



UNIVERSIDADE ESTADUAL DE CAMPINAS

Faculdade de Ciências Aplicadas

Giovani Teixeira de Azevedo

Otimização de parâmetros de processo para injeção de rodas de alumínio: uma abordagem orientada a dados por meio de modelos substitutos

Limeira

2022

Giovani Teixeira de Azevedo

Otimização de parâmetros de processo para injeção de rodas de alumínio: uma abordagem orientada a dados por meio de modelos substitutos

Dissertação apresentada à Faculdade de Ciências Aplicadas da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia de Produção e de Manufatura, na Área de Pesquisa Operacional e Gestão de Processos.

Orientador: Prof. Dr. Leonardo Tomazeli Duarte

Coorientador: Dr. Filipe Corrêa Pinto

Este exemplar corresponde à versão final da dissertação defendida pelo aluno Giovani Teixeira de Azevedo, e orientado pelo Prof. Dr. Leonardo Tomazeli Duarte.

Limeira

2022

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Faculdade de Ciências Aplicadas
Ana Luiza Clemente de Abreu Valério - CRB 8/10669

Az25o Azevedo, Giovani Teixeira de, 1994-
Otimização de parâmetros de processo para injeção de rodas de alumínio :
uma abordagem orientada a dados por meio de modelos substitutos / Giovani
Teixeira de Azevedo. – Limeira, SP : [s.n.], 2022.

Orientador: Leonardo Tomazeli Duarte.
Coorientador: Filipe Corrêa Pinto.
Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade
de Ciências Aplicadas.

1. Aprendizado de máquina. 2. Otimização. 3. Fundição. I. Duarte,
Leonardo Tomazeli, 1982-. II. Pinto, Filipe Corrêa. III. Universidade Estadual de
Campinas. Faculdade de Ciências Aplicadas. IV. Título.

Informações Complementares

Título em outro idioma: Optimization of process parameters for injection of aluminum
wheels : a data-driven approach through surrogate models

Palavras-chave em inglês:

Machine learning

Optimization

Founding

Área de concentração: Pesquisa Operacional e Gestão de Processos

Titulação: Mestre em Engenharia de Produção e de Manufatura

Banca examinadora:

Leonardo Tomazeli Duarte [Orientador]

Romis Ribeiro de Faissol Attux

Washington Alves de Oliveira

Data de defesa: 15-12-2022

Programa de Pós-Graduação: Engenharia de Produção e de Manufatura

Identificação e informações acadêmicas do(a) aluno(a)

- ORCID do autor: 0000-0003-1128-1939

- Currículo Lattes do autor: <https://lattes.cnpq.br/6277297647433311>

Autor: Giovani Teixeira de Azevedo.

Título: Otimização de parâmetros de processo para injeção de rodas de alumínio: uma abordagem orientada a dados por meio de modelos substitutos.

Natureza: Dissertação.

Instituição: Universidade Estadual de Campinas.

Data da Defesa: Limeira, 15 de Dezembro de 2022.

BANCA EXAMINADORA

Prof. Dr. Leonardo Tomazeli Duarte (Orientador)

Prof. Dr. Romis Ribeiro de Faissol Attux

Prof. Dr. Washington Alves de Oliveira

A Ata da Defesa, com as respectivas assinaturas dos membros, encontra-se no processo de vida acadêmica do aluno.

Agradecimentos

Satisfação e gratidão, simples palavras que expressam sentimentos de felicidade de uma vida inteira. Felicidade essa que segundo Mahatma Gandhi não existe um caminho certo para percorrer, mas sim, o estado de ser feliz é o caminho. Com esse ideal que dedico este primeiro parágrafo de agradecimento a vida, que seja o destino ou Deus, sempre nos proporciona muito mais do que somos gratos, e por todas oportunidades que trilhou meu caminho até hoje, obrigado.

Mesmo com toda dificuldade financeira, minha família sempre esteve ao meu lado apoiando e motivando minhas decisões. Agradeço muito a minha mãe Lourença Isabel (mais conhecida pelos amigos como Bel), que me defende até hoje como melhor eletricitista de Cruzeiro e que nunca me negou nenhum favor que estivesse ao seu alcance. Na realidade, possuo 4 mães, sendo as outras três minhas irmãs, que são tão preocupadas comigo quanto a Bel. Foram elas que iniciaram todo meu desenvolvimento, me convencendo a ir a escola, comprando minhas roupas ou mesmo falando com meu chefe da casa de frios no meu primeiro emprego com 15 anos, para explicar que eu não tinha experiência, como também me motivar a estudar para ser piloto da Aeronáutica. Por todas essas etapas que passaram ao meu lado, meus sinceros agradecimentos.

Oportunidade são caminhos que aparecem em sua vida e que depende apenas de você aceitá-las ou descartá-las. Só foi possível finalizar todas as disciplinas e pré-requisitos do curso de mestrado com imenso apoio da área atual que trabalho, mais específico, do diretor Carlos Eduardo Lopes da área de engenharia avançada. Foi o Carlos que primeiramente me incentivou a buscar o mestrado e continuar os estudos sempre com um pé na universidade, para estar atento em todas as novidades do mundo da pesquisa. Muito obrigado por liberar horas para realizar as disciplinas tanto em Campinas, quanto em Limeira, além da disponibilidade os dados para realizar toda pesquisa. Somente com tal oportunidade foi possível chegar nesta etapa do trabalho.

Aqui cabe um agradecimento especial ao meu Co-orientador e supervisor Dr. Filipe Correa. Que além de amigo foi a pessoa que me encontrou como um simples estagiário em Cruzeiro SP e me forneceu todo apoio, incentivo, desafios e direções, tal como incentivos para continuar estudando e crescendo de forma humilde mas persistente. Sem suas boas analogias e dicas provavelmente não estaria aqui hoje, e sou extremamente grato por isso.

Todo crescimento pessoal e profissional só é alcançado com erros que são apontados, anotados e trabalhados para desenvolvimento. Dito isto, gostaria de agradecer imensamente aos professores doutores Romis Ribeiro e Washington Alves, que não só me ensinou muito nas disciplinas que cursei, como também adicionaram ótimos comentários

do meu trabalho na qualificação, que me possibilitou chegar no nível de trabalho que tenho hoje. Um ponto importante que também deve ser comentado, é o tempo de almoço separado e dedicado pelo Prof. Romis em 2019 com dúvidas que eu havia dentro de projeto da empresa, fiquei muito surpreso com sua disposição e vontade em querer me auxiliar mesmo sem ter almoçado naquele dia. Por último agradeço a paciência com relação as datas de defesa que precisei mudar em 2022 e também no envio do trabalho final.

Por fim e não menos importante, gostaria de agradecer ao Dr. Leonardo Tomazeli, meu orientador nesse trabalho e um dos melhores professores que tive durante o mestrado. Muitas vezes eu sei que devo ter cobrado muito e mandado diversas mensagens, porém acredito que no fim está dando tudo certo. Agradeço por todo rigor técnico e detalhes que corrigiu desse imenso trabalho. Por favor, continue sempre assim e faça com que novos alunos tenham oportunidade de aprender como eu tive, obrigado.

Resumo

O presente trabalho de mestrado se propõe, a partir de uma abordagem orientada a dados, construir modelos substitutos de aprendizado de máquina com intuito principal de utilizá-los no contexto da otimização de parâmetros do processo de injeção de rodas de alumínio. Neste sentido, foi coletado um conjunto de dados multivariados do processo de injeção e refrigeração nos moldes das rodas de alumínio de uma empresa do setor automotivo totalizando 65 variáveis. Em seguida esses dados passaram por uma série de tratamentos até ser utilizados no treinamento de modelos de aprendizado de máquina. No que se refere ao problema de otimização do processo, foram consideradas quatro funções objetivos. Uma dessas funções está associado a um problema de classificação, no qual se deseja discriminar entre rodas aprovadas ou reprovadas no processo de inspeção. As demais funções objetivos estavam relacionadas a problemas de regressão cujas saídas a serem explicadas estavam associados a parâmetros do processo. Em nosso estudo, investigamos quatro métodos de aprendizado de máquina, sendo eles: regressão logística (apenas para classificação), regressão linear, floresta aleatória e redes neurais. Quando otimizados com todos objetivos em conjunto, os resultados não foram satisfatórios, porém quando focado em apenas uma função objetivo, como da redução do tempo de ciclo, as curvas de otimização se mostraram mais condizentes com a realidade. Por fim, realizamos análises de interpretabilidade de modo a entender a relevância dos atributos considerados na modelagem do problema.

Abstract

The present master's work proposes, from a data-oriented approach, to build substitute models of machine learning with the main intention of using them in the context of the optimization of parameters of the injection process of aluminum wheels. In this way, a set of multivariate data from the injection and cooling process in the molds of aluminum wheels from automobilistic company was collected, totaling 65 variables. Then these data went through a series of treatments until they were used in the training of machine learning models. With regard to the process optimization problem, four objective functions were considered. One of these functions is associated with a classification problem, in which you want to discriminate between approved or failed wheels in the inspection process. The other objective functions were related to regression problems whose outputs to be explained were associated with process parameters. In our study, we investigated four machine learning methods, namely: logistic regression (only for classification), linear regression, random forest and neural networks. When optimized with all objectives together, the results were not satisfactory, but when focused on only one objective function, such as the cycle time reduction, the optimization curves were more consistent with reality. Finally, we performed analyzes of interpretability in order to understand the relevance of the attributes considered in modeling the problem.

Lista de ilustrações

Figura 1 – Classificação dos métodos mais importantes (comuns e especiais) para a fundição de metais e ligas metálicas (adaptado de (OCZOŚ; KAWALEC, 2012)).	22
Figura 2 – Diagrama esquemático mostrando o custo de fundição de produtos de liga de alumínio de acordo com o tipo de processo de fabricação e volume de produção (adaptado de (ASHTON <i>et al.</i> , 2009))	23
Figura 3 – Esquemático de uma máquina de injeção a baixa pressão para fabricação de rodas de alumínio. 1 - Molde; 2 - Forno de espera; 3 - Sistema de abastecimento e 4 material fundido.	24
Figura 4 – Peças produzidas pelo método <i>LPDC</i> a) radiador de óleo; b) peça de um motor de avião; c) roda de caminhão em alumínio (adaptado de (CZEKAJ <i>et al.</i> , 2016)).	24
Figura 5 – Comparação de técnicas de fundição com base em 4 fatores para aplicações automotivas (adaptado de (RAJAGOPAL, 1981)).	25
Figura 6 – Exemplificação de ferramenta para um processo <i>LPDC</i> em rodas (adaptado de (ZHANG <i>et al.</i> , 2005)).	25
Figura 7 – Exemplo de um conjunto de macroporosidade em uma roda: (a) uma imagem de raio-x macroscópica da área do cubo mostrando conjunto de macroporosidade; (b) um raio seccionado mostrando um conjunto de macroporosidade; (c) uma projeção de raios-x bidimensional da macroporosidade (adaptado de (ZHANG <i>et al.</i> , 2005)).	26
Figura 8 – a) Variação na solubilidade do hidrogênio com a temperatura em alumínio líquido e sólido; b) um exemplo de microporosidade de hidrogênio na roda <i>LPDC</i> A356. (adaptado de (ZHANG <i>et al.</i> , 2005)).	26
Figura 9 – Variação anual da pressão parcial ambiente da água a uma operação de fundição de rodas típica da América do Norte (adaptado de (ZHANG <i>et al.</i> , 2005)).	27
Figura 10 – Principais subáreas do aprendizado de máquina.	30
Figura 11 – Representação gráfica da função sigmoide.	32
Figura 12 – Exemplo da construção de uma árvore de decisão para o dataset Iris, utilizando apenas duas variáveis, comprimento e largura da pétala.	34
Figura 13 – Esquema geral de uma floresta aleatória.	35
Figura 14 – Modelo Perceptron desenvolvido elaborado por McCulloch e Pitts (1943).	36
Figura 15 – Representação gráfica das funções de ativação.	37
Figura 16 – Modelo básico de uma MLP.	37
Figura 17 – Três formatos de modelos treinados de ML.	38

Figura 18 – Diferentes curvas de erro de treinamento para o conjunto de dados de teste e treino.	39
Figura 19 – Exemplo de uma matriz de confusão com sua legenda ao lado.	40
Figura 20 – Exemplo de curva ROC construída ponto a ponto.	43
Figura 21 – Curva ROC para diferentes passos, juntamente com seus respectivos AUC.	43
Figura 22 – Diferentes formas de corrigir conjunto de dados desbalanceados.	44
Figura 23 – As explicações locais baseadas no TreeExplainer permitem uma ampla variedade de novas maneiras de entender a estrutura do modelo global. a) Uma explicação local baseada na atribuição de uma medida numérica de crédito a cada recurso de entrada. b) Ao combinar muitas explicações locais, pode-se representar a estrutura global mantendo a fidelidade local ao modelo original.	46
Figura 24 – Exemplos de ferramentas como saídas do pacote SHAP. a) Gráfico de força para apenas uma previsão do modelo b) Gráfico resumo para todas as previsões.	47
Figura 25 – Áreas para coletar dados.	49
Figura 26 – Principais classes de algoritmos meta-heurísticos.	50
Figura 27 – Principais passos de um processo de otimização evolutiva orientada a dados.	52
Figura 28 – a) Fluxograma para otimização com dados sintéticos; b) Fluxograma para otimização com dados reais de produção.	55
Figura 29 – Aproximações da função Ackley por uma rede MLP considerando diferentes número de amostras.	56
Figura 30 – Ciclo utilizado para cada iteração de um dado conjunto de amostras.	56
Figura 31 – Diagrama da aplicação do SHAP com modelo de floresta aleatória.	59
Figura 32 – Diagrama de testes projetado com Orange para treinamento do modelo de classificação AI1.	61
Figura 33 – Diagrama de testes projetado com Orange para treinamento dos modelos de regressão AI2, AI3 e AI4.	61
Figura 34 – Exemplo de distribuição dos pontos de otimização ao longo das iterações para um conjunto de 10^6 amostras: a) Conjunto de pontos aleatórios no início da otimização; b) Conjunto de pontos na 10° iteração; c) Conjunto de pontos na 45° iteração.	64
Figura 35 – Curva de otimização ao longo das 1000 iterações realizada pelo DE para um N dimensões.	65
Figura 36 – Mapa de calor da correlação entre as variáveis de vazão.	66

Figura 37 – a) Histograma de 5 variáveis de refrigeração com alta correlação; b) Histograma de 2 variáveis de refrigeração com alta correlação e mesma distribuição.	67
Figura 38 – Distribuição da varável sequência de produção para os grupos de rodas aprovadas e reprovadas.	67
Figura 39 – Temperatura dentro do molde em um ponto específico do ciclo para todo o conjunto de 14 mil rodas.	68
Figura 40 – Classificação das variáveis mais relevantes segundo coeficiente de SHAP.	69
Figura 41 – Classificação das variáveis mais relevantes segundo coeficiente de SHAP para predição de uma roda reprovada.	69
Figura 42 – Curvas de temperatura para o ciclo de apenas uma roda manufaturada.	70
Figura 43 – Matriz de confusão para cada modelo avaliado: a) Floresta Aleatória; b) Redes Neurais; c) Regressão Logística.	71
Figura 44 – Curva ROC para os modelos: a) Floresta Aleatória; b) Redes Neurais; c) Regressão Logística.	72
Figura 45 – Otimização final com DE para função objetivo completa.	74
Figura 46 – Otimização com apenas objetivo de redução da porcentagem de sucata - AI1.	75
Figura 47 – Otimização com apenas objetivo de redução do tempo de ciclo - AI4.	75

Lista de tabelas

Tabela 1 – Ilustração da probabilidade atribuída pelo classificador para o conjunto de dados de teste do exemplo.	42
Tabela 2 – Grupo de Atributos	58
Tabela 3 – Funções objetivos por modelo substituto	60
Tabela 4 – Objetivo alcançado versus quantidade de amostras.	63
Tabela 5 – Objetivo alcançado versus quantidade de amostras.	64
Tabela 6 – Melhor combinação de hiper-parâmetros para AI1.	72
Tabela 7 – Resultados do modelo substituto para o classificador AI1.	72
Tabela 8 – Resultados do modelo substituto AI2.	73
Tabela 9 – Resultados do modelo substituto AI3.	73
Tabela 10 – Resultados do modelo substituto AI4.	73

Lista de abreviaturas e siglas

IoT:	<i>Internet of Things</i>
AI:	<i>Artificial Intelligence</i>
GPU:	<i>Graphics Processing Unit</i>
ML:	<i>Machine Learning</i>
EA:	<i>Evolutionary Algorithms</i>
DE:	<i>Diferencial Evolution</i>
DDDM:	<i>Data Driven Decision Making</i>
FEA:	<i>Finite Element Analysis</i>
DOE:	<i>Design of Experiments</i>
LPDC:	<i>Low Pressure Die Casting</i>
MIT:	<i>Massachusetts Institute of Technology</i>
LR:	<i>Logistic Regression</i>
RF:	<i>Random Forest</i>
MLP:	<i>Multi Layer Perceptron</i>
TP:	<i>True Positive</i>
TN:	<i>True Negative</i>
FN:	<i>False Negative</i>
FP:	<i>False Positive</i>
TPR:	<i>True Positive Rate</i>
TNR:	<i>True Negative Rate</i>
ROC:	<i>Receiver Operator Characteristic</i>
FPR:	<i>False Positive Rate</i>
AUC:	<i>Area Under the Curve</i>

SMOTE: *Synthetic Minority Oversampling Technique*

LIME: *Local Interpretable Model-Agnostic Explanations*

DDEO: *Data-Driven Evolutionary Optimization*

PdM: *Predictive Maintenance*

RMSE: *Root Mean Squared Error*

Sumário

1	Introdução	17
1.1	Motivação e justificativa	18
1.2	Objetivos	19
1.3	Organização	19
2	Processo de fundição	21
2.1	Fundição a baixa pressão	22
2.2	Principais problemas de fundição de rodas	24
3	Decisão orientada a dados: um breve panorama	28
3.1	Visão geral da área de ciência de dados	28
3.2	Aprendizado de máquina	29
3.2.1	Métodos de aprendizado supervisionado	31
3.2.1.1	Regressão logística	31
3.2.1.2	Árvore de decisão	33
3.2.1.3	Floresta aleatória	34
3.2.1.4	Redes neurais artificiais	36
3.2.2	Métricas de avaliação	37
3.2.2.1	Sobre-ajuste e sub-ajuste	38
3.2.2.2	Matriz de confusão, tipos de erro e RMSE	39
3.2.2.3	Curva ROC	41
3.2.2.4	Conjunto de dados desbalanceados	42
3.2.2.5	Interpretabilidade de modelos	44
3.3	Otimização	48
3.3.1	Descrição do problema de otimização	48
3.3.2	Otimização por meta-heurística	49
3.3.3	Modelos substitutos	51
3.3.4	Otimização em fundição	51
4	Metodologia	54
4.1	Proposta de Solução	54
4.2	Experimentos com dados sintéticos: função Ackley	55
4.2.1	Experimento 1: Modelo Substituto Aplicado na Função Ackley Bi-dimensional	55
4.2.2	Experimento 2: Modelo Substituto Aplicado na Função Ackley N-Dimensões	57
4.3	Modelo substituto para otimização de parâmetros na injeção de rodas de alumínio	57
4.3.1	Preparação das variáveis	57

4.3.2	Estudo de relevância das variáveis	58
4.3.3	Função multiobjetivo	59
4.3.4	Avaliação dos modelos de ML	60
4.3.5	Otimização com DE	62
5	Experimentos e resultados	63
5.1	Experimentos com dados sintéticos	63
5.1.1	Experimentos com função Ackley bidimensional	63
5.1.2	Experimentos com função Ackley n-dimensões	64
5.2	Experimentos com dados reais	65
5.2.1	Análise Exploratória	66
5.2.2	Relevância das variáveis	68
5.2.3	Modelos substitutos	70
5.2.4	Otimização com DE	73
5.2.4.1	Otimização considerando 4 modelos	74
5.2.4.2	Otimização considerando a função AI1	74
5.2.4.3	Otimização considerando a função AI4	75
5.3	Resumo do capítulo	76
6	Conclusões e Perspectivas	77
	Referências	79
	Appendices	85
A	Funções Python	86
A.1	Função Multiobjetivo	86
A.2	Função DE	86

1 Introdução

A utilização de dados como arcabouço metodológico e estratégico na tomada de decisões empresariais já é algo relativamente comum e presente nos mais diversos setores tecnológicos e industriais. Nos últimos anos, a utilização de técnicas mais sofisticadas, o maior volume de dados (*Big Data*) e potencial de processamento das informações, trouxeram luz à chamada ciência de dados moderna que vem assumindo um protagonismo tanto na academia quanto na indústria.

A geração e a disponibilização de dados são a base para o desenvolvimento dessas novas técnicas. Atualmente existem diversas tecnologias que contribuem para essa gigantesca e complexa geração de informações que adquirem dados sobre o que fazemos, compramos, medimos e até pensamos. Como exemplo de principais fontes de informação temos: web sites, aplicativos, bancos de dados públicos, casas inteligentes e sistemas de manufatura inteligente. Estes dois últimos itens foram possibilitados por meio do que é chamado de internet das coisas (IoT, do inglês *internet of things*), termo utilizado pela primeira vez em 1999 por Kevin Ashton ([ASHTON et al., 2009](#)), que denomina a tendência de que tudo ao nosso redor esteja conectado, produzindo dados a todo momento ([MCAFEE et al., 2012a](#)) ([WAMBA et al., 2015](#)). Assim, esse volume elevado e diversificado de dados tem servido como arcabouço principal na ciência de dados moderna, auxiliando nas tomadas de decisões empresariais e acadêmicas.

A inteligência artificial (AI, do inglês *artificial intelligence*) não é um conceito novo. Warren McCulloch e Walter Pitts, em 1943, foram fortes pioneiros nesse tema, a partir da elucidação de redes neurais artificiais com intuito de simular o funcionamento do cérebro humano ([PALM, 1986](#)). No entanto, somente com as novas tecnologias de unidade de processamento gráfico (GPUs, do inglês *graphics processing unit*) e o surgimento de técnicas mais avançadas de aprendizado de máquina (ML, do inglês *machine learning*), além de outras ferramentas de análise, observou-se um ganho substancial de desempenho, que possibilitou convergir toda essa gama de informação a fim de extrair tópicos relevantes para aplicações atuais ([GHAHRAMANI, 2015](#))([VAPNIK, 2013](#)).

Outra área já muito explorada na literatura científica, conhecida como algoritmos evolutivos (EA, do inglês *evolutionary algorithms*), se refere a um conjunto de métodos de otimização cujo funcionamento é inspirado na evolução biológica. Essas técnicas se baseiam em princípios de sistemas adaptativos que direcionam a busca com o intuito alcançar boas soluções. Um dos métodos mais conhecidos e discutidos na literatura, o algoritmo genético, foi introduzido por Holland in 1970 ([WHITLEY et al., 1996](#)). Outro algoritmo muito comum e de simples utilização é conhecido como evolução diferencial (DE,

do inglês *differential evolution*), proposto em 1997 por Storn e Price (STORN; PRICE, 1997), e que visa minimizar funções espaciais contínuas, não lineares e não diferenciáveis (PRICE, 2013).

Esta dissertação, conforme discutido na sequência, se fundamentará em uma abordagem orientada a dados, com a integração de conceitos de aprendizado de máquina e de um algoritmo evolutivo, visando a resolução de um problema prático da indústria automotiva. Estas técnicas serão aplicadas no processo de fabricação de rodas de alumínio na fundição sob baixa pressão com o objetivo de otimizar os parâmetros de entrada da máquina injetora, de modo que a otimização do processo se insira no contexto de uma abordagem cada vez difundida na indústria e que vem sendo referenciada como decisão orientada a dados (DDDM, do inglês *Data Driven Decision Making*) (PROVOST; FAWCETT, 2013).

1.1 Motivação e justificativa

Na última década, a aplicação de ciência de dados e métodos de otimização para resolver problemas relacionados à saúde, indústria, mercado financeiro, dentre outros, cresceu de maneira expressiva. Exemplos vão desde aplicações mais simples e cotidianas como recomendações de novos produtos, sugeridos de forma individual de acordo com o perfil do usuário, bem como modelos capazes de distinguir o COVID-19 da pneumonia a partir de imagens de tomografia computadorizada do tórax (LI *et al.*, 2020).

Do ponto de vista de aplicação industrial, a grande competitividade do mercado globalizado tem influenciado cada vez mais a busca por técnicas no auxílio de tomada de decisões das organizações, no intuito de otimizar tempo, recursos, qualidade e encontrar melhor combinação homem e máquina (SHRESTHA *et al.*, 2019). Nesse cenário, a motivação deste trabalho é buscar uma forma de otimizar os parâmetros para um processo de alta complexidade referente à fabricação de rodas de alumínio. Tal processo, que é realizado em alta temperatura (700°C), possui mais de 100 variáveis controláveis e não controláveis, que influenciam diretamente na qualidade da roda produzida. Nesse sentido, desenvolver uma abordagem sistêmica que seja capaz de sugerir parâmetros otimizados é importante tanto do ponto de vista acadêmico quanto industrial. Para superar esse desafio, novas ferramentas de otimização têm sido estudadas, envolvendo simulações, algoritmos evolutivos e aprendizado de máquina (GONDKAR; INAMDAR, 2014).

Um ponto central em nossa proposta é o uso da otimização evolutiva. Tais técnicas são especialmente de interesse em problemas nos quais a definição da função objetivo é complexa; tal complexidade pode ser decorrente de diversos aspectos, como a dificuldade de obtenção de uma expressão analítica. Por exemplo, é comum em problemas na indústria considerar simulações baseadas em análises de elementos finitos (FEA, do inglês *finite*

element analysis) ou realizar etapas de planejamento de experimentos (DOE, do inglês *design of experiments*). Em ambas as situações, a definição de uma função custo não é uma tarefa trivial. Uma alternativa em tais casos é, a partir dos dados observados desses experimentos ou simulações, considerar um modelo dito substituto, também conhecido como modelo de *surrogate* (EMMERICH *et al.*, 2002) ou metamodelo (ONG *et al.*, 2003) no presente trabalho, adotaremos o termo modelo substituto. Tais modelos servem como função objetivo, e podem, por exemplo, serem abordados por algoritmos evolutivos, que são capazes de conduzir um processo de otimização mesmo em situações complexas (JIN *et al.*, 2018).

Outros desafios que merecem ser destacados estão ligados ao alto custo computacional para o desenvolvimento de modelos de aprendizado de máquina em dados sintéticos de FEA ou também aos custos para se projetar experimentos em produção com DOE, ambos requerem a realização de processos de homologação para a sua implementação. Dessa forma, esta proposta de pesquisa visa lidar com tais dificuldades, a partir de otimização evolutiva orientada a dados históricos advindos do próprio processo produtivo. De modo mais específico, o objetivo da presente pesquisa é estabelecer uma metodologia para determinação dos parâmetros de um processo relacionado à injeção e solidificação de rodas de alumínio. Para tanto, é necessário inicialmente estabelecer um modelo substituto, o que é feito com auxílio de modelos de aprendizado de máquina. Em seguida, é proposta uma etapa de otimização considerando a meta-heurística evolução diferencial. Os dados utilizados para pesquisa e desenvolvimento foram disponibilizados por uma empresa do setor automotivo.

1.2 Objetivos

De forma generalista o objetivo do presente trabalho é desenvolver uma metodologia para otimizar os parâmetros do processo de fundição sob baixa pressão para rodas de alumínio. Para isso, em específico serão aplicadas técnicas de aprendizado de máquina em dados históricos do processo produtivo, para atuar em conjunto com uma meta-heurística para otimização. Esses modelos serão utilizados como função objetivo de um algoritmo evolutivo chamado de evolução diferencial. Este último buscará a otimização dos parâmetros de processo, sendo direcionado pelos pesos de cada objetivo do modelo de aprendizado de máquina supracitado. Por fim, esses objetivos serão ajustados para garantir redução do tempo de ciclo, estabilidade produtiva e qualidade na fabricação das rodas.

1.3 Organização

Este trabalho será organizado conforme disposto a seguir. O Capítulo 2 apresentará um breve panorama dos principais processos de fundição existentes focando no processo

utilizado na manufatura de rodas. O Capítulo 3 apresentará uma visão geral sobre as principais técnicas de ciência de dados utilizadas para auxílio na tomada de decisão focando nas técnicas aplicadas neste trabalho de pesquisa. Em seguida, o Capítulo 4 evidencia toda metodologia e experimentos da abordagem adotada. Os resultados e discussões são apresentados no Capítulo 5. Por fim, as conclusões obtidas neste trabalho serão exibidas no Capítulo 6.

2 Processo de fundição

A manufatura de peças pelo processo de fundição tem sido usada para fabricar armas, joias e utensílios desde civilizações antigas, justamente por ter a habilidade de produzir componentes de formato complexo, boa acuracidade dimensional e bom acabamento superficial (PATTNAIK *et al.*, 2012). Durante a segunda guerra mundial, a alta demanda por peças complexas nos Estados Unidos foi um forte motivador para o avanço tecnológico do processo de fundição. Um exemplo disso é a produção de pás das turbinas de aeronaves, que já não conseguia ser atendida pelas tradicionais ferramentarias da época (BARNETT, 1988). Desde então, diversas pesquisas têm sido conduzidas com intuito de desenvolver novas ligas (MERLIN *et al.*, 2009), novos processos, novas formas de caracterização, bem como entender as fontes de defeitos e discontinuidades que influenciam nas propriedades das peças (ZHANG *et al.*, 2005).

Atualmente, existem diversos processos de fundição, cuja seleção ocorre devido a uma série de características, tais como: tamanho da peça, complexidade da geometria, tipo de material, volume de produção, disponibilidade de equipamentos e principalmente, custos de produção. A Figura 1 apresenta os principais processos de fundição classificados de acordo com a característica de preenchimento dos moldes. O processo por gravidade ocorre utilizando a força gravitacional para auxiliar o preenchimento e solidificação das peças. Tal processo pode ser subdividido de acordo com o tipo de molde utilizado. Moldes permanentes, por exemplo, são utilizados em processos contínuo ou de alta produtividade. Já os moldes descartáveis ou colapsáveis, são fabricados utilizando materiais descartáveis, tais como areia ou cerâmica. Por outro lado, os processos de fundição sob pressão aplicam uma força adicional para auxiliar no preenchimento das peças e também para compensar a contração durante o processo de solidificação. (CZEKAJ *et al.*, 2016).

Um importante fator para auxílio na escolha do tipo de processo de fundição é a correlação de aspectos econômicos e tecnológicos. Assim, o volume a ser produzido da peça em questão tem alta influência no custo e na técnica a ser utilizada. A Figura 2 mostra um diagrama comparando o custo de fabricação de uma peça de acordo com o volume produzido. Como pode ser observado, para baixos volumes o método de fundição por moldes de areia é o de menor custo. Entretanto, em casos de grandes volumes (maiores que 1000 peças), a técnica mais econômica para ligas de alumínio é o de fundição sob alta pressão. O custo do processo de fundição sob baixa pressão (*LPDC*, do inglês *Low Pressure Die Casting*) é ligeiramente mais alto que o processo de alta pressão. Porém características tais como, geometrias do molde, características de equipamento e também microestrutura final do fundido tornam esse processo ideal para a fabricação de rodas (CZEKAJ *et al.*, 2016). Sendo assim, será dada maior ênfase a este processo produtivo

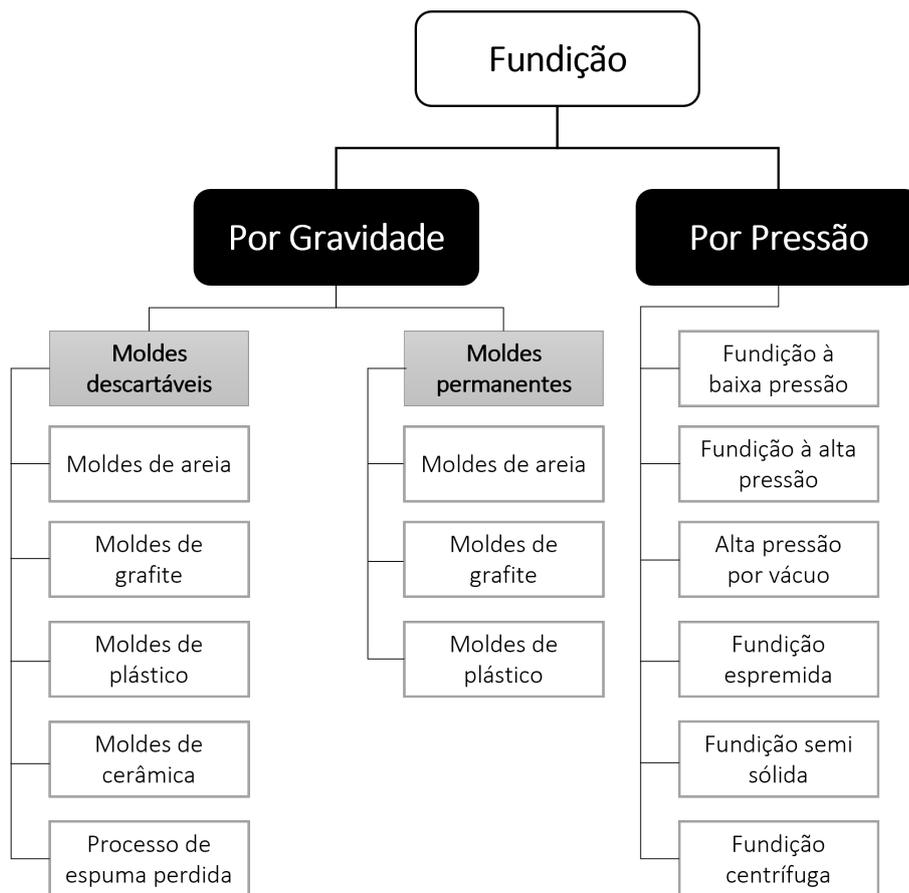


Figura 1 – Classificação dos métodos mais importantes (comuns e especiais) para a fundição de metais e ligas metálicas (adaptado de (OCZOŚ; KAWALEC, 2012)).

no decorrer deste trabalho.

2.1 Fundição a baixa pressão

Fundição sob baixa pressão é o termo utilizado para descrever o processo de manufatura que ocorre mediante a transferência do material fundido para o interior de um molde por meio da pressurização do forno de espera utilizando ar comprimido. Desta forma, o sistema consegue preencher regiões restritas dos moldes, mantendo contato direto para facilitar a solidificação, reduzir gases, bom refinamento da microestrutura e garantir as propriedades mecânicas. Além disso, existem um alto aproveitamento do material, uma vez que por meio deste processo é possível utilizar uma menor quantidade de sobremetal para posterior usinagem.

De forma geral, o processo de injeção e solidificação ocorre entre uma faixa de pressão de 0.2 a 1 *bar*. Uma máquina injetora de baixa pressão é apresentada na Figura 3. Este processo pode ser dividido em 4 partes principais, conforme apresentado a seguir: 1 ferramenta com o design da peça a ser produzida; 2 forno de espera com material fundido; 3 sistema de abastecimento (cadinho nesse caso) e 4 material fundido. Para que ocorra

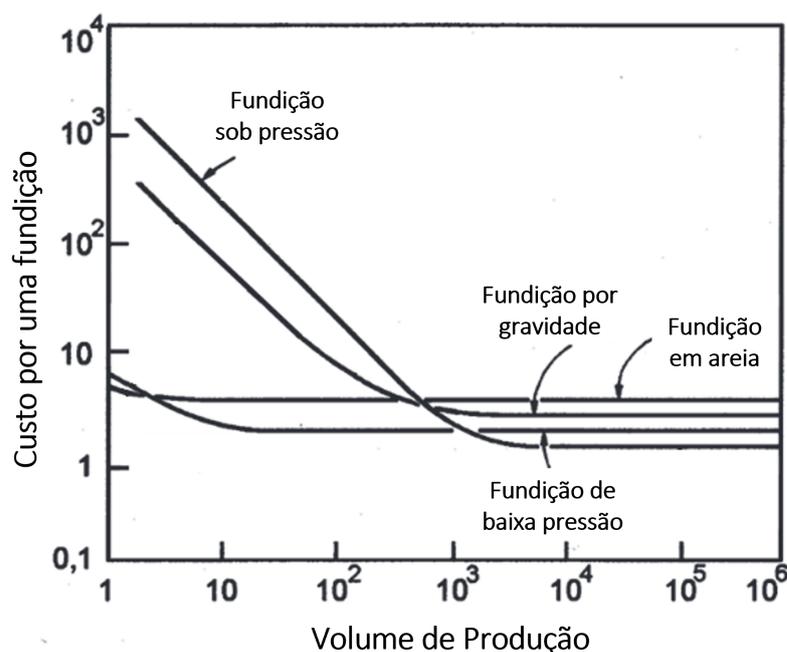


Figura 2 – Diagrama esquemático mostrando o custo de fundição de produtos de liga de alumínio de acordo com o tipo de processo de fabricação e volume de produção (adaptado de (ASHTON *et al.*, 2009))

a injeção de peças por esse sistema, ar comprimido (representado pelas setas azuis) é inserido no sistema com o intuito de pressionar o líquido forçando-o a subir o tubo e preencher o molde (ferramenta). Em seguida, o sistema de refrigeração que utiliza ar e água atua com objetivo de solidificar o produto. Após o resfriamento até uma temperatura pré estabelecida, a peça é extraída do molde e um novo ciclo se inicia.

Apesar do molde da Figura 3 representar uma roda, diversos outros produtos podem ser manufaturados com esse modelo de máquina, basta o redimensionamento do molde e a escolha apropriada do material a ser fundido (PATTNAIK *et al.*, 2012). A Figura 4 apresenta três exemplos de peças produzidas pelo processo de fundição a baixa pressão: a) é um radiador de óleo; b) parte de um motor de avião e c) roda de uma caminhão em alumínio. Percebe-se que ambas as peças possuem complexas geometrias e necessitam de boas propriedades mecânicas, importantes fatores para escolha do método de produção.

NO contexto do setor automotivo, uma variedade de peças podem ser produzidas pelo método de LPDC. A Figura 5, apresenta uma comparação de quatro técnicas de produção por fundição avaliadas em quatro fatores, com relação a aplicações no setor automotivo. Percebe-se que LPDC tem fortes pontos positivos com relação à complexidade e à economia. Por outro lado, deixa a desejar na segurança e flexibilidade de design (RAJAGOPAL, 1981).

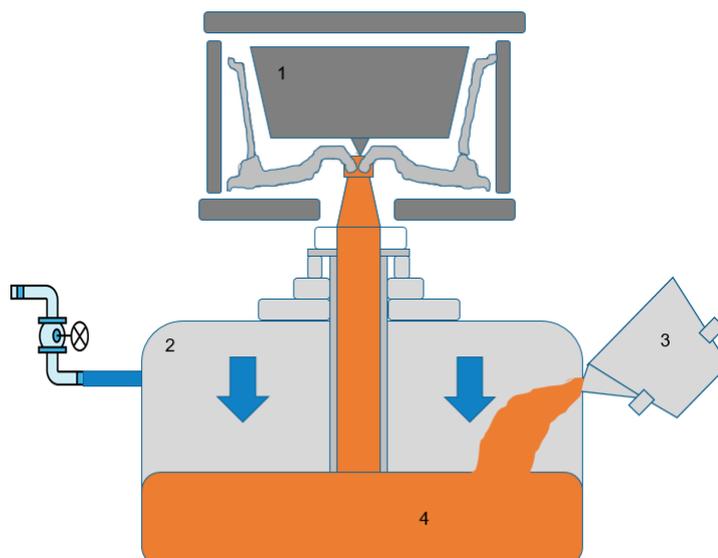


Figura 3 – Esquemático de uma máquina de injeção a baixa pressão para fabricação de rodas de alumínio. 1 - Molde; 2 - Forno de espera; 3 - Sistema de abastecimento e 4 material fundido.

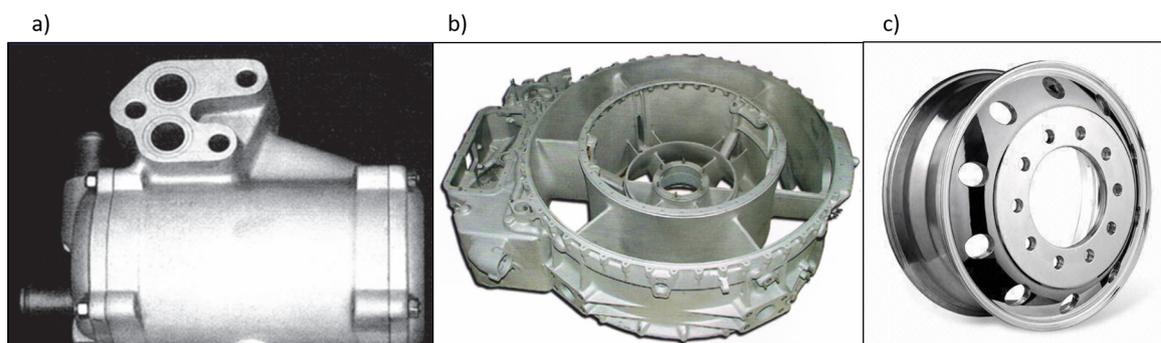


Figura 4 – Peças produzidas pelo método *LPDC* a) radiador de óleo; b) peça de um motor de avião; c) roda de caminhão em alumínio (adaptado de (CZEKAJ *et al.*, 2016)).

2.2 Principais problemas de fundição de rodas

Mesmo com todas as vantagens supracitadas em relação ao processo *LPDC*, manter tal produção em larga escala estável não é uma tarefa simples. Isso ocorre devido ao complexo sistema de injeção e solidificação direcional que deve acontecer. A Figura 6 representa o desenho esquemático de uma ferramenta utilizada para o processo de fabricação de rodas, normalmente feitas de aço e ferro fundido. Quatro partes principais compõem o molde: matriz (*botton core*), tobeira (*sprue or metal inlet*), gavetas (*side core*) e macho (*top core*). O sistema de injeção pressuriza o alumínio pela tobeira que depois passa pela matriz até o enchimento das laterais do aro. Em seguida, refrigerações atuam para solidificar direcionalmente do aro para o cubo to o material. Tubos de refrigeração a ar e água atuam em diversas configurações e posições para otimizar tal processo. Até

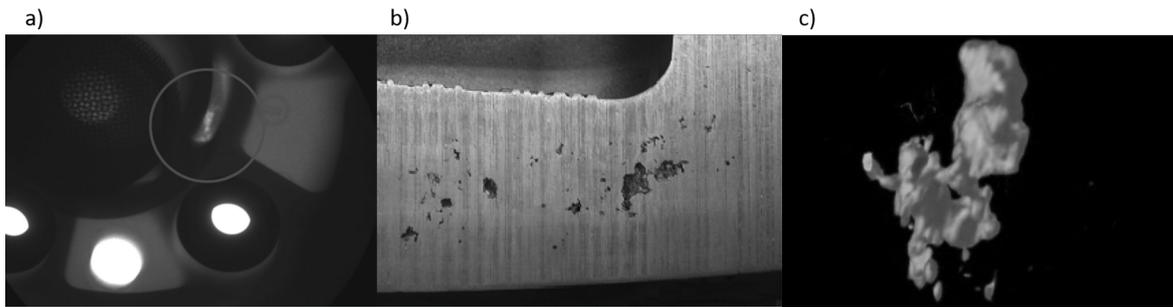


Figura 7 – Exemplo de um conjunto de macroporosidade em uma roda: (a) uma imagem de raio-x macroscópica da área do cubo mostrando conjunto de macroporosidade; (b) um raio seccionado mostrando um conjunto de macroporosidade; (c) uma projeção de raios-x bidimensional da macroporosidade (adaptado de (ZHANG *et al.*, 2005)).

processo podem influenciar na formação das microporosidades no produto fundido, sendo a captação de hidrogênio do ar pelo alumínio líquido o principal responsável por este tipo de defeito. A variação de solubilidade do hidrogênio comparado com a temperatura é apresentado na Figura 8 a). Percebe-se que, na fase líquida, o alumínio possui grande afinidade com o oxigênio e é capaz de decompor as moléculas do vapor de água da atmosfera conforme reação química 2.1. Assim, ocorre uma captação gradual do hidrogênio com maior exposição ao ar ambiente até que o limite de solubilidade seja alcançado na faixa de $0.69 \text{ cc } H_2/100g \text{ Al}$. Uma vez que o limite é alcançado, nucleações podem ocasionar pequenas porosidades de gás hidrogênio no sólido, como mostra a Figura 8 b). Zhang et al. identificou uma maior incidência de defeitos associados à microporosidade entre os meses de julho e setembro em uma empresa instalada na América do Norte (ZHANG *et al.*, 2005).

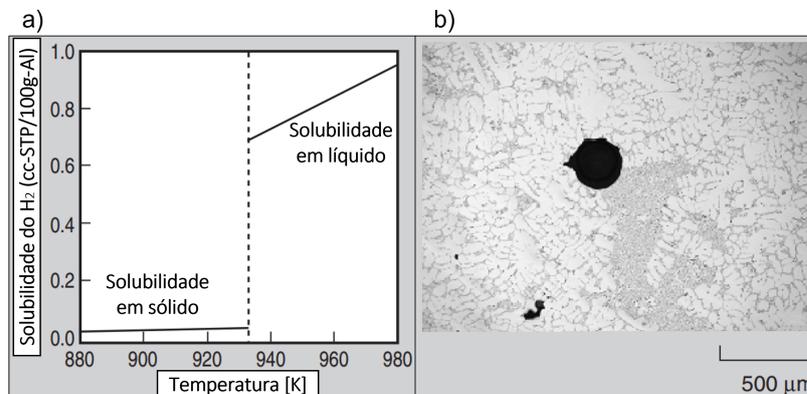


Figura 8 – a) Variação na solubilidade do hidrogênio com a temperatura em alumínio líquido e sólido; b) um exemplo de microporosidade de hidrogênio na roda LPDC A356. (adaptado de (ZHANG *et al.*, 2005)).

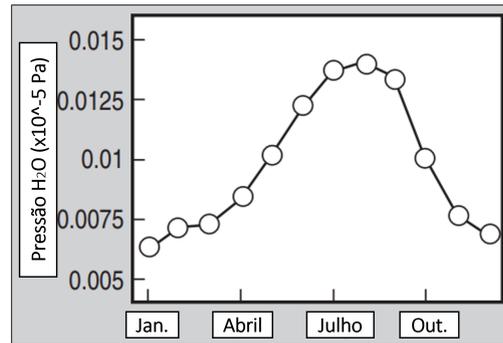


Figura 9 – Variação anual da pressão parcial ambiente da água a uma operação de fundição de rodas típica da América do Norte (adaptado de (ZHANG *et al.*, 2005)).

De acordo com o autor, isso ocorre devido à variação de temperatura ambiente e umidade relativa ao longo do ano, que implica diretamente na pressão parcial da água no ar, como representado pela Figura 9. Ou seja, durante o verão, a existência de chuvas é maior, tal como altas temperaturas, fatores importantes que influenciam a tendência de formação do gás hidrogênio na liga de alumínio.

Essas condições de clima que variam ao longo do ano compõem as chamadas variáveis não controláveis do processo. Elas impactam direta ou indiretamente no processo e até mesmo no desenvolvimento de parâmetros como entrada da máquina injetora podem ser afetadas por essas variáveis. Desta forma, definir uma folha de processo para um ciclo rápido e com qualidade na manufatura não é uma tarefa trivial, uma vez que trata-se de um problema de otimização multiobjetivo com diversas variáveis. Essas variáveis têm por objetivos definir os parâmetros de injeção e controle de temperatura do metal fundido, configuração do sistema de válvulas que controlam a solidificação e controle da temperatura da tobeira já que esta possui uma resistência própria.

Na literatura, os principais trabalhos estão relacionados ao desenvolvimento de parâmetros de processo por meio de simulações FEA com o intuito de se encontrar a melhor condição, em questão de tempo de processo e menor probabilidade de porosidade no fundido (DHISALE *et al.*, 2022). Em outros casos, utiliza-se aprendizado de máquina em conjunto com simulações FEA na construção de um modelo substituto para otimização (SHAHANE *et al.*, 2020) e (ZHANG *et al.*, 2007b). No entanto, trabalhar com modelos de ML (para atuarem como modelos substitutos) com base em dados de simulações é custoso computacionalmente e difícil quando se tem poucas amostras (SHAN; WANG, 2010). Por outro lado, desenvolver experimentos no próprio ambiente de fundição demanda tempo e dinheiro das empresas. Nesse contexto, desenvolver um método que utiliza dados provenientes do chão de fábrica aliado a algoritmos de aprendizado de máquina e de otimização se mostram relevantes tanto do ponto de vista industrial quanto acadêmico.

3 Decisão orientada a dados: um breve panorama

Esse capítulo discorrerá sobre aspectos atuais da ciência de dados e suas interações com a área de auxílio na tomada de decisão. Inicialmente, será apresentada uma visão geral histórica das tecnologias que culminaram na ciência de dados atual. Em seguida, serão discutidas as principais ferramentas de aprendizado de máquina e suas métricas de avaliação dos modelos. Por fim, serão discutidas as abordagens de otimização consideradas nesse trabalho para tratar o modelo substituto fornecido por técnicas de tipos de aprendizado de máquina.

3.1 Visão geral da área de ciência de dados

De fato, o *Big Data*, termo que define um imenso conjunto de dados caracterizado com elevado volume, velocidade, variedade e valor (CHEN *et al.*, 2014), tem papel fundamental no desenvolvimento dessas e outras tecnologias para trabalho na área dados. Uma vez que o intuito é ter ferramentas adequadas para extração, processamento e modelagem dessas informações, com o intuito de atuar na tomada decisão seja empresarial ou acadêmica. Por isso, o *Big Data* tem sido visto como a nova fronteira de inovação, competitividade e produtividade (MANYIKA *et al.*, 2011).

Nos últimos 10 anos os setores industriais também se interessaram pelo potencial que o *Big Data* pode trazer de retorno. Porém, frequentemente essas empresas também precisam se preocupar com novos investimentos em sensoriamento, automação e poder computacional. Principalmente empresas de manufatura que ainda detêm tecnologias antigas, que não fazem parte dessa nova era de indústria 4.0, precisam se adequar, a fim de gerar novas análises para redução de custo e otimização da produção (LASI *et al.*, 2014). Algumas dessas adequações são justamente novos investimentos em manufatura aditiva, internet das coisas (*IoT*), realidade aumentada, simulações, computação em nuvem, coleta de dados do processo e segurança da informação (CHEN *et al.*, 2014).

Como citado por Andrew McAfee and Erik Brynjolfsson na *Harvard Business Review* (MCAFEE *et al.*, 2012b), o desempenho das empresas podem melhorar radicalmente com a exploração de informações advindas desse grande fluxo de dados. Em outras palavras, trabalhar na cultura de tomada de decisão orientada a dados (DDDM) pode ser fator essencial para manter competitividade no mercado. Fawcett e Provost (2018) mostraram que as empresas dos EUA que trabalham com essa filosofia orientada a dados possui uma produtividade cerca de 4 a 6% maior que as demais. Fato que evidencia o quão importante

é instaurar essa nova tendência de dados dentro das empresas.

Com esses dados disponíveis em mãos, é necessário ter acesso a arcabouço de ferramentas corretas para interpretação e análise. De fato, é importante recorrer a ferramentas de diversas áreas como por exemplo de aprendizado de máquina, pesquisa operacional e meta-heurística, tais como outros segmentos, essenciais para solução dos problemas de pesquisa e desenvolvimento. Cabe destacar que a matemática e estatística funcionam como a base teórica fundamental dessas áreas, seguindo princípios lógicos para melhor adequação do sistema final (MAURO *et al.*, 2015).

Em seguida, também se fazem necessárias habilidades de ciência da computação para servir como ambiente de trabalho. Isso porque todas as técnicas supracitadas precisam de um local para serem aplicadas, que muitas vezes acontecem por linguagem de programação Python, R, Java, SQL, ou mesmo softwares *Drag and Drop* da atualidade como ML Azure Studio ou Orange. Somente com essas ferramentas é possível começar a divagar no oceano de dados e extrair *insights* para auxílio na tomada de decisão ou mesmo automação de processo (MCAFEE *et al.*, 2012b).

Por fim, toda essa tecnologia será direcionada para solucionar problemas diversos na sociedade. Assim, conhecimento da área a ser explorada para projetos é essencial para modelagem do problema de negócio, uma vez que um dos trabalhos do profissional da área de dados é levantar informações mínimas de coleta de dados para desenvolvimento do projeto. Alguns importantes exemplos dessas aplicações são: imagem do buraco negro criada a partir de 8 petabytes de dados de diferentes telescópios (FALCKE, 2017); visão computacional aplicada a diversas áreas da medicina (SHEN *et al.*, 2017); sistemas de recomendação em diferentes segmentos de negócio (MA; SUN, 2020); e também otimização de processos em manufatura (WANG *et al.*, 2020).

Um das áreas promissoras para aumentar a eficiência de empresas de manufatura é justamente aprendizado de máquina. Isso ocorre devido a capacidade dos algoritmos de "aprender" com os dados de entrada e auxiliarem na predição e entendimento dos problemas. Muitos casos são relatado para manutenção preditiva (SUSTO *et al.*, 2014), interpretação do processo de manufatura (UYAN *et al.*, 2022), otimização de parâmetros (DHISALE *et al.*, 2022), entre outros que visam melhoria na produção. O fato é que essas novas tecnologias voltadas as ciências de dados vieram para permanecer nessa era da inteligência artificial (DAUGHERTY; WILSON, 2018).

3.2 Aprendizado de máquina

O termo aprendizado de máquina (ML, do inglês *machine learning*) foi utilizado pela primeira vez em 1959 por Arthur Samuel, engenheiro do MIT (*Massachusetts Institute of Technology*). Segundo ele, essa subárea da inteligência artificial se conceitua como

“o campo de estudo que os computadores têm habilidade de aprender sem que sejam programados para tal”. Em outras palavras, *ML* é um ramo em evolução de algoritmos computacionais projetados para emular a inteligência humana aprendendo com dados do ambiente ao redor (MOHRI *et al.*, 2012).

Um fator importante é conhecer como essas máquinas são capazes de compreender os dados, encontrar padrões e depois predizê-los. Por trás de todo modelo de *ML* existe um algoritmo lógico com ferramentas estatísticas e matemáticas que direcionam a aprendizagem. Muitas vezes esses algoritmos fazem parte de bibliotecas construídas com linguagem de programação, tal como *Python*, *Java*, *R*, *C*, *C++*, *JavaScript*, *Escala*, entre outras que servem como base fundamental para desenvolvimento dos modelos.

Dentro do aprendizado de máquina existem diferentes formas nos quais os modelos podem aprender, seja por diferentes conjuntos de dados ou área de aplicação. Alguns dos principais tipos de *ML* são apresentados na Figura 10 (KOTSIANTIS *et al.*, 2006b) e, detalhados na sequência:

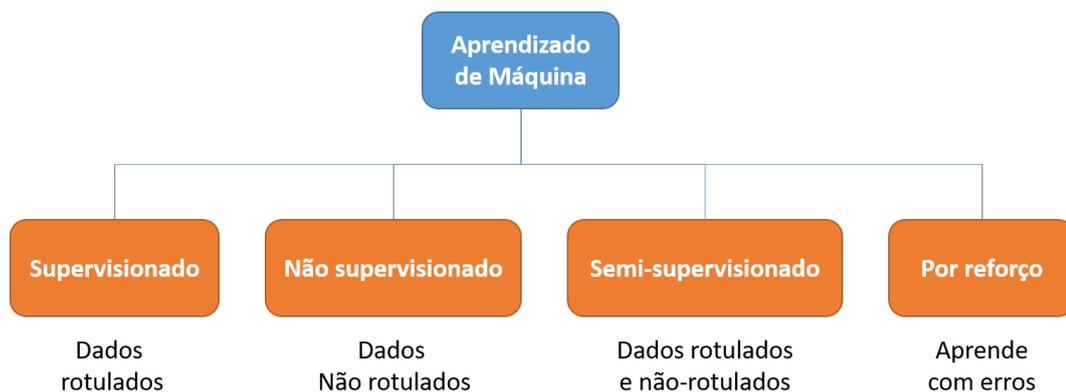


Figura 10 – Principais subáreas do aprendizado de máquina.

- **Supervisionado:** Neste tipo de aprendizado, no momento de treinamento é fornecido ao modelo o conjunto de dados de entrada e também o resultado (dados rotulados), que é chamada de variável alvo. Uma analogia simples, é quando você ensina uma criança o que é uma laranja, mostrando a cor, formato, gosto, cheiro e em um próximo evento ela saberá como definir se uma fruta é laranja ou não. Dentro desse segmento, podem haver diferentes problemas como de regressão e classificação, no qual o primeiro tenta prever uma faixa numérica (variável alvo), como por exemplo preços de terrenos e o segundo prevê categorias, ou seja, se um cliente deve receber ou não um crédito por exemplo.
- **Não supervisionado:** Neste tipo de aprendizado, a variável alvo não é apresentada, é selecionado apenas o conjunto de dados com todas as características. Em alguns casos, são fornecidos indícios para o modelo de como deve ser a saída. Por exemplo,

o algoritmo de clusterização tem o papel de dividir os dados em grupos, porém, via de regra, com o conhecimento prévio do número de grupos.

- **Semi-supervisionado:** Ocorre quando existem dados rotulados e não rotulados para treinamento do modelo. Via de regra, a proporção de dados rotulados é consideravelmente menor que a proporção de dados não rotulados.
- **Por Reforço:** No aprendizado por reforço, o algoritmo aprende com respostas a partir do próprio ambiente. Ou seja, ao longo do treinamento o modelo, já produz alguns resultados que servem para ajustes de parâmetros em um novo treinamento até chegar no melhor resultado. Um exemplo clássico se refere ao aprendizado do jogo xadrez, a partir da exploração de recompensas decorrentes de movimentos considerados exitosos (MARSLAND, 2011).

A metodologia de aprendizado de máquina é selecionada com base em diferentes fatores, a começar pela definição do problema a ser resolvido (FAWCETT; PROVOST, 2018). A etapa seguinte trata-se do levantamento dos dados relacionados a tal problema, como por exemplo imagens, dados estruturados ou não. Em algumas situações, pode-se mostrar necessária a coleta de novas variáveis, que segundo o problema define como relevantes para tal sistema.

3.2.1 Métodos de aprendizado supervisionado

No caso do presente trabalho o problema principal a ser estudado é a otimização de parâmetros de produção de rodas de alumínio no ambiente de fundição. Os dados disponíveis nessa aplicação constam de coletas em série temporal ao longo da manufatura, tal como do molde projetado que é utilizado nas máquinas injetoras de baixa pressão. Assim, tem-se características específicas de produção e molde, mas também rótulos de saída com classes aprovadas e reprovadas de rodas, tal como estabilidade do processo. Dessa forma, modelos supervisionados para prever tais saídas podem ser desenvolvidos. Por esse motivo são apresentados em seguida quatro técnicas de aprendizado escolhidas para o desenvolvimento desse trabalho.

3.2.1.1 Regressão logística

A regressão logística (LR, do inglês *Logistic Regression*), também conhecida como modelo logit, é muito utilizada para construir modelos preditivos com saídas categóricas, a partir de variáveis contínuas ou discretas. Na grande maioria dos estudos, a variável alvo é binária, porém, quando essa variável não é binária, pode ser utilizada a regressão logística multi-nominal (DREISEITL; OHNO-MACHADO, 2002). A LR pode ser vista como uma técnica de classificação em aprendizado de máquina, pois possui como objetivo

classificar instâncias de acordo com um conjunto de atributos (variáveis do sistema). A fundamentação matemática principal para melhor entendimento dessa técnica é a função logística (ilustrada graficamente na Figura 11) (JR *et al.*, 2013). A Equação (3.1) representa a função inversa da função logit, no qual t é a combinação linear das variáveis de entrada do modelo e os parâmetros do modelo (conhecidos como pesos).

$$f(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}} \quad (3.1)$$

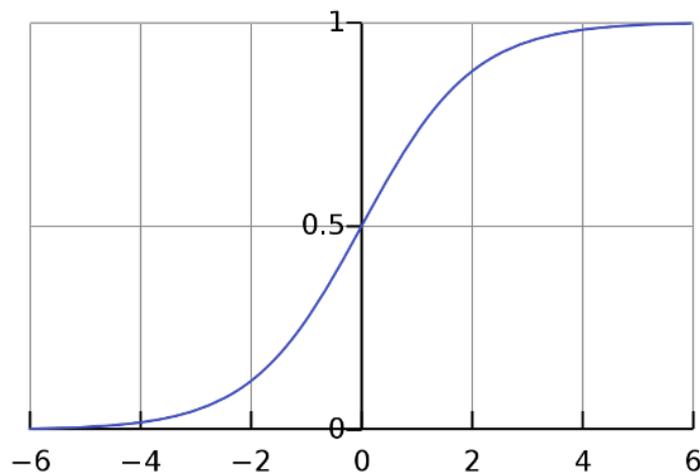


Figura 11 – Representação gráfica da função sigmoide.

A Equação (3.1) também denominada função sigmoide, é uma curva "S", delimitada entre intervalo 0 e 1, para qualquer t pertencente aos reais. Isso garante que a saída $f(t)$ do modelo permaneça no intervalo e tenha uma interpretação probabilística. As áreas nas quais pequenas variações nos valores de x causam grandes mudanças em valores de y representam áreas de maior probabilidade de mudança de estado da variável y em função de x (ZANINI, 2007).

A regressão logística pode ser considerada como uma das ferramentas mais clássicas em aprendizado de máquina para a tarefa de classificação, e vem sendo utilizada em um vasto número de aplicações, em áreas como: medicina, permitindo por exemplo determinar os fatores que caracterizam um grupo de indivíduos doentes em relação a indivíduos saudáveis, ciências atuariais, econometria dentre outras.

Um ponto a ser destacado é a robustez que a técnica regressão logística pode ter para *outliers*, termo esse utilizado para caracterizar pontos discrepantes em relação à distribuição dos dados. Tal robustez fortalece a segurança do modelo com relação a sensibilidade, que pode ocasionar uma explicação ruim dada pela função regressão de saída, fazendo com que o modelo não represente as variáveis analisadas do problema (FAWCETT; PROVOST, 2018).

3.2.1.2 Árvore de decisão

As árvores de decisão em aprendizado de máquina são ferramentas muito populares e que apresentam uma característica interessante que é a possibilidade de inspeccionamento de todas as camadas de segmentação, uma vez que é possível seguir uma classificação desde a raiz até as folhas de forma transparente. A ideia básica do algoritmo é dividir a classificação com base em características dos dados de entrada começando pela raiz (base), até chegar as folhas no qual possui a resposta final que deseja-se (MARS LAND, 2011).

Existem diferentes formas desse algoritmo, porém elas derivam do mesmo princípio, no qual constroem a árvore em “*Greedy Manner*” (busca gananciosa) começando então pela raiz e escolhendo a variável mais informativa a cada passo. A primeira variável é mais importante e leva em consideração todos os pontos do espaço, tentando encontrar o ponto que maximiza a pureza dos dados de acordo com um critério, que será explicado mais adiante.

Para ilustrar como a árvore de decisão particiona o espaço, será utilizado como exemplo o conjunto de dados de flores iris (FISHER, 1988), utilizado usualmente no contexto de classificação de três espécies: Setosa, Virgínica e Versicolor. Tomemos como exemplo a realização da classificação considerando duas variáveis de entrada(ou atributos): *petal length* (comprimento da pétala) e *petal width* (largura da pétala). Suponha então que o primeiro ponto de corte seja o comprimento em 2,45 cm (eixo x), conseguindo separar apenas a espécie Setosa abaixo desse valor, representado pela cor marrom da Figura 12. Entretanto as outras duas espécies ainda continuam misturadas, logo, deve-se utilizar outro ponto de corte. Outra opção seria a largura de 1,75 cm (eixo y), identificando melhor a classe Versicolor (verde) e Virgínica (roxa). Assim, o algoritmo continua a classificação até algum critério de parada (profundidade, folhas entre outros), como mostra a Figura 12.

Portanto, de maneira resumida, a árvore pode ser entendida como uma série de perguntas, cujo objetivo é estreitar a resposta final. Assim, uma métrica importante para divisão dos nós da árvore de decisão é o ganho de informação, cálculo que indica a quantidade de informação ganha após a inserção de um nó na árvore que efetua uma decisão a partir de um atributo, isto é, o “benefício” obtido ao se utilizar este atributo (variável) ao invés dos outros no conjunto (MARS LAND, 2011).

Uma forma muito conhecida de fazer o cálculo do ganho de informação é a partir do conceito conhecido como entropia, calculada da seguinte maneira:

$$Entropy(p) = - \sum_i p_i \log_2 p_i. \quad (3.2)$$

A entropia pode ser utilizada como uma medida de informação, de modo que, no contexto

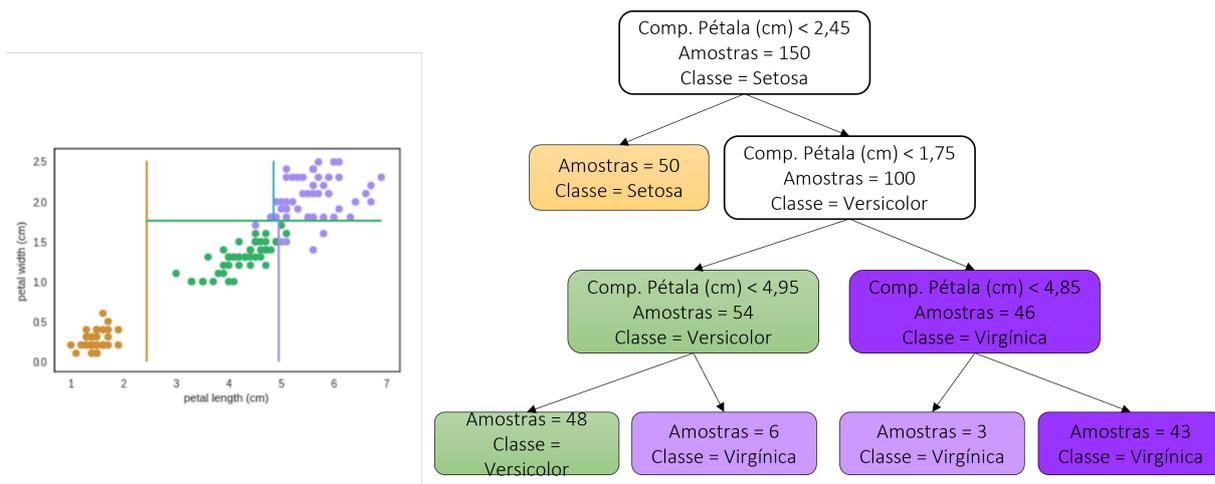


Figura 12 – Exemplo da construção de uma árvore de decisão para o dataset Iris, utilizando apenas duas variáveis, comprimento e largura da pétala.

Fonte: [Medium](#).

de árvore de decisão, a ramificação que tiver maior entropia é escolhida (QUINLAN, 1986)(QUINLAN, 1993).

Outra métrica popular para cálculo do ganho de informação é o coeficiente Gini, que basicamente tenta medir o grau de impureza do sistema antes e depois da ramificação. O coeficiente de Gini é utilizado nos algoritmos CART (BREIMAN, 2017) e IBM IntelligentMiner (CABENA *et al.*, 1999).

Finalmente, cabe destacar que a árvore de decisão é uma importante ferramenta de aprendizado de máquina que fornece base para o entendimento de outros algoritmos, tal como os conhecidos métodos *ensemble*. Alguns desses são floresta aleatória (RF, do inglês *Random Forest*) além de *AdaBoost* e *Gradient Boosting*, também são conhecidos como “*Decision by Committee*”, no qual um conjunto de classificadores treinados individualmente cooperam com suas decisões no resultado final (MARQUÉS *et al.*, 2012).

3.2.1.3 Floresta aleatória

A ideia principal do conceito conhecido como floresta aleatória é a utilização de diferentes árvores de decisão, em conjunção com a introdução de um componente aleatório, cujo objetivo é diminuir a variância, sem criar nenhuma tendência e também retirando a necessidade de “podar” (definir parâmetro de profundidade) a árvore (MARS LAND, 2011). Um problema ainda recorrente para esse tipo de modelo é o parâmetro que define a quantidade de árvores que serão consideradas.

Uma vez que o modelo de floresta é constituído de diversas árvores de decisão, a saída do classificador requer a definição de um comitê, que visa agregar os resultados individuais de cada árvore. A Figura 13 ilustra o conceito geral da floresta aleatória, em que para uso em sistemas de classificação o resultado é, via de regra, escolhido em votação

majoritária. Já para sistemas de regressão utilizada agregação, geralmente é feita a partir de uma média aritmética das saídas de todas as árvores.

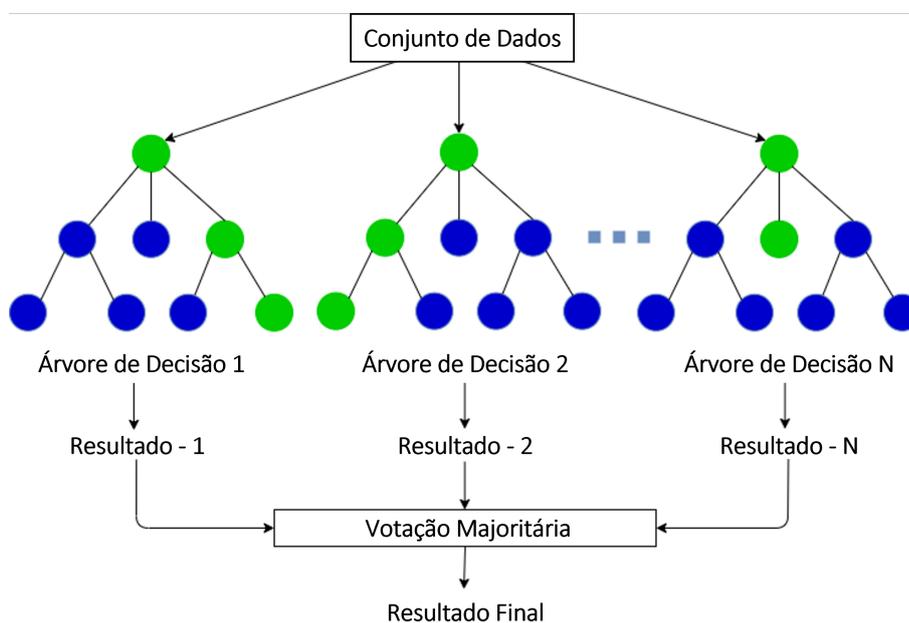


Figura 13 – Esquema geral de uma floresta aleatória.

Fonte: Adaptado de [Medium](#)

Em termos de execução, a floresta aleatória pode ser resumida pelos seguintes passos:

- Seja N o número de árvores selecionadas, para cada n pertencente N têm:
 - Criar uma nova amostra do conjunto (com uso de Bootstrap) de treinamento.
 - Utilizar essa amostra para treinar o modelo n da árvore de decisão.
 - Em cada nó específico da árvore, será selecionado de forma aleatória um número m de variáveis e calculado o ganho de informação (ou Índice Gini) para apenas esse conjunto, em seguida é escolhido a de maior valor para ser um nó.
 - Repete o último item até terminar de toda ramificação da árvore. Quando finalizado, uma nova árvore n começará a ser construída.

Trabalhos recentes sugerem que bons resultados podem ser esperados utilizando a técnica de floresta aleatória. Por exemplo, esse tipo de classificador tem se tornado popular dentro da comunidade de sensoriamento remoto (BELGIU; DRĂGUȚ, 2016). Além disso, o modelo também demonstra ser robusto em conjunto de dados com alta dimensionalidade e multicolinearidade, além é claro de ter rápido treinamento e apresentar uma boa robustez ao problema de sobreaprendizado.

3.2.1.4 Redes neurais artificiais

Sistemas matemáticos bioinspirados são comuns na ciência. Exemplos incluem os populares algoritmos genéticos, algoritmos otimização de colônia de formigas, e as redes neurais artificiais. Uma das inspirações para essas redes vem do cérebro humano, que é composto por um conjunto expressivo de conexões de neurônios, unidades fundamentais de processamento. De fato, apenas um neurônio pode ser entendido como um sistema que recebe um sinal e fornece uma resposta como saída. Entretanto, quando pensamos em uma rede totalmente conectadas por neurônios, complexas atividades podem ser realizadas, tal como comunicação e visão, de uma forma muito rápida como o próprio reconhecimento de uma face pelo olho humano (ZOU *et al.*, 2008).

Foi inspirado nesse neurônio do cérebro humano que um modelo artificial simples chamado *Perceptron* foi elaborado por McCulloch e Pitts (1943). Na Figura 14 um perceptron de três entradas é ilustrado. Tal sistema recebe sinais de entrada, denotados por x_1, x_2 e x_3 , com seus respectivos pesos w_1, w_2 e w_3 , analogamente aos sinais das sinapses. Em seguida, é realizada uma soma ponderada, que a partir de um limite (*threshold*) a unidade entrega uma saída binária 0 ou 1, dada por um função de ativação, que, no caso, clássico do perceptron é simplesmente a comparação com um limiar dado por T . Interessante notar que a função sigmoide, mencionado na seção anterior referente à regressão logística, pode ser vista como uma versão suavizada da função original de ativação.

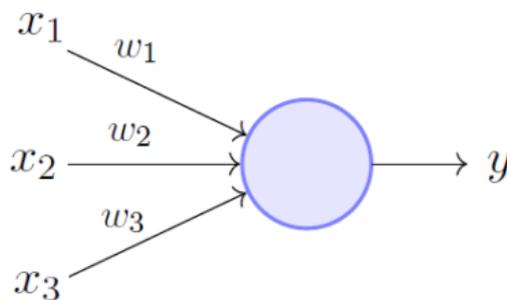


Figura 14 – Modelo Perceptron desenvolvido elaborado por McCulloch e Pitts (1943).

$$y = \begin{cases} 0, & \text{se } \sum_{i=0}^n w_i x_i > T \\ 1, & \text{se } \sum_{i=0}^n w_i x_i < T \end{cases} \quad (3.3)$$

Assim como o cérebro, conforme mais neurônios fizerem parte da rede, processos mais complexos podem ser efetuados. Um modelo famoso de aprendizado de máquina é o *Perceptron* de múltiplas camadas (MLP, do inglês *Multi layer perceptron*), representado pela Figura 16. Tem-se primeiro a camada de entrada, em seguida as camadas escondidas e por fim a camada de saída. O processo de aprendizado acontece com ajuste dos pesos de acordo com os padrões de entrada, ou seja, de acordo com os exemplos.

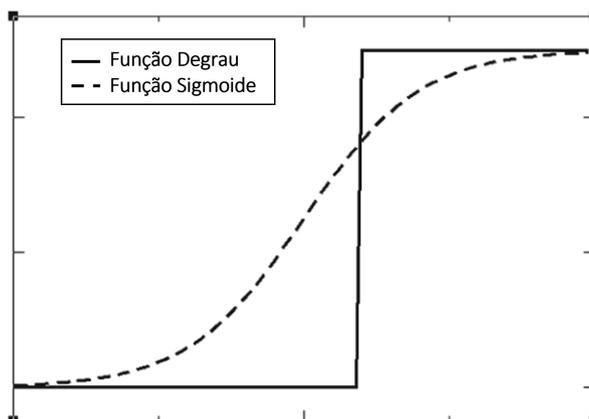


Figura 15 – Representação gráfica das funções de ativação.

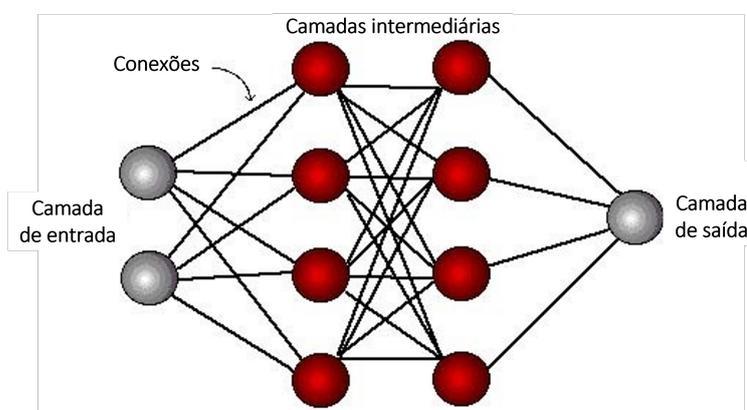


Figura 16 – Modelo básico de uma MLP.

Um importante conceito relacionado às redes MLP refere-se ao processo de ajuste de seus pesos, que é, via de regra, feito pelo algoritmo conhecido como *backpropagation*, abreviação de *backward propagation of errors*. Em breves termos, o backpropagation pode ser visto como aplicação do método do gradiente descendente com ferramenta do problema de otimização associado ao aprendizado feito pela MLP. Mais precisamente, dada uma rede neural artificial e uma função de erro, o método calcula o gradiente da função de erro (também conhecida como função perda), o método backpropagation calcula o gradiente da função de erro em relação aos pesos da rede neural.

3.2.2 Métricas de avaliação

Uma etapa fundamental para a aplicação de modelos de aprendizado é a validação dos resultados obtidos por meio de métricas específicas de avaliação do rendimento das predições dos modelos. O estabelecimento de métricas não é uma tarefa simples e depende diretamente do problema em questão. Desta forma, definir uma boa métrica significa saber como direcionar o aprendizado dos modelos e garantir melhor resultado final. As métricas utilizadas em aprendizado de máquina servem também como uma importante maneira de identificar a ocorrência de problemas usuais no treinamento, incluindo sobre-

aprendizado, sub-aprendizado e problemas relacionados à presença de conjuntos de dados desbalanceados. Na sequência, faremos uma discussão sobre alguns desses problemas e apresentaremos as métricas e estratégias de avaliação mais utilizadas em aprendizado de máquina, incluindo a matriz de confusão e a curva ROC.

3.2.2.1 Sobre-ajuste e sub-ajuste

Espera-se que todo modelo de aprendizado de máquina seja de certa forma generalizável, pois assim no momento em que for apresentado um novo conjunto de dados, será possível fazer previsões de forma abrangente. Entretanto, usualmente isso não acontece devido a dois fatores principais, são eles o sobre-ajuste e o sub-ajuste, também comumente referenciados pelas palavras em inglês *overfitting* e *underfitting*, respectivamente. A Figura 17 ilustra de forma simples esses problemas no contexto de regressão. Primeiramente quando a função gerada pelo treinamento do modelo não consegue se moldar aos pontos do conjunto de dados, diz-se que ocorreu *underfitting*. Já quando o treinamento ocorre de forma demasiada (*overfitting*), diz-se que o modelo aprendeu muito, até mesmo os ruídos e imprecisões do conjunto de dados. Dado esses problemas, um dos desafios em aprendizado de máquina é a busca por modelos que apresentem uma boa capacidade de generalização. Em termos do exemplo ilustrativo da Figura 17, a subfigura central representa um modelo que com bom potencial de generalização.



Figura 17 – Três formatos de modelos treinados de ML.

Fonte: Adaptado de [Hackernoon](#).

Nessa linha de raciocínio, o questionamento a se levantar é como evitar esses dois tipos de situação. Para o *underfitting*, é importante testar diferentes técnicas de aprendizado de máquina, como, por exemplo aquelas apresentadas anteriormente nesta seção. No caso do *overfitting* a solução se torna um pouco mais complexa, procedimentos como a escolha do modelo também são importantes. Além disso, estratégias de treinamento como a parada antecipada, ilustrada na Figura 18, pode contribuir com a robustez ao sobreaprendizado em que a ideia é parar o treinamento. Finalmente, outra forma de evitar o *overfitting* é dispor de uma maior quantidade de dados nas etapas de treinamento.

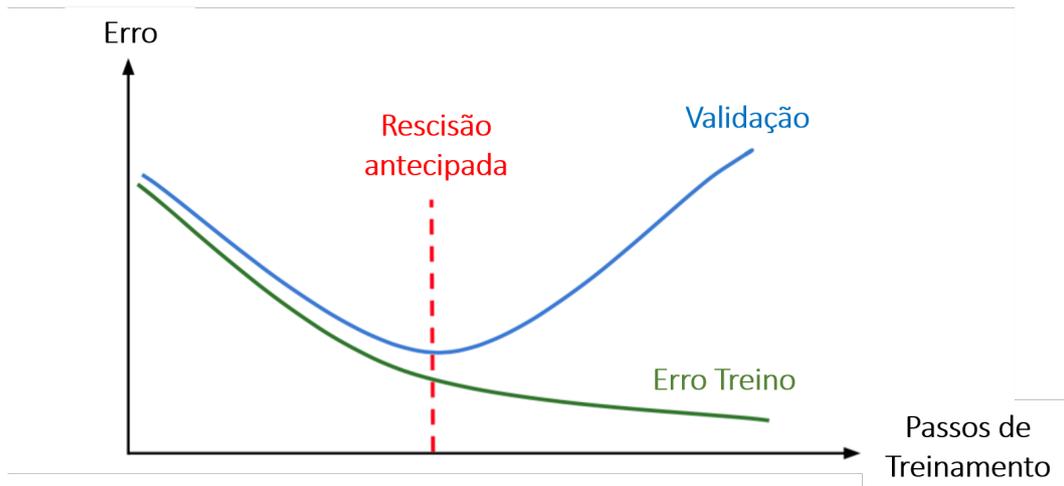


Figura 18 – Diferentes curvas de erro de treinamento para o conjunto de dados de teste e treino.

Fonte: Adaptado de [Hackernoon](#).

3.2.2.2 Matriz de confusão, tipos de erro e RMSE

Uma importante maneira de inspecionar os resultados de um modelo de classificação em um cenário supervisionado é a chamada matriz de confusão. No caso de modelos de regressão, nos quais em que a variável alvo é contínua, a métrica usual é a raiz do erro quadrático médio (RMSE, do inglês *root mean squared error*). O RMSE é dado pela seguinte expressão:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2}, \quad (3.4)$$

onde y_i é o valor previsto pelo modelo para a amostra i , enquanto x_i é o resultado real.

Com relação à matriz de confusão, a ideia principal é comparar a predição do modelo com o valor real das respostas, porém para dados categorizados, visto que são aplicadas em modelos de classificação. A Figura 19 ilustra um exemplo simples, utilizando um conjunto de dados de pessoas diabéticas e sem a patologia, de um estudo realizado pela Universidade de Washington ([KAHN, 2018](#)), no qual a ideia era tentar classificar se uma pessoa teria chance de desenvolver diabetes ou não.

Neste conjunto de teste existem 181 perfis sendo avaliados, em que 119 não possui diabetes e 62 que possuem a doença (conjunto de dados exclusivos para teste). A diagonal principal mostra os verdadeiros positivos e falsos, ou seja, o modelo previu que 108 pessoas não seriam diabéticas e acertaram as respostas (TP, do inglês *True Positive*). Além disso, modelo estimou que 36 pacientes desenvolveriam a doença e todas corretas (TN, do inglês *True Negative*). Entretanto, dado que nenhum modelo é perfeito, é usual que ocorram erros. No exemplo em questão, 26 candidatos foram previstos como sem possuírem diabetes, porém eles chegaram a desenvolver a doença (FN, do inglês *False Negative*). Além disso,

11 perfis foram previstos diabéticos e realmente não tinham a doença (FP, do inglês *False Positive*). Apesar do exemplo ter sido para uma classificação binária (desenvolveriam ou não diabetes), a matriz de confusão também serve para validação de maiores classes.

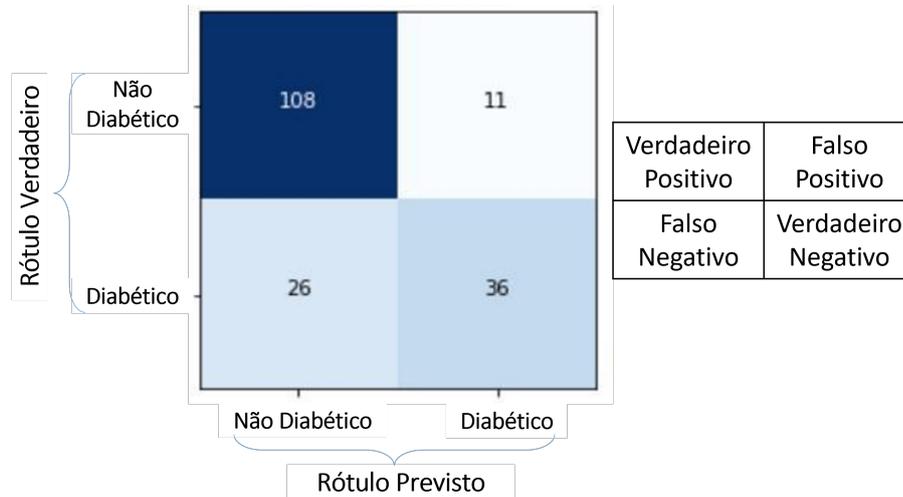


Figura 19 – Exemplo de uma matriz de confusão com sua legenda ao lado.

Fonte: [Heartbeat](#).

Associado à matriz de confusão, podemos então definir várias métricas de desempenho. Por exemplo, a acurácia de um modelo de aprendizado de máquina é definida como a soma dos verdadeiros positivos (TP) e verdadeiros falsos (FP), dividido pelo total de exemplos avaliados, ou seja:

$$Acurácia = \frac{TP + FP}{TP + FP + TN + FN}. \quad (3.5)$$

Cabe ressaltar, no entanto, que a avaliação exclusiva da acurácia não é suficiente para analisar o desempenho de um classificador. Por exemplo, no caso de dados desbalanceados, ou seja, quando há muito mais amostras de uma das classes a acurácia pode ser enganosa, uma vez que não necessariamente fornece uma ideia dos erros cometidos em cada uma das classes. Assim, para ajudar na interpretação dos resultados, é possível definir outras métricas de qualidade, incluindo as seguintes.

$$Sensibilidade = Revocação = \frac{TP}{TP + FN} \quad (3.6)$$

$$Especificidade = \frac{TN}{TN + FP} \quad (3.7)$$

$$Precisão = \frac{TP}{TP + FP} \quad (3.8)$$

A sensibilidade, também conhecida como taxa de verdadeiros positivos (TPR, do inglês *True Positive Rate*), é a relação do número de reais verdadeiros acertados no

conjunto comparado ao que foi previsto como positivos. Já a especificidade, chamada de taxa de verdadeiros negativos (TNR, do inglês *True Negative Rate*), compara o quanto o modelo acertou dos valores negativos com relação ao que foram classificados como negativos. Resumindo o TPR é a taxa de acertos das previsões positivas para o que foi classificado como positivos, enquanto o TNR é a quantidade de acertos das previsões negativas para os casos que foi classificado como negativo.

A métrica conhecida como precisão é a relação do quanto de acertos existentes nos casos positivos comparado com o que realmente são positivos (note a sutil diferença com TPR, que é em relação ao que tenta-se prever, e não os valores reais do conjunto de dados). Já a revocação, tem o mesmo significado que a TPR, sendo o quanto foi acertado dos positivos para toda classificação do que foi julgado como positivo.

3.2.2.3 Curva ROC

Quando se trata da visualização do desempenho do modelo de um classificador, a curva ROC (ROC, do inglês *Receiver Operator Characteristic*) é uma boa ferramenta para análise. Essa curva trata-se de um gráfico que ilustra a probabilidade de verdadeiros positivos no eixo y em relação aos falsos positivos no eixo x, ao longo da predição do conjunto de teste. Ou seja, é a sensibilidade (TPR) no eixo y pela taxa de falsos positivos ($FPR = 1 - \text{especificidade}$, do inglês *False Positive Rate*) no eixo x. É muito fácil confundir esses parâmetros com as Equações (3.6 e 3.7). Para ilustrar essa sutileza, considere o exemplo apresentado na Tabela 1, que representa um conjunto de 20 pontos a ser avaliado, no qual a coluna Classe representa o valor real (p – positivo e n - negativo) e a coluna Predição a probabilidade que o modelo previu para cada ponto. Percebe-se que a tabela está em ordem decrescente em relação as predições, no qual todos os valores maiores que 0,5 foram classificados como positivos (VUK; CURK, 2006).

Na Figura 20, tem-se todos os pontos referente a Tabela 1, de modo que o modelo aumenta a TPR, há um aumento na TNR. Quanto mais próximo do canto superior esquerdo a curva ROC estiver, melhor será o desempenho do modelo. Uma vez que o gráfico foi construído de forma discreta, existem diversas formas de moldar uma curva em cima desses pontos, porém a mais usual é a *Convex Hull*, basicamente é construído uma curva com relação à intersecção dos menores conjuntos convexos de pontos.

Finalmente, pode-se retirar um pouco mais de informação da curva ROC a partir da área embaixo da curva (AUC, do inglês *Area Under the Curve*). Essa métrica na verdade trata-se da porcentagem da área em baixo da curva ROC em relação ao total possível ideal, que demonstra o quanto o modelo é capaz de fazer uma distinção entre as classes de saída. A Figura 21 ilustra de forma simples essa ideia, no qual da esquerda para direita tem-se o primeiro resultado como excelente, em que $AUC = 100\%$ e consegue separar totalmente os casos negativos e positivos (fato que dificilmente acontece na realidade).

Tabela 1 – Ilustração da probabilidade atribuída pelo classificador para o conjunto de dados de teste do exemplo.

Índice	Classe	Predição
1	p	0,9
2	p	0,8
3	n	0,7
4	p	0,6
5	p	0,55
6	p	0,54
7	n	0,53
8	n	0,52
9	p	0,51
10	n	0,505
11	p	0,4
12	n	0,39
13	p	0,38
14	p	0,37
15	n	0,36
16	n	0,35
17	p	0,34
18	n	0,33
19	p	0,3
20	n	0,1

Fonte: Adaptado de (VUK; CURK, 2006).

Já o segundo gráfico demonstra $AUC = 75\%$, resultado relativamente bom, porém ainda existem pontos indiferentes no conjunto de dados. O terceiro possui $AUC = 50\%$, tendo nenhuma distinção o conjunto, ou seja, o modelo não consegue diferenciar os resultados de forma eficiente. E no último, o modelo errou todos os resultados de forma oposta.

3.2.2.4 Conjunto de dados desbalanceados

Até agora fizemos menção à métrica de acurácia considerando que a proporção das classes no conjunto de dados é igual (dados balanceados). No entanto, na maior parte dos casos práticos isso não acontece, de modo que a situação mais usual é um desbalanço entre as classes. Conjuntos de dados desbalanceados podem gerar diversos problemas de treinamento e avaliação do modelo, uma vez que pode ocorrer um enviesamento com relação à classe com maior quantidade de amostras. Tal situação de desbalanceamento ocorre no presente estudo, visto que o objetivo é detecção de sucata, que corresponde a cerca de 4–6% do conjunto de dados total.

O problema de desbalanceamento das classes requer alguns cuidados nas etapas de tratamento de dados e de análise das soluções obtidas. De fato, um primeiro ponto importante em tal situação é que pode ser necessário realizar um balanceamento das clas-

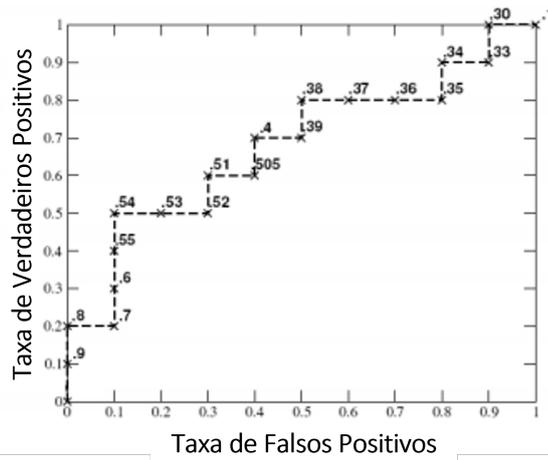


Figura 20 – Exemplo de curva ROC construída ponto a ponto.
 Fonte: Adaptado de (VUK; CURK, 2006).

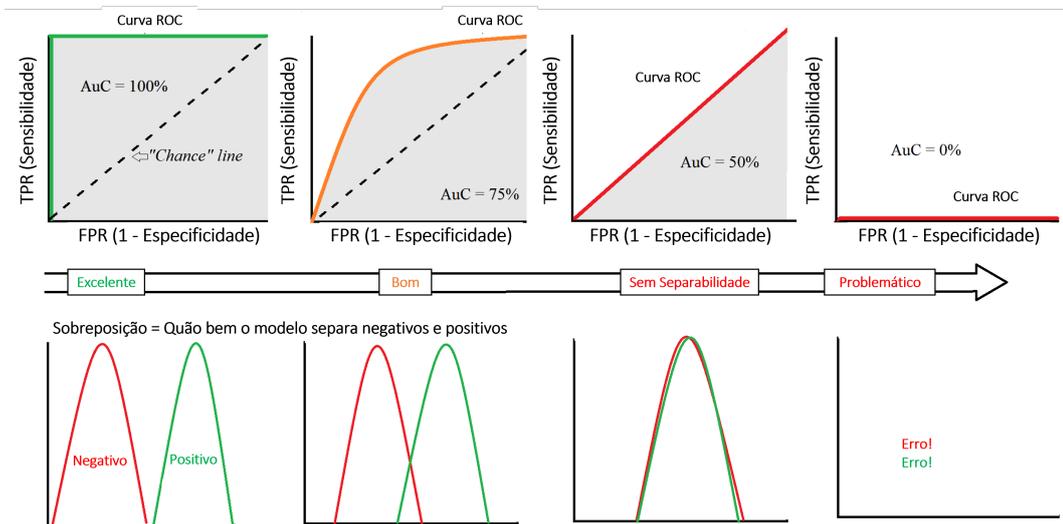


Figura 21 – Curva ROC para diferentes passos, juntamente com seus respectivos AUC.
 Fonte: Adaptado de Data Science Central.

ses. Um segundo aspecto ligado ao desbalanceamento é a escolha de métricas específicas para avaliar o modelo. Por exemplo, uma alternativa à acurácia, que pode ser enganosa em situações com dados desbalanceados, é a média entre as medidas de sensibilidade e especificidade. Outra possibilidade é o cálculo do coeficiente de correlação de Matthew’s (BOUGHORBEL *et al.*, 2017), expresso da seguinte maneira:

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{3.9}$$

Ainda com relação às estratégias de balanceamento existem duas formas principais, subamostragem (*Undersampling*) e sobreamostragem (*Oversampling*). O primeiro caso, visa eliminar de forma aleatória amostras da classe majoritária do conjunto de dados. Um dos problemas evidentes dessa técnica é que ela pode descartar amostras potencial-

mente importantes para o aprendizado (KOTSIANTIS *et al.*, 2006a). No segundo caso, a ideia vai justamente no outro sentido, ou seja, aumentar a quantidade de dados da classe minoritária, que pode ser feito a partir da replicação aleatória dos dados por meio da abordagem SMOTE, (do inglês *Synthetic Minority Oversampling Technique*), que basicamente cria novos pontos a partir da interpolação de instâncias da classe minoritária (CHAWLA *et al.*, 2002). A Figura 22 ilustra as estratégias de balanceamento supracitadas.

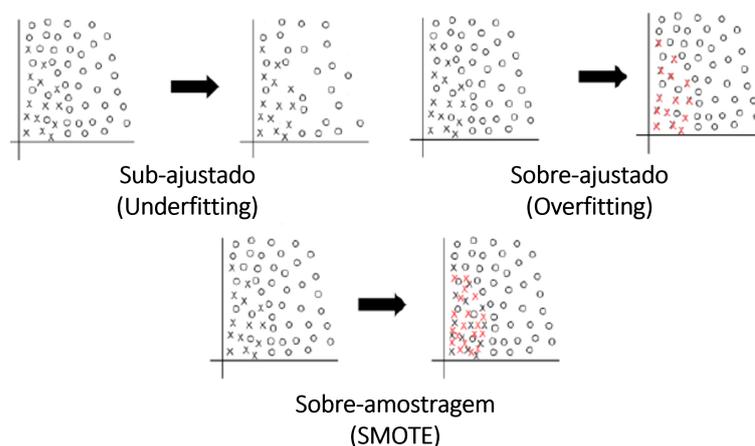


Figura 22 – Diferentes formas de corrigir conjunto de dados desbalanceados.
Fonte: Adaptado de (JR; MASTELINI, 2018).

3.2.2.5 Interpretabilidade de modelos

Treinar modelos de aprendizado de máquina para atingir melhores métricas não é o único objetivo perseguido em muitas situações. De fato, muitos problemas exigem, além de uma bom desempenho do método, uma compreensão acerca da influência dos atributos (entradas) no processo de predição da saída. A dificuldade neste caso é que modelos de ML, sobretudo os mais complexos, se comportam como verdadeiros *black box models* (modelos caixas-pretas), no sentido de não ser trivial estabelecer, por meio de simples inspeções, relações entre as variáveis de entrada e saída. Como um exemplo prático, no caso da avaliação de perfil para empréstimo no banco, é natural supor que o cliente tem o direito de saber o porquê um empréstimo solicitado não foi liberado. Se tal análise é feita por meio de uma abordagem de ML, torna-se então necessário dispor de ferramentas de interpretabilidade. Um segundo exemplo é o problema de churn (cancelamento de clientes de um dado serviço). Pode ser de grande interesse à operadora desse serviço compreender quais atributos são mais relevantes, de modo a entender quais os motivos levaram o cliente a cancelar o seu serviço.

A busca por modelos explicáveis/interpretáveis de ML motivou a pesquisa em ferramentas distintas, capazes de permitir algum tipo de entendimento entre as relações entradas-saída de um modelo de ML. No caso de modelos lineares, por exemplo, basta observar os coeficientes de cada atributo, dado que tais coeficientes quantificam, de modo

comparativo, a influência do respectivo atributo. No entanto, tal tarefa começa a se tornar um pouco mais complicada no caso de modelos não lineares como as redes neurais e as florestas aleatórias. Tal dificuldade bem motivando o desenvolvimento de ferramentas mais gerais de interpretabilidade. Um primeiro exemplo muito conhecido na literatura se baseia nos chamados valores de Shapley, que foram introduzidos por Lloyd Shapley, economista ganhador do prêmio Nobel em 2012, no contexto da teoria dos jogos. Considerando um jogo cooperativo, os valores de Shapley distribuem de forma justa uma contribuição específica de cada jogador. Analogamente pode-se pensar em uma partida de futebol, no qual o gol não pertence apenas ao atacante, mas sim para os demais da equipe que contribuíram de forma indireta para o gol.

Imagine agora que você precise montar um time de futebol para competir. Sua chance de inicial de ganhar é zero com zero jogadores. Você então primeiramente contrata um jogador muito habilidoso, de modo que claramente sua nota de contribuição será maior que a dos demais a serem contratados. Na situação contrária, começando apenas com os jogadores menos habilidosos e no final adicionar o melhor, pode ser que esse último não tenha a mesma nota comparado com a situação em que ele foi o primeiro escolhido. O exemplo ilustra de forma geral os problemas da distribuição das importâncias de cada jogador: os times podem estar mais propensos a pagar mais nas primeiras contratações do que nas últimas.

Dado o exemplo supracitado, em que a ordem de escolha dos jogadores (variáveis) é fortemente relacionado à sua importância para previsão do modelo, é preciso ter uma forma justa e geral de realizar o cálculo. Neste caso, os valores de Shapley são calculados a partir da média desses valores (importância ou nota) de cada variável para todas as ordens possíveis de escolhas. Formalmente esse processo é definido já simplificado por:

$$\varphi_i(v) = \frac{1}{n} \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n - |S| - 1)!}{n!} (v(S \cup \{i\}) - v(S)) \quad (3.10)$$

tal que φ_i é a contribuição do jogador i , N é o conjunto de todos os jogadores, v é a função que calcula a contribuição de uma equipe, n é o número total de jogadores, S é um subconjunto de N , e o somatório itera em todos os subconjuntos de N que não contém o jogador i . De forma mais intuitiva, a contribuição de cada jogador é a média de sua contribuição em relação a todas as permutações de times que não o incluem. Entretanto, é custoso fazer os cálculos para todas as possibilidades de combinação, assim existem diversas formas simplificadas utilizadas no pacote Python SHAP como o desenvolvido por Erik Štrumbelj (ŠTRUMBELJ; KONONENKO, 2014).

No caso de modelos de aprendizado de máquina derivados de métodos em árvores, como floresta aleatória, um importante trabalho é o de Lundberg *et al.* (2020). Nesta pesquisa, os autores propõem diversas ferramentas para interpretabilidade global do modelo baseado na combinação de muitas explicações locais de cada previsão. A Figura 23

a) apresenta um exemplo local de importâncias com bases nas variáveis de entrada, para o conjunto de dados com intuito de prever a chance de mortalidade. Em 23 b) temos o diagrama de como é criado a explicação global com base ao se combinar explicações locais. Isso foi demonstrado usando três conjuntos de dados médicos para treinar árvores de decisão impulsionadas por gradiente e, em seguida, computar explicações locais com base em valores SHAP (LUNDBERG *et al.*, 2020). A computação de explicações locais em todas as amostras em um conjunto de dados permite o desenvolvimento de muitas ferramentas para entender a estrutura do modelo global.

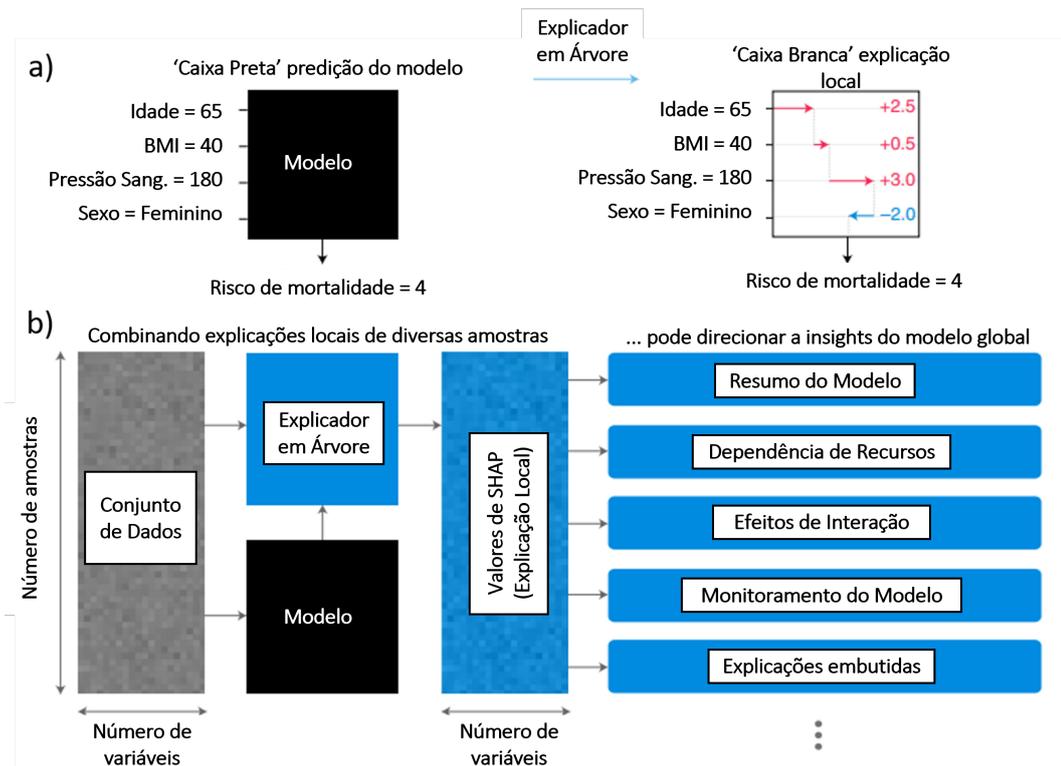


Figura 23 – As explicações locais baseadas no TreeExplainer permitem uma ampla variedade de novas maneiras de entender a estrutura do modelo global. a) Uma explicação local baseada na atribuição de uma medida numérica de crédito a cada recurso de entrada. b) Ao combinar muitas explicações locais, pode-se representar a estrutura global mantendo a fidelidade local ao modelo original.

Fonte: (LUNDBERG *et al.*, 2020).

Diversas ferramentas foram implementadas para fornecer explicações das previsões como denotado por Lundberg *et al.* (2018), para aplicações médicas. A Figura 24 a) mostra uma explicação local (apenas uma previsão) para um modelo de ML treinado com a base de dados *boston house*, com a variável alvo sendo os preços das casas de Boston. O valor 24,41 representa o preço previsto do modelo de ML. Logo abaixo, para cada variável usada na predição, tem-se uma barra, acompanhada do nome da variável e um número. A barra indica o tamanho do efeito, e a cor o sua influência positiva ou negativa. A barra da variável LSTAT, por exemplo, indica que o fato de, para este exemplo, esta variável assumir o valor 4,98 fez com que o modelo atribuísse uma contribuição positiva

de aproximadamente 5,57 (comprimento da barra) à resposta do modelo. Este é o valor SHAP desta variável. Ou seja, a importância local da variável LSTAT neste exemplo é de +5,57. Se a variável resposta corresponde a milhares de dólares, isto equivale a +5.570,00.

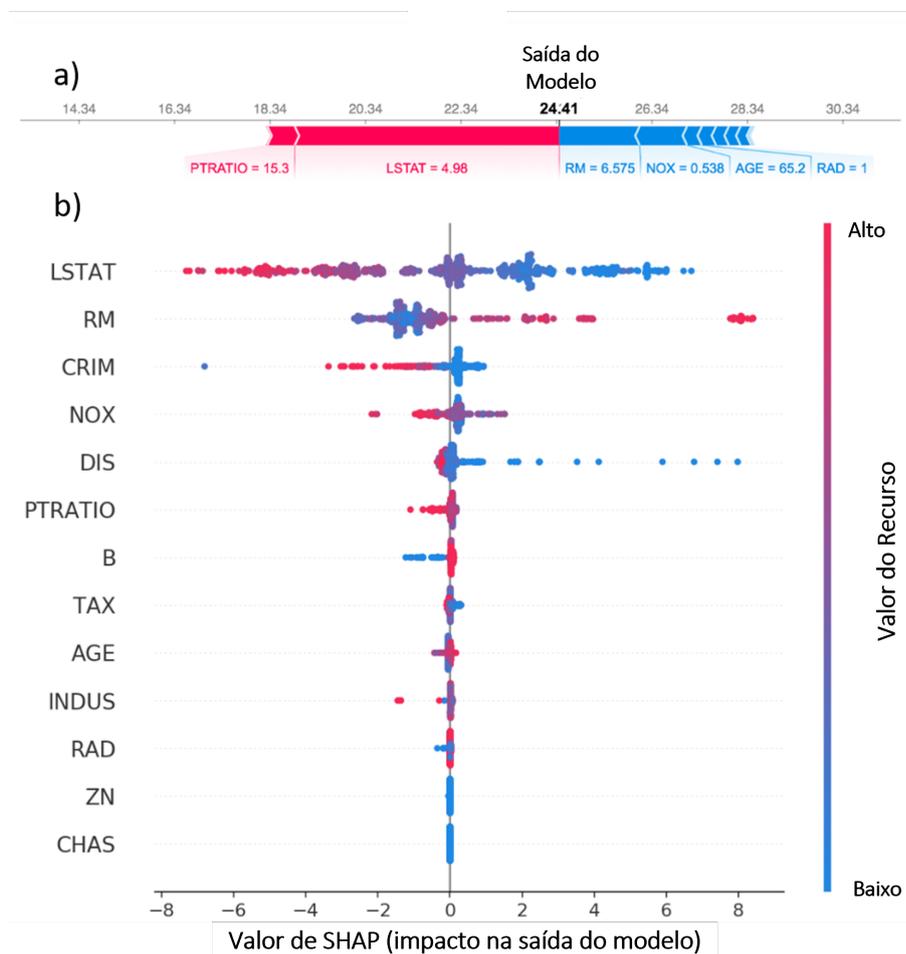


Figura 24 – Exemplos de ferramentas como saídas do pacote SHAP. a) Gráfico de força para apenas uma previsão do modelo b) Gráfico resumo para todas as previsões.

Fonte: Adaptado de <https://medium.com/big-data-blog/shap-o-que->

O exemplo anterior apresentou apenas a explicação para uma previsão do modelo de ML treinado, assim como faria uma outra técnica chamada LIME (do inglês *Local Interpretable Model-Agnostic Explanations*) (RIBEIRO *et al.*, 2016). Entretanto, no caso de explicações globais existe o gráfico em resumo (*summary plot*) apresentado na Figura 24 b), que expressa as principais variáveis influenciadoras na decisão do modelo de ML. Se o ponto está azul, significa que para aquele ponto, aquela variável tem um valor baixo (quando considerando todos os valores que aquela variável assume base de dados). Quanto mais vermelho o ponto é, mais alto é o valor. A posição do ponto no eixo horizontal indica o efeito, o valor de SHAP, daquele ponto. Quanto mais a direita, mais positiva é a contribuição daquela variável naquele ponto.

É importante denotar que diferentes valores numéricos de cada variável influenciam

diferentes formas a previsão, e que normalmente as variáveis que tem maiores distribuições como a LSTAT tem mais impacto. Assim, esses conjuntos de ferramentas servem como base para entendimento das variáveis mais importantes de forma global ou local, seja para entendimento dos dados ou para seleção de variáveis que mais podem auxiliar no treinamento dos modelos.

3.3 Otimização

Problemas de otimização se encontram em diversos segmentos atuais, tais como medicina, engenharia, transporte e relações públicas. Normalmente esses problemas são formulados com base em algum alvo que deseja-se minimizar ou maximizar, em alguns casos podem haver multi-objetivos. Uma variedade de exemplos podem ser citados, tal como redução de peso em peças, aumento de lucro, trajetória mais curta, redução da porcentagem de reprovações entre outros. Entretanto escolher a metodologia para solução do problema não é uma tarefa simples, uma vez que a escolha dessa está relacionado com a formulação da função objetivo e qual seu aspecto. Por esse motivo as próximas seções irão explicar um pouco melhor a descrição do problema de otimização, como trabalhar com meta-heurísticas e otimização orientada a dados.

3.3.1 Descrição do problema de otimização

Em um sistema produtivo de máquinas injetoras tal como demonstrado na Seção 2.1, diversas variáveis influenciam na qualidade da roda, sejam elas temperatura, pressão ou tempo de enchimento. Toda nova roda a iniciar uma produção é preciso passar por um período de teste (chamado de *tryout*), em que o engenheiro de processo precisa ajustar os parâmetros de entrada da máquina antes de validar uma folha de processo (receita) para o produto e iniciar realmente a produção do mesmo. Neste *tryout* a experiência do engenheiro faz diferença para definição final, pois são inúmeras a quantidade de combinações que se pode haver. Mesmo assim, não é possível afirmar que os valores finais definidos são realmente otimizados para aumentar a estabilidade produtiva, diminuir tempo de ciclo e reduzir a probabilidade de rodas reprovadas.

Dito isso, formular um problema de otimização com base neste contexto é essencial para a busca de um melhores conjuntos de parâmetros para produção antes mesmo da etapa de teste em linha. Porém, para essa questão é necessário entender as principais áreas envolvidas no resultado final em produção. A Figura 25 representa os três segmentos que influenciam diretamente na composição da receita de processo de uma roda, no qual o design, formato do molde e parâmetros de processo são essenciais para a qualidade da roda final. Nessa linha de raciocínio trabalhar com as variáveis advindas de cada umas dessas áreas a ponto de otimizá-las antes mesmo do sistema entrar em teste é fundamental.

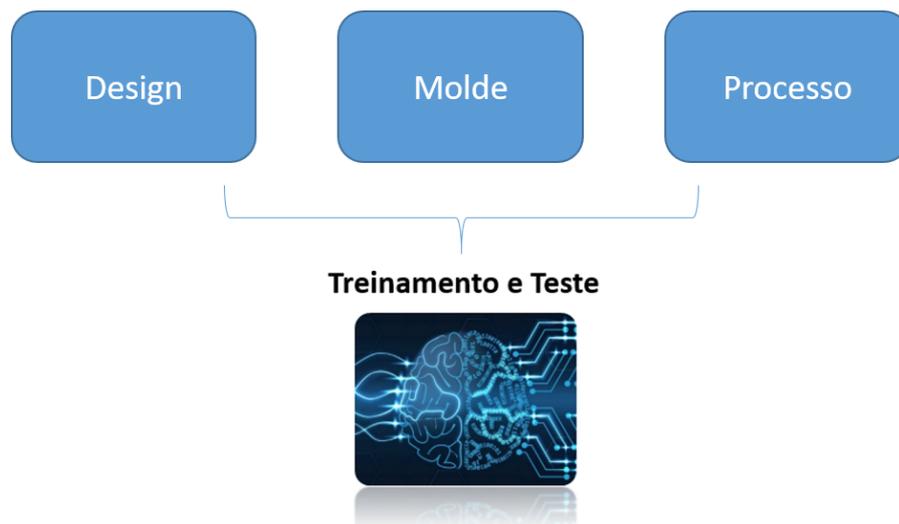


Figura 25 – Áreas para coletar dados.

Um ponto importante de salientar é como será trabalhado o problema de otimização com essas variáveis para se encontrar a melhor combinação entre elas. Neste caso, formular uma equação matemática que represente os objetivos desse problema é complexo, porém uma alternativa tem sido utilizar modelos de aprendizado de máquina para extração dessas características com alguns alvos de treinamento. Após essa etapa é possível otimizar o próprio modelo de ML, que, no presente trabalho, é feita com uma meta-heurística, no intuito de buscar a melhor combinação dos parâmetros e por fim sugerir novos valores para a receita final.

3.3.2 Otimização por meta-heurística

Nas últimas décadas, muitos algoritmos de otimização, incluindo algoritmos exatos e aproximados, foram propostos para resolver problemas das mais diversas áreas. Na classe de algoritmos de otimização exata, o projeto e implementação de algoritmos geralmente são baseados em métodos como programação dinâmica, backtracking e métodos branch-and-bound (BEHESHTI; SHAMSUDDIN, 2013). Mesmo que esses algoritmos tenham um bom desempenho em muitas aplicações (JANSSON; KNÜPPEL, 1995), (TOROSLU; COSAR, 2004), (MARTÍ *et al.*, 2010), não são eficientes na resolução de problemas com larga quantidade de variáveis e de funções não-lineares particulares. Justamente pelo fato de que o espaço de busca aumenta exponencialmente com o tamanho do problema e a busca exaustiva é impraticável nestes problemas. Além disso, métodos aproximados tradicionais, como algoritmos gulosos, geralmente exigem várias suposições que podem não ser fáceis de validar em problemas reais.

Nessa linha de raciocínio, um conjunto de algoritmos mais adaptáveis e flexíveis é necessário para superar essas limitações. Com base nessa motivação, diversos algoritmos normalmente inspirados em fenômenos naturais têm sido propostos na literatura. Dentre

eles, alguns algoritmos baseados em meta-heurística de busca têm mostrado capacidades satisfatórias para lidar com problemas de otimização de alta dimensão. As palavras “meta” e “heurística” são gregas onde “meta” é “nível superior” ou “além” e heurística significa “encontrar”, “conhecer”, “guiar uma investigação” ou “descobrir” (LAZAR, 2002). Heurísticas são métodos para encontrar soluções boas (quase ótimas) a um custo computacional razoável sem garantir viabilidade ou otimalidade (RUSSELL; NORVIG, 2002). Em outras palavras, meta-heurísticas são um conjunto de estratégias inteligentes para aumentar a eficiência de procedimentos heurísticos.

De acordo com Voß *et al.* (2012), uma meta-heurística pode ser definido como: “*um processo mestre iterativo que orienta e modifica as operações de heurísticas subordinadas para produzir eficientemente soluções de alta qualidade. Ele pode manipular uma única solução completa (ou incompleta) ou uma coleção de soluções por iteração. As heurísticas subordinadas podem ser procedimentos de alto (ou baixo) nível, ou uma simples busca local, ou apenas um método de construção.*”

Uma série de classificações de algoritmos meta-heurísticos são apresentados na Figura 26. No qual as mais importantes são, incluindo inspirada e não inspirada na natureza, baseada em populações contra busca de ponto único, dinâmica contra função objetiva estática, vizinhança única contra várias estruturas de vizinhança e uso de memória contra métodos sem memória. Em que, as mais famosas retratam algoritmos evolutivos, de natureza inspirada e implícita, tal como algoritmos genéticos, programação evolutiva/genética, evolução diferencial entre outros (BEHESHTI; SHAMSUDDIN, 2013).

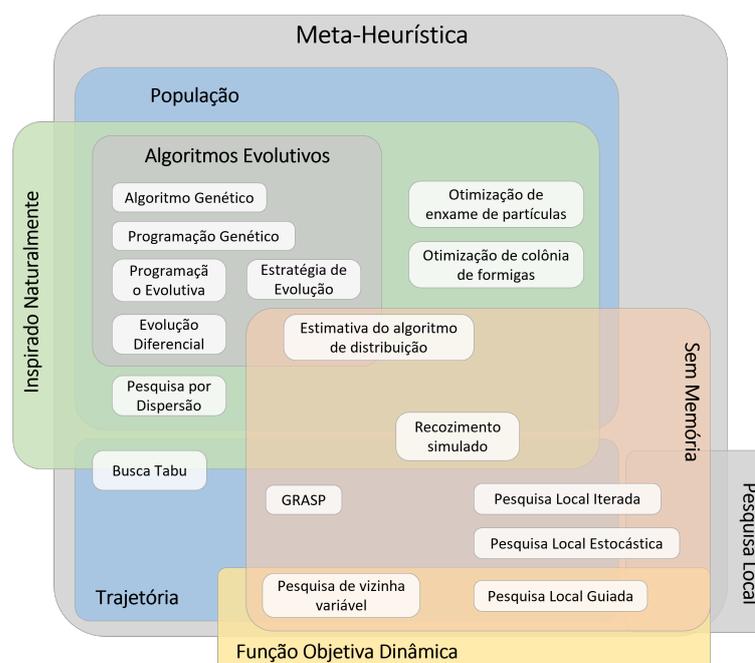


Figura 26 – Principais classes de algoritmos meta-heurísticos.

Fonte: Adaptado de (BEHESHTI; SHAMSUDDIN, 2013).

No caso do presente trabalho, com um conjunto grande de variáveis de três áreas,

um método simples foi escolhido para primeiros testes em conjunto com os modelos de ML: a evolução diferencial (DE, do inglês *Differential Evolution*). Tal método foi escolhido justamente por sua simplicidade de análise, formulação e execução. O DE é um algoritmo evolutivo guloso (*greedy*) que se baseia nos mecanismos de seleção natural e na genética de populações, e utiliza operadores de mutação, cruzamento e seleção para gerar novos indivíduos em busca do mais adaptado (FEOKTISTOV, 2006).

No DE, inicialmente é realizada uma escolha aleatória com distribuição uniforme de uma população composta por N_p indivíduos, que são denominados vetores, os quais devem cobrir todo o espaço de busca. Em seguida é feita a avaliação dos indivíduos, onde é medido o valor de aptidão dos mesmos, o qual é obtido pela avaliação do indivíduo por meio da função a ser otimizada, neste caso trata-se do modelo de ML substituto. Após isso, ocorre a mutação, onde estes vetores sofrem modificações, o que faz surgir novos indivíduos, denominados vetores doadores, pela adição da diferença ponderada entre dois indivíduos escolhidos aleatoriamente da população inicial a um terceiro indivíduo que também é escolhido de forma aleatória com distribuição uniforme da população original.

3.3.3 Modelos substitutos

Conforme apresentado nas seções anteriores o fenômeno do *Big Data* em conjunto com técnicas de aprendizado de máquina tem mudado a forma como é tomado decisões na atualidade. Uma outra área em forte crescimento é voltada à otimização evolutiva orientada a dados (DDEO, do inglês *Data-Driven Evolutionary Optimization*). Esses dados podem ser estruturados ou não-estruturados, imagens, vídeos, áudio tal como também podem ser sintéticos, através de simulações com FEA ou de processo produtivo (software Promodel por exemplo).

A partir desses dados, podem ser treinados modelos de aprendizado de máquina, também conhecidos como modelo substituto ou *Surrogate Model* (JIN *et al.*, 2018). No presente trabalho será utilizado a nomenclatura como modelo substituto. A Figura 27 apresenta um diagrama geral desde a coleta de dados históricos ou simulados, depois pré-processamento com preparação das variáveis alvos, em seguida treinamento com modelos de ML. Neste caso, pode-se ter os modelos substitutos diretos e indiretos, no qual o último depende da mineração de dados. Por fim, esses modelos são utilizados como função objetivo da computação evolutiva, no qual passam por diversas iterações até atingir o conjunto de parâmetros finais otimizados.

3.3.4 Otimização em fundição

Dada a complexidade do processo de fundição e solidificação, existem diversas pesquisas envolvendo a modelagem matemática desse tipo de problema a partir de ferra-

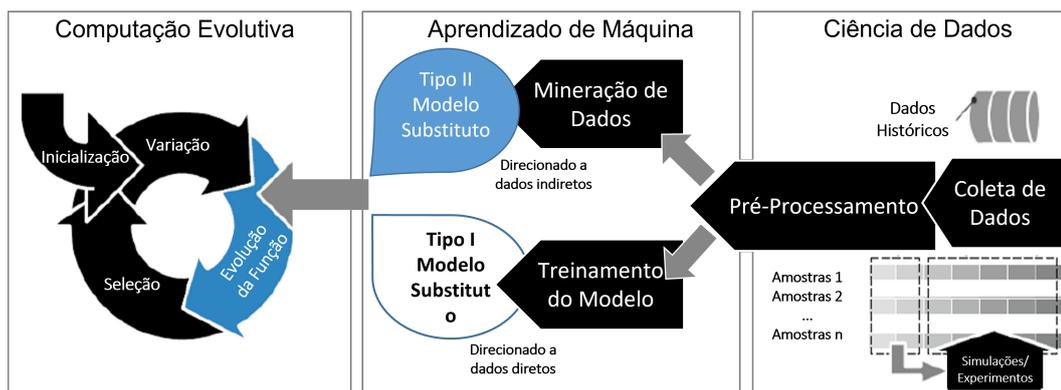


Figura 27 – Principais passos de um processo de otimização evolutiva orientada a dados. Fonte: Adaptado de (JIN *et al.*, 2018).

mentas computacionais dedicadas à simulação. Em (THOMAS, 2018), são apresentados alguns exemplos de tais softwares, com ênfase nos avanços mais recentes na simulação da produção contínua do aço por fundição. Há também soluções para realização de simulações na produção de rodas de alumínio (GUOFA *et al.*, 2009) (ZHANG *et al.*, 2007a), bem como para rodas de magnésio (WANG *et al.*, 2007). Um dos desafios presentes nesse ferramental é que as simulações são baseadas em FEA, que, via de regra, são custosas, podem culminar em simulações que duram horas ou mesmo dias dependendo da complexidade e quantidade dos nós nas malhas e da configuração do computador no qual se realiza simulação.

A despeito da complexidade computacional supracitada, algumas pesquisas visam otimizar parâmetros com base na simulação dos processos, como aplicado por (GONDKAR; INAMDAR, 2014). Neste caso, utiliza-se apenas ferramentas do próprio software com diversas iterações do modelo, com o intuito de gerar amostras. Também há casos na literatura para o desenvolvimento do modelo substituto, no qual experimentos realizados com FEA servem como dados de entrada para treinamento de um modelo de aprendizado de máquina, modelo esse utilizado como função objetivo que pode ser abordado por uma meta-heurística na tarefa de otimização. Por exemplo, no caso de (ZHANG *et al.*, 2007b) e (SHABANI; MAZAHERY, 2012), os autores consideram redes neurais artificiais em conjunto com algoritmos genéticos. Também é possível conduzir otimização dos parâmetros dos processos considerando experimentos reais em produção, como os trabalhos de (TSOUKALAS, 2008) e (ZHENG *et al.*, 2009), nos quais foram projetados experimentos para obter maior variabilidade de dados e assim poder utilizá-los para treinamento do modelo de aprendizado de máquina, que é então utilizado como função objetivo de um algoritmo evolutivo.

Nesta etapa, é importante ressaltar que os trabalhos encontrados na literatura atuais que envolvem otimização com peças fundidas e moldadas apenas servem para um produto específico. Justamente por que cada tipo de molde tem suas características indi-

viduais, são feitas otimizações específicas por produto. Além disso, simulação FEA tende a ser demorada dependendo da complexidade da malha projetada, e testes em produção consomem alto volume de tempo e dinheiro da empresa. Diante dessas dificuldades, abordagens de aprendizado de máquinas têm sido desenvolvidas utilizando dados do processo produtivo, tais abordagens são comumente conhecidos como gêmeo digital de processo (JONES *et al.*, 2020). Nessas abordagens, um dos objetivos vislumbrados é alcançar uma réplica virtual do sistema, como ocorre em manutenção preditiva (PdM, *Predictive Maintenance*), para prever possíveis falhas nas máquinas antes mesmo de haver as paradas em produção (SUSTO *et al.*, 2014).

No caso do presente trabalho, os dados para desenvolvimento dos modelos advêm da própria variabilidade do sistema produtivo, acompanhado da padronização de algumas variáveis para diferentes modelos de roda (moldes diferentes). A ideia é realmente não ter que utilizar dados sintéticos advindos de simulações, custosas computacionalmente e nem mesmo dados advindos de testes com alto custo de material e tempo. Assim, com a própria variabilidade de linha e diferenças entre produtos (design e molde) acredita-se ser possível desenvolver modelos de ML (substitutos) para otimização em uma nova roda a ser produzida.

4 Metodologia

Neste capítulo, será discutida a metodologia proposta para desenvolvimento do trabalho. Inicialmente, uma proposta de solução será apresentada. Em seguida, apresentaremos os experimentos sintéticos desenvolvidos para avaliação da metodologia de otimização orientada a dados. Por fim, será detalhado o passo a passo da preparação dos experimentos com dados reais advindos do processo produtivo, tal como os modelos de aprendizado de máquina escolhidos para avaliação.

4.1 Proposta de Solução

No presente projeto, a base de nossa proposta se fundamenta em técnicas de aprendizado de máquina para identificação dos comportamentos entrada-saída de processos relacionados à injeção de rodas de alumínio. Entretanto, antes de trabalhar com o problema real em questão, elaboramos 2 experimentos para validação da solução proposta em cenários sintéticos. Tal etapa teve como objetivo um entendimento mais refinado do funcionamento da otimização orientada a dados, para que, desse modo, se tornasse possível partir para análises com dados reais.

Conforme pode ser observado na Figura 28 *a*), o estudo com dados sintéticos constará da geração de dados a partir da função Ackley, no qual o experimento 1 considera um problema de otimização com apenas 2 variáveis, enquanto o experimento 2 será realizado para N dimensões. Em seguida, esses dados servirão de entrada para um modelo de rede neural artificial, que terá como alvo a própria resposta da função Ackley, porém com amostras aleatórias dentro de um espaço delimitado. Por fim, o modelo será utilizado como função objetivo da meta-heurística evolução diferencial.

Já a Figura 28 *b*), representa o mesmo fluxo, porém com dados reais da produção de rodas de alumínio. Dados esses majoritariamente advindos de parâmetros de processo das injetoras e também da rastreabilidade das rodas no raio X. Além disso, neste caso tem-se 4 modelos de aprendizado de máquina, que serão denominados AI1, AI2, AI3 e AI4, cada um com seu objetivo específico otimização, com base no problema de interesse.

Na sequência, será detalhada a metodologia utilizada na preparação dos dados, construção da função multiobjetivo a partir dos 4 modelos de aprendizado de máquina, testes dos modelos preditivos e otimização final com evolução diferencial. Os experimentos foram implementados na linguagem de programação Python.

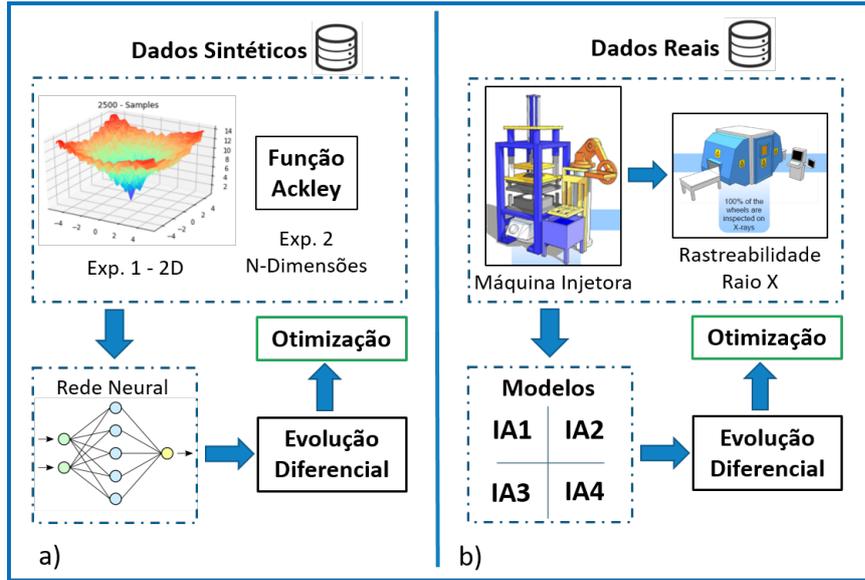


Figura 28 – a) Fluxograma para otimização com dados sintéticos; b) Fluxograma para otimização com dados reais de produção.

4.2 Experimentos com dados sintéticos: função Ackley

Para testar a ideia de otimização evolutiva orientada a dados foi considerada uma função muito utilizada na literatura: a função Ackley (DIETERICH; HARTKE, 2012). Matematicamente, a função Ackley n dimensional é definida por:

$$f(x_0, x_1, \dots, x_n) = a \exp\left(b \sqrt{\frac{1}{n} \sum_{i=0}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=0}^n \cos(cx_i)\right) + a + e. \quad (4.1)$$

Os limites usuais dessa função se encontram no hipercubo $-32 \leq x_i \leq 32$. Além disso, tal função apresenta valor mínimo igual a zero, que é atingido pelo vetor $\mathbf{x} = [0, 0, \dots, 0]$. Usualmente são escolhidas as constantes $a = 20$, $b = 0.2$ e $c = 2\pi$ para testes com otimização (YANG, 2010).

4.2.1 Experimento 1: Modelo Substituto Aplicado na Função Ackley Bidimensional

Neste primeiro experimento, foi considerado uma situação com apenas 2 dimensões, que resulta na equação $f(x_1, x_2)$ final:

$$f(x_1, x_2) = -20 \exp\left[-0.2 \sqrt{0.5(x_1^2 + x_2^2)}\right] - \exp[0.5(\cos(2\pi x_1) + \cos(2\pi x_2))] + 20 + e. \quad (4.2)$$

A restrição do espaço nesse caso é dada por $-5 \leq (x_1, x_2) \leq 5$, de modo que o valor ótimo da função é atingido em $f(0, 0) = 0$.

Com base nesse limite de (x_1, x_2) , foi projetado 5 pares de conjuntos amostrais aleatórios a partir da função (4.2), nos tamanhos de: 100, 400, 2500, 10^4 e 10^6 . A Figura

29 mostra a aproximação da função Ackley a partir de uma rede neural MLP para cada um dos cenários com esses diferentes números de amostras. É interessante analisar que, com o aumento do número de amostras, a função objetivo estimada demonstra uma maior definição dos mínimos e máximos locais, tal como ao longo de sua superfície, o que sugere que a qualidade da aproximação da função aumenta conforme o número de amostras também aumenta.

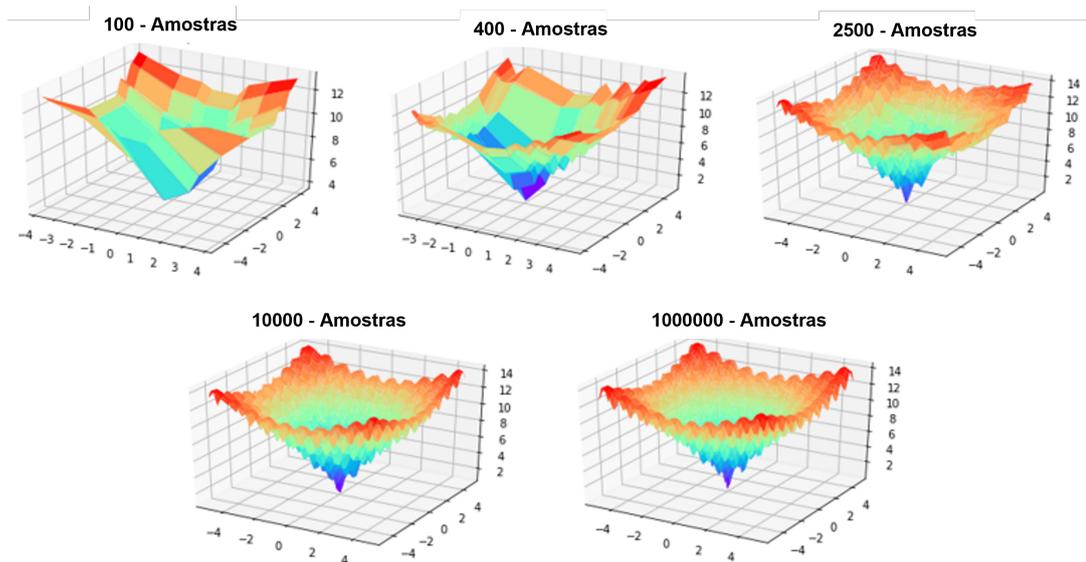


Figura 29 – Aproximações da função Ackley por uma rede MLP considerando diferentes número de amostras.

Para cada um dos 5 conjuntos de amostras exibidos na Figura 29, foram executadas 10 iterações com uma sequência de passos evidenciada na Figura 30. Tal sequência é iniciada com a geração aleatória de amostras (x_1, x_2) , distribuída uniformemente ao longo do intervalo supracitado. Em seguida, para as amostras simuladas, calcula-se a função Ackley, apresentada em (4.2), de maneira que o resultado desse cálculo nos fornece um conjunto de dados entrada-saída. Esses dados então são utilizados para treinamento de uma rede neural MLP. Depois desta etapa, a rede MLP treinada é considerada como função objetivo a ser otimizada pelo algoritmo DE ao longo de 500 iterações.



Figura 30 – Ciclo utilizado para cada iteração de um dado conjunto de amostras.

4.2.2 Experimento 2: Modelo Substituto Aplicado na Função Ackley N-Dimensões

Nesse segundo experimento, o objetivo é identificar o comportamento da curva de otimização do DE para n dimensões na função Ackley (4.1). A sequência de passos seguidos neste caso é análoga ao representada na Figura 30, diferindo apenas em 2 passos. A primeira diferença é a escolha do número de dimensões como primeira etapa do ciclo, que pode ser $n = [2, 3, 5, 10, 20, 30, 40, 50]$. Uma segunda diferença é que, para cada dimensão, são feitas apenas 5 iterações com o número fixo de 10^4 amostras aleatórios, com mesmo processo de distribuição uniforme anterior.

Os próximos passos seguem a mesma lógica da Seção 4.2.1, ou seja, a rede neural MLP é treinada com n dimensões para cada uma das 5 interações e o melhor valor encontrado é salvo, tal como a curva de otimização ao longo das 1000 iterações com o DE.

4.3 Modelo substituto para otimização de parâmetros na injeção de rodas de alumínio

O processo de manufatura de uma roda de alumínio requer a execução de uma série de etapas até o produto acabado. As principais são: preparação do liga fundida, injeção e solidificação do metal no molde, tratamento térmico, usinagem, pintura e embalagem (quando modelo é espelhado, é preciso passar novamente por usinagem e depois verniz). Todavia, muitos problemas (rodas sucateadas ou retrabalhadas) encontrados na usinagem ou pintura derivam de variações no processo durante as etapas de fundição, preenchimento e solidificação das rodas.

Assim, buscar entender e otimizar as etapas supracitadas é um processo desejável com vistas à melhoria desse processo. Entretanto, por se tratar de um processo complexo, com muitas variáveis que influenciam o sistema, é difícil obter todas respostas somente a partir da observação humana. É justamente por esse motivo que propomos no presente trabalho uma estratégia de otimização a partir de dados coletados das máquinas injetoras da fundição.

4.3.1 Preparação das variáveis

O conjunto de dados disponibilizados por um empresa do setor automotivo contém um total de 14 mil amostras (1 ponto para cada ciclo da roda) de 4 modelos de rodas diferentes. Tal conjunto contém características do produto e parâmetros de processo, totalizando 65 variáveis (atributos) de 5 grupos diferentes apresentados na Tabela 2. Cabe salientar que os atributos relacionados ao *Design* somente diferem entre os produtos, enquanto temperatura, pressão, refrigeração e qualidade possuem variabilidade entre os modelos de rodas, mas também durante o processo produtivo.

Tabela 2 – Grupo de Atributos

Grupo	Quantidade
Design	17
Temperatura	4
Pressão	10
Refrigeração	33
Qualidade	1
Total	65

Como usual em processos de análise de dados, é necessário realizar uma etapa de pré-processamento, que, em nosso trabalho, foi composta da seguinte sequência de tarefas:

1. Limpeza dos números nulos (vazios);
2. Limpeza dos *Outliers*;
3. Teste de variabilidade para cada atributo;
4. Teste de correlação entre variáveis;
5. Preparação das variáveis alvo.

Na etapa 3, foram apenas estudados os mínimos e máximos de cada variável a ponto de entender se os valores dentro do conjunto de dados estavam corretos e o que deveríamos fazer nessa situação. Na etapa 4, foi utilizado o coeficiente de Pearson para teste da relação estatística entre as variáveis contínuas. Por fim, na etapa 5, fizemos uma engenharia de atributos para criação das variáveis alvos a serem treinadas pelos modelos. Após a realização de tais procedimentos, foi possível iniciar demais estudos com modelos de aprendizado de máquina, como discutido na sequência.

4.3.2 Estudo de relevância das variáveis

Uma vez que os dados estavam pré-processados e limpos, um primeiro estudo realizado buscou identificar as variáveis com maior relevância para geração de rodas reprovadas. Para isso, o método de floresta aleatória foi utilizado inicialmente com 65 variáveis de entrada e 1 variável alvo com indicação da qualidade da roda, ou seja, aprovada ou reprovada (0/1). Mesmo sendo um método derivado de árvores de decisão, de fácil explicação, não é tão simples entender como ocorreram as previsões do modelo.

Assim, a técnica de interpretabilidade com coeficiente de Shapley foi utilizada não somente para fornecer variáveis relevantes do processo, como também indicar quais delas poderiam ter bons resultados com os modelos. A Figura 31 esboça o diagrama geral de treinamento e avaliação. Foi utilizado a biblioteca SHAP, implementada na linguagem

de programação Python, com o intuito de quantificar a relevância dos atributos para detecção de sucata considerando um modelo de classificação baseada em floresta aleatória (LUNDBERG *et al.*, 2020)(LUNDBERG *et al.*, 2020). Por fim, uma análise dessas variáveis mais importantes foram avaliadas para seleção ou não nos modelos de processo, tal como comparações com processo real produtivo.

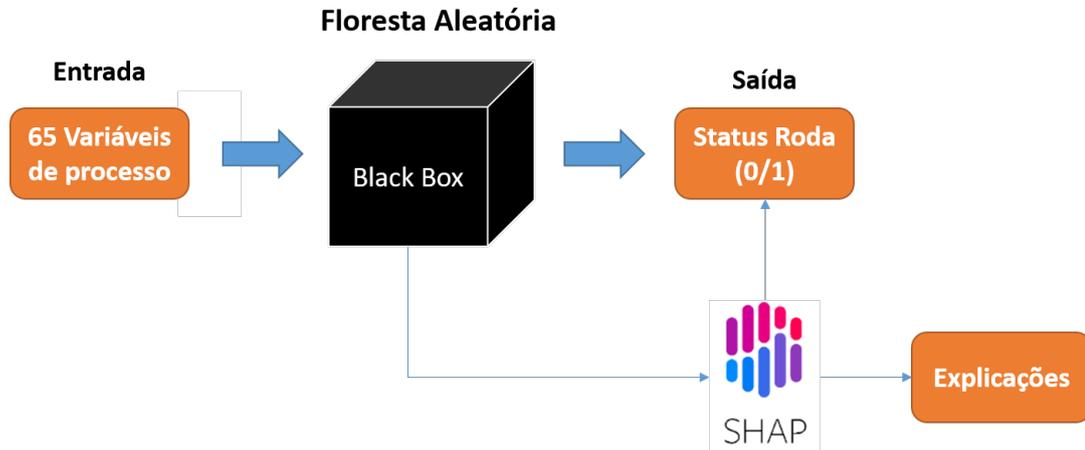


Figura 31 – Diagrama da aplicação do SHAP com modelo de floresta aleatória.

4.3.3 Função multiobjetivo

Para cada novo produto projetado pela engenharia da empresa, uma nova folha de processo com parâmetros de controle produtivo das máquinas deve ser criada. Assim sendo, com base na experiência do engenheiro de processo, é projetada uma versão primária do conjunto de parâmetros a ser testado em produção, processo conhecido como *tryout*, que leva em consideração uma série de fatores, como quantidade de refrigerações no molde, limitações de válvulas da máquina e aumento de produtividade.

Visando então obter uma melhor folha de processo (otimizada) antes mesmo de realizar o *tryout*, que uma função multiobjetivo foi projetada, sendo evidenciada em (4.3). Trata-se de uma função simples combinação linear com pesos w_1, w_2, w_3, w_4 para cada uma das funções objetivos associadas aos modelos de aprendizado $AI1, AI2, AI3, e AI4$. Esses são os modelos substitutos de funções objetivos que aprenderam (foram treinados) com dados advindos do processo e variável alvo pré-estabelecida. A função objetivo do problema é, portanto, dada por:

$$f(x) = w_1 AI1 + w_2 AI2 + w_3 AI3 + w_4 AI4, \quad (4.3)$$

onde x representa uma variável de decisão que contém todos os atributos do processo.

Conforme mencionado na Seção 2, o objetivo principal na produção de rodas de alumínio é fabricar um produto rápido, com baixo custo e, o mais importante, com qualidade. Com isso em mente, a Tabela 3 apresenta um detalhamento das funções objetivos

consideradas, considerando tanto o problema de aprendizado associado (classificação ou regressão) e a variável alvo a ser predita, além do objetivo específico. O primeiro problema, expresso por *AI1*, visa alcançar a menor probabilidade de se obter sucata. Ter um processo estável em ambiente de fundição é não ter grandes variações de temperatura, por isso *AI2* e *AI3* buscam reduzir o desvio padrão da variável de temperatura. Por fim, gostaríamos de ter mais rodas sendo produzidas a cada momento; nesse sentido, *AI4* objetiva aumento da produtividade com redução do tempo de ciclo para cada produto.

Tabela 3 – Funções objetivos por modelo substituto

Modelo	Target	Tipo	Objetivo
AI1	0/1	Classificação	Minimizar a probabilidade de sucata
AI2	Temperatura	Regressão	Minimizar o Desvio padrão
AI3	Temperatura	Regressão	Minimizar o Desvio padrão
AI4	Tempo de ciclo	Regressão	Minimizar o tempo de ciclo

A ideia é conseguir a partir dos pesos controlar qual objetivo seria mais compensado, levando em consideração o objetivo principal da roda perante ao interesse do engenheiro de processo, que poderia optar por uma roda com maior quantidade de sucata, porém com alta produtividade, ou vice versa. Portanto, depois da aplicação do algoritmo de otimização, que no presente trabalho foi o algoritmo DE, seria possível sugerir as melhores configurações de parâmetros de produção.

4.3.4 Avaliação dos modelos de ML

Conforme mencionado na Seção 4.3.1, todo o conjunto de dados foi pré-processado utilizando a linguagem de programação Python. Para facilitar o treinamento e testes de diversos modelos de aprendizado de máquina foi utilizada a ferramenta de código aberto (*Open Source*) Orange Canvas. Tal ferramenta fornece uma solução simples do tipo "*drag and drop*" para realização de testes rápidos e comparações de métricas dos modelos. A Figura 32 mostra todo fluxo de trabalho projetado para teste do modelo de classificação *AI1*. Percebe-se neste caso a diversidade de algoritmos utilizados para treinamento, sendo eles regressão logística, floresta aleatória e redes neurais.

Já a Figura 33, mostra o fluxo de teste projetado para os modelos de regressão *AI2*, *AI3* e *AI4*. Nesse caso, consideramos uma regressão linear, além da regressão via floresta aleatória e rede neural.

Com os fluxos de testes prontos, foi preciso definir os dados de entrada e configurar métricas de comparação. Para todos modelos foi utilizado o método de treinamento com validação cruzada (*Cross Validation*), separado por 10 pastas (*folds*), no qual os modelos são sempre treinados e testados na proporção 9×1 com 10 combinações diferentes. Essa metodologia foi escolhida justamente para proporcionar maior segurança na asser-

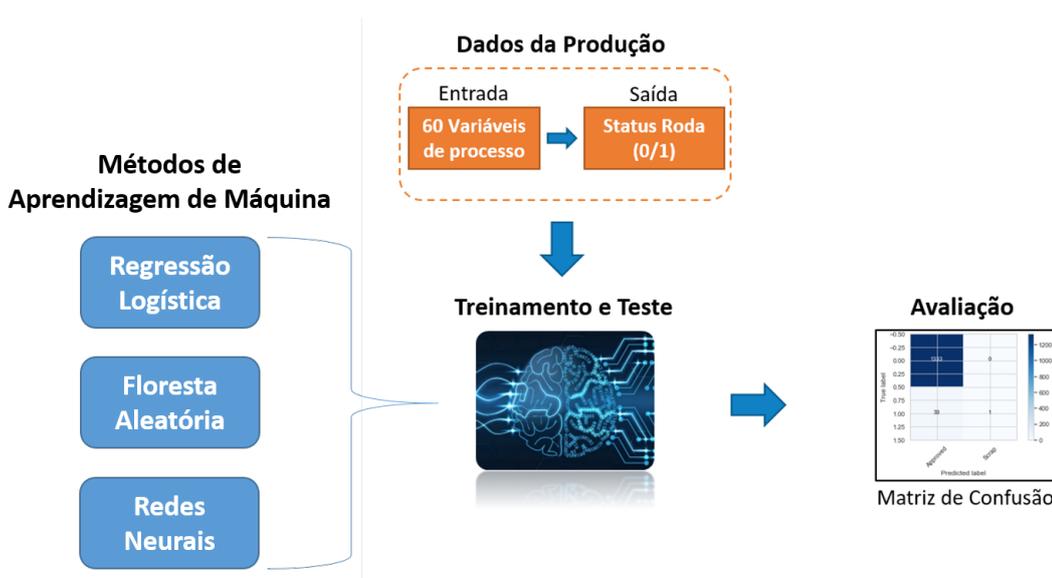


Figura 32 – Diagrama de testes projetado com Orange para treinamento do modelo de classificação AI1.

tividade do modelo de ML, quando comparada a outra forma convencional com divisões por exemplo 70/30%. Em seguida, são apresentadas as métricas para avaliação. Devido ao desbalanceamento dos dados com relação à porcentagem de sucata de produção, para o modelo de classificação observamos particularmente a precisão como métrica de avaliação. Já no caso da regressão foi escolhida a raiz do erro quadrático médio (RMSE, do inglês *root mean squared error*).

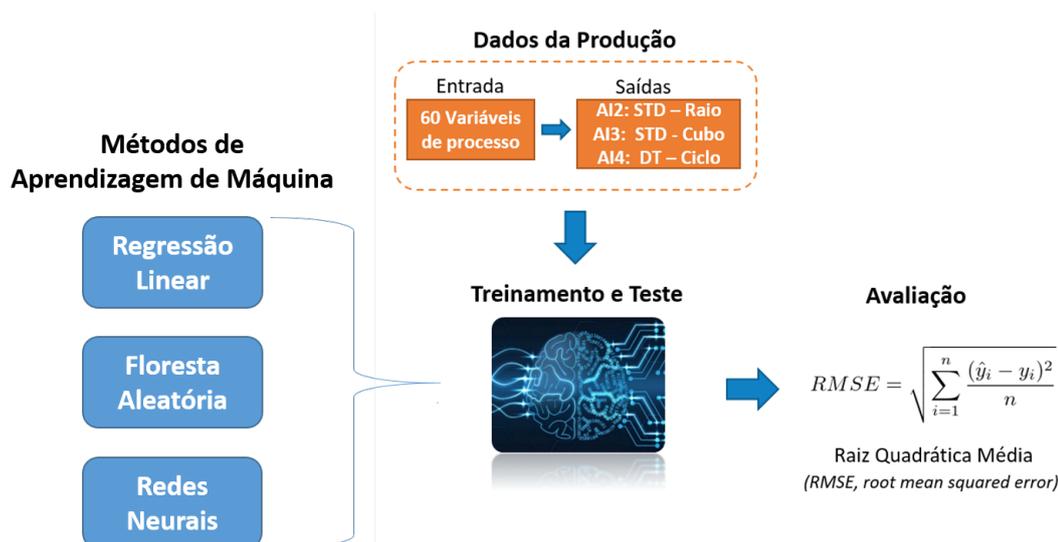


Figura 33 – Diagrama de testes projetado com Orange para treinamento dos modelos de regressão AI2, AI3 e AI4.

4.3.5 Otimização com DE

Uma vez que os modelos de aprendizado de máquina foram escolhidos como explicado na Seção 4.3.4, somente aqueles que obtiveram melhor desempenho foram selecionados para serem treinados com linguagem de programação Python com auxílio da biblioteca específica Scikit-learn. Com todos os 4 modelos de aprendizado de máquina treinados, foi desenvolvido uma função Python para representar a função objetivo (4.3), o código é apresenta no Anexo A.1. Neste caso foram definidos os pesos igualmente para avaliação inicial do otimizador.

Além disso, também foi necessário desenvolver uma função Python para representar o algoritmo evolutivo DE, apresentado no Anexo A.2. Essa função foi um pouco mais elaborada com alguns hiper-parâmetros, que são variáveis de entrada para controle e testes da otimização com evolução diferencial. Tem-se por exemplo "*fobj*" que representa a função objetivo a ser otimizada e "*bounds*" a serem os limites das fronteiras de decisão que o otimizador DE irá percorrer até chegar na solução ótima. Os demais hiper-parâmetros como probabilidade de mutação ("*mut*"), cruzamento ("*crossp*"), tamanho da população ("*popsize*") e quantidade de iterações ("*its*") foram mantidos fixos em 0.8, 0.7, 20 e 1000, respectivamente.

5 Experimentos e resultados

Após a realização dos procedimentos descritos na Seção 4, foi obtida uma série de resultados apresentados na sequência. Inicialmente, na Seção 5.1, serão apresentados os resultados para experimentos com dados sintéticos gerados pela função Ackley. Já a Seção 5.2 abordará os experimentos com dados reais advindos de uma empresa do setor automotivo. Além disso, analisaremos a escolha dos modelos de aprendizado de máquina associados a cada um dos quatro objetivos definidos na Tabela 3. Por fim, a Seção 5.3 apresenta um resumo dos principais pontos observados nos experimentos.

5.1 Experimentos com dados sintéticos

Os resultados dos experimentos sintéticos foram divididos em duas partes. Primeiro, abordamos a otimização da função Ackley bidimensional. Em seguida, será investigada a mesma função, porém com n -dimensões, e com foco na curva de aprendizado associado ao processo iterativo da otimização.

5.1.1 Experimentos com função Ackley bidimensional

Como mencionado na Seção 4.2.1, o primeiro experimento foi projetado para verificar a otimização do modelo da função Ackley 2D em função da quantidade de amostras utilizadas para treinamento. A Tabela 4 apresenta o melhor resultado encontrado pelo DE ao longo da otimização, utilizando uma rede MLP treinada para a devida quantidade de amostras. Percebe-se que, com o aumento do número de amostras, o modelo substituto apresenta melhores resultados tomando como referência o verdadeiro mínimo da função Ackley, que é de $f(x_1, x_2) = 0$ para $(x_1, x_2) = (0, 0)$.

Tabela 4 – Objetivo alcançado versus quantidade de amostras.

Amostras	$F(x, y)$
100	3.4955
400	3.6365
2500	2.9539
10^4	2.4994
10^6	1.6264
10^8	0.5122

Um outro ponto importante a ser observado é como ocorre a busca pela otimização com o algoritmo DE em termos da evolução das variáveis de otimização ao longo das iterações. A partir da Figura 34, nota-se que, apesar do DE ter executado ao todo 1000

iterações, esta meta-heurística converge para uma vizinhança muito próxima aos pontos de mínimos da função, com pequenas mudanças nos valores até o final. Na sequência, estendemos esses tais para o caso n-dimensional

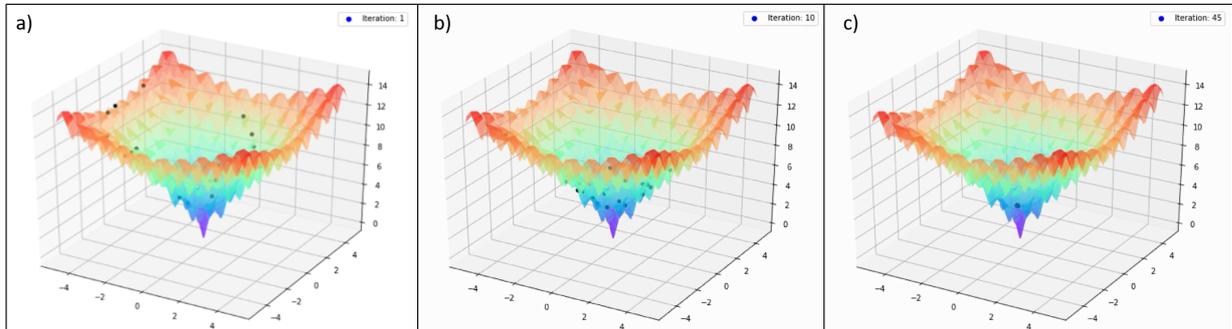


Figura 34 – Exemplo de distribuição dos pontos de otimização ao longo das iterações para um conjunto de 10^6 amostras: a) Conjunto de pontos aleatórios no início da otimização; b) Conjunto de pontos na 10^{a} iteração; c) Conjunto de pontos na 45^{a} iteração.

5.1.2 Experimentos com função Ackley n-dimensões

Observando agora para um conjunto de amostras fixos de 10^6 , variando apenas as dimensões como mencionado na Seção 4.2.2. A Tabela 5 também mostra o valor alcançado na minimização com algoritmo DE, porém para diferentes dimensões. Nota-se que, conforme as dimensões aumentam, a dificuldade do modelo substituto de aprender e encontrar o ponto ótimo da função com o DE aumenta. Esses resultados corroboram com afirmação de que problemas com altas dimensões (grande número de variáveis) torna difícil a busca por soluções ótimas até mesmo com otimização orientada a dados. Uma vez que, para o DE otimizar, é necessário que o modelo substituto consiga representar corretamente a função objetivo.

Tabela 5 – Objetivo alcançado versus quantidade de amostras.

Dimensões	$F(x_n)$
2	2.1153
3	2.9027
5	3.6310
10	4.7661
20	5.2965
30	5.2123
40	5.6533
50	6.0329

Para facilitar o entendimento do problema, a Figura 35 evidencia a busca pelo ponto ótimo ao longo das 1000 iterações do DE. Nesta figura cada cor do gráfico representa

uma dimensão, variando entre 2 ($d2$) até 50 ($d50$). Percebe-se que, conforme esse número aumenta, é preciso mais iterações para que o DE possa alcançar o valor mínimo. Neste gráfico fica claro que para altas dimensões o modelo tem certa dificuldade de aprendizado, logo demanda mais iterações do DE para encontrar ponto ótimo.

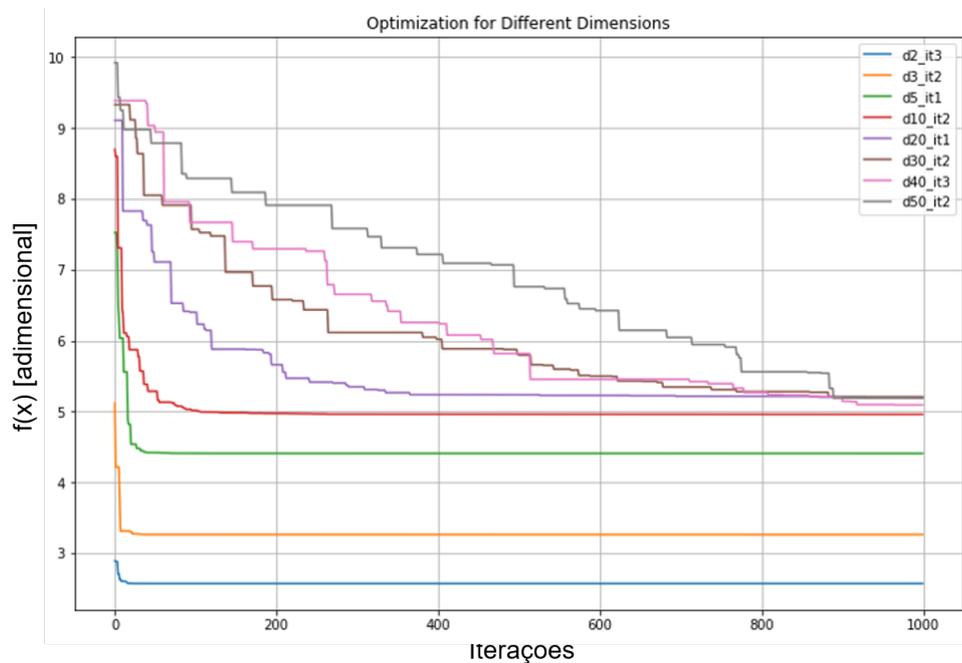


Figura 35 – Curva de otimização ao longo das 1000 iterações realizada pelo DE para um N dimensões.

Também é possível, na Figura 35, analisar a convergência do processo de otimização via DE. Comparando com a Figura 34 para o experimento de 2 dimensões, pode-se verificar que até 10 dimensões (curva vermelha da Figura 35), a convergência se dá de maneira rápida com valor inferior a 100 iterações. Somente acima de 20 dimensões, quando a complexidade da função Ackley aumenta, que o algoritmo de DE demonstrou necessitar de mais iterações como para 50 dimensões, que ficou na casa de 900 iterações necessárias para atingir ponto mínimo.

5.2 Experimentos com dados reais

Esta etapa dos resultados foi dividida em alguns pontos para facilitar nossa análise. O primeiro deles, apresentado na Seção 5.2.1, aborda uma análise exploratória dos dados e pequenas limpezas necessárias. Em um segundo ponto, é apresentado na Seção 5.2.2 um estudo sobre a relevância das variáveis do sistema. Em seguida, na Seção 5.2.3, apresentamos uma discussão sobre as métricas e resultados dos modelos substitutos. Por fim, serão analisados três cenários variando a quantidade de modelos substitutos para otimização com DE.

5.2.1 Análise Exploratória

Como mencionado na Seção 4.3.1, antes de definir todas as variáveis para trabalhar com os modelos de aprendizado de máquina (apresentadas na Tabela 2), foram realizados alguns estudos de correlação e variabilidade dos dados. A Figura 36 por exemplo retrata um gráfico em formato de mapa de calor da análise de correlação apenas das variáveis vazão como entrada da máquina. Os números no eixo x e y representam o registro da máquina injetora. Nota-se que de 3 a 7 existe uma alta correlação positiva, isso significa que, conforme uma variável aumentava, a outra seguia na mesma direção. Enquanto o registro 8 demonstra exatamente o contrário em relação os registros 9 e 10.

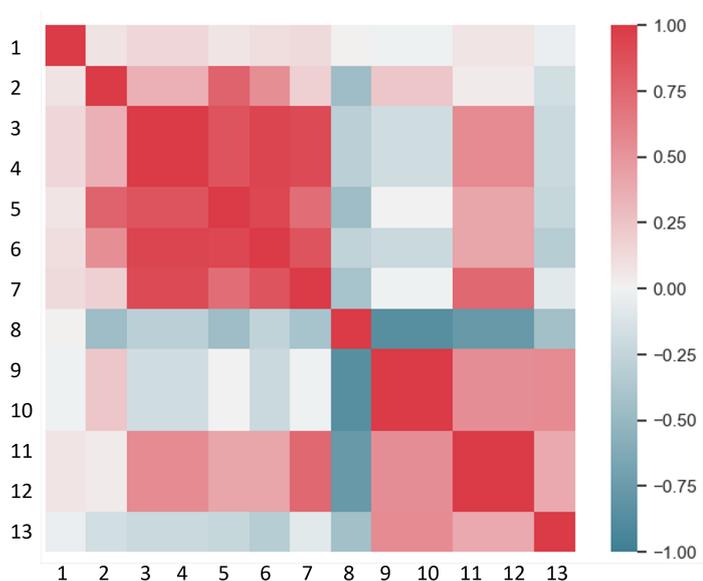


Figura 36 – Mapa de calor da correlação entre as variáveis de vazão.

Esse fenômeno é facilmente explicável devido às configurações em pares que acontece dentro do molde em relação ao sistema de refrigeração. De fato, em alguns casos, mais de uma válvula precisa atuar em conjunto em um mesmo ponto que estima-se solidificar. Note na Figura 37 a) como a distribuição de frequência dos registros 3 a 7 são muito similares, e que, no caso do registro 9 e 10, são exatamente iguais, conforme ilustrado na Figura 37 b). Esse mesmo estudo foi realizado para cada grupo de variáveis da Tabela 2, pois analisar um mapa de calor com correlação de todas as variáveis em conjunto não seria viável. Em seguida, foram retiradas do grupo de análise tais variáveis com alta correlação e que forneciam o mesmo tipo de variabilidade para o conjunto de dados.

Uma outra análise exploratória importante de ser citada remete à variável alvo do modelo de classificação, que é categoricamente caracterizada em rodas aprovadas e reprovadas. Nesta etapa, o intuito foi entender diferentes distribuições dessas duas classes mencionadas para diversas variáveis do processo. Dentre elas, a mais interessante é apresentada na Figura 38, que é a sequência de produção. Essa variável nada mais é que uma contagem de cada roda produzida ao longo de cada turno, que normalmente pode

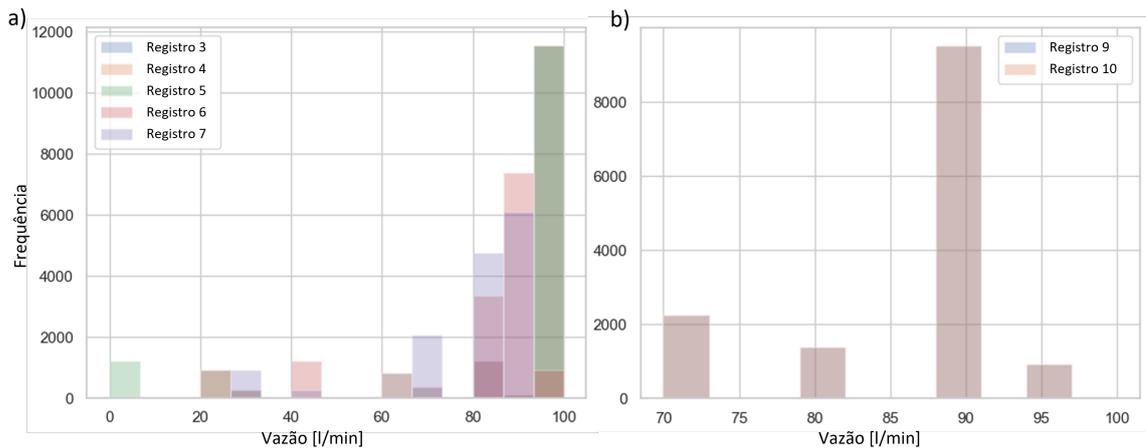


Figura 37 – a) Histograma de 5 variáveis de refrigeração com alta correlação; b) Histograma de 2 variáveis de refrigeração com alta correlação e mesma distribuição.

variar entre 1 a 110. Percebe-se que existe uma alta distribuição de rodas reprovadas para sequências menores que 20; isso ocorre justamente devido a variáveis externas e modificações nos parâmetros das máquinas ao longo do início de cada turno. Por esse motivo, uma variável de tempo de máquina aberta foi adicionada ao conjunto de dados para tentar absorver esse tipo de variabilidade do sistema e poder contribuir para treinamento dos modelos.

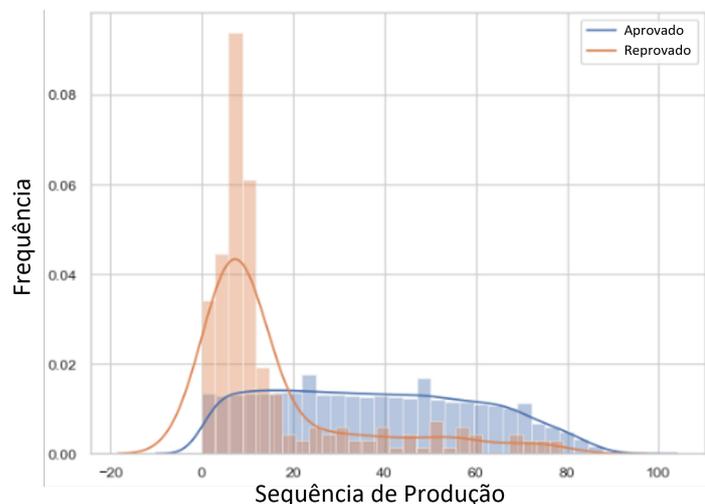


Figura 38 – Distribuição da variável sequência de produção para os grupos de rodas aprovadas e reprovadas.

Por fim, alguns gráficos de dispersão foram analisados comparativamente com as classes de rodas boas e ruins a fim de entender se poderia haver alguma tendência. Cada ponto da Figura 39 apresenta a temperatura dentro do molde em um estágio específico do ciclo, enquanto as cores representam rodas aprovadas e reprovadas. Percebe-se que existe uma faixa normal de trabalho próxima dos 450°C , no qual 90% das rodas permanecem e alguns pontos fora dessa região. É evidente também que diversas rodas reprovadas se

encontram abaixo da temperatura ideal de 400°C . Essa observação vai de encontro mais uma vez com a variável mencionada anteriormente de tempo de ciclo maiores que o esperado, o que, conseqüentemente, quebra o ritmo de produção, aumentando a probabilidade de geração de rodas reprovadas.

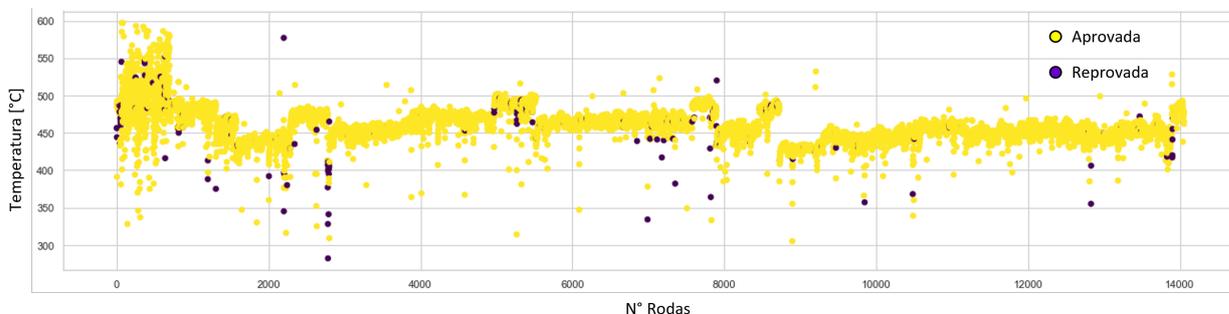


Figura 39 – Temperatura dentro do molde em um ponto específico do ciclo para todo o conjunto de 14 mil rodas.

5.2.2 Relevância das variáveis

Antes de começar o treinamento de modelos de aprendizado de máquina, foi idealizado um experimento como descrito na Seção 4.3.2 para o método de classificação, chamado floresta aleatória, no intuito de entender as variáveis mais relevantes para o processo produtivo de rodas de alumínio e possíveis seleções para demais etapas. A Figura 40 representa as classificações das variáveis mais relevantes segundo valores de Shapley. Para manter sigilo, o nome das variáveis foi omitido a pedido da empresa que forneceu os dados.

Esse gráfico retrata uma avaliação de interpretabilidade global do modelo substituído, tendo a variável alvo como a classe de qualidade da roda, ou seja, rodas reprovadas ou aprovadas. Cada ponto no gráfico é uma amostra e neste caso foi selecionado apenas 17 variáveis, pois, a partir da número 12, elas são pouco relevantes. Já a variável número 1, 3, 4 e 5 pode-se dizer que estão influenciando positivamente para predição de rodas aprovadas. Enquanto a variável 2, possui alta influência negativa para rodas aprovadas, porém significa que tem mais impacto nas rodas reprovadas. Por fim, as variáveis 6 e 7 também influenciam a previsão de rodas reprovadas, porém com menor coeficiente que a variável 2.

Essa interpretação geral dos modelos caixa preta, conhecida também como abordagem agnóstica, com a utilização dos valores de SHAP, sugere que, de forma global, poucas variáveis tem alto impacto no modelo. Porém, cabe destacar que essas variáveis estão diretamente relacionadas com as métricas de aprendizado do modelo, e, logo, se o modelo de forma geral teve baixa precisão, a análise de influência das variáveis não é tão generalizável. Em nosso caso, o valor de precisão para esse modelo de teste foi de 62%.

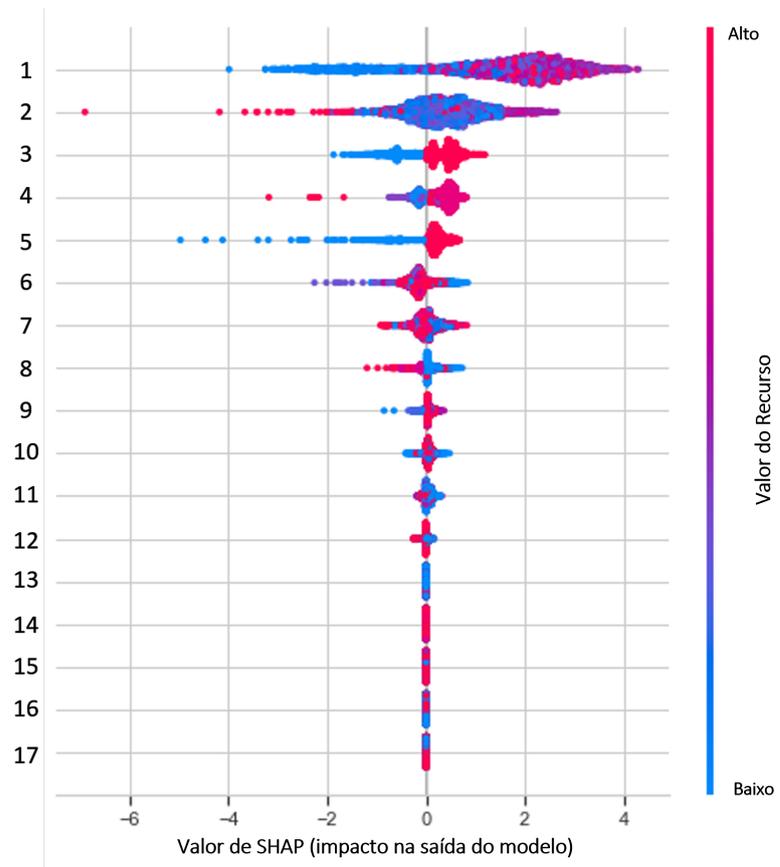


Figura 40 – Classificação das variáveis mais relevantes segundo coeficiente de SHAP.

Também é interessante considerar a análise local com SHAP, o que permite avaliar apenas as características de uma roda específica. A Figura 41 apresenta uma classificação específica para uma roda reprovada do conjunto de dados, e quais as variáveis mais relevantes para predição daquela predição realizada pelo modelo. Neste caso, tem-se como exemplo de 1 a 5 com fortes influências negativas (comprimento da barra azul) para predição da roda reprovada, e poucas variáveis influenciando a aprovação da roda (faixas em vermelho).

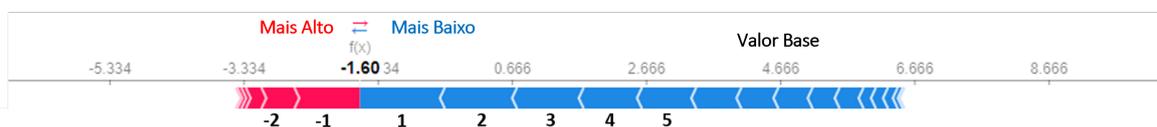


Figura 41 – Classificação das variáveis mais relevantes segundo coeficiente de SHAP para predição de uma roda reprovada.

Essa mesma técnica com coeficiente de Shapley, também foi utilizado por (UYAN *et al.*, 2022) em um estudo realizado com dados do ambiente de fundição também na produção de rodas de alumínio. Neste caso, o autor preparou 36 variáveis de entrada, com a variável alvo de duas classes, rodas aprovadas e reprovadas. Um modelo XGBoost foi implementado e dentro das 5 variáveis classificadas como mais importantes, 3 delas estavam com métricas relacionadas ao sistema de refrigeração e 2 delas correlatas à variável

referente ao metal. Algumas dessas mesmas variáveis são utilizadas no presente trabalho, porém outras como densidade da liga de alumínio não se têm controle na montagem experimental considerada.

No início da pesquisa, esses experimentos foram importantes para eliminar algumas variáveis com baixa relevância aos modelos. Interessante notar também que, ao longo do processo de escolhas das variáveis, algumas variáveis referentes ao desenho da roda começaram a se apresentar como relevantes para o aprendizado, o que evidencia que não somente variáveis de processo influenciam o sistema como um todo.

5.2.3 Modelos substitutos

Nos problemas de otimização discutidos anteriormente, há um componente relacionado às curvas de temperatura associadas a ciclos do processo. A Figura 42 apresenta um conjunto de curvas de temperatura (T1, T2, T3, T4 e T5) dentro do molde (ferramenta) para apenas um ciclo de uma roda produzida. Note que cada curva tem sua característica específica, e utilizar todos os pontos como alvo de predição seria inviável. Logo, nesta etapa foi decidido escolher a métrica de desvio padrão para que trouxesse em apenas um valor todo o formato da curva, e, a partir disso, orientar o treinamento do modelo de regressão representado pelas funções AI2 e AI3 para minimização dos desvios dessas curvas.

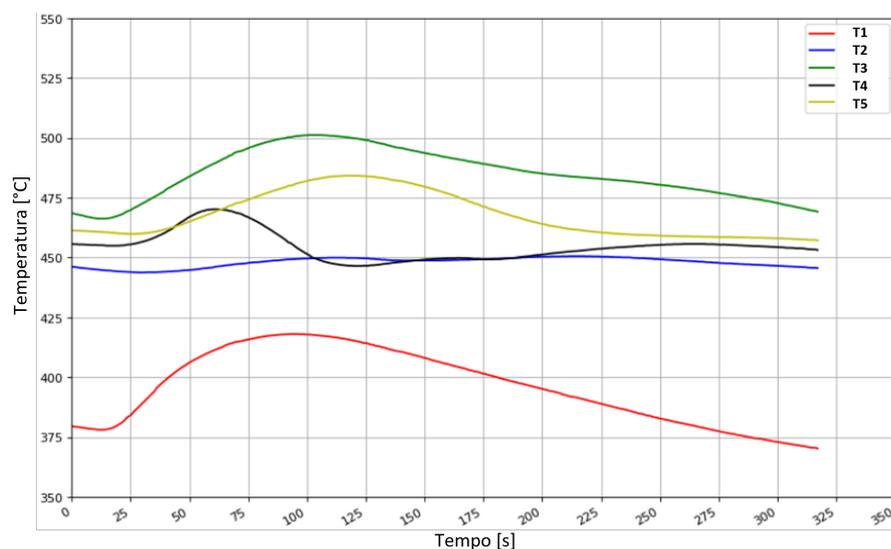


Figura 42 – Curvas de temperatura para o ciclo de apenas uma roda manufaturada.

Com as variáveis alvo prontas, definir métricas específicas para cada tipo de modelo de aprendizado de máquina (ML) é essencial para correta interpretação da eficiência do aprendizado. No caso do modelo de classificação, serão utilizadas as métricas precisão, acurácia, curva ROC e matriz de confusão. Já no caso de modelos de regressão, será

utilizada a métrica RMSE. Assim, nesta seção, serão apresentados diversos resultados comparativos para os modelos *AI1*, *AI2*, *AI3* e *AI4*, tal como descrito na Seção 4.3.3.

Iniciando então com o classificador *AI1*, a Figura 43 mostra uma comparação de 3 algoritmos de aprendizado de máquina utilizados para se treinar o modelo substituto. A variável alvo neste caso trata-se da qualidade da roda com status de aprovada ou reprovada pelo sistema de inspeção com Raio-X. Como em média a porcentagem de sucata na fundição fica em torno de 10%, temos um conjunto de dados desbalanceados. Por esse motivo, a métrica principal escolhida para avaliação foi a precisão do modelo, uma vez que foca exatamente na predição correta das rodas reprovadas; outras métricas também serão analisadas.

A Figura 43 mostra a matriz de confusão para cada modelo de ML avaliado. Considerando 0 como classe de roda reprovada, percebe-se que o modelo de floresta aleatória teve 102 rodas previstas como sucata, porém somente 77 foram realmente reprovadas. Já no caso de redes neurais, esse número caiu para 52, o que condiz com menor taxa de acertos. Por fim, o modelo de regressão logística não previu nenhuma roda como reprovada.

		Previsto		
		0.0	1.0	Σ
Real	0.0	77	153	230
	1.0	25	13790	13815
Σ		102	13943	14045

		Previsto		
		0.0	1.0	Σ
Real	0.0	33	197	230
	1.0	19	13796	13815
Σ		52	13993	14045

		Previsto		
		0.0	1.0	Σ
Real	0.0	0	230	230
	1.0	0	13815	13815
Σ		0	14045	14045

Figura 43 – Matriz de confusão para cada modelo avaliado: a) Floresta Aleatória; b) Redes Neurais; c) Regressão Logística.

Considerando o caso da curva ROC o melhor desempenho foi das redes neurais com $AUC = 0.904$, pelo gráfico da Figura 44 pode-se observar que ficou maior que as demais, enquanto a floresta aleatória parece ter uma taxa de acerto alta no começo, porém depois de um período ficou estagnada. Por fim, observa-se na Tabela 7 que o modelo de floresta aleatória obteve o melhor desempenho entre os demais com relação à precisão = (0, 755). E como precisão é o fator mais relevantes dentre essas três métricas, foi escolhido o modelo de floresta aleatória para ser utilizado com a função objetivo final.

Uma vez que foi definido o algoritmo de floresta aleatória, realizamos uma otimização dos hiper-parâmetros por meio de uma técnica conhecida como *grid-search*. Essa estratégia, considera uma grade com valores pré-definidos de busca para algumas variáveis de entrada do modelo de aprendizado de máquina (BERGSTRA; BENGIO, 2012), selecionando assim somente a combinação com melhor desempenho. Neste caso, a Tabela 6 apresenta a configuração final otimizada dos hiper-parâmetros, que percorreu a busca por tais contornos:

- n estimators = [100, 300, 500, 800, 1200]
- max depth = [5, 8, 15, 25, 30]
- min samples split = [2, 5, 10, 15, 100]
- min samples leaf = [1, 2, 5, 10].

Tabela 6 – Melhor combinação de hiper-parâmetros para AI1.

Hiper-parâmetros	Valor Selecionado
Nº estimatiors	500
Max depth	30
Min samples split	2
Min samples leaf	1

Tabela 7 – Resultados do modelo substituto para o classificador AI1.

Algoritmos	Precisão	Acurácia	AUC
Floresta Aleatória	0.755	0.987	0.840
Redes Neurais	0.635	0.985	0.904
Regressão Logística	0.000	0.984	0.837

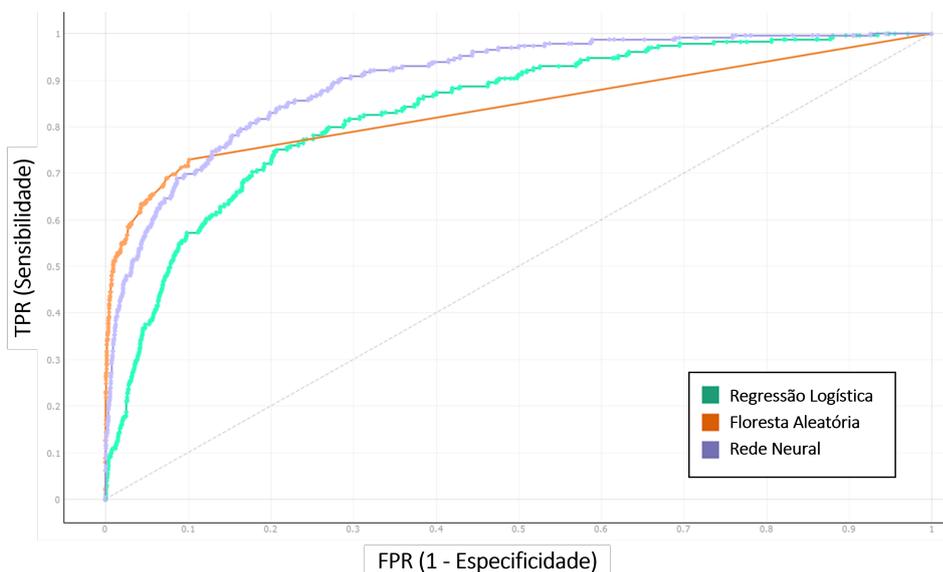


Figura 44 – Curva ROC para os modelos: a) Floresta Aleatória; b) Redes Neurais; c) Regressão Logística.

Já para os modelos de regressão expressos por $AI2$ e $AI3$, com variável alvo de desvio padrão da temperatura para cada ciclo de produção da roda, foram treinados três algoritmos de ML diferentes para comparação. No caso, o modelo de redes neurais (MLP) apresentou a menor raiz do erro quadrático médio (RMSE) de 5.64 e 3.09 respectivamente. Ambos resultados podem ser verificados nas Tabelas 8 e 9.

Tabela 8 – Resultados do modelo substituto AI2.

Algoritmos	RMSE
Redes Neurais	5.644
Floresta Aleatória	5.803
Regressão Linear	6.317

Tabela 9 – Resultados do modelo substituto AI3.

Algoritmos	RMSE
Redes Neurais	3.096
Regressão Linear	3.231
Floresta Aleatória	3.256

Depois de definidas as redes neurais para serem utilizados como AI2 e AI3, ambos os modelos foram novamente treinados, porém dessa vez por código Python. Em ambos os casos foi configurado o número máximo de iterações em 500, justamente com base em pesquisas de como essas entradas tem impacto no aprendizado das redes neurais (BREUEL, 2015). Neste caso, até foram testadas algumas formas de otimização dos hiperparâmetros, porém nenhuma das estratégias apresentou melhorias nas métricas analisadas, e, sendo assim, mantivemos uma configuração fixa (configuração *default* da biblioteca).

Por fim, o último modelo AI4, treinado com variável alvo de tempo total de ciclo, mostrou melhor desempenho com o algoritmo de regressão linear, com erro aproximado de zero (Tabela 10). Isso pode ser explicado justamente por se tratar de uma somatória do parâmetro de tempo para cada um dos cinco estágios sendo eles: quatro passos de injeção do metal líquido, solidificação da roda, abertura do molde e extração da roda, o que resulta em uma previsão 100% correta. Esse modelo também foi passado para o código Python, porém foi utilizada a configuração padrão da própria biblioteca com relação aos hiperparâmetros. Somente com todos os 4 modelos em treinados em linguagem Python, que foi possível integrar tudo na função objetivo final e utilizar em conjunto com o otimizar DE.

Tabela 10 – Resultados do modelo substituto AI4.

Algoritmos	RMSE
Regressão Linear	0.000
Redes Neurais	0.235
Floresta Aleatória	0.283

5.2.4 Otimização com DE

Conforme apresentado na Seção 4.3.5, uma função objetivo foi elaborada com base nos quatro modelos de ML mencionados na Seção 4.3.4 para atuar diretamente

com o algoritmo evolucionário DE (modelo substituto). Nesta etapa do trabalho, foram projetados três cenários principais para realização dos testes:

1. Função objetivo com os 4 modelos de IA
2. Função objetivo com apenas AI1 (prever sucata).
3. Função objetivo com apenas AI4 (tempo de ciclo).

5.2.4.1 Otimização considerando 4 modelos

A Figura 45 representa a curva de otimização com saídas da função multiobjetivo utilizando os modelos substitutos *AI1*, *AI2*, *AI3* e *AI4*, ambos com mesmo peso $w_1 = w_2 = w_3 = w_4 = 0.25$. Percebe-se a rápida minimização da função antes mesmo de 200 iterações. No entanto, quando observado o último indivíduo evoluído da iteração 1000, os valores previstos de *IA2* e *IA3*, foram de -201.5 e -220.3 respectivamente, não fizeram sentido prático, uma vez que a variável alvo consiste no desvio padrão do ciclo de temperatura para cada roda produzida, que deveria além de ser positivo, estar próximo de zero. Cabe ressaltar, que essa é uma limitação encontrada para o formato do modelo substituto projetado, sugerindo que *IA2* e *IA3*, na realidade poderia estar dificultando a otimização com o DE.

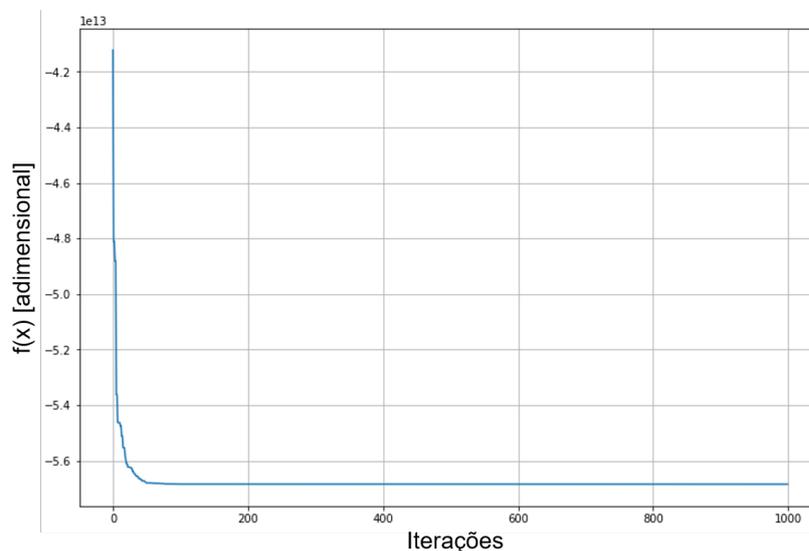


Figura 45 – Otimização final com DE para função objetivo completa.

5.2.4.2 Otimização considerando a função AI1

Diante do resultado obtido na seção anterior, optamos por realizar outros 2 testes. O primeiro deles considera apenas o modelo de classificação *AI1*, que visa prever a probabilidade da roda ser reprovada. A Figura 46, representa novamente o gráfico de minimização da função objetivo, porém com apenas 500 iterações. Neste caso, é observado

um número próximo de zero para o melhor caso, que condiz com a realidade esperada da previsão do modelo.

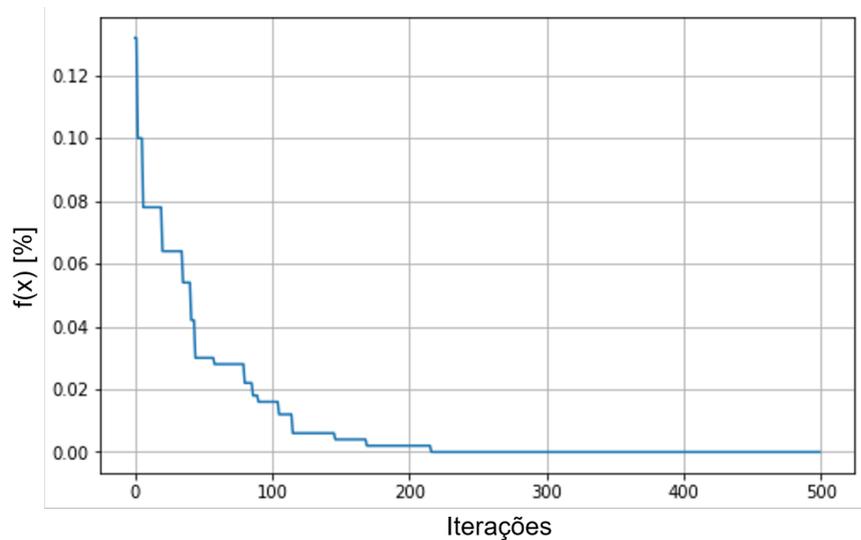


Figura 46 – Otimização com apenas objetivo de redução da porcentagem de sucata - AI1.

5.2.4.3 Otimização considerando a função AI4

Já o segundo teste com modelo de regressão expresso por *AI4*, representado pela Figura 47, e realizado para um conjunto de 500 iterações também gerou resultados que nos parecem interessantes. De fato, houve uma minimização próxima de 200 segundos, que apesar de ser um tempo de ciclo relativamente curto quando comparado com o tempo médio de 300 segundos das rodas usuais de produção, pode apresentar uma boa oportunidade sugestiva para o planejamento do processo.

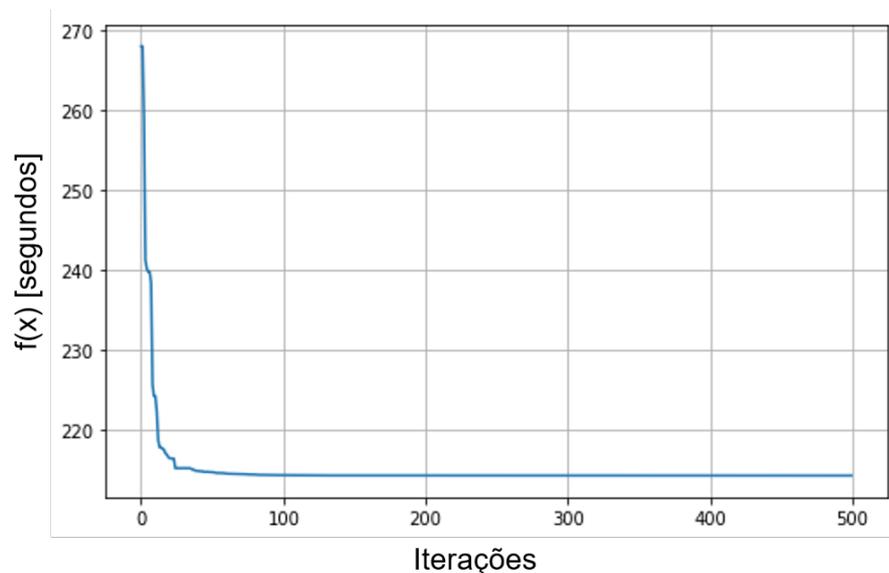


Figura 47 – Otimização com apenas objetivo de redução do tempo de ciclo - AI4.

5.3 Resumo do capítulo

O capítulo 5 foi dividido em duas partes principais: *i*) experimentos com dados sintéticos e *ii*) experimentos com dados reais advindos de uma empresa do setor automobilístico. A primeira parte com dados sintéticos teve como objetivo principal testar e validar a metodologia de trabalho com modelos de ML em conjunto com meta-heurística conhecida como otimização orientada a dados, o que de fato mostrou resultados positivos com relação ao aprendizado e também com relação a otimização.

Em seguida, a Seção 5.2 abordou experimentos utilizando dados reais. Primeiramente foi realizado uma análise exploratória dos dados, com mapa de calor da correlação das variáveis, etapa importante para exclusão de algumas variáveis com mesmo significado no conjunto de dados. Além disso, também foram apresentados alguns estudos comparando algumas variáveis com a classes de rodas boas ou ruins, tal como seu impacto.

Na sequência, foi realizado um estudo da relevância das variáveis com valores de Shapley (biblioteca SHAP) para entender quais poderiam realmente ser utilizadas para treinamento dos modelos substitutos. Em seguida, um grupo de 65 variáveis foram selecionadas para treinamento dos modelos *AI1*, *AI2*, *AI3* e *AI4*. Cada modelo substituto foi avaliado com três modelos de aprendizado diferentes, no qual apenas aquele com melhor desempenho foi escolhido para trabalhar com o DE diretamente em linguagem Python.

Por fim, esses modelos substitutos, foram testados com em três cenários de otimização. O primeiro deles a função objetivo foi configurado com os quatro modelos substitutos. O segundo cenário a função objetivo foi utilizado apenas *AI1* que é modelo substituto para prever rodas reprovadas e aprovadas. Por fim o último cenário a função objetivo utilizou apenas *AI4*, que é o modelo para prever menor tempo de ciclo.

6 Conclusões e Perspectivas

O presente trabalho de pesquisa visou desenvolver uma metodologia de otimização de uma função objetivo construída a partir de quatro modelos substitutos para uma estratégia de otimização orientada a dados. Em particular, buscou-se otimizar parâmetros do processo de máquinas injetoras para produção de rodas de alumínio. Entretanto, com intuito inicial de validar a metodologia, consideramos experimentos com dados sintéticos obtidos a partir da função conhecida Ackley. Esse experimento foi subdividido em duas etapas, primeiramente com a função Ackley bidimensional e em seguida com a função n -dimensional. No primeiro caso, foi possível observar o mínimo local alcançado pelo DE diminuindo conforme o número de amostras foi aumentando. No segundo caso, concluiu-se que conforme a quantidade de dimensões aumenta, o mínimo local alcançado precisava de mais iterações para ser encontrado, além de se obter valores maiores que para baixas dimensionalidades.

Depois de validado o método de otimização evolutiva orientada a dados com o experimento sintético, foi iniciada a preparação e análise exploratória dos dados reais. Diversas técnicas estatísticas foram utilizadas para limpeza e comparação das variáveis. Uma análise de correlação possibilitou encontrar variáveis similares que não agregando ao aprendizado dos modelos. Assim, nesta etapa foi eliminada algumas variáveis redundantes. Por fim, foram selecionados apenas os 65 atributos altamente relevantes para treinamento dos modelos, ou seja, que fornecem importantes informações a respeito do processo de fabricação de rodas de alumínio e relevantes para estabilidade do processo produtivo.

Uma vez com os dados limpos e preparados, foi possível iniciar o treinamento para escolha dos melhores modelos de ML para representar $AI1$, $AI2$, $AI3$ e $AI4$. No caso do classificador $AI1$, o algoritmo floresta aleatória obteve melhor desempenho que os demais no que se refere à precisão. Já para os modelos substitutos $AI2$ e $AI3$, ambos foram representados pelas redes neurais justamente por apresentar a melhor métrica RMSE. Por fim, o algoritmo de regressão linear obteve melhores resultados para representar $AI4$.

Depois de definido os modelos substitutos, três cenários foram projetados para testes de otimização. O primeiro cenário, representado pela função objetivo (4.3), teve pesos iguais, e foi possível concluir que os modelos $AI2$ e $AI3$, treinados previamente para prever o desvio padrão da temperatura, demonstraram resultados previstos com valores negativos e próximos de zero, o que é fora do esperado para a variável alvo projetada. Assim, a ideia inicial de criar $AI2$ e $AI3$, como representação da estabilidade de processo a partir de variáveis alvo relacionadas a temperatura, não se mostrou eficiente.

Já o segundo cenário, no qual a função objetivo (4.3) foi reformulada apenas com

o modelo *AI1* (classificador para prever sucata), gerou resultados mais condizentes com o esperado no trabalho de pesquisa. De fato, obteve-se valores que indicam uma redução de probabilidade de geração de sucata. No terceiro cenário, a otimização considerando apenas o modelo substituto *AI4* também apresentou bons resultados com relação à predição da redução do tempo de ciclo, que foi relativamente baixo quando comparado aos tempos usuais das demais rodas da produção.

Trabalhar com otimização evolutiva orientada a dados para o processo de produção de roda de alumínio provou não ser uma tarefa trivial. Isso porque o processo é influenciado por diversas variáveis controláveis e não controláveis ao sistema. Essa abordagem foi escolhida ao invés de simulações FEA ou experimentos com DOE, por conta de dois fatores principais: tempo e custo. Futuros trabalhos poderiam abranger testes em produção utilizando o conjunto de parâmetros otimizados, valores esses encontrados com o DE considerando apenas os modelos substitutos *AI1* e *AI4*. Outro ponto que também pode ser trabalhado é na melhoria da representação da estabilidade de processo pelos modelos substitutos por *AI2* e *AI3*, mudando talvez a variável alvo para outra métrica relacionado a curva de temperatura.

Referências

- ASHTON, K. *et al.* That ‘internet of things’ thing. *RFID journal*, v. 22, n. 7, p. 97–114, 2009. [9](#), [17](#), [23](#)
- BARNETT, S. Investment casting—the multi-process technology. *Foundry Trade J. Int.*, v. 11, n. 3, p. 33, 1988. [21](#)
- BEHESHTI, Z.; SHAMSUDDIN, S. M. H. A review of population-based meta-heuristic algorithms. *Int. J. Adv. Soft Comput. Appl*, v. 5, n. 1, p. 1–35, 2013. [49](#), [50](#)
- BELGIU, M.; DRĂGUȚ, L. Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, Elsevier, v. 114, p. 24–31, 2016. [35](#)
- BERGSTRA, J.; BENGIO, Y. Random search for hyper-parameter optimization. *Journal of machine learning research*, v. 13, n. 2, 2012. [71](#)
- BOUGHORBEL, S.; JARRAY, F.; EL-ANBARI, M. Optimal classifier for imbalanced data using matthews correlation coefficient metric. *PloS one*, Public Library of Science, v. 12, n. 6, p. e0177678, 2017. [43](#)
- BREIMAN, L. *Classification and regression trees*. [S.l.]: Routledge, 2017. [34](#)
- BREUEL, T. M. The effects of hyperparameters on sgd training of neural networks. *arXiv preprint arXiv:1508.02788*, 2015. [73](#)
- CABENA, P.; CHOI, H. H.; KIM, I. S.; OTSUKA, S.; REINSCHMIDT, J.; SAARENVIRTA, G. Intelligent miner for data applications guide. *IBM RedBook SG24-5252-00*, v. 173, 1999. [34](#)
- CHAWLA, N. V.; BOWYER, K. W.; HALL, L. O.; KEGELMEYER, W. P. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, v. 16, p. 321–357, 2002. [44](#)
- CHEN, M.; MAO, S.; LIU, Y. Big data: A survey. *Mobile networks and applications*, Springer, v. 19, n. 2, p. 171–209, 2014. [28](#)
- CZEKAJ, E.; PYSZ, S.; GARBACZ-KLEMPKA, A.; ŻUCZEK, R. An introduction to the problem of the conversion of structural components, including components manufactured by casting techniques. *Metallurgy and Foundry Engineering*, v. 42, n. 1, p. 21–21, 2016. [9](#), [21](#), [24](#)
- DAUGHERTY, P. R.; WILSON, H. J. *Human+ machine: Reimagining work in the age of AI*. [S.l.]: Harvard Business Press, 2018. [29](#)
- DHISALE, M.; VASAVADA, J.; TEWARI, A. An approach to optimize cooling channel parameters of low pressure die casting process for reducing shrinkage porosity in aluminium alloy wheels. *Materials Today: Proceedings*, Elsevier, 2022. [27](#), [29](#)

- DIETERICH, J. M.; HARTKE, B. Empirical review of standard benchmark functions using evolutionary global optimization. *arXiv preprint arXiv:1207.4318*, 2012. 55
- DREISEITL, S.; OHNO-MACHADO, L. Logistic regression and artificial neural network classification models: a methodology review. *Journal of biomedical informatics*, Elsevier, v. 35, n. 5-6, p. 352–359, 2002. 31
- EMMERICH, M.; GIOTIS, A.; ÖZDEMIR, M.; BÄCK, T.; GIANNAKOGLU, K. Metamodel—assisted evolution strategies. In: SPRINGER. *International Conference on parallel problem solving from nature*. [S.l.], 2002. p. 361–370. 19
- FALCKE, H. Imaging black holes: past, present and future. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2017. v. 942, n. 1, p. 012001. 29
- FAWCETT, T.; PROVOST, F. *Data Science para negócios: O que você precisa saber sobre mineração de dados e pensamento analítico de dados*. [S.l.]: Alta Books Editora, 2018. 28, 31, 32
- FEOKTISTOV, V. *Differential evolution*. [S.l.]: Springer, 2006. 51
- FISHER, R. *Iris Data Set*. 1988. <<https://archive.ics.uci.edu/ml/datasets/iris.html>>. Acessado: 2022-06-10. 33
- GHAHRAMANI, Z. Probabilistic machine learning and artificial intelligence. *Nature*, Nature Publishing Group, v. 521, n. 7553, p. 452–459, 2015. 17
- GONDKAR, V. S.; INAMDAR, K. Optimization of casting process parameters through simulation. *International Journal of Application or Innovation in Engineering and Management*, v. 3, n. 6, p. 276–283, 2014. 18, 52
- GUOFA, M.; XIANGYU, L.; KUANGFEI, W.; HENGZHI, F. Numerical simulation of low pressure die-casting aluminum wheel. *Department of Mechanical Engineering, Chengde Petroleum College, Chengde*, Citeseer, v. 67000, 2009. 52
- JANSSON, C.; KNÜPPEL, O. A branch and bound algorithm for bound constrained optimization problems without derivatives. *Journal of Global Optimization*, Springer, v. 7, n. 3, p. 297–331, 1995. 49
- JIN, Y.; WANG, H.; CHUGH, T.; GUO, D.; MIETTINEN, K. Data-driven evolutionary optimization: An overview and case studies. *IEEE Transactions on Evolutionary Computation*, IEEE, v. 23, n. 3, p. 442–458, 2018. 19, 51, 52
- JONES, D.; SNIDER, C.; NASSEHI, A.; YON, J.; HICKS, B. Characterising the digital twin: A systematic literature review. *CIRP Journal of Manufacturing Science and Technology*, Elsevier, v. 29, p. 36–52, 2020. 53
- JR, D. W. H.; LEMESHOW, S.; STURDIVANT, R. X. *Applied logistic regression*. [S.l.]: John Wiley & Sons, 2013. v. 398. 32
- JR, S. B.; MASTELINI, S. M. Tópicos especiais em reconhecimento de padrões [2cop329]. 2018. 44
- KAHN, M. *Diabetes Data Set*. 2018. <<https://archive.ics.uci.edu/ml/datasets/diabetes.html>>. Acessado: 2019-03-28. 39

- KOTSIANTIS, S.; KANELLOPOULOS, D.; PINTELAS, P. *et al.* Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering*, v. 30, n. 1, p. 25–36, 2006. [44](#)
- KOTSIANTIS, S. B.; ZAHARAKIS, I. D.; PINTELAS, P. E. Machine learning: a review of classification and combining techniques. *Artificial Intelligence Review*, Springer, v. 26, n. 3, p. 159–190, 2006. [30](#)
- LASI, H.; FETTKE, P.; KEMPER, H.-G.; FELD, T.; HOFFMANN, M. Industry 4.0. *Business & information systems engineering*, Springer, v. 6, n. 4, p. 239–242, 2014. [28](#)
- LAZAR, A. Heuristic knowledge discovery for archaeological data using genetic algorithms and rough sets. In: *Heuristic and optimization for knowledge discovery*. [S.l.]: IGI Global, 2002. p. 263–278. [50](#)
- LI, L.; QIN, L.; XU, Z.; YIN, Y.; WANG, X.; KONG, B.; BAI, J.; LU, Y.; FANG, Z.; SONG, Q. *et al.* Artificial intelligence distinguishes covid-19 from community acquired pneumonia on chest ct. *Radiology*, Radiological Society of North America, 2020. [18](#)
- LUNDBERG, S. M.; ERION, G.; CHEN, H.; DEGRAVE, A.; PRUTKIN, J. M.; NAIR, B.; KATZ, R.; HIMMELFARB, J.; BANSAL, N.; LEE, S.-I. From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, Nature Publishing Group, v. 2, n. 1, p. 2522–5839, 2020. [45](#), [46](#), [59](#)
- LUNDBERG, S. M.; NAIR, B.; VAVILALA, M. S.; HORIBE, M.; EISSES, M. J.; ADAMS, T.; LISTON, D. E.; LOW, D. K.-W.; NEWMAN, S.-F.; KIM, J. *et al.* Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature Biomedical Engineering*, Nature Publishing Group, v. 2, n. 10, p. 749, 2018. [46](#)
- MA, L.; SUN, B. Machine learning and ai in marketing—connecting computing power to human insights. *International Journal of Research in Marketing*, Elsevier, v. 37, n. 3, p. 481–504, 2020. [29](#)
- MANYIKA, J.; CHUI, M.; BROWN, B.; BUGHIN, J.; DOBBS, R.; ROXBURGH, C.; BYERS, A. H. *et al.* *Big data: The next frontier for innovation, competition, and productivity*. [S.l.]: McKinsey Global Institute, 2011. [28](#)
- MARQUÉS, A.; GARCÍA, V.; SÁNCHEZ, J. S. Exploring the behaviour of base classifiers in credit scoring ensembles. *Expert Systems with Applications*, Elsevier, v. 39, n. 11, p. 10244–10250, 2012. [34](#)
- MARSLAND, S. *Machine learning: an algorithmic perspective*. [S.l.]: Chapman and Hall/CRC, 2011. [31](#), [33](#), [34](#)
- MARTÍ, R.; GALLEGRO, M.; DUARTE, A. A branch and bound algorithm for the maximum diversity problem. *European Journal of Operational Research*, Elsevier, v. 200, n. 1, p. 36–44, 2010. [49](#)
- MAURO, A. D.; GRECO, M.; GRIMALDI, M. What is big data? a consensual definition and a review of key research topics. In: AIP. *AIP conference proceedings*. [S.l.], 2015. v. 1644, n. 1, p. 97–104. [29](#)

- MCAFEE, A.; BRYNJOLFSSON, E.; DAVENPORT, T. H.; PATIL, D.; BARTON, D. Big data: the management revolution. *Harvard business review*, Cambridge, v. 90, n. 10, p. 60–68, 2012. [17](#)
- MCAFEE, A.; BRYNJOLFSSON, E.; DAVENPORT, T. H.; PATIL, D.; BARTON, D. Big data: the management revolution. *Harvard business review*, v. 90, n. 10, p. 60–68, 2012. [28](#), [29](#)
- MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, Springer, v. 5, n. 4, p. 115–133, 1943. [9](#), [36](#)
- MERLIN, M.; TIMELLI, G.; BONOLLO, F.; GARAGNANI, G. L. Impact behaviour of a356 alloy for low-pressure die casting automotive wheels. *Journal of materials processing technology*, Elsevier, v. 209, n. 2, p. 1060–1073, 2009. [21](#)
- MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. *Foundations of machine learning*. [S.l.]: The MIT Press, 2012. [30](#)
- OCZOŚ, K. E.; KAWALEC, A. *Kształtowanie metali lekkich*. [S.l.]: Wydawnictwo Naukowe PWN Warszawa, Poland, 2012. [9](#), [22](#)
- ONG, Y. S.; NAIR, P. B.; KEANE, A. J. Evolutionary optimization of computationally expensive problems via surrogate modeling. *AIAA journal*, v. 41, n. 4, p. 687–696, 2003. [19](#)
- PALM, G. Warren mcculloch and walter pitts: A logical calculus of the ideas immanent in nervous activity. In: *Brain Theory*. [S.l.]: Springer, 1986. p. 229–230. [17](#)
- PATTNAIK, S.; KARUNAKAR, D. B.; JHA, P. Developments in investment casting process—a review. *Journal of Materials Processing Technology*, Elsevier, v. 212, n. 11, p. 2332–2348, 2012. [21](#), [23](#)
- PRICE, K. V. Differential evolution. In: *Handbook of optimization*. [S.l.]: Springer, 2013. p. 187–214. [18](#)
- PROVOST, F.; FAWCETT, T. Data science and its relationship to big data and data-driven decision making. *Big data*, Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA, v. 1, n. 1, p. 51–59, 2013. [18](#)
- QUINLAN, J. Program for machine learning. *C4. 5*, Morgan Kaufmann Pub, 1993. [34](#)
- QUINLAN, J. R. Induction of decision trees. *Machine learning*, Springer, v. 1, n. 1, p. 81–106, 1986. [34](#)
- RAJAGOPAL, S. Squeeze casting: a review and update. *Journal of applied metalworking*, Springer, v. 1, n. 4, p. 3–14, 1981. [9](#), [23](#), [25](#)
- RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. "why should i trust you?" explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. [S.l.: s.n.], 2016. p. 1135–1144. [47](#)
- RUSSELL, S.; NORVIG, P. Artificial intelligence: a modern approach. 2002. [50](#)

- SHABANI, M. O.; MAZAHERY, A. Optimization of process conditions in casting aluminum matrix composites via interconnection of artificial neurons and progressive solutions. *Ceramics International*, Elsevier, v. 38, n. 6, p. 4541–4547, 2012. 52
- SHAHANE, S.; ALURU, N.; FERREIRA, P.; KAPOOR, S. G.; VANKA, S. P. Optimization of solidification in die casting using numerical simulations and machine learning. *Journal of Manufacturing Processes*, Elsevier, v. 51, p. 130–141, 2020. 27
- SHAN, S.; WANG, G. G. Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. *Structural and multidisciplinary optimization*, Springer, v. 41, n. 2, p. 219–241, 2010. 27
- SHEN, D.; WU, G.; SUK, H.-I. Deep learning in medical image analysis. *Annual review of biomedical engineering*, NIH Public Access, v. 19, p. 221, 2017. 29
- SHRESTHA, Y. R.; BEN-MENACHEM, S. M.; KROGH, G. V. Organizational decision-making structures in the age of artificial intelligence. *California Management Review*, SAGE Publications Sage CA: Los Angeles, CA, v. 61, n. 4, p. 66–83, 2019. 18
- STORN, R.; PRICE, K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, Springer, v. 11, n. 4, p. 341–359, 1997. 18
- ŠTRUMBELJ, E.; KONONENKO, I. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, Springer, v. 41, n. 3, p. 647–665, 2014. 45
- SUSTO, G. A.; SCHIRRU, A.; PAMPURI, S.; MCLOONE, S.; BEGHI, A. Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Transactions on Industrial Informatics*, IEEE, v. 11, n. 3, p. 812–820, 2014. 29, 53
- THOMAS, B. G. Review on modeling and simulation of continuous casting. *steel research international*, Wiley Online Library, v. 89, n. 1, p. 1700312, 2018. 52
- TOROSLU, I. H.; COSAR, A. Dynamic programming solution for multiple query optimization problem. *Information Processing Letters*, Elsevier, v. 92, n. 3, p. 149–155, 2004. 49
- TSOUKALAS, V. Optimization of porosity formation in AlSi9Cu3 pressure die castings using genetic algorithm analysis. *Materials & Design*, Elsevier, v. 29, n. 10, p. 2027–2033, 2008. 52
- UYAN, T. Ç.; OTTO, K.; SILVA, M. S.; VILAÇA, P.; ARMAKAN, E. Industry 4.0 foundry data management and supervised machine learning in low-pressure die casting quality improvement. *International Journal of Metalcasting*, Springer, p. 1–16, 2022. 29, 69
- VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer science & business media, 2013. 17
- VOSS, S.; MARTELLO, S.; OSMAN, I. H.; ROUCAIROL, C. *Meta-heuristics: Advances and trends in local search paradigms for optimization*. [S.l.]: Springer Science & Business Media, 2012. 50

- VUK, M.; CURK, T. Roc curve, lift chart and calibration plot. *Metodoloski zvezki, Anuska Ferligoj*, v. 3, n. 1, p. 89, 2006. [41](#), [42](#), [43](#)
- WAMBA, S. F.; AKTER, S.; EDWARDS, A.; CHOPIN, G.; GNANZOU, D. How 'big data' can make big impact: Findings from a systematic review and a longitudinal case study. *International Journal of Production Economics*, Elsevier, v. 165, p. 234–246, 2015. [17](#)
- WANG, C.; TAN, X.; TOR, S.; LIM, C. Machine learning in additive manufacturing: State-of-the-art and perspectives. *Additive Manufacturing*, Elsevier, v. 36, p. 101538, 2020. [29](#)
- WANG, Y.-c.; LI, D.-y.; PENG, Y.-h.; ZENG, X.-q. Numerical simulation of low pressure die casting of magnesium wheel. *The International Journal of Advanced Manufacturing Technology*, Springer, v. 32, n. 3-4, p. 257–264, 2007. [52](#)
- WHITLEY, D.; RANA, S.; DZUBERA, J.; MATHIAS, K. E. Evaluating evolutionary algorithms. *Artificial intelligence*, Elsevier, v. 85, n. 1-2, p. 245–276, 1996. [17](#)
- YANG, X.-S. Firefly algorithm, levy flights and global optimization. In: *Research and development in intelligent systems XXVI*. [S.l.]: Springer, 2010. p. 209–218. [55](#)
- ZANINI, A. Fea/ufjf 007/2007. 2007. [32](#)
- ZHANG, B.; COCKCROFT, S.; MAIJER, D.; ZHU, J.; PHILLION, A. Casting defects in low-pressure die-cast aluminum alloy wheels. *Jom*, Springer, v. 57, n. 11, p. 36–43, 2005. [9](#), [21](#), [25](#), [26](#), [27](#)
- ZHANG, B.; MAIJER, D.; COCKCROFT, S. Development of a 3-d thermal model of the low-pressure die-cast (lpdc) process of a356 aluminum alloy wheels. *Materials Science and Engineering: A*, Elsevier, v. 464, n. 1-2, p. 295–305, 2007. [52](#)
- ZHANG, X.; TONG, S.; XU, L.; YAN, S. Optimization of low-pressure die casting process with soft computing. In: IEEE. *2007 International Conference on Mechatronics and Automation*. [S.l.], 2007. p. 619–623. [27](#), [52](#)
- ZHENG, J.; WANG, Q.; ZHAO, P.; WU, C. Optimization of high-pressure die-casting process parameters using artificial neural network. *The International Journal of Advanced Manufacturing Technology*, Springer, v. 44, n. 7, p. 667–674, 2009. [52](#)
- ZOU, J.; HAN, Y.; SO, S.-S. Overview of artificial neural networks. *Artificial Neural Networks*, Springer, p. 14–22, 2008. [36](#)

Appendices

A Funções Python

A.1 Função Multiobjetivo

```
#  $Fit(x) = W1*AI_1 + W2*AI_2 + W3*AI_3 + W4*AI_4$ 
def fit_function(parameters):
    w1 = w2 = w3 = w4 = 0.25
    # Rejection Prediction
    scrap_proba = model_RFC.predict_proba([parameters])[0][0]
    # Time Cycle Prediction
    time_cycle = model_LRdt.predict([parameters])[0]
    # STD Temperature Prediction for Spoke
    std_spoke = model_LRspoke.predict([parameters])[0]
    # STD Temperature Prediction for Sprue Bush
    std_sprue = model_LRsprue.predict([parameters])[0]

    return w1*scrap_proba + w2*time_cycle + w3*std_spoke + w4*std_sprue
```

A.2 Função DE

```
def de(fobj, bounds, mut=0.8, crossp=0.7, popsize=20, its=1000):
    dimensions = len(bounds)
    pop = np.random.rand(popsize, dimensions)
    min_b, max_b = np.asarray(bounds).T
    diff = np.fabs(min_b - max_b)
    pop_denorm = min_b + pop * diff
    fitness = np.asarray([fobj(ind) for ind in pop_denorm])
    best_idx = np.argmin(fitness)
    best = pop_denorm[best_idx]
    for i in range(its):
        for j in range(popsize):
            idxs = [idx for idx in range(popsize) if idx != j]
            a, b, c = pop[np.random.choice(idxs, 3, replace = False)]
            mutant = np.clip(a + mut * (b - c), 0, 1)
            cross_points = np.random.rand(dimensions) < crossp
            if not np.any(cross_points):
                cross_points[np.random.randint(0, dimensions)] = True
```

```
trial = np.where(cross_points, mutant, pop[j])
trial_denorm = min_b + trial * diff
f = fobj(trial_denorm)
if f < fitness[j]:
    fitness[j] = f
    pop[j] = trial
    if f < fitness[best_idx]:
        best_idx = j
        best = trial_denorm
yield best, fitness[best_idx]
```