade do gás mais estático é o leito e quanto maior a velocidade maior a turbulência, podendo chegar a se tornar um regime borbulhante. Outro fator relevante, é a queda de pressão relacionada com a velocidade do gás, quando o gás passa pela placa perfurada acontece uma queda de pressão. A pressão cai linearmente com a velocidade do gás, com o aumento gradual da velocidade do gás a queda de pressão provoca o equilíbrio do balanço de forças das partículas provocando a fluidização do leito, essa velocidade de gás que provoca o equilíbrio é chamada de velocidade mínima de fluidização (u_{mf}). Quando a queda de pressão se mantém constante com o aumento da velocidade do gás, é estabelecido o regime borbulhante. A queda de pressão diminui quando a velocidade do gás se torna maior que a velocidade terminal

das partículas e isso ocorre quando o regime de transporte pneumático começa. (PHILIPPSEN, 2015). É possível observar os diferentes regimes de fluidização de acordo com a velocidade do fluido e diferença de pressão na figura 2.

Figura 2 - Regimes de fluidização do leito em função da velocidade superficial do fluido.



Fonte: NITZ, 2008

A vazão de gás ideal de modo que mantenha o regime do leito no estado de fluidização mínima onde a troca de calor entre o gás e as partículas seja máxima, evitando com que a perda de calor pela saída dos gases. Em regime borbulhante a troca de calor entre o gás e as partículas é minimizada fazendo com que a temperatura da saída dos gases seja maior. A temperatura do ar de entrada é inversamente proporcional ao tempo de secagem, porém deve ser suficiente para secar a areia temperaturas do ar muito elevadas geram desperdício de energia térmica (BODHANWALLA, 2017).

2.3 INTELIGÊNCIA ARTIFICIAL

Inteligência artificial (IA) é a inteligência atribuída para as máquinas. Na ciência da computação, campo de pesquisa de IA define-se como o estudo de "agentes inteligentes": qualquer dispositivo que percebe seu ambiente e toma ações que maximizam sua chance de sucesso em algum objetivo. Coloquialmente, o termo "inteligência artificial" é aplicada quando uma máquina imita funções "cognitivas" que os humanos associam a mente humana como "aprender" e "resolver problemas". Atualmente é classificado como IA a compreensão da fala humana por uma máquina, sistemas que competem em alto nível em jogos estratégicos (como xadrez e Go), carros autônomos, roteamento inteligente em redes de entrega de conteúdo, simulações militares e interpretar dados complexos (INTERNATIONAL CONFERENCE ON ICT AND KNOWLEDGE ENGINEERING, 2017).

2.4 APRENDIZADO DE MÁQUINA

O aprendizado de máquina é definido como o campo de estudo que dá aos computadores a capacidade de aprender sem serem programados explicitamente. O aprendizado de máquina (*machine learning*) é usado para ensinar as máquinas a lidar com os dados com mais eficiência. Às vezes, depois de visualizar os dados, não é possível interpretar as informações extraídas dos dados, por muitas vezes uma variável tem forte influência na outra e quando se avalia mais de 3 variáveis existem limitações nas visualizações destes dados. Nesse caso, aplicamos o aprendizado de máquina. Com a abundância de conjuntos de dados disponíveis, a demanda por aprendizado de máquina está aumentando. Muitos setores aplicam o aprendizado de máquina para extrair dados relevantes. O objetivo do aprendizado de máquina é aprender com os dados. Muitos estudos foram feitos sobre como fazer as máquinas aprenderem por si mesmas sem serem explicitamente programadas. Muitos matemáticos e programadores aplicam várias abordagens para encontrar a solução deste problema que possuem enormes conjuntos de dados (MAHESH, 2020).

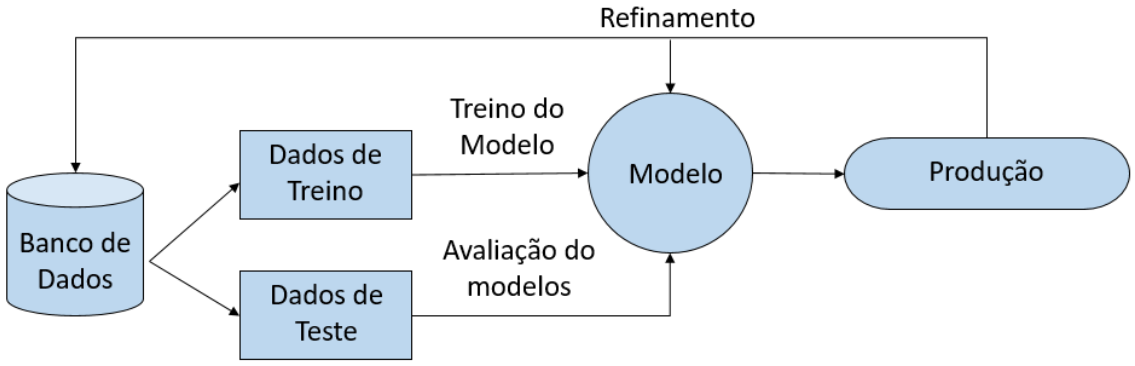
O aprendizado de máquina depende de diferentes algoritmos para resolver problemas de dados. O tipo de algoritmo empregado depende sobre o tipo de problema que você deseja resolver, o número de variáveis, o tipo de modelo que melhor se adequa a ele e assim por diante (BOUSQUET, 2003).

2.5 TIPOS DE APRENDIZADO DE MÁQUINA

2.5.1 APRENDIZADO SUPERVISIONADO

A aprendizagem supervisionada é um tipo de aprendizado de máquina onde uma função mapeia uma entrada para uma saída com base noz exemplos de pares de entrada-saída fornecida pelo usuário. Os algoritmos supervisionados de aprendizado de máquina são aqueles que precisam de ajuda externa. O conjunto de dados de entrada é dividido em dados de treino e dados teste. O conjunto de dados do treino tem uma variável de saída que precisa ser previsto ou classificado. Todos os algoritmos aprendem alguns tipos de padrões do conjunto de dados de treinamento e os aplica ao conjunto de dados de teste para previsão ou classificação. O fluxo de trabalho de algoritmos de aprendizado de máquina supervisionado é dado pela Figura 3 (MAHESH, 2020).

Figura 3 - Fluxograma de aprendizado de máquina supervisionado.



Fonte: MAHESH, 2020 com tradução de elaboração própria

2.5.2 APRENDIZAGEM NÃO SUPERVISIONADA

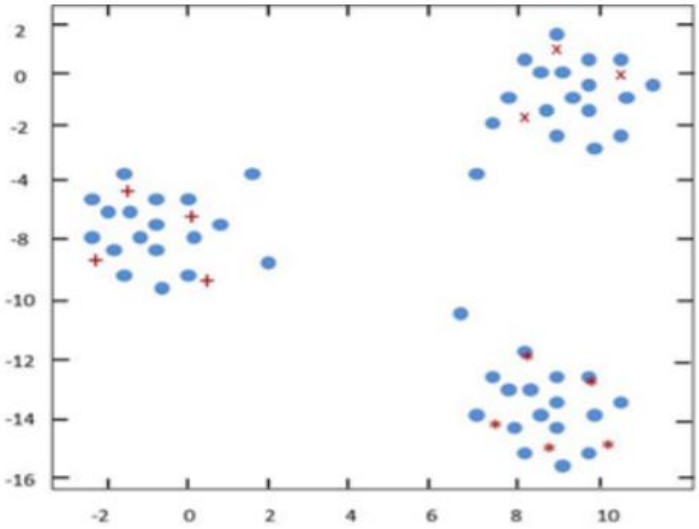
Imagine uma máquina que recebe alguma sequência de entradas x_1, x_2, x_3, ..., onde x_t é a entrada sensorial no tempo t. Essas entradas ou dados, pode corresponder a uma imagem na retina, aos pixels de uma câmera ou a uma forma de onda de som. Na aprendizagem não supervisionada, a máquina recebe entradas x_1, x_2, x_3, ..., mas não obtém resultados alvo supervisionados, nem recompensas de seu ambiente. Pode parecer um tanto misterioso imaginar o que a máquina poderia aprender, uma vez que não recebe nenhum feedback de seu ambiente. No entanto, é possível

desenvolver uma estrutura formal para aprendizagem não supervisionada com base na noção de que o objetivo da máquina é construir representações da entrada que podem ser usadas para a tomada de decisão, prever entradas futuras, comunicar eficientemente as entradas para outra máquina, etc. Em certo sentido, a aprendizagem não supervisionada pode ser considerada como encontrar padrões nos dados acima e além do que seria considerado puro ruído não estruturado. Dois exemplos clássicos muito simples de aprendizagem não supervisionada são agrupamento e redução de dimensionalidade (BOUSQUET, 2003).

2.5.3 APRENDIZADO SEMI-SUPERVISIONADO

O aprendizado semi-supervisionado é um tipo de técnica de aprendizado de máquina. É o meio caminho entre o aprendizado supervisionado e o não supervisionado. O conjunto de dados parcialmente rotulado é mostrado na Figura 4. O principal objetivo do aprendizado semi-supervisionado é superar as desvantagens da aprendizagem supervisionada e não supervisionada. O aprendizado supervisionado requer uma grande quantidade de dados de treinamento para classificar os dados de teste, o que é um processo econômico e demorado. Por outro lado, a aprendizagem não supervisionada não requer nenhum dado rotulado, o que agrupa os dados com base na similaridade nos pontos de dados usando agrupamento ou abordagem de máxima verossimilhança. A principal desvantagem desta abordagem é que ela não consegue agrupar dados desconhecidos com precisão. Para superar esses problemas, o aprendizado semi-supervisionado foi proposto pela comunidade de pesquisa, que pode aprender com uma pequena quantidade de dados de treinamento e pode rotular os dados de teste desconhecidos. A aprendizado semi-supervisionado constrói um modelo com poucos padrões rotulados como dados de treinamento e trata o restante dos padrões como dados de teste (REDDY, 2018).

Figura 4 - Dados semi-classificados.



Fonte: REDDY, 2018

2.6 MÉTODOS COMUNS DE APRENDIZADO DE MÁQUINA

2.6.1 ÁRVORE DE DECISÃO

A aprendizagem por árvore de decisão é um dos algoritmos de aprendizagem mais populares, devido às suas várias características que o tornam atrativo: simplicidade, compreensibilidade, sem parâmetros e capacidade de lidar com dados de diferentes tipos como objetos, números e datas. Na aprendizagem da árvore de decisão, uma árvore de decisão é induzida a partir de um conjunto de instâncias de treinamento rotuladas representado por um *turple* de valores de atributo e um rótulo de classe. Por causa do vasto espaço de busca, a aprendizagem da árvore de decisão é normalmente um processo ganancioso, de cima para baixo e recursivo começando com todos os dados de treinamento e uma árvore vazia. Um atributo que melhor particiona os dados de treinamento é escolhido como a divisão para a raiz e os dados de treinamento são então particionados em subconjuntos separados que satisfazem os valores do atributo de divisão. Para cada subconjunto, o algoritmo prossegue recursivamente até que todas as instâncias em um subconjunto pertençam a mesma classe (SU, 2006).

Figura 5 - Representação da árvore de decisão.

Fonte: Adaptado de GAMA 2011.

2.6.2 NAIVE BAYERS

O classificador de Naive Bayers é uma técnica de classificação baseada no Teorema de Bayes com forte suposição de independência entre os recursos. Um classificador Naive Bayes assume que a presença de um recurso específico em uma classe não está relacionada à presença de qualquer outro recurso. Naive Bayes visa principalmente a classificação e é utilizado principalmente para realizar *clusters*, onde o propósito da classificação depende da probabilidade condicional de acontecer (MAHESH, 2020).

Seja x = (x_1, x_2, ..., x_n) um vetor de características e C = (c_1, c_2, ..., c_m) denotar todas as categorias possíveis às quais os vetores podem pertencer. O princípio de um classificador Naive Bayes é calcular as probabilidades p_1, p_2, ..., p_n para cada x onde p_j é a probabilidade que x pertença à categoria c_j. Determinando o valor máximo (p_1, p_2, ⋯, p_j), saberemos a qual categoria o vetor de características x pertence. Portanto, o problema de classificação pode ser considerado como encontrar o valor máximo da equação 1:

$$P(c_j|x_1, \dots, x_n) = \frac{P(x_1, x_2, \dots, x_n)P(c_j)}{P(c_1, c_2, \dots, c_n)} \quad (1)$$

Onde P (c_j) denota a probabilidade de que uma amostra aleatória pertença à categoria c_j. P (x_1, x_2, ..., x_n | c_j) é a probabilidade de que a categoria c_j contenha o

vetor de características x = (x_1, x_2, ..., x_n), se já sabemos que a amostra de treinamento está em c_j. P (c_1, c_2, ..., c_n) é a probabilidade conjunta de todas as categorias possíveis (FENG, 2016).

2.6.3 MÁQUINA DE VETORES DE SUPORTE

A máquina de vetores de suporte se tornou popular devido à sua habilidade de resolver problemas de classificação não linear. O princípio da máquina de vetores de suporte é construir um hiperplano para garantir que a distância entre os dois tipos de estrutura seja maximizada. Enquanto maximiza a distância, a máquina de vetores de suporte ainda precisa reduzir a chance de que os pontos sejam classificados em grupos errados para minimizar as punições. Desta forma, podemos ver que a máquina de vetores de suporte é essencialmente um método de otimização quadrática (DING, 2001).

Usando (x_i, y_i) para denotar uma amostra do conjunto de treinamento, sabemos $x_i \in R^d$ e $y_i \in \{-1, +1\}$ onde d é a dimensão do vetor x_i e y_i denota o resultado da classificação do recurso vetor x_i. Se o hiperplano puder dividir o conjunto de amostra, tem-se a seguinte condição satisfeita:

$$y_i[(w \cdot x_i) + b] - 1 \geq 0\,, i = 1, 2, \ldots, n. \quad (2)$$

Para maximizar a distância entre esses dois conjuntos de amostras, é preciso minimizar

$$\varphi(\mathrm{w}) = \frac{1}{2}\left|\left|w\right|\right|^2 \quad (3),$$

Resolvendo o problema de otimização, é obtido o seguinte resultado

$$\sum_{i=1}^{n} y_i a_i^0 (x \cdot x_i) + b = 0 \quad (4),$$

Onde a_i^0 é um multiplicador de Lagrange.

Se o conjunto de treinamento não for divisível, um fator de relaxamento i ≥ 0 e um parâmetro de penalidade C serão introduzidos. A função objetivo ideal é então transformada em

$$\varphi(\mathrm{w}) = \frac{1}{2}\left|\left|w\right|\right|^2 + c\sum_{i=1}^{n} \epsilon_i \quad (5)$$

Com as seguintes restrições:

$$y_i[(w \cdot x_i) + b] - 1 + \epsilon_i \geq 0\,, i = 1, 2, \ldots, n. \quad (6)$$

A máquina de vetores suportados recebe os sinais de $\{y_i\}$ para distinguir amostras de diferentes categorias.

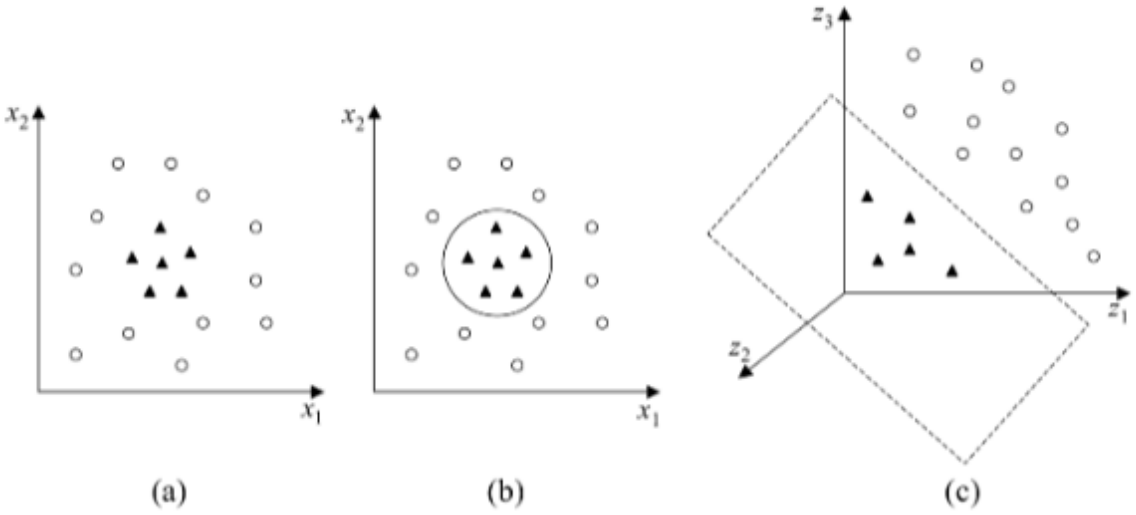
$$I(x) = sgn\left(\sum_{i=1}^{n} y_i a_i^0 (x \cdot x_i) + b\right) \quad (7)$$

Usando funções de Kernel, a máquina de vetores de suporte é capaz de construir uma conexão entre um conjunto de vetores de baixa dimensão e um conjunto de vetores de alta dimensão. Para problemas que não são divisíveis em um espaço de baixa dimensão, a máquina de vetores de suporte o transforma em um espaço de alta dimensão e, portanto, o torna divisível, se possível. Uma função kernel é definida como $\mathrm{K(x,y)} = \varphi(\mathrm{x}) \cdot \varphi(\mathrm{y})$ onde φ é a função de mapeamento entre o vetor de baixa dimensão x e o vetor de alta dimensão y. Se a função do Kernel for selecionada apropriadamente, a máquina de vetores suportados obtém uma função de classificadora como:

$$I(x) = sgn(\textstyle\sum_{i=1}^{n} y_i a_i^0 (\varphi(x) \cdot \varphi(x_i)) + b) \quad (8),$$

Onde φ (x) é um vetor de dimensão superior. Porque o Kernel função K (x, y) se relaciona apenas a x e y, onde x e y são ambos vetores de baixa dimensão. A Figura 5 apresenta diferentes separações por máquina de vetor suportado (LORENA, 2017 e FENG, 2016).

Figura 6 - (a) Conjunto de dados não linear; (b) Fronteira não linear no espaço de entradas; (c) Fronteira linear no espaço de características.

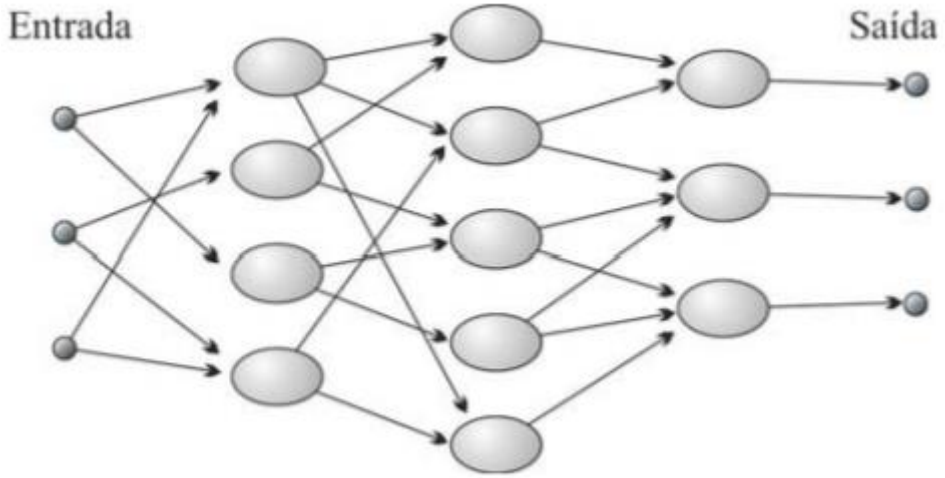


Fonte: LORENA, 2017.

2.6.4 REDES NEURAIS

Uma rede neural artificial (ou simplesmente rede neural) consiste em um modelo de aprendizado de máquina baseado no complexo funcionamento do cérebro humano. A rede artificial se baseia em uma camada de entrada de neurônios (ou nós, unidades), uma ou duas (ou até mesmo três) camadas ocultas de neurônios e uma camada final de neurônios de saída. A Figura 7 mostra uma arquitetura típica, onde linhas conectando neurônios também são mostradas (FERNEDA, 2006).

Figura 7 - Representação de um sistema de redes neurais artificial.



Fonte: FERNEDA, 2016

Cada conexão está associada a um número numérico denominado peso. A saída, hi, do neurônio i na camada oculta é,

$$h_i = \sigma\left(\sum_{j=1}^{N} V_{ij}x_j + T_i^{hid}\right) \quad (9)$$

Onde $\sigma()$ é chamado de função de ativação ou de transferência, N o número de *input* de neurônios, V_{ij} os pesos, x_j os *inputs* dos *inputs* dos neurônios, e o T_i^{hid} o *threshold* dos neurônios da camada oculta. O objetivo da função de ativação é, além de introduzir a não linearidade na rede neural, limitar o valor do neurônio para que a rede neural não seja paralisada por neurônios divergentes. Foi demonstrado que uma rede neural construída da maneira acima pode aproximar qualquer função computável com uma precisão arbitrária. Os números dados aos neurônios de entrada são variáveis independentes e aqueles retornados dos neurônios de saída são variáveis

dependentes da função sendo aproximada pela rede neural. As entradas e saídas de uma rede neural podem ser binárias (como sim ou não) ou mesmo símbolos (verde, vermelho, ...) quando os dados são codificados apropriadamente. Esse recurso confere uma ampla gama de aplicabilidade às redes neurais (WANG, 2003).

2.6.5 BOOSTING

O aprendizado de máquina *Boosting*, é baseado na observação de que encontrar muitas regras básicas pode ser muito mais fácil do que encontrar uma única regra de previsão altamente precisa. Para aplicar o *boosting*, inicia-se com um método ou algoritmo para encontrar regras. O algoritmo de impulso chama esse algoritmo de aprendizado "fraco" ou "básico" repetidamente, cada vez que o alimenta com um subconjunto diferente dos exemplos de treinamento (ou, para ser mais preciso, uma distribuição ou ponderação diferente sobre os exemplos de treinamento). Cada vez que é chamado, o algoritmo de aprendizado básico gera uma nova regra de previsão fraca e, após muitas rodadas, o algoritmo de impulso deve combinar essas regras fracas em uma única regra de previsão que será muito mais precisa do que qualquer uma das regras fracas (SCHAPIRE, 2003).

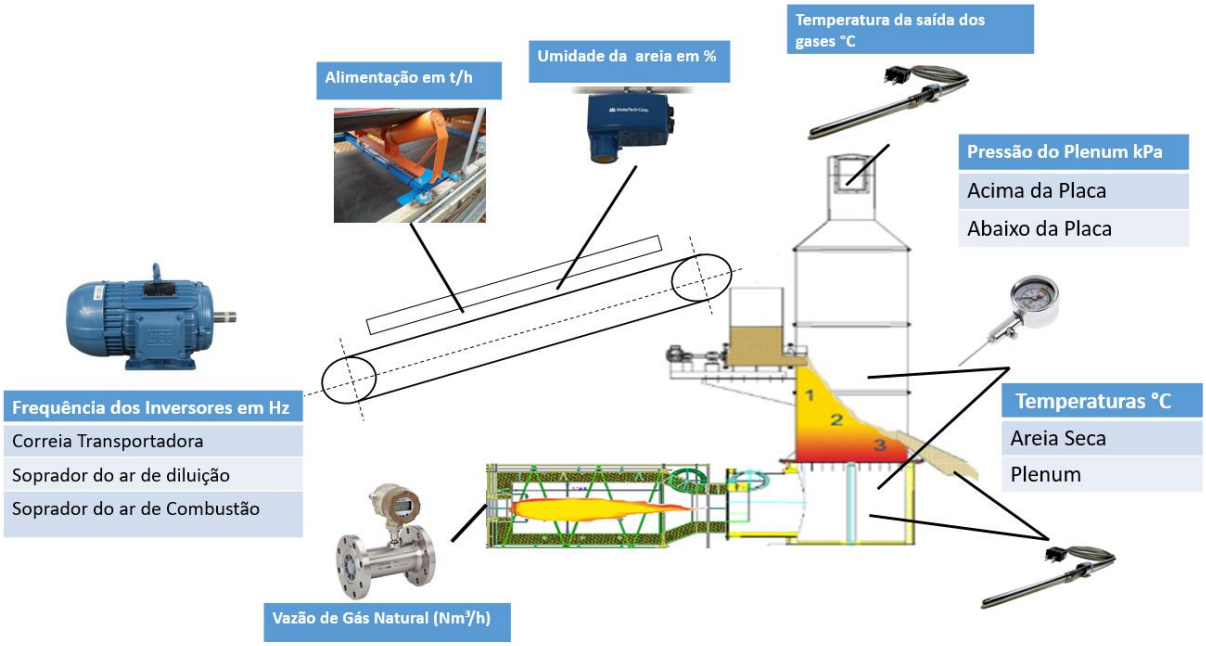
3 MATERIAIS E MÉTODOS

A arquitetura da Indústria 4.0 pode ser estratificada em 5 camadas, camada de aquisição de dados, camada de manipulação dos dados, camada de armazenamento de dados, análise de dados, camada de acesso do usuário. Essas camadas são integradas e personalizadas para serem adotadas de acordo com os requisitos das indústrias (JANA, 2019). O sucesso de implementação de um projeto de aprendizado de máquina está estritamente ligado a qualidade dessas camadas.

3.1 AQUISIÇÃO DOS DADOS

Os dados foram obtidos de uma fábrica de produtos para construção civil líder no seguimento de argamassa e rejuntes. Os dados do processo fazem parte do processo da secagem de areia para se produzir argamassas. Nesse processo é vital que a areia esteja completamente seca antes de ir para as etapas seguintes do processo. As variáveis de processo foram coletadas a partir de um Controlador Lógico Programável (CLP) integrado com um sistema para coletas e análises de dados. A Figura 8 mostra a disposição dos sensores integrados com o CLP.

Figura 8 - Sensores e dados coletados do processo de secagem.



Fonte: Elaboração própria

As variáveis coletadas do processo estão compiladas na tabela 1:

Tabela 1 - Variáveis coletadas do sistema.

Variável	Unidade	*Feature*	Origem
Frequência do Inversor do Ar de Diluição	Hz	hz_ventdil	Inversor
Frequência do Inversor do Ar de Combustão	Hz	hz_ventcomb	Inversor
Frequência do Inversor da Correia Transportadoria	Hz	hz_tc	Inversor
Alimentação do Secador (Instântanea)	t/h	alimentacao_inst	Balança Integradora
Umidade da Areia Alimentada	%	umd_entrada	Medidor de Umidade
Temperatura do Plenum	°C	plenum_temp	Termopar
Temperatura da Areia Seca	°C	sand_tempout	Termopar
Temperatura de Saída dos Gases	°C	ar_tempout	Termopar
Pressão Abaixo da Placa Perfurada	kPa	pressao_plenumup	Barômetro
Pressão Acima da Placa Perfurada	kPa	pressao_plenumdown	Barômetro
Vazão de Gás Natural	Nm^3/h	vazao_gas	Medidor de Vazão
Carga do Queimador	%	queimador_carga	CLP

Fonte: Elaboração própria

Os dados foram exportados no formato csv divido por virgulas. Com 12 variáveis e 21.858 registros válidos, ou amostras.

3.2 PRÉ-PROCESSAMENTO DOS DADOS

O pré-processamento se inicia logo quando os dados são coletados e organizados em conjunto, com o objetivo de resolver problemas comuns encontrados em conjuntos de dados. A qualidade do projeto de *machine learning* está diretamente ligada a qualidade dos dados utilizados fazendo desta etapa do processo fundamental. Normalmente dados apresentam os seguintes problemas, grande quantidade de valores desconhecidos, ruídos, desproporcionalidade da variável de saída, entre outros. Quando estes problemas são tratados antes de serem inseridos nos modelos o tornam mais confiáveis (BATISTA et al., 2003).

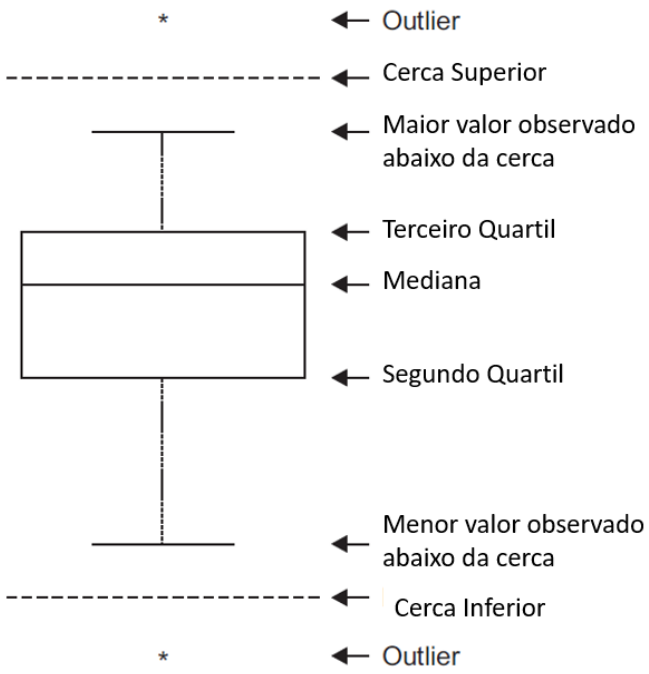
Foram adotadas as seguintes técnicas para solucionar os problemas encontrados nos dados:

3.2.1 DETECÇÃO DE OUTLIERS

Outliers são pontos de dados distantes da maioria dos outros pontos de dados, os *outliers* nos dados que não são normalmente distribuídos não requerem identificação. Como a maioria dos testes estatísticos presume que os dados são normalmente distribuídos, a identificação de valores discrepantes deve preceder a análise dos dados. Os gráficos *boxplots* podem ser usados para identificar os *outliers*.

Neste gráfico de caixa, quaisquer dados que estejam fora das linhas da cerca superior ou inferior são considerados discrepantes (KWAK, 2017).

Figura 9 - Bloxplot.



Fonte: KWAK,2017 tradução de elaboração própria

Os *outliers* excluídos do *dataframe* foram aqueles cuja leitura no processo foi considerada impossível de ser alcançada, possíveis erros de leituras dos sensores, e não todos os *outliers* detectados. A detecção dos outliers é importante para entender as métricas de desempenho dos modelos gerados.

3.2.2 DADOS AUSENTES

Os dados vazios não podem entrar no modelo eles devem ser excluídos ou preenchidos. Os valores foram preenchidos realizando a interpolação dos dados válidos mais próximos no intervalo de 4 minutos, caso o intervalo fosse maior que 4 minutos os dados não eram preenchidos e posteriormente excluídos.

3.2.3 NORMALIZAÇÃO DOS DADOS

A normalização é realizada com o objetivo resolver o problema de desproporcionalidade entres os dados, a normalização consiste em transformar os valores dos atributos para um intervalo específico, mais comumente entre [0,1] ou [-1,1]. A normalização se faz necessária uma vez que, muitos dos métodos de

aprendizado de máquina usados assumem que todos os recursos estão centrados em torno de zero e têm variância na mesma ordem. Se uma característica tem uma variância com uma ordem superior a outras, ela pode dominar a função objetivo e tornar o estimador incapaz de aprender com outras características (SHANKER, 1996).

O método utilizado para normalização foi o z-score:

$$z = \frac{x - u}{s} \quad (10)$$

Onde z será o valor atribuído, u é a média das amostras, x o valor do dado antes de ser substituído e s o desvio padrão. A normalização foi aplicada a todas as variáveis do *dataframe* utilizada nos modelos para resolver o problema de proporcionalidade entre os dados.

3.3 ANÁLISE ESTATISTICA

Antes de se iniciar a modelagem é necessário, antes de tudo, conhecer os dados e detectar e tratar os *outliers*, identificar se há dados faltantes e preencher os vazios de modo que não haja enviesamento da amostra de dados. A Figura 10, mostra a descrição dos dados obtidos:

Figura 10 - Descrição das features.

	count	mean	std	min	25%	50%	75%	max
sand_tempout	229150.0	76.631604	9.518285	20.870000	72.89000	74.27000	76.62000	278.850
umd_entrada	229150.0	1.718624	13.413277	-1.990000	1.15000	1.38000	1.79000	3868.000
ar_tempout	229150.0	14.418336	150.470653	-263.180000	69.43000	86.90000	97.87000	850.080
plenum_temp	229150.0	491.493064	52.571407	-450.370000	461.03000	498.95000	535.16000	584.190
queimador_carga	144815.0	31.575464	9.550888	0.000000	27.60000	32.00000	38.30000	70.000
arcomb_temp	146812.0	45.386590	4.218190	-263.051300	43.65176	45.07859	47.86739	251.774
vazao_gas	146765.0	168.669887	23.349267	2.583236	159.59530	169.79580	184.57390	370.795
vazao_gastotal	146812.0	324583.063162	76381.087724	193748.000000	260765.75000	323916.50000	382802.50000	474614.000
pressao_plenoup	229150.0	554.604953	302.004342	-17526.590000	342.70000	513.69000	673.38000	2208.040
pressao_plenodown	229150.0	1051.200638	213.956526	98.540000	927.73250	1021.37000	1098.66000	3392.640
hz_ventdil	144689.0	48.305559	1.592698	0.000000	47.00000	48.00000	49.00000	55.000
hz_ventcomb	144689.0	55.992930	0.441569	0.000000	56.00000	56.00000	56.00000	56.000
hz_tc	229150.0	50.005067	9.171902	0.000000	43.00000	52.00000	59.00000	60.000
nv_sl500	229150.0	269.237784	120.343483	17.860000	183.64000	284.54000	363.56000	485.920
hz_tcmax	229150.0	58.481026	4.618995	30.000000	60.00000	60.00000	60.00000	60.000
hz_tcmin	229150.0	30.919424	5.515938	20.000000	25.00000	30.00000	30.00000	59.000
alimentacao_acumulada	53144.0	9127.411900	5624.755745	839.000000	4135.00000	8169.50000	14099.00000	19592.000
alimentacao_inst	40074.0	25.728117	4.489203	-1.090000	24.08000	26.35000	28.31000	77.680
umd_ent_manual	52565.0	4.064011	2.635669	3.000000	3.20000	4.10000	4.40000	50.000

Fonte: elaboração própria

Uma matriz de correlação das *features* foi feita para verificar a correlação linear entre os pares das variáveis estudadas. A correlação utilizada foi a de Pearson (p) este coeficiente quantifica a força de associação entre duas variáveis e assume valores entre -1 e 1. Valores próximo a 1 indicam forte correlação positiva, ou seja, as variáveis são diretamente dependentes. Os valores que se aproximam de -1, indicam forte correlação, porém negativa, sendo essas variáveis inversamente dependentes (GALARÇA, 2010). O coeficiente de Pearson é calculado da seguinte forma:

$$p = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad (11)$$

3.4 RELAÇÃO TEMPORAL

Para verificar a relação temporal entre a carga do queimador e a temperatura de saída da areia foram criadas variáveis novas da carga do queimador denominadas queimador_carga-lagN, sendo o N número de minutos de atraso em relação a *feature* original, conforme a Figura 11.

Figura 11 - Conceito de lag.

timestamp0	queimador_carga	queimador_carga-lag1	queimador_carga-lag2	queimador_carga-lag3
20/08/2021 16:31	19	-1 min	-2 min	-3 min
20/08/2021 16:32	19	19		
20/08/2021 16:33	19	19	19	
20/08/2021 16:34	19	19	19	19
20/08/2021 16:35	18	19	19	19
20/08/2021 16:36	18	18	19	19
20/08/2021 16:37	19	18	18	19
20/08/2021 16:38	22	19	18	18
20/08/2021 16:39	25	22	19	18
20/08/2021 16:40	28	25	22	19
20/08/2021 16:41	30,4	28	25	22
20/08/2021 16:42	30,4	30,4	28	25
20/08/2021 16:43	30,4	30,4	30,4	28
20/08/2021 16:44	28,4	30,4	30,4	30,4
20/08/2021 16:45	25,3	28,4	30,4	30,4
20/08/2021 16:46	22,3	25,3	28,4	30,4

Fonte: Elaboração própria

Após a criação das *features* com *lag* foi considerado como tempo presente o lag-3. As *features* sem o atraso, são as *features* objetivo onde serão mantidas apenas

as variáveis em que é possível colocá-las como *setpoint* ou aquelas as quais são parâmetro de qualidade do processo, neste caso a temperatura de saída da areia. O modelo conseguindo prever a carga no *lag-3* tem o potencial de criar sugestões de carga que possam alcançar os parâmetros de qualidade previamente determinados. As variáveis com o lag2 e lag1 foram excluídas do *dataframe*, pois quando o modelo for aplicado esses dados não existem, por serem informações futuras e o objetivo é determinar as condições 3 minutos a frente do tempo da sugestão de carga.

3.5 VALIDAÇÃO CRUZADA

A validação cruzada é um método estatístico de avaliação e comparação de algoritmos de aprendizagem, dividindo os dados em dois segmentos: um usado para aprender ou treinar um modelo e outro usado para validar o modelo. Na validação cruzada típica, os conjuntos de treinamento e validação devem se cruzar em rodadas sucessivas de modo que cada ponto de dados tenha uma chance de ser validado (REFAEILZADEH, 2016).

O *dataframe* foi separado em dois grupos, um de treino e outro de teste. O grupo de dados de teste contém 70% dos dados coletados aleatoriamente do *dataframe*, restando então 30% dos dados para teste, também coletados de forma aleatória.

3.6 APLICAÇÃO DOS MODELOS

Os algoritmos de aprendizado de máquina, como *gradiente boost*ing, *random florest* e redes neurais para regressão e classificação, envolvem vários hiperparâmetros que devem ser definidos antes de executá-los. Em contraste com os parâmetros padrões dos modelos de primeiro nível, que são determinados durante o treinamento, esses parâmetros de ajuste de segundo nível geralmente precisam ser cuidadosamente otimizados para atingir o desempenho máximo (PROBST, 2019).

O processo de otimização utilizado foi o *RandomizedSearchCv*, este método funciona em dois módulos, um de ajuste e outro de pontuação. É estabelecido um conjunto de hiperparâmetros, onde é realizado a validação cruzada com estes parâmetros de forma aleatória, então um sistema de pontuação é formulado baseado

no desempenho de cada conjunto de hiperparametros, por fim é selecionado o conjunto com a melhor pontuação (IZHAR, 2021).

Foram utilizados os seguintes hiperparâmetros para os modelos:

1. Rede Neural Artificial – Rede Neural Estatística Multicamadas (*Multylayer perceptron*)
    a. Número de Camada Ocultas e Neurônios (*hidden_layer_sizes*): (100, 75, 50, 25)
    b. Solucionador (solver): ‘adam’. Refere-se a um otimizador baseado em gradiente estocástico
    c. Número máximo de interações (*max_inter*): 1000
    d. Taxa de aprendizagem (*learning_rate*): ‘*invscaling*’
    e. Alpha: 5e-06
    f. Ativação (*activation*): ‘tanh’

    Random Florest :

    g. Número de Estimadores (*n_estimators*): 500
    h. Mínimo de divisões de amostra (*min_samples_split*): 2
    i. Máximo de *features* (*max_features*): *‘auto’*
    j. Bootstrap: ‘*True*’

    Gradient Boosting:

    k. Número máximo de folhas das árvores da árvore de decisão (*num_leaves*): 302
    l. Profundidade máxima da arvore de regras básicas (*max_depth*): 9
    m. Taxa de aprendizagem do Boost (*learning_rate*): 0,3
    n. Tipo de Boosting (*boosting_type*) = ‘dart’, encontram múltiplas árvores de regressão aditiva

3.7 VALIDAÇÃO DOS MODELOS

Para avaliar o desempenho dos modelos de aprendizado de máquina é necessário comparar os resultados preditos pelos modelos com os dados reais que não fizeram parte do conjunto de treinamento do modelo, e a partir dessa comparação são calculadas as métricas de desempenho para avaliar a confiabilidade dos modelos (MONARD,2003).

3.7.1 *MÉTRICAS DE DESEMPENHO*

As métricas de desempenho de um modelo indicam a qualidade do ajuste, dando uma medida do desempenho do modelo em predizer as amostras que foram previstas (PEDREGOSA, 2011). As principais métricas utilizadas para modelos de regressão são: coeficiente de regressão, erro médio absoluto e erro médio quadrático (CHOU, 2018).

1. Coeficiente de Determinação (R^2) é descrito como uma medida de variância explicada ou incerteza reduzida. Dado que $\sum_{i=1}^{n}(y_i - \bar{y}_i)^2$ é a incerteza remanescente após aplicação do modelo, então $\sum_{i=1}^{n}(y_i - \bar{y}_i)^2 - \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ é a incerteza explicada pelo modelo de regressão, utilizando a proporção entre essas duas incertezas obtém a medida de qualidade de ajuste do R² (LATTIN, 2011).

$$R^2 = \frac{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2 - \sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \quad (12)$$

2. Erro Médio Quadrático (MSE) é a média do quadrado dos módulos das diferenças entre os valores preditos e os valores originais. Isso faz com que o MSE penalize mais o modelo por predições mais distantes do modelo.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (13)$$

3. Desvio Médio Quadrático (RMSE) é a raiz quadrada do erro médio quadrático, essa medida é menos sensível aos *outliers* preditos, suavizando os desvios provocados por eles.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (14)$$

4. Erro Médio Absoluto (MAE) é média do erro da previsão e do dado original, ele fornece a ideia da distância média da previsão em relação ao dado original.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i) \quad (15)$$

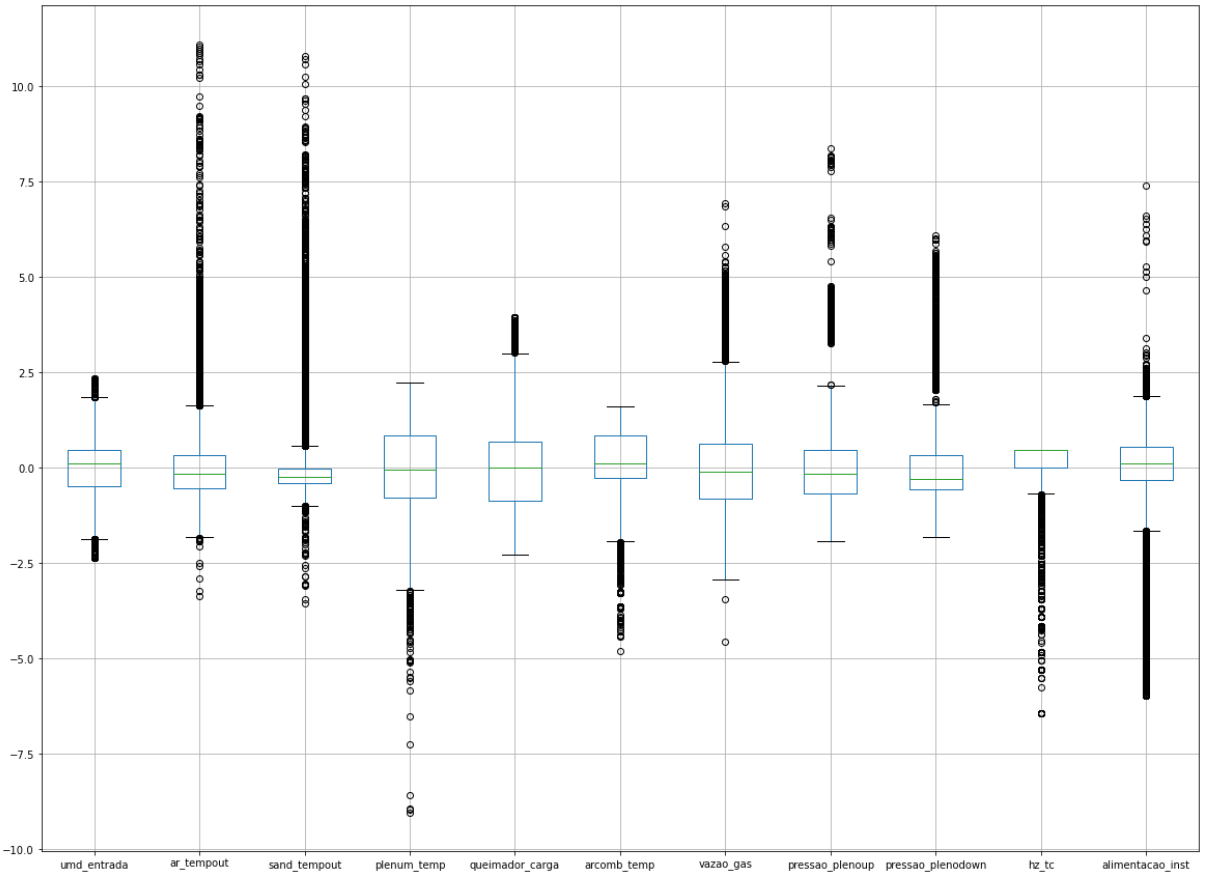
.

4 RESULTADOS E DISCUSSÕES

4.1 ANÁLISE ESTATISTICA

Na Figura 12 é apresentado um *BoxPlot* com os dados normalizados, utilizado para visualização de *outliers* e variabilidade dos dados.

Figura 12 - Boxplot das features coletadas normalizadas.



Fonte: Elaboração própria

A partir da análise do *boxplots* e da Tabela 2, é possível observar que o processo as variáveis sofrem grandes variações, com exceção da variável controlada pela malha de controle. A malha de controle controla o processo para que a temperatura da areia ao sair do secador seja 75°C, o que explica a baixa distância interquartis e a pouca variação. A umidade alimentação (umd_entrada) também é bem-comportada, uma vez que existem uma série de protocolos a serem seguidos antes de alimentar o secador, como deixar a pilha de areia “descansar” para perder

umidade. A pressão abaixo da placa perfurada do plenum sofre grandes variações, devido a entupimentos esporádicos que ocorrem ao longo do processo além das variações da quantidade de areia alimentada por hora.

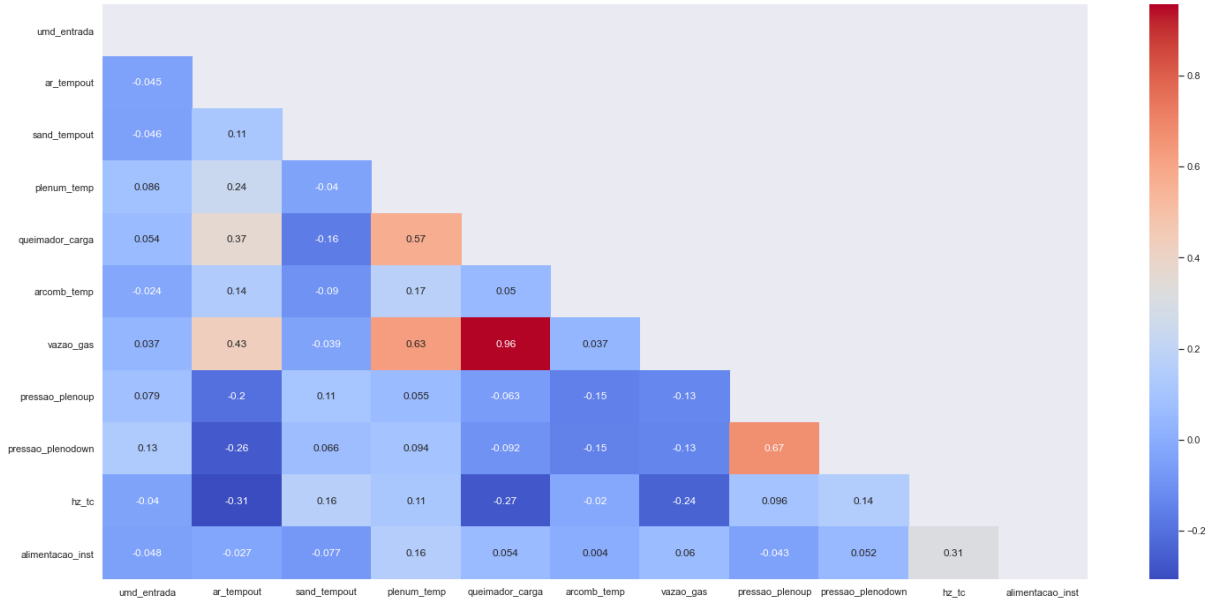
Tabela 2 - Estatística descritiva dos dados.

Features	Média	Desvio Padrão	Mínimo	1º Quartil	2º Quartil	3º Quartil	Máximo
umd_entrada	4,41	0,86	2,38	4,00	4,50	4,80	6,41
ar_tempout	83,37	14,11	37,94	74,80	81,77	90,18	129,84
sand_tempout	76,26	4,87	38,73	73,51	75,30	77,53	100,00
plenum_temp	495,75	37,84	153,59	466,05	493,82	527,58	580,02
queimador_carga	25,71	11,24	0,00	16,00	25,70	33,40	70,00
arcomb_temp	48,85	3,63	31,39	47,87	49,23	51,89	54,68
vazao_gas	157,85	30,74	17,78	133,25	154,88	177,33	370,80
pressao_plenoup	590,11	80,71	434,63	536,00	577,58	627,63	1265,20
pressao_plenodown	1088,49	379,23	397,68	869,75	973,36	1212,34	3392,64
hz_tc	57,98	4,34	30,00	58,00	60,00	60,00	60,00
alimentacao_inst	26,33	4,41	0,00	24,84	26,86	28,75	58,94

Fonte: Elaboração própria

Dado que a influência entre as variáveis é um fator importante para selecionar as *features* para assim obter um modelo de maior qualidade. A Figura 13 apresenta uma matriz com a correlação de Pearson entre as variáveis utilizadas neste trabalho.

Figura 13 – Matriz correlação de Pearson.



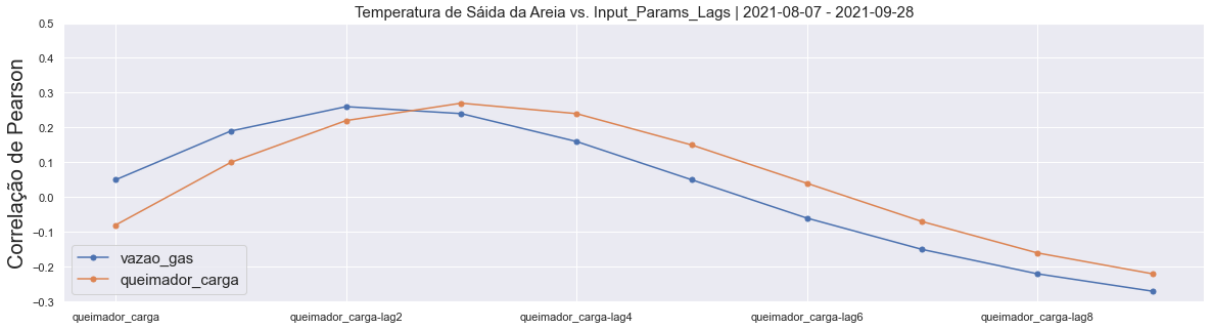
Fonte: Elaboração própria

É possível observar fortíssima correlação entre a carga do queimador e a vazão de gás, uma vez que a carga do queimador é o *setpoint* para determinar a vazão do gás natural que entra no queimador, por possuírem essa fortíssima correlação não é interessante que a vazão do gás entre no modelo para determinar a carga do

queimador. Por ser um processo multivariado e as variáveis variarem em conjunto as correlações par a par entre elas não são muito altas, porém algumas correlações chamam mais atenção, a correlação entre a carga do queimador e a temperatura do plenum possui uma correlação de 0,57, o que significar que aumentar a carga do queimador aumenta a temperatura do plenum e uma correlação de 0,37 com a temperatura de saída do ar, o que indica que essas temperaturas são mais sensíveis as mudanças das vazões de gás.

Como a temperatura de saída de areia é a variável de controle do processo, e esta apresenta a correlação baixa com a carga do queimador que é um dos *setpoints* do processo, foi investigado se a carga em momentos antes tem aumento na correlação na temperatura da saída da areia a fim de determinar uma relação temporal entre essas variáveis, para isso foi elaborada a Figura 14.

Figura 14 – Correlação da temperatura de saída da areia versus a carga do queimador e vazão de gás em diferentes *lags*.



Fonte: elaboração própria

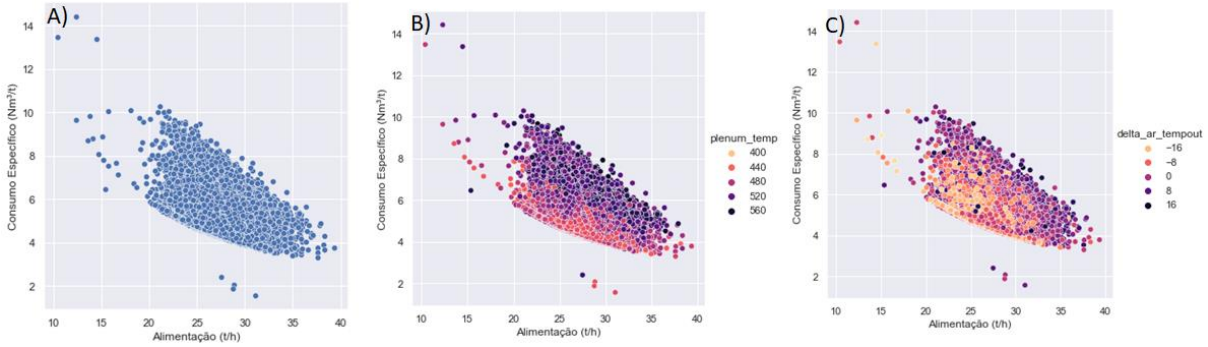
A Figura 14 foi elaborada filtrando os dados para deixar apenas nas condições de trabalho do equipamento, ignorando os dados de *ramp-up* e eventuais *outliers,* foi observado no *lag*3 (3 minutos antes da medida da temperatura de areia) a correlação de aproximadamente 0,3, a maior em relação aos outros tempos. Logo foi considerado que há uma inércia de 3 minutos do aumento da carga do queimador até acontecer o aumento na temperatura de saída da areia. A partir disso, se estabeleceu o modelo deve predizer a carga do queimador no lag3.

4.2 VISUALIZAÇÕES DOS DADOS

A visualização dos dados é uma etapa importante para gerar *insights* preciosos de um processo. Nessa etapa foi calculado o consumo específico dividindo a

alimentação horária (t/h) pela vazão de gás (Nm³/h). A partir dessa nova variável foi gerado visuais para entender quais as condições do processo que reduzem o consumo específico. Foram gerados gráficos de dispersão entre a alimentação horária e o consumo específico explicada por diferentes variáveis.

Figura 15 - Alimentação Vs. Consumo Específico. (A) Gráfico de dispersão simples, (B) Explicado pela temperatura do plenum e (C) Explicado pela diferença entre a temperatura de saída dos gases e a temperatura de saída da areia seca.



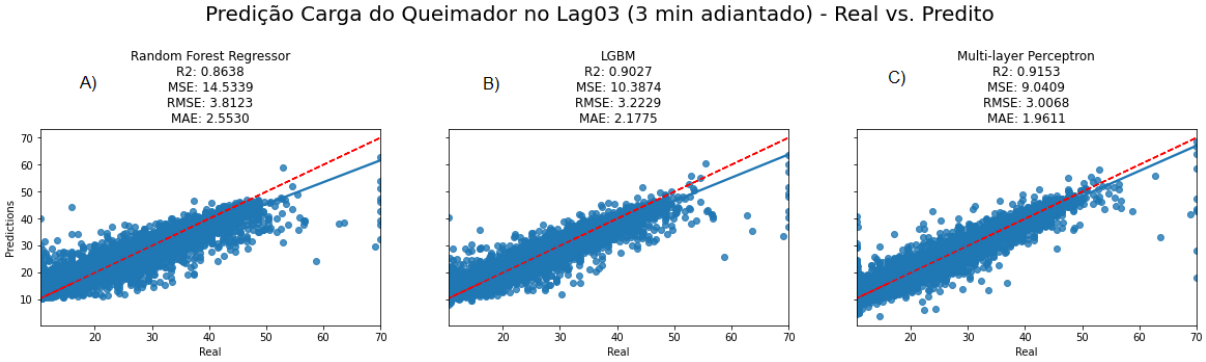
Fonte: Elaboração própria

A partir desses gráficos foi possível concluir que a relação entre a alimentação e o consumo específico não é linear. Quando observado o gráfico B da Figura 15 é fácil observar que em temperaturas mais baixas do plenum o consumo específico é menor e observando o gráfico C é perceptível que quanto menor e mais negativa for a diferença entre a temperatura de saída dos gases e a temperatura da saída da areia seca mais eficiente é o processo em relação ao consumo de combustível.

4.3 APLICAÇÃO E VALIDAÇÃO DOS MODELOS

Foram ensaiados 3 modelos, utilizando 3 métodos de *machine learning* diferentes, o *Random Florest*, modelo baseado na arvore de decisão, o LGBM (*Light Gradient Boosting*), baseado em na técnica de *Boosting* e o por último o Multi-Layer Perceptron, baseado em Redes Neurais. A Figura 16 mostra o desempenho dos modelos em predizer os dados, a partir dos dados de teste separados para esta avaliação.

Figura 16 - Gráficos dispersão Real vs. Preditos. A) Modelo de Random Florest. B) Modelo LGBM C) Multi-Layer Perceptron.



Fonte: Elaboração própria.

Os modelos apresentaram um bom resultado, com R^2 entre 0,86 até 0,92, mostrando alta capacidade de previsão. Os modelos apresentaram baixos MAE, porém quando comparamos o MAE com o RMSE é possível notar uma diferença devido a *outliers* remanescentes no conjunto de dados de teste. A Tabela 3 compila as métricas de desempenho dos modelos.

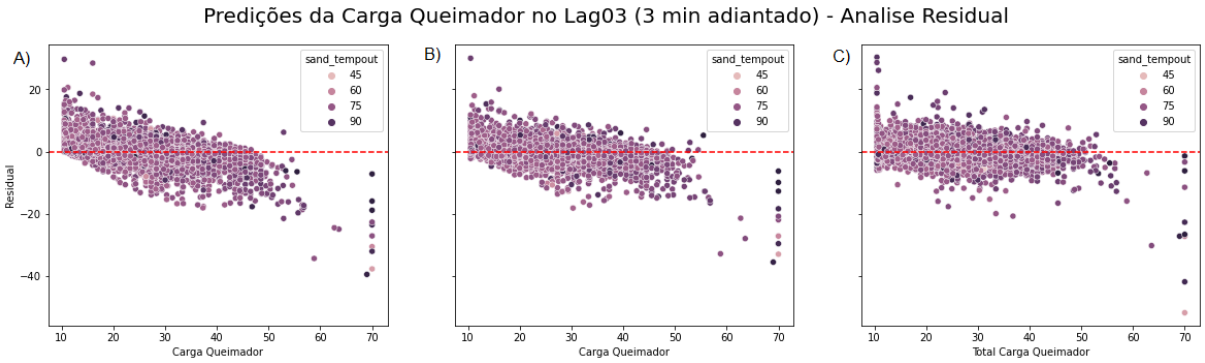
Tabela 3 - Métrica de desempenho dos modelos.

Modelo de Aprendizado	R^2	MSE	RMSE	MAE
Random Florest Regressor	0,86	14,53	3,81	2,55
Light Gradient Boosting Machine	0,90	10,39	3,22	2,18
Multi-Layer Perceptron	0,92	9,04	3,00	1,96

Fonte: Elaboração própria

Para melhor compreensão das métricas foi elaborada a análise residual, onde o resíduo é a diferença do previsto em relação ao dado original. A Figura 17 mostra a análise residual.

Figura 17 -Residuos Vs. Carga no lag 3 A) Modelo de Random Florest. B) Modelo LGBM C) Multi-Layer Perceptron.

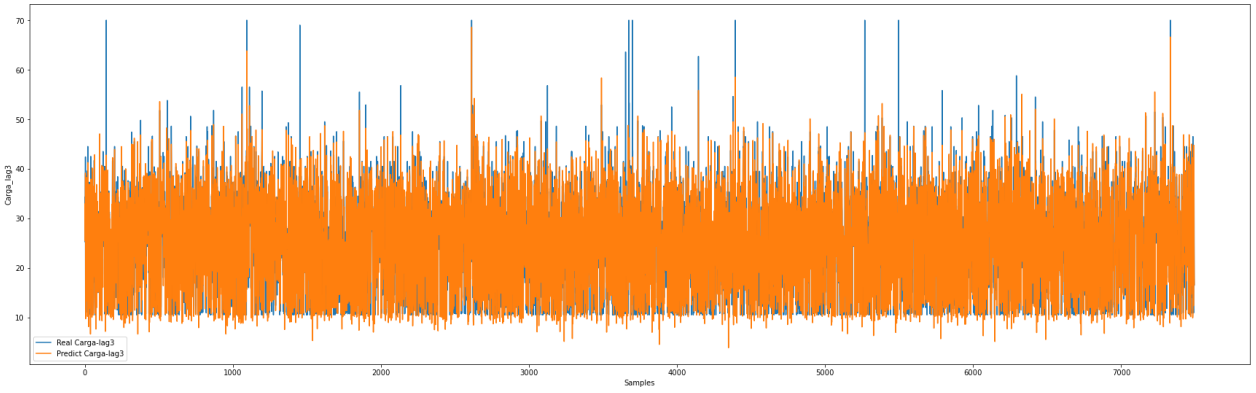


Fonte: Elaboração própria

É possível observar na Figura 16 que o modelo *Multi-Layer Perceptron* o que apresentou melhores métricas de desempenho apresenta os resíduos, conforme a figura 17, mais compactados em relação ao eixo x.

A Figura 18 apresenta a distribuição dos dados coletados e dos pontos preditos pelo modelo. É notável que os valores preditos apresentam valores próximos aos reais.

Figura 18 - Comparativo entre os valores preditos e os valores experimentais.



Fonte: Elaboração própria

5 CONCLUSÕES

O processo de secagem de areia possui muitas variáveis que podem afetar o desempenho do processo e impactar o consumo de combustível do processo. A análise de dados associada aos modelos de *machine learning* pode ajudar a compreender melhor os processos. Além de oferecer sugestões de condições de trabalhos que tem o potencial de otimizar o consumo de combustível, ajudando a atingir as metas de redução de carbono.

As visualizações dos dados sugerem condições de trabalho que podem reduzir o consumo específico. Operar com temperaturas do plenum em faixas menores que 500 ºC e reduzir a temperatura da saída dos gases em relação à temperatura de saída da areia pode gerar forte redução do consumo específico.

Com essas condições estabelecidas é possível utilizar os modelos preditivos e estabelecer como entradas dos modelos algumas condições de processo e setups de do processo para sugerir uma carga do queimador ideal para atingir os parâmetros desejados.

Entres os modelos propostos, o modelo baseado em redes neurais, o *Multi-layer perceptron,* foi o que apresentou os melhores parâmetros de desempenho. Este modelo apresentou R^2 de 0,92, que indica uma boa capacidade de predição. Além disso apresentou um MAE 1,96 indicando boa assertividade média, porém apresentou um MSE de 9,02 indicando que algumas medidas estavam muito distantes do real. De fato, há grande erros pertos dos extremos do intervalo avaliado da carga do queimador, evidenciando a presença dos *outliers* no conjunto de dados de teste.

.

REFERÊNCIAS

JENA, M. C.; MISHRA, S. K.; MOHARANA H. S. Application of Industry 4.0 to enhance sustainable manufacturing, **Environmental Progress & Sustainable Energy**, India, v.39, 2020.

ALLWOOD, J. M.; CULLEN, J. M.; MILFORD, R. L. Options for Achieving a 50% Cut in Industrial Carbon Emissions by 2050**, Environmental Science & Technology**, Reino Unido, v. 44, p. 1888-1894, 2010.

ALMEIDA, S. L. M.; LUZ, A. B. da (Ed.).**Manual de agregados para construção civil**. Rio de Janeiro: CETEM, p 228, 2019, il. ISBN 9788561121457

DAVIS, L. L; TEPORDEI, V. V. **Sand and gravel. In : Mineral Facts and Problems**, 1995 Edition, Bureau of Mines, Preprint from Bulletin 675, 15p.

JIBIN, A; SHYAMKUMAR, M.B . Study on Sand Particles Drying in a Fluidized Bed Dryer using CFD, **International Journal of Engineering Studies**, India, v.8, 2016. p 129-145.

GATTI & RESCHINI LTDA. **Manual de Instrução Secador Modelo** SLF1800ES. 2019

PHILIPPSEN, C.G; VILELA, A C F; ZEN, L D. Fluidized bed modeling applied to the analysis of processes: review and state of the art. **Journal of Materials Research and Technology**, Rio Grande do Sul, v. 4, p. 208 – 216, 2015.

NITZ, M.; GUARDANI, R. Fluidização Gás-Sólido: Fundamentos e Avanços. **Revista Brasileira de Engenharia Química**, São Paulo. 2008.

BODHANWALLA, H; RAMACHANDRAN, M. Parameters affecting the Fluidized bed performance: A review. **Journal on Emerging trends in Modelling and Manufacturing**, India, v. 3, p. 17-21, 2017.

INTERNATIONAL CONFERENCE ON ICT AND KNOWLEDGE ENGINEERING, 15., Bangkok, 2017. **Artificial Intelligence, Machine Learning and Deep Learning**. Bangkok, Siam University, 2017.

MAHESH, B. Machine Learning Algorithms - A Review. **International Journal of Science and Research**, v. 9, p. 381-386, 2020.

BOUSQUET, O. et al. **Advanced Lectures on Machine Learning**. Canberra. Springer-Verlag Berlin Heidelberg, 2004. p. 72–112. Disponível em <https://link.springer.com/content/pdf/10.1007%2Fb100712.pdf>. Acesso em 27 Set. 2021.

REDDY, Y. C. A.; VISWANATH, P.; REDDY, B. E.; Semi-supervised learning: a brief review, **International Journal of Engineering & Technology**, v. 7, p. 81-85, 2018.

SU, JIANG; ZHANG, HARRY. A Fast Decision Tree Learning Algorithm. **American Association for Artificial Intelligence**, p. 500-505, 2006.

GAMA, J.; FACELI, K.; LORENA, A.C.; DE CARVALHO, A.C.P.L.F. **Inteligência artificial: uma abordagem de aprendizado de máquina**. [S.l.]: Grupo Gen – LTC.

DING, C. H.; DUBCHAK, I. Multi-class protein fold recognition using support vector machines and neural networks. **Bioinformatics**, v. 17, n. 4, p. 349–358, 2001.

FENG, W; SUN, J; ZHANG, L; CAO, C; YANG, Q., A support vector machine based naive Bayes algorithm for spam filtering, **2016 IEEE 35th International Performance Computing and Communications Conference (IPCCC)**, 2016, p. 1-8, doi: 10.1109/PCCC.2016.7820655.

LORENA, A. C; CARVALHO, A. C. P. L. F. Uma Introdução às *Support Vector Machines*, **Revista de Informática Teórica e Aplica**, v. 14(2), p. 44-67 ,2007.

FERNEDA, E. Redes neurais e sua aplicação em sistemas de recuperação de informação, **Ciência da Informação**. 2006, v. 35, n. 1, p. 25-30. Disponível em: <https://doi.org/10.1590/S0100-19652006000100003>. Acesso em 30 Set 2021.

WANG, S. **Interdisciplinary computing in JAVA programming**. Nova York. Springer Science+Business Media, 2003. p. 81–84.

SCHAPIRE, R. E. **The Boosting Approach to Machine Learning: An Overview.** In: DENISON**,** D.D, et al (eds). Nonlinear Estimation and Classification. Lecture Notes in Statistics, vol 171. Springer, Nova York, 2003, p 149-172.

BATISTA et al. **Pré-processamento de dados em aprendizado de máquina supervisionado**. 2003. Tese (Doutorado) – Universidade de São Paulo.

SHANKER, M.; HU, M. Y.; HUNG, M. S. Effect of data standardization on neural network training, **Omega**, Grã Bretanha, v. 24(4), p. 385-397, 1996.

GALARÇA, S. P.; LIMA, C. S. M.; SILVEIRA, G; RUFATO, A R. Correlação de Pearson e análise de trilha identificando variáveis para caracterizar porta-enxerto de Pyrus communis L. **Ciência e Agrotecnologia**, Brasil, v. 34, n. 4, p. 860–869, 2010.

REFAEILZADEH P.; TANG L.; LIU H. Cross-Validation. In: Liu L., Özsu M. (eds) **Encyclopedia of Database** Systems. Springer, Nova York, 2016, p. 216-223.

PROBST, P.; BOULESTEIX A.; BISCHL, B. Tunability: Importance of Hyperparameters of Machine Learning Algorithms, **Journal of Machine Learning Research,** Alemanha, v.2, p. 1-31, 2019.

MOHAMMED, S.; BILAL, N.; KULSUM S.; **Random Forest Regressor Machine Learning Model Developed for Mental Health Prediction Based on Mhi-5, Phq-9 and Bdi Scale**. Disponível em: <https://ssrn.com/abstract=3867416>. Acesso em 04 de outubro de 2021.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos Sobre Aprendizado de Máquina. In: **SISTEMAS Inteligentes Fundamentos e Aplicações**. 1. ed. Barueri-SP: Manole Ltda, p. 89–114 2003.

PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

LATTIN; J. M. **Análise de Dados Multivariados**. São Paulo: CENGAGE Learnig**,** 2011**.** 455 p.