UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA DEPARTAMENTO DE QUÍMICA

PEDRO HENRIQUE FERREIRA LIMA

DATA SCIENCE: UM GLOSSÁRIO PARA PROFISSIONAIS DA ÁREA DE QUÍMICA

SÃO CARLOS

2022

UNIVERSIDADE FEDERAL DE SÃO CARLOS
PEDRO HENRIQUE FERREIRA LIMA

DATA SCIENCE: UM GLOSSÁRIO PARA PROFISSIONAIS DA ÁREA DE QUÍMICA

Trabalho de conclusão de curso apresentado ao programa de graduação em Química Tecnológica do Departamento de Química da Universidade Federal de São Carlos para obtenção do título de bacharel em química tecnológica

Orientador: Prof. Dr. Edenir Rodrigues Pereira Filho

SÃO CARLOS

2022

FUNDAÇÃO UNIVERSIDADE FEDERAL DE SÃO CARLOS

DEPARTAMENTO DE QUÍMICA - DQ/CCET

Rod. Washington Luís km 235 - SP-310, s/n - Bairro Monjolinho, São Carlos/SP, CEP 13565-905

Telefone: (16) 33518206 - http://www.ufscar.br

DP-TCC-FA nº 15/2022/DQ/CCET

Graduação: Defesa Pública de Trabalho de Conclusão de Curso

Folha Aprovação (GDP-TCC-FA)

FOLHA DE APROVAÇÃO

PEDRO HENRIQUE FERREIRA LIMA

DATA SCIENCE: UM GLOSSÁRIO PARA PROFISSIONAIS DA ÁREA DE QUÍMICA

Trabalho de Conclusão de Curso

Universidade Federal de São Carlos – Campus São Carlos

São Carlos, 09 de setembro de 2022

ASSINATURAS E CIÊNCIAS

Cargo/Função	Nome Completo
Orientador	Prof. Dr. Edenir Rodrigues Pereira Filho
Membro da Banca 1	M.Sc. Sileine Costa Rodrigues
Membro da Banca 2	Dra. Jeyne Pricylla Castro Castilho

Documento assinado eletronicamente por **Caio Marcio Paranhos da Silva**, **Professor(a)**, em 26/09/2022, às 22:20, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site https://sei.ufscar.br/autenticacao, informando o código verificador **0828690** e o código CRC **95E3CDC6**.

Referência: Caso responda a este documento, indicar expressamente o Processo nº 23112.035401/2022-32

SEI nº 0828690

Modelo de Documento: Grad: Defesa TCC: Folha Aprovação, versão de 02/Agosto/2019

Agradecimentos:

Inicialmente, agradeço aos meus pais, Francisco e Patrícia, por tudo que me forneceram ao longo da vida. Mas neste caso em especial, agradeço-os por todo suporte oferecido e que tornou possível a realização desta graduação.

Agradeço à minha irmã, Beatriz, que sempre foi uma grande companheira e amiga em minha vida. Assim como, agradeço à minha namorada, Larissa, pelo companheirismo e extensivo suporte oferecido nesta etapa final de minha formação.

Agradeço ao professor Dr. Edenir, por despertar em mim, através da matéria eletiva ministrada de Introdução à Quimiometria, o interesse pelos conceitos de *Design of Experiments* e *Data Science*. Além disso, agradeço ainda ao professor Dr. Edenir por todo apoio, conhecimento e ajuda, que tornaram este projeto viável.

Agradeço à Universidade Federal de São Carlos e seu corpo docente, por toda estrutura fornecida e todo conhecimento transferido. Assim como, agradeço à UFSCar por todas pessoas que conheci durante a graduação. Das quais, agradeço especialmente meus grandes amigos André e Neri, que estiveram comigo durante a graduação, desde o primeiro dia. Ambos foram meus companheiros ao longo de todos os desafios desta etapa, não somente os desafios acadêmicos como as provas, trabalhos, seminários e afins, mas também pelos desafios da mudança à uma nova cidade e um novo ciclo.

Agradeço, por fim, à LabWare Brasil pela oportunidade de estagio, que além de me agregar uma série de conhecimentos, também inspirou parcialmente a realização deste projeto. Assim como, agradeço à minha gestora de estágio, Sileine, por aceitar previamente compor a banca avaliadora.

Resumo:
O *Data Science* visa fornecer estratégias e ferramentas capazes de otimizar processos e aumentar a robustez dos resultados, sendo um mecanismo poderoso e determinante para potencializar a capacidade de pesquisa e produção, assegurando a conformidade dos centros tecnológicos e laboratórios. Os métodos estatísticos compõem a base que fundamenta a ciência de dados, em suas variadas aplicações e distinções, sendo de grande valia para experimentalistas de todos os setores. Diversas ferramentas e linguagens computacionais se destacam com a finalidade de capacitar ou facilitar o estudo dos dados. Estas ferramentas atuam em conjuntura com o *Design of Experiments* (planejamento experimental), sendo fundamentais para obtenção conclusiva de dados e de decisões assertivas. A otimização de processos, atrelada à redução de custos e desperdícios materiais, levam à conformidade dos centros produtivos e geram a interface entre *Data Science, Design of Experiments* e os processos de controle da qualidade. As ferramentas de programação para *Data Science* atuam desde o início do processo de análise exploratória dos dados, iniciando-se com a leitura e interpretação de arquivos de texto (txt), CSV (*comma-separated values*), XML (*extensible markup language*) e outros. Em sequência, os dados obtidos são tratados e modelados para o uso dos métodos matemáticos e estatísticos, obtendo informações determinantes para conclusões acerca dos experimentos, como parâmetros de estatística descritiva e inferencial, testes de hipótese, ANOVA e regressão linear. Este trabalho se concentra na descrição das seguintes ferramentas do *Data Science* e métodos do controle da qualidade: R, Python, C^{++}, Matlab, Octave, SQL, 6σ e 5S.

Palavras-chave: *Design of Experiments*, Análise de dados, *Data Science*, *Data Mining*, Linguagem R, Python, C^{++}, Matlab, Octave, SQL, Controle da qualidade.

Abstract:
Data Science aims to provide strategies and tools capable of optimizing processes and increasing results strength, being a powerful and decisive mechanism to enhance the research and production capability, which ensures the conformity of technology centers and laboratories. The statistical methods comprise the basis which underpin data science, in its varied applications and distinctions, being invaluable for researchers of all sectors. Several tools and computing languages stand out as they aim to enable or facilitate the data analysis. Those tool work together with the Design of Experiments, being fundamental for obtain conclusive data and making assertive decisions. The optimization processes, along with cost and waste reduction, lead to standardization of the production center and allow the interface between Data Science, Design of Experiments and quality control processes. The programming tools for Data Science play a roll since the beginning of the exploratory analysis of the data project, starting with the reading and comprehension of text files (txt), CSV (comma-separated values), XML (extensible markup language) and others. These obtained data are treated and modeled, becoming appropriate to go through mathematical and statistical methods, obtain specific information for conclusions about experiments, based on descriptive and inferential statistics, hypothesis tests, ANOVA and linear regression. This work is centered on the description of the following programming tools and quality control methods: R, Python, C++, Matlab, Octave, SQL, 6σ and 5S.

***Key-Words*:** Design of Experiments, Data Analysis, Data Science, Data Mining, R, Python, C^{++}, Matlab, Octave, SQL, Quality Assurance.

Lista de figuras

Lista de tabelas

SUMÁRIO

1 – Introdução

1.1 – Sobre a empresa LabWare e gestão dos dados laboratoriais

A LabWare é uma empresa estadunidense fundada em 1987 por Vance Kershner, na cidade de Wilmington em Delaware. Kershner, durante sua experiência trabalhando na empresa Dupont, vislumbrou a oportunidade de atuar em um mercado de automação e gerenciamento de dados laboratoriais. Uma das propostas iniciais, e central, da empresa era a de automatizar o registro de dados dos instrumentos e equipamentos, permitindo sua inserção automática dentro de um sistema. Partindo desse pressuposto, a empresa inicia suas atividades na produção e fornecimento de softwares LIMS e, posteriormente, ELN [1].

Um software LIMS (*laboratory information management system*) tem a finalidade de armazenar, gerenciar, tratar e auditar todos os dados produzidos em um laboratório, fornecendo ferramentas que irão acompanhar os dados em todo seu ciclo de vida, desde o registro até a obtenção de resultados e relatórios a partir dos mesmos. Já o software ELN (*Electronic laboratory notebook*) fornece uma visualização computacional amigável, para que sejam inseridos os dados obtidos por experimentos e ensaios, usualmente decorrentes de processos de pesquisa, entregando *templates* configuráveis, que se adequam ao fluxo de experimentação dos pesquisadores. Assim, é possível obter e organizar os dados, sejam estes registrados automaticamente dos instrumentos ou inseridos manualmente, para que por fim possam ser tratados e analisados dentro das funcionalidades do *software* LIMS, até que gerem resultados e relatórios [1].

Empresas e centros de pesquisas cada vez mais geram e armazenam dados, tornando crescente a busca por sistemas e ferramentas capazes de armazenar e gerenciar os mesmos. Além disso, é requerido que os procedimentos sejam práticos, ágeis e que empreguem ferramentas capazes de extrair o máximo de informação, bem como realize o armazenamento de acordo com a segurança dos dados.

Tendo em vista o apresentado, a área de atuação da empresa, que atua diretamente com a obtenção e gerenciamento de dados, foram também um dos motivadores para realização deste projeto.

1.2 – Definindo experimentos e variáveis do processo

Experimentos são ensaios, ou conjuntos de ensaios, nos quais se executam variações premeditadas nas variáveis que compõem o sistema experimental. Essa variação premeditada possibilita inferir de qual maneira as variáveis controláveis do processo afetam a variável resposta do experimento [2].

Em um experimento existem variáveis controláveis que afetam a variável resposta. A variável resposta é aquela que se refere ao dado que se busca em um ensaio. As variáveis controláveis (fatores) são aquelas que planejadamente variam ao longo de um ensaio, possibilitando que os experimentalistas entendam o comportamento que esta variação impacta sobre a variável resposta. Os fatores se apresentam em diferentes valores, denominados níveis. Quando os fatores são quantitativos, estes níveis podem ser classificados como altos ou baixos, isto é, maior e menor valor em escala gradativa. A combinação dos fatores forma o que se conhece como tratamento [2].

Existe ainda, o erro residual, uma variável incontrolável que compõem o dado obtido como resposta. O resíduo é inerente ao experimento, sua presença se dá ao acaso e pode ser provocado por qualquer componente do sistema, como: instrumento utilizado, flutuações experimentais, clima externo, posição dentro do campo de coleta (fator presente em estudos de campo) e outros [2].

Exemplificando: Suponha que se deseje determinar de que forma a temperatura e a composição afetam a medida de pH de uma mistura. Para executar esse estudo, suponha ainda que a composição da mistura seja proveniente de dois componentes A e B, neste caso distribuindo 6 tubos de ensaio com as seguintes composições:

Frasco	Componente A (%)	Componente B (%)
A1	50	50
A2	50	50
B1	70	30
B2	70	30
C1	30	70
C2	30	70

Tabela 1 – Disposição dos frascos nas diferentes composições

Em seguida, separando os tubos A1, B1 e C1 em um conjunto (bloco 1) e os tubos A2, B2 e C2 em outro conjunto (bloco 2), realiza-se a medida de pH dos

tubos do bloco 1 à temperatura de 20°C e a medida de pH dos tubos do bloco 2 à temperatura de 80°C.

Neste caso hipotético, os fatores seriam a temperatura e a composição da mistura, a variável resposta seria o pH obtido, os valores de temperatura de 20°C e 80°C seriam os níveis baixo e alto da variável temperatura, respectivamente. Os tratamentos seriam os 6 conjuntos estudados para obtenção da variável resposta, como mostrado a seguir:

Tratamento	Composição	Temperatura
Tratamento 1	Frasco A1	20°C
Tratamento 2	Frasco A2	80°C
Tratamento 3	Frasco B1	20°C
Tratamento 4	Frasco B2	80°C
Tratamento 5	Frasco C1	20°C
Tratamento 6	Frasco C2	80°C

Tabela 2 – Tratamentos do experimento genérico

Realizada a medida para cada tratamento, obtém-se o valor da variável resposta nas diferentes composições propostas, e nas duas diferentes temperaturas. Pode-se então decidir um método ou ferramenta adequada para tratar as informações coletadas e inferir de que forma cada uma afeta o valor de pH.

Algumas considerações podem ser feitas sobre este exemplo, note que o fator composição possui 3 níveis, enquanto que o fator temperatura apresenta 2 níveis, o número de tratamentos realizados sempre será igual a multiplicação da quantidade de níveis dos dois fatores, como apresentado, o exemplo conta com 6 tratamentos. Note que se fossem feitas réplicas dos tratamentos, o valor do total de tratamentos seria obtido pela multiplicação da quantidade de níveis de cada fator e pela quantidade de réplicas.

O experimento genérico apresentado foi planejado para ser executado em uma análise multivariada de fatores, na qual ao longo do experimento foram testados todos os níveis de composição, e em todos os níveis de temperatura pré-estabelecidos. Essa escolha ocorre para que sejam consideradas as interações entre as variáveis. As interações são flutuações que podem interferir nos valores previstos da variável resposta, a depender do comportamento de um fator na presença do outro. Em outras palavras, não se pressupõem que um fator não afeta

ao outro, e que consequentemente, não haverá flutuações na variável resposta. A análise OVAT (*one variable at time*) ignora a ocorrência das interações, tal escolha pode tornar estudos falhos em suas conclusões [3].

1.3 – Design of Experiments (DoE): Vantagens de um delineamento assertivo

Em uma fábrica são produzidos diversos produtos diariamente, sobre estes devem ser realizados uma série de testes e análises para assegurar que possuam a qualidade esperada. O controle da qualidade de uma empresa é um setor de grande importância, e que quando funcional, gera consideráveis ganhos de confiabilidade por parte de seus clientes. Com a finalidade de melhorar a qualidade industrial, a produtividade, o desempenho do produto final, os custos das operações, entre outras características, as empresas realizam vários experimentos para identificar os níveis ótimos dos parâmetros que regulam seus processos de fabricação [4].

Em muitas ocasiões é inviável ou dificultada a realização adequada dos testes necessários por conta da quantidade de esforços, materiais (padrões e reagentes) e custos para tal. Estes problemas podem ser contornados quando os experimentos são planejados e analisados com métodos e técnicas estatísticas adequadas. O DoE, também conhecido na literatura como planejamento ou delineamento de experimentos, é uma ciência que consiste em um conjunto de técnicas de análises estatísticas e experimentais, que fundamentam profissionais das mais diversas áreas a planejar, modelar e analisar experimentos, gerando maior confiabilidade dos resultados obtidos, apoiando as tomadas de decisões, otimizando e agilizando seu processo [4].

O planejamento e delineamento de experimentos objetivam aumentar a capacidade de conclusão acerca dos experimentos, reduzindo de forma considerável o efeito do erro residual, ou seja, visa eliminar a ocorrência da variação ao acaso. O delineamento é a etapa do projeto em que o experimentalista irá decidir de que forma será executado o experimento, definindo como ele irá organizar os tratamentos dentro das unidades experimentais.

Existem três principais delineamentos experimentais: Delineamento completamente casualizado (DCC), delineamento em blocos casualizados (DBC) e delineamento em quadrado latino (DQL). Estes formatos de experimentação são

baseados em três principais componentes que asseguram a validade, a possibilidade de execução e a efetividade dos ensaios experimentais, garantindo também a assertividade dos testes estatísticos e análise de dados que sucedem os ensaios experimentais. Esses componentes são: Casualização, repetição e o controle da unidade experimental (blocos) [2].

Casualização:

A casualização consiste em realizar uma distribuição estocástica dos tratamentos dentro das unidades experimentais, garantindo que todos os tratamentos possuam probabilidades equalitárias de ocupar qualquer unidade. Através da casualização se pode garantir uma distribuição independente do erro residual [2].

Repetição:

A repetição (réplica) consiste na quantidade de vezes que um tratamento é replicado dentro do experimento. Em outras palavras, a repetição não corresponde a um novo e diferente tratamento, mas sim a replicar um tratamento, com a intenção de aumentar a confiabilidade do resultado obtido e verificar o erro aleatório inerente às medidas. A repetição permite estimar o erro experimental, uma vez que existirá mais de um resultado referente a um mesmo tratamento, de forma que variações na variável resposta dentro das réplicas podem expressar o efeito do erro experimental [2].

Blocos:

O controle das unidades experimentais, ou controle da distribuição dentro de blocos, consiste em dividir estrategicamente a unidade experimental para evitar influências externas. Tomando como exemplo a distribuição do gradiente de temperatura dentro de um forno, as posições mais próximas à fonte de calor serão submetidas a uma maior influência do gradiente do que aquelas mais distantes, tornando interessante subdividir as posições dentro desse forno em blocos.

Como descrito anteriormente, o controle dos blocos é uma ferramenta poderosa para evitar que fatores externos interfiram nos valores obtidos nos ensaios, garantindo a homogeneidade do experimento. Contudo essa técnica deve ser aplicada com grande cautela, sendo de extrema importância que o

experimentalista conheça amplamente o processo de estudo, os equipamentos e o local do experimento antes de aplicar sua estratégia de distribuição de blocos [2].

Delineamento completamente casualizado:

O delineamento completamente casualizado, como o nome indica, é aquele no qual não há restrições para a alocação dos tratamentos dentro das unidades experimentais. Neste formato todas as entradas possuem a mesma probabilidade de ocupar uma unidade experimental.

O pressuposto desse delineamento é de que não existam vícios nos valores aleatórios obtidos para alocação dos tratamentos, sendo garantida a estocasticidade da distribuição. A garantia de que não exista uma tendencia forçada dentro da alocação é um pressuposto fundamental para sustentar os testes estatísticos do processo.

Por conta das características deste delineamento, recomenda-se seu uso quando não existe nenhuma condição ou especificação singular para a unidade experimental, assegurando que todas as entradas sejam homogêneas [2].

Delineamento em blocos casualizados:

O delineamento em blocos casualizados é aplicado quando existe uma fonte conhecida de variância dentro das unidades experimentais, sendo necessário que exista um controle sobre a alocação dos tratamentos. Diferentemente do caso completamente casualizado, não há uma garantia de homogeneidade das possíveis entradas, contudo, é possível que sejam manipuladas e criadas divisões internas dentro da unidade que garantam a homogeneidade interna a cada bloco.

Voltando ao exemplo do forno citado anteriormente, no qual as áreas próximas a fonte de calor sofriam maior influência desta energia. Imagine que a fonte de calor seja localizada exatamente no centro deste forno que possui um formato quadrado. Dessa forma, pode-se inferir uma divisão para que sejam alocadas amostras ao centro (maior influência ao calor), às proximidades do centro (influência intermediária do calor) e que sejam alocadas amostras próximas as paredes deste forno (menor influência ao calor). Quando os blocos são comparados entre si, fica evidente que existe um fator externo de variação atuando

sobre eles, porém quando se pensa internamente aos blocos criados, nota-se que estes não possuem influência desta fonte de variação conhecida.

O experimento realizado em DBC deve possuir número de réplicas igual ao número de blocos existentes, uma vez que para garantir a validade estatística do processo todos tratamentos devem igualitariamente passar pelo ensaio dentro de cada bloco. A casualização ocorre devido a probabilidade, de que cada tratamento ocupe cada um dos blocos existentes, seja a mesma [2].

Delineamento em quadrado latino:

O delineamento em quadrado latino é aplicado quando existem duas fontes de variações conhecidas, que atuam sobre a unidade experimental. Visando homogeneizar os blocos, deve-se visualizar a tabela de tratamentos como uma matriz, na qual um tratamento aparece apenas uma vez em cada linha e cada coluna, isto é, um tratamento experimenta os efeitos de variação externa necessariamente, e somente, uma vez.

Como o nome do delineamento e o formato de visualização sugerem, por se tratar de um quadrado e uma matriz quadrada respectivamente, uma limitação para a aplicação do DQL é a de que o número de tratamentos deve ser o mesmo que o número de réplicas, tornando este processo muitas vezes inviável por conta da quantidade de execuções que podem ser necessárias para experimentos com muitos tratamentos [2].

Planejamento fatorial:

Além dos delineamentos experimentais, as práticas do *Design of Experiments* englobam também as técnicas de planejamento fatorial. O planejamento fatorial de experimentos é uma técnica analítica que tem duas principais finalidades. Uma destas, é a de capacitar os experimentalistas em determinar quais são as variáveis (fatores) que possuem maior nível de significância para a execução do sistema experimental. O nível de significância de um fator, conhecido como o efeito de um fator, é medido através das análises dos dados coletados da execução do sistema experimental [2].

A outra finalidade do planejamento fatorial está relacionada ao número de experimentos necessário em uma prática experimental. Conforme abordado

anteriormente, o número de experimentos necessários é obtido através da multiplicação da quantidade de fatores existentes pela quantidade de níveis dos fatores. Tendo isto em vista, pode-se presumir que experimentos com maior quantidade de fatores e níveis irão ocasionar em uma quantidade inviável de experimentos para serem realizados [2].

Nesse sentido, desenvolveu-se a técnica do planejamento fatorial 2^k. Um experimento fatorial 2^k é desenhado para que cada fator possua apenas dois níveis, um alto e outro baixo. Dessa forma, neste sistema o k representará o número de fatores existentes, por tanto, a quantidade de experimentos necessários será sempre obtida pela expressão 2^k [2].

Os principais planejamentos fatoriais existentes são: Fatorial completo, fatorial fracionário, Doehlert e Box-Behnken. Os quais serão descritos a seguir:

Planejamento Fatorial Completo

O planejamento fatorial completo consiste, como o nome sugere, na execução de planejamento fatorial em si, sem necessariamente aplicar nenhuma regra ou especificação, diferenciando-o dos demais, como será abordado. Por tanto, a quantidade de experimentos executadas segue a regra da expressão 2^k [2].

Para elaborar a tabela de planejamento experimental em sistema fatorial, o experimentalista cria um rotulo para designar nível alto e baixo, a maneira mais comumente aplicada é chamar o nível baixo de -1 e alto de +1. Dessa forma, cria-se uma matriz X, em que cada coluna representa os fatores utilizados. As linhas representarão os diferentes tratamentos, ou seja, todas as possíveis combinações existentes para todos os fatores em todos níveis, obtida pela expressão 2^k. O preenchimento da planilha pode seguir de diversas maneiras, não havendo uma regra ou requisito critico, além é claro da obrigatoriedade de se garantir que todas as combinações sejam executadas. Terminado o preenchimento da matriz X, e uma vez executado os experimentos planejados, as variáveis respostas obtidas compõem o que chamados de vetor Y, que deve ser posicionado à direita da matriz X [2].

Planejamento Fatorial Fracionário

O planejamento fatorial fracionário consiste em reduzir a quantidade de execuções necessárias de forma planejada, para tal, a execução seguira a quantidade pré-estabelecida pela expressão 2^{k-p}, na qual o k segue representando a quantidade de fatores e o p será uma constante arbitrariamente definida pelo experimentalista [2].

Exemplificando, suponha que se deseja realizar um experimento fatorial com 4 fatores, o planejamento infere a escolha de apenas dois níveis, gerando a quantidade de ensaios, seguindo a expressão 2^k, de 16 experimentos. Um fatorial fracionário consiste em escolher um valor para p e então obter a nova quantidade de experimentos. Suponha ainda que definimos o valor 1 para p, portanto, a nova quantidade de ensaios será 8. Desta situação podemos extrair uma vantagem e uma desvantagem, sendo estas a diminuição da quantidade de ensaios e a diminuição de informações para determinação do efeito dos fatores, respectivamente [2].

Como descrito, é evidente que a redução da quantidade de fatores impossibilita combinar todos níveis de todos fatores. Dessa forma, existirão varias formas de combinar os fatores e seus níveis para gerar o planejamento experimental reduzido. Uma das maneiras de distribuir os ensaios é levar em conta o valor obtido da expressão k-p e formar a tabela para um fatorial completo 2^{k-p}. Em outras palavras, considerando o exemplo anterior, pode-se construir uma tabela de fatorial completo para os 3 primeiros fatores e que terá 8 entradas, e posteriormente aleatorizar a entrada dos níveis do quarto fator.

Planejamento Doehlert

O planejamento Doehlert é um dos métodos analíticos que empregam as técnicas de MSR (Metodologias de Superfície de Resposta). As metodologias MSR consistem no uso de aplicações algébricas e estatísticas para obtenção, em estudos com várias variáveis, de sistemas visuais, em suma, os gráficos de superfície, que indicam quais são as variáveis de maior influência. As superfícies amplificam a capacidade analítica acerca das influências e interações fatoriais [2].

A aplicação do planejamento Doehlert ocorre pelo uso da matriz de Doehlert, para definição da distribuição dos coeficientes matriciais, levando-se em conta a posição do ponto central, que é definido pela quantidade de fatores

estudados. O domínio da distribuição em torno do ponto central é definido pela quantidade de fatores estudadas, sendo uma distribuição esférica para dois fatores e hiperesférica para mais fatores [2].

Box-Behken

O experimento executado em Box-Behnken, assim como o Doehlert, usa as técnicas de MSR. A obtenção do planejamento experimental segue através do uso do cubo experimental para definir os experimentos, isto é, a definição dos níveis e consequentemente, a formação dos tratamentos, é feita pelas posições definidas dentro do cubo experimental [2].

O planejamento é realizado para estudo de três fatores ou mais, considerando-se as três dimensões do cubo. As posições estabelecidas dentro da superfície do cubo inferem que o experimento seja realizado com dois fatores em seus níveis extremos, isto é, nível alto e baixo, e que o terceiro fator seja testado em uma posição central, ou seja, um valor intermediário [2].

Estratégia de experimentação e escolha dos parâmetros:

Conhecidos os parâmetros que compõem uma experimentação e as técnicas que garantem a diminuição dos erros experimentais, deve-se pensar em aplicar tais conhecimentos dentro de uma estratégia de experimentação, para tal, algumas garantias devem ser verificadas [3]:

- Conhecimento preciso do experimento: Conhecer plenamente o objeto de estudo, o processo experimental e, principalmente, qual finalidade do experimento é o princípio para uma estratégia efetiva. O bom entendimento dessas etapas torna mais preciso o julgamento do experimentalista acerca das necessidades de seu processo, assim como quais são as possíveis e principais fontes de variações e causadoras do erro residual.
- Determinar a variável resposta e fatores do experimento: Deve-se, desde o princípio do projeto, definir qual a variável resposta buscada. Esta decisão deve preceder a determinação dos fatores, uma vez fixado qual o objetivo, deve-se basear nisso a determinação dos fatores, seguindo para a determinação de quais serão os níveis dos mesmos. Dessa forma, ficam estabelecidas quais serão as variáveis

controláveis do processo, as quais se deseja conhecer a influência sobre a variável resposta. Essa cautela permite garantir que os fatores estudados são de fato influentes sobre a variável resposta, da mesma forma que garante que os níveis escolhidos são cabíveis dentro da análise executada.

- Determinar o delineamento e planejamento fatorial no qual será executado o experimento: Uma vez que o experimentalista conhece por completo o processo, que determinou os fatores, seus níveis e a variável resposta, pode-se partir para etapa de determinar qual será o delineamento experimental e se será executado algum planejamento fatorial para reduzir a quantidade de experimentos.
- Realização do experimento, coleta e análise dos dados: Após planejar o procedimento experimental, chega a hora de executar. É de conhecimento de todos os laboratoristas que esta etapa requer imensa atenção, para que não sejam perdidas informações e cautela para executar os ensaios dos tratamentos, dentro dos parâmetros planejados. Finalizado os ensaios e coletados os dados, deve-se passar para etapa de tratamento e análise. Para tal, deve-se voltar para as bases teóricas da estatística e da matemática, concentrando-se principalmente nos cálculos de análise de variância (ANOVA) [3], testes de comparações múltiplas (testes de médias) e testes de regressão linear (previsibilidade de resultados).

A estratégia determinada pelo experimentalista influencia diretamente sobre como será realizada a análise dos dados. A depender do caso estudado, da quantidade de informações a ser analisada, no tempo desprendido para realiza-las e também na comodidade, surge o *Data Science*. Um conjunto de práticas que aumenta a capacidade e velocidade da extração de informações acerca de um conjunto de dados [4].

O *Data Science* é proposto como o 4º paradigma da ciência, sendo uma etapa que entrega as ferramentas necessárias para extrair informações úteis dos dados. O crescente uso do *Data Science* apresenta uma mudança e avanço nos métodos científicos, se colocando como um novo passo na evolução científica, seguindo os três paradigmas anteriores em sequência, evidências empíricas

(experimentais), teorias e modelos científicos, e ciência computacional, ver Figura 1 para mais detalhes [5].

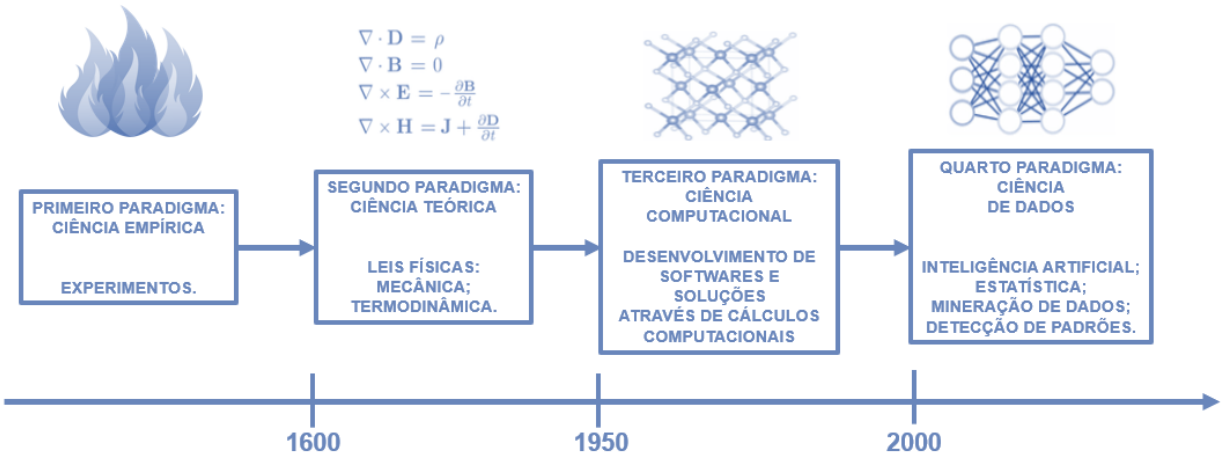

Figura 1 – Os quatro paradigmas do método científico.

2 – Objetivo

O avanço tecnológico, aliado às novas tendências da indústria 4.0, tornam imprescindível que os pesquisadores, experimentalistas ou laboratoristas, adaptem-se a esse novo momento histórico da indústria. Existem diversas razões que fundamentam que estes profissionais se atualizem e estejam preparados para as novas demandas que possam surgir. Dentre as quais, destacam-se a capacidade de utilizar habilmente os computadores, visando extrair deles seu potencial e, aliá-los aos processos de análises experimentais, aumentando a produtividade, agilidade e confiabilidade da obtenção de resultados e relatórios. Seja em processos de pesquisa e desenvolvimento (P&D), ou em processos de garantia da qualidade (QA), as ferramentas do *Data Science* são capazes de oferecer soluções práticas, ágeis e confiáveis.

Tendo em vista o apresentado, este trabalho se dispõe a apresentar e descrever, para pesquisadores e cientistas, as principais ferramentas, métodos e terminologias do *Data Science* e dos métodos de controle da qualidade.

3 – Discussão

3.1 – Data Science

Para começar um descritivo sobre *Data Science*, pode-se iniciar definindo o objeto central de estudo, o dado. Os dados são todos os componentes coletados e armazenados em determinado processo. Assim, seu propósito é representar os fatos coletados em um experimento dentro de um contexto computacional, que uma vez analisado de forma coesa, levará à obtenção de uma informação. Definimos a informação como um dado que após ser processado permite que sejam obtidos significados e conclusões acerca do processo. Em maior hierarquia encontra-se o conhecimento, que corresponde aos padrões e resultados tratados e obtidos no processo [6].

Informações e conhecimento sustentam uma tomada de decisão. Em muitas ocasiões um projeto de *Data Science* é realizado direcionando a sua aplicação para formular um sistema de apoio a decisão (SAD). Estas são modelagens estratégicas que apoiam cientistas em obter a maior conjuntura de informações e conhecimento antes de prosseguir para a etapa de conclusão e tomada de decisão.

Com o avanço da tecnologia e a crescente produção de informações que precisariam ser armazenadas e posteriormente tratadas, surge um novo conceito e uma nova necessidade de expandir as ferramentas existentes para o estudo, a coleta, a manipulação e a análise de dados. O *Data Science* é uma ciência multidisciplinar entre a estatística, a ciência da computação e as práticas experimentais. Através da correlação entre as bases teóricas e técnicas destes pilares, aliadas com métodos, tais quais o *Machine Learning* e o *Data Mining*, o experimentalista é capaz de estudar o dado em todo seu ciclo de vida [7]. A Figura 2 apresenta uma visão macro das etapas de um processo *Data Science*.

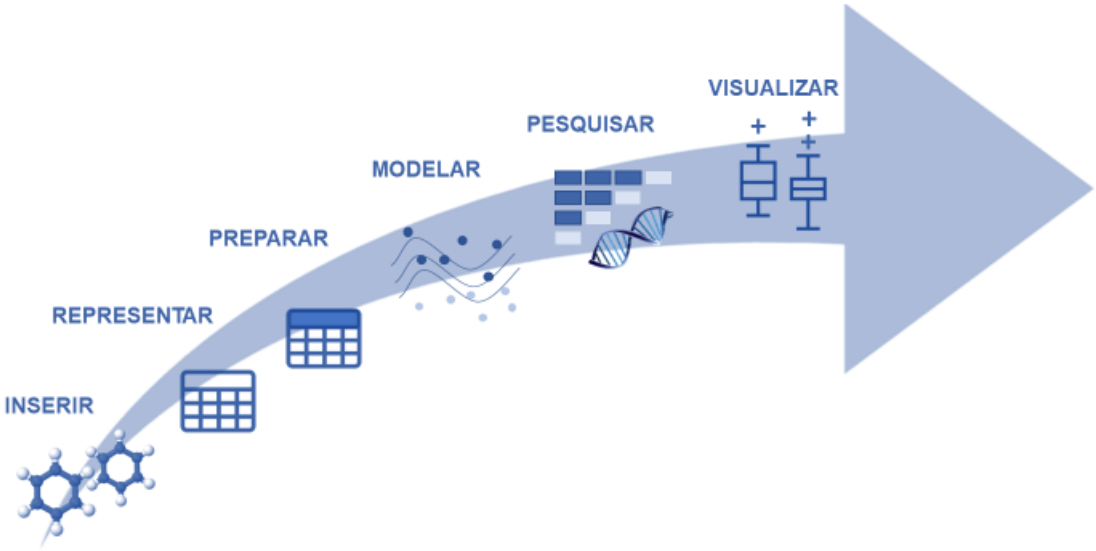

Figura 2 – Etapas e procedimento de um projeto de *Data Science.*

Não é incomum que o *Data Science* seja erroneamente associado apenas às técnicas estatísticas, sendo presumido que sua aplicação seja voltada exclusivamente à etapa de análise de dados. Muito além disso, o Data Science engloba todas as etapas do ciclo de vida do dado, desde a importação até a obtenção de informações, seguindo genericamente o fluxo da Figura 3.

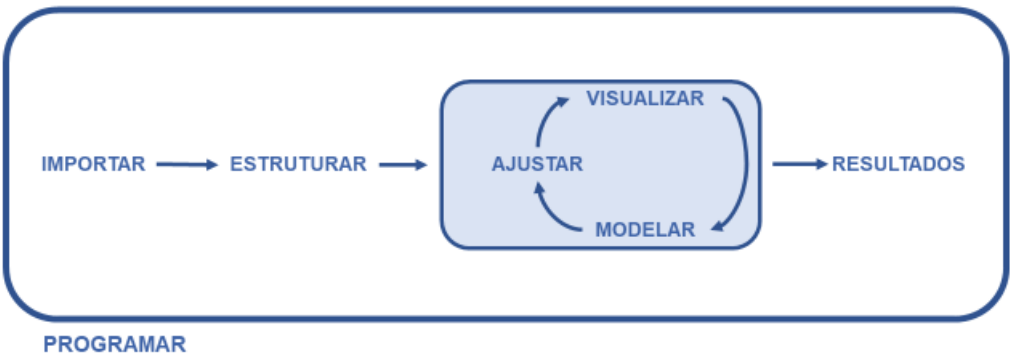

Figura 3 – Fluxograma base para análise de dados em Data Science.

Sintetizando, a ciência de dados é uma extensa conjuntura de práticas, o experimentalista que busca nessa área obter informações suficientes para elucidar e apoiar seus processos, precisará conhecer eficientemente as técnicas necessárias que precedem e sucedem a análise de dados. Em outras palavras, o processo não se resume apenas a coletar os resultados do laboratório e realizar alguma análise de dados sem critério [8]. Um processo bem sucedido de *Data Science*, para aplicações em pesquisas e laboratórios, inicia-se com a execução

correta do experimento, usando os fatores corretos e o delineamento adequado, uma coleta assertiva dos dados e, quando necessário, a aplicação apropriada de técnicas de amostragem. Obtido o conjunto de dados, é necessário perícia para: Importá-lo e estruturá-lo coesamente, dentro do *software* de preferência, possibilitando que sejam obtidas informações deste conjunto reestruturado de dados.

As informações que se podem obter são tanto aquelas referente as análises estatísticas dos dados como ANOVA, teste de comparações múltiplas e regressão linear, como já citadas anteriormente, mas também informações visuais. O *Data Science* e os *softwares* de linguagem computacional são ferramentas poderosas para a obtenção de gráficos, das mais diversas finalidades, possibilitando que o experimentalista não somente aumente a sua própria visualização dos resultados obtidos, como também capacite que outros profissionais leigos sobre o assunto possam compreender.

Como ilustrado na Figura 3, o trabalho do cientista de dados se inicia na importação dos dados, segue para uma etapa de grande impacto para o processo, e de grande interesse comercial, que compreende a modelagem e organização dos dados [7]. Um conjunto de dados devidamente estruturado é capaz de fornecer informações valiosas, assim como torna todo processo subsequente de análise extremamente mais ágil e confiável. A potencialização da obtenção de padrões e correlações, dentro de conjuntos de dados, gera informações de grande valor. Fato que impulsionou a exploração do *Data Mining*, parcela do *Data Science* que explora a produção de *insights* (perspicácia ou capacidade de tomar conclusões assertivas) acerca do conjunto de dados [7].

3.2 – Data Mining

O grande volume de dados armazenados, atrelado a quantidade de informações inseridas dos bancos de dados, tornou ainda mais significante a capacidade de transformar dados brutos em *insights*. O *Data Mining* é a ferramenta responsável por visualizar e obter padrões comportamentais dentro das *databases*. A capacidade de destacar esses padrões e estabelecer correlações é o que definimos como os *insights*. A eficiência do *Data Mining* e consequentemente a confiabilidade dos *insights* obtidos depende intimamente da

eficácia das etapas de transformação e modelagem dos dados. Dados que foram devidamente “limpos” e apresentam uma estrutura coesa, tornam possível a investigação e comparação interna por um *software* adequado [9].

O *Data Mining* objetiva identificar características estruturais do conjunto de dados, sejam estas tendências, correlações ou dependências. Diferentemente da análise de dados estatística mais purista do *Data Science*, na qual o experimentalista define o problema que quer estudar, seleciona os dados de seu interesse e decide quais métodos de estudo de hipótese serão utilizados, o *Data Mining* faz uso de um sistema computacional de pesquisa interno aos dados, que retorna as semelhanças e anomalias que foram observadas, indicando ao pesquisador quais serão os problemas que ele terá de identificar e solucionar [9].

A análise e o planejamento exploratório dos dados, aliados à modelagem, constituem a base do processo ao qual o data mining está inserido: O processo KDD (*Knowledge Discovery in Database*). O KDD consiste no vasto conceito de encontrar correlações, informações e conhecimento dentro de um banco de dados. Uma definição comum proposta ao termo é de que: “KDD é um processo não trivial, interativo e iterativo, para identificações de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados” [10].

Conforme a descrição do processo KDD, o fator iterativo do processo indica a possibilidade de repetir o processo de exploração dos dados inúmeras vezes. Assim, é possível obter *insights* suficientes sobre o conjunto de dados para formar informações fundamentadas destes. Por outro lado, o fator interativo indica a participação humana no processo, refletindo na necessidade de que o experimentalista controle o processo. A prática e repetição são fundamentais para aumentar a perspicácia do pesquisador, potencializando a sua capacidade de reunir *insights* assertivos. A Figura 4 ilustra um fluxo genérico de um projeto KDD e como o *Data Mining* está inserido neste processo.

Figura 4 – Etapas de um processo KDD e *Data Mining*

A prática de data mining está intimamente relacionada ao conceito de *Big Data*. A definição de *Big Data* está relacionada aos “3 Vs”, que são: Volume, velocidade e variedade. São dados que possuem maior variedade, que são entregues em maiores volumes e com velocidades crescentes. Em outras palavras são dados maiores e complexos, que cada vez mais possuem maior variedade de informação. Alguns especialistas e literaturas adicionaram mais “2 Vs”, o valor e a veracidade. Estes dois surgem, pois, dados possuem valor intrínseco, sendo necessário verificar a veracidade dos mesmos [7].

3.3 – Ferramentas do Data Science

O experimentalista que deseja encontrar no *Data Science* as ferramentas necessárias para potencializar, agilizar e aumentar a confiabilidade de seu processo, precisará inevitavelmente se lançar ao aprendizado de alguma linguagem de programação. Existem diversas ferramentas, *softwares* e linguagens de programação que apoiam um cientista de dados, muitas destas compartilham diversas semelhanças, mas ao mesmo tempo possuem singularidades e vantagens únicas entre si. De certa forma, caberá ao pesquisador optar por aquela que mais lhe agrada, excetuando-se poucos casos particulares nos quais somente uma ferramenta especifica será funcional para o seu problema.

Diversos padrões de programação em linguagem computacional surgiram para viabilizar a comunicação entre humanos e as máquinas. É de conhecimento geral que os computadores são máquinas com potenciais incríveis e incansáveis, capazes de solucionar problemas imensos que tomariam horas ou dias para um humano comum, tais características torna de grande valia que seja possível uma comunicação com estes objetos. Principalmente, para que através de tal, sejam

gerados códigos que comandam tarefas para estas máquinas de tamanha capacidade [11].

Evidentemente, a necessidade de aprender uma linguagem computacional assombra alguns exploradores de primeira viagem, entretanto, esta não deve ser uma grande preocupação. Tais *softwares* são constantemente atualizados, sendo desenvolvidos diversos pacotes e funcionalidades regularmente, que trazem funções cada vez mais simples de resolver longos problemas. A Figura 5 mostra um exemplo de execução em linguagem R. Este código de 8 linhas de comando (as demais são linhas de comentários) é capaz de retornar informações valiosas sobre um experimento fatorial realizado em delineamento de blocos casualizados. Como indicado na Figura 5, o programa retorna o resultado da análise de variância (ANOVA), dos testes de comparação múltiplas e a análise de variância para os desdobramentos.

- ## Instalar e carregar pacote necessário para os testes
- install.packages("easyanova")
- library(easyanova)
- ## Indicar endereço do arquivo e importar
- setwd("C:/Users/pedro/OneDrive/Desktop/UFSCar/TCC/DoE/R")
- D=read.table("FatDBC.txt",head=TRUE)
- ## Calcular ANOVA e comparações de médias
- ea2(D,design = 2)
- ## Instalar e carregar pacote necessário para ANOVA do desdobramento
- install.packages("ExpDes.pt")
- library(ExpDes.pt)
- ## Calcular ANOVA do desdobramento
- fat2.dbc(D$FatorA,D$FatorB,D$Bloco,D$Prod)

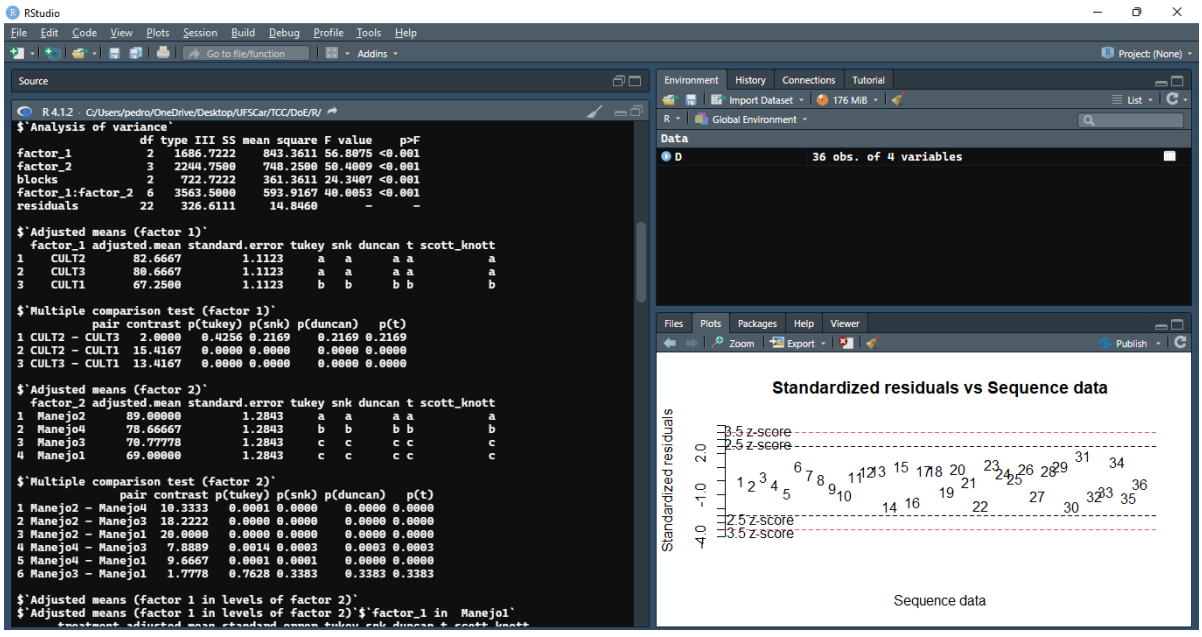

Figura 5 – Print de tela obtido do programa R na visualização Rstudio

Tal qual a aplicação apresentada, existem diversas outras que podem auxiliar experimentalistas a resolver problemas estatísticos e matemáticos, sem grande necessidade de escrever códigos longos e complexos. Todavia, é claro que conhecer profundamente uma ferramenta, possibilita que sejam resolvidos problemas cada vez mais desafiadores. Além disso, o *software* é capaz de responder aquilo que lhe é requerido, em outras palavras, é de extrema importância que o cientista de dados possua grande perspicácia e intuição para realizar as modelagens adequadas, para extrair a informação correta do banco de dados e também que saiba interpretar com destreza os resultados gerados pelo software. Concluindo, a ferramenta apresenta grande poder em agilizar, facilitar e otimizar os processos de *Data Science*, porém, ela não o faz assertivamente sem que lhe sejam informados os comandos e informações corretos, nem tampouco ela lhe entregará as conclusões aos resultados que ela obteve.

Utilizar o Python ou R requer um baixo nível de abstração, que torna muito mais intuitivo o aprendizado e sua aplicação. Por conta desta característica, estas linguagens não são referências de velocidade de processamento, principalmente quando comparadas ao C^{++}, um exemplo claro de linguagem de programação orientada a objetos, que requer uma indicação da classificação das estruturas que são inseridas no código. Esta diminuição na velocidade do processamento ocorre, pois o computador é obrigado a calcular e compilar essas indicações pelo usuário [11].

Enquanto a programação em C^{++} requer maior perícia do programador para especificar esse nível de estrutura, o R ou Python o fazem pelo usuário, gastando maior tempo de processamento. Contudo, apesar dessa pontuação referente as linguagens R ou Python, estas ainda se mostram muito mais eficazes que programas de uso ainda mais simples, como o Excel, sendo possível trabalhar com grande quantidade de dados. Para trabalhos com *Big Data* é altamente recomendado o uso do C^{++}, que funciona de forma muito mais eficiente que as demais [11].

Matlab e Octave utilizam uma linguagem direcionada com um formato que se assemelha à escrita de expressões algébricas, classificando as estruturas dos dados em grupos de unidades matemáticas, como vetores e matrizes. Essa

especificidade de linguagem dos dois *softwares* os tornam menos símil ao R ou Python, por exemplo. Estas duas últimas, quando comparadas entre si, apresentam uma diferença estrutural de linguagem muito menor do que quando comparadas ao Matlab ou Octave. A classificação *open source* do R e Python, aliada às suas características e longo desenvolvimento como *softwares* para análises de dados, entrega maior variedade e poder para execução de testes estatísticos. Em contrapartida, *softwares* como Matlab e Octave se destacam por entregar ambientes de visualização dos dados mais desenvolvidos e amigáveis, oferecendo ferramentas práticas e robustas para projetar e plotar os conjuntos de dados [12].

As linguagens computacionais mais comuns são, em sua maioria, definidas dentre duas formas de programação principais, sendo estas: Programação orientada a objetos (POO) e programação estruturada. Na programação orientada a objetos, com intuito de facilitar a definição das componentes do código, são definidas classes dentro desta programação. A orientação a objetos busca relacionar as classes de objetos dentro da programação com classificações semelhantes àquelas do mundo real, tornando as mais próximas da nossa realidade e facilitando a estruturação e classificação [13].

A programação estruturada por sua vez, possui como característica particular a sua divisão em três tipos básicos de estrutura de execução que se interligam, sendo estas: sequências, condições e repetições (loops). Define-se cada uma delas da seguinte forma:

- Sequências: Compostas pelos comandos que serão processados pelo software.
- Condições: Comandos que somente são processados se dada condicional imposta se verificar ou não.
- Repetições: Comandos que são processados contínua e repetidamente até que uma dada condicional imposta se verifique.

A tabela 3 demonstra as principais diferenças entre a programação orientada a objetos e a programação estruturada.

Programação orientada a objetos	Programação estruturada
Métodos	Procedimentos e funções
Instâncias de variáveis	Variáveis
Mensagens	Chamadas a procedimentos e funções
Classes	Tipo de dados definidos pelo usuário
Herança	Não disponível
Polimorfismo	Não disponível

Tabela 3 – Comparativo entre programação orientada a objetos e programação estruturada. Obtida do portal Devmedia

3.4 – Linguagem R

O software R é uma das principais ferramentas para Data Science, trata-se de uma linguagem aberta e gratuita, capaz de resolver problemas de cálculos computacionais matemáticos e de estatística, assim como a plotagem de gráficos, fornecendo um conjunto de funcionalidades inovadoras para análise de dados. Por se tratar de um sistema *open source* (sistema aberto que possibilita que desenvolvedores externos criem novas funcionalidades, pacotes e atualizações), a linguagem possui uma gama de adeptos, uma vez que tal formato permite que continuamente sejam lançados novos pacotes, novas funções e ferramentas que facilitam ou ampliam o uso e aplicação do sistema [8].

O Rstudio é um ambiente de desenvolvimento integrado do R, sendo a interface de uso recomendada para a linguagem R. O Rstudio utiliza a linguagem R para seu funcionamento, possuindo à sua disposição todos os quase 20 mil pacotes publicados no CRAN (*The comprehensive R archive network*). O Rstudio facilita o uso do R e o torna mais amigável para trabalhar, assim como torna mais intuitiva a preparação dos códigos, isto porque apresenta uma visualização muito mais limpa e organizada, principalmente para gerar relatórios e textos para publicação externa. A interface Rstudio funciona em linguagem C^{++} e usa o framework QT como interface gráfica [14].

Diferentemente do R comum, que apresenta apenas o console de comando na visualização, quando o usuário acessar o Rstudio se deparará com a tela inicial do ambiente que é dividida em três ou quatro painéis (a depender da preferência do usuário, como será detalhado a seguir), cada um destes possui uma finalidade diferente, sendo também possível distribui-los de acordo com a preferência, assim como, é também possível mudar a quantidade de subjanelas por painel. A

descrição a seguir se refere à forma padrão do programa, os quatro painéis são [8]:

1 – *Source*: Neste painel o usuário pode escrever livremente seu código, podendo estruturá-lo de formas mais amigáveis à leitura, principalmente por possibilitar pular linhas sem necessariamente executar algum comando, assim como possibilita comentar o código livremente. Essa funcionalidade simplifica o compartilhamento de um código para execução de terceiros.

2 – *Console*: Neste painel o usuário também pode escrever seus comandos e gerar seu código, fica a critério de preferência do mesmo escolher se prefere trabalhar diretamente no console ou utilizar o *Source*. Também é no console que são gerados os *outputs* dos comandos, em outras palavras, se o usuário escreve e executa um comando, é gerada uma nova linha no console com o resultado (*output*) do comando. O console é a central de comandos do R e a mais recorrente ferramenta do programa.

3 – *Enviroment* e *history*:

- Na aba *enviroment* são lançados os dados estruturados gerados durante a sessão. Por exemplo, se o usuário importar um arquivo CSV (*comma separated values*) e nomear o dado gerado de “EX”, aparecerá nesta aba informações relativas a EX, como sua quantidade de entradas, linhas, colunas. Fica disponível também uma opção para visualizar este conjunto em uma nova tela (a visualização é disponibilizada para estruturas de data frame).
- Na aba *history* podemos consultar todo histórico de comandos utilizados.

4 – *Files*, *plots*, *packages*, *help* e *viewer*:

- Na aba *files* encontra-se arquivos recorrentes de uso e acessos rápidos aos diretórios.
- Na aba *plots* são gerados os gráficos obtidos através dos comandos do R.
- Na aba *packages* são apresentados todos pacotes já instalados no R, também é indicado quais estão ativados naquela sessão.

- Na aba *help* são gerados os tutoriais de cada pacote ou função do R. Quando o usuário escreve no console alguma função com uma interrogação precedendo-a, ele está solicitando que seja demonstrado o tutorial daquela função (exemplo: ?read.table).
- Na aba *viewer* é disponibilizado conteúdos provenientes da internet.

A versatilidade e atualização constante do software R, aliadas à gratuidade do serviço, são possivelmente seus maiores atrativos para profissionais do *Data Science*. Através do programa R, o pesquisador é capaz de realizar todas as etapas envolvidas em um roteiro de análise de dados, desde a importação e leitura de arquivos externos em diferentes formatos, manipular e modelar os dados para que apresentem uma estrutura adequada para análise, seguindo para obtenção dos resultados dos testes estatísticos e, por fim, plotar gráficos personalizados com diversas finalidades [8].

A versão base do Rstudio possui prontamente diversas funções básicas para solucionar tarefas como: Análises de estatística descritiva e inferencial, cálculos matemáticos e manipulações de matrizes, *arrays* (conjunto de matrizes) e *dataframes* (semelhante a matrizes, porém não se limitam a dados numéricos), plotagem de gráficos e outros. Entretanto, conhecer e utilizar os pacotes disponíveis aumentam o volume de funcionalidades, ou até mesmo facilitam e aprimoram aquelas já existentes. Dentre os diversos pacotes disponíveis, pode-se citar: *Tidyverse*, *GGPLOT2*, *DPLYR*, *MarkDown*, *forcats*, *shiny* e muitos outros.

3.5 – Python

O Pyhton, assim como o R, também é uma linguagem aberta e gratuita, que possui grande similaridade na capacidade de solucionar problemas estatísticos e matemáticos. Contudo, diferentemente da programação em R, o Python se faz amplamente presente em sistemas de automatização e formação de *scripts*, sendo utilizado por vários ramos da engenharia de *software*, como na produção de *frameworks* que formam *websites*. Essa amplitude de aplicação do Python torna comum que as equipes de desenvolvimentos de *software* de empresas possuam certo conhecimento prévio desta linguagem, facilitando uma implantação de uso para análise de dados [15].

A ampla e desenvolvida biblioteca de pacotes do Python, aliadas à sua grande facilidade de interfaceamento e comunicação com diversos outros *softwares* e linguagens, tais quais o R, SQL, C^{++}, Julia e outros, o torna de grande interesse para aplicações de *Data Science*. Algumas destas bibliotecas possuem grande arsenal de funções estatísticas e matemáticas para análises experimentais, como serão apresentadas mais adiante.

Para fazer uso da programação em Python, o usuário precisará fazer o *download* de um distribuidor de linguagens. O distribuidor de linguagens mais recomendado para análise de dados é, possivelmente, o ANACONDA. Um distribuidor de linguagens é uma central de interfaceamento que permite ao usuário acessar uma série de ferramentas e *softwares*, por exemplo, ao acessar o ANACONDA o programador encontrará acesso tanto ao Python quanto ao R, não só isso, como o distribuidor trará diversas interfaces de uso destas linguagens, pacotes e bibliotecas de uso variado. Dentre elas, tratando-se das aplicações para análise de dados em Python, destaca-se a interface com o *Jupyter Notebook*, e as bibliotecas *pandas*, *seaborn*, *matplotlib* e *numpy* [16].

Escrever um código extenso para análise exploratória de dados é um processo complexo, que requer muita atenção e inserção de uma série de comentários ao longo do processo, para torná-lo compreensível para leitura e facilitar seu compartilhamento. A ideia de um notebook para programação é de justamente tornar esse processo mais simples e fluido, fragmentando o código e possibilitando que este seja escrito com uma estrutura lógica e dividida em partes, que transcreva uma noção de início, meio e fim para funcionalidade do código. *Notebooks* como o Jupyter entregam a possibilidade de criar blocos de código, cada bloco apresenta uma função específica, e quando os blocos atuam sincronamente, entregam uma funcionalidade conjunta [16].

O Jupyter entrega as funcionalidades de um notebook com uma visualização simples e intuitiva, através deste é possível modelar e organizar conjuntos de dados, analisar e plotar gráficos personalizados. A vantagem do uso de um sistema *notebook* é a possibilidade de armazenar os resultados de cada etapa da análise exploratória, realizada em cada bloco de código, não sendo necessário realizar o processamento do código todo a cada vez que for realizar

um projeto. Essas funcionalidades, como citado anteriormente, estão atreladas ao uso das bibliotecas disponíveis para Python. A seguir encontra-se uma breve descrição das principais bibliotecas para *Data Science* [15].

- *Pandas*: O *pandas* é uma biblioteca licenciada para uso em Python e de código aberto, sua ampla quantidade de funções são em sua maioria voltadas para análises estatísticas e exploratórias de dados.
- *Seaborn*: O *seaborn* é uma biblioteca de visualização de dados em Python, suas aplicações são voltadas para elaboração de gráficos com alto potencial informativo.
- *Matplotbit*: O *matplotbit*, tal qual o *seaborn*, é uma biblioteca de visualização de dados, seu conjunto de funções é voltado para a produção de gráficos 2D. Suas plotagens podem compor scripts Python para distribuição em diversas aplicações em Python e alimentar servidores web.
- *Numpy*: O *numpy* é uma biblioteca voltada para o processamento de dados em larga escala, que apresentam arranjos multidimensionais e de grande tamanho. Suas funções são aplicadas para análise estatística e matemática dos dados.

3.6 – C++

A programação em C++ é um exemplo de linguagem de programação orientada ao objeto (POO), ela sucedeu a versão anterior C. Por se tratar de uma POO, a linguagem C++ possui grande potencial e agilidade para que sejam escritos códigos e programas em grande escala. O C++ é amplamente presente em diversos *softwares* gráficos e controles de *hardware*. O fato de ser uma POO, a torna uma grande candidata para lidar com problemas de *Big Data*, principalmente por ser a única programação capaz de trabalhar com dados de mais de um gigabyte em um segundo [17].

A linguagem C++ acaba recebendo pouca atenção como ferramenta para *Data Science* quando comparada ao Python e R, principalmente porque requer maior perícia em programação dos usuários. Contudo, sua aplicação está longe de ser inferior, pelo contrário, por conta de seu enorme potencial de processamento é capaz de entregar resultados muito mais rapidamente, e até

mesmo realizar cálculos em conjuntos de dados maiores que aqueles que as demais linguagens suportariam. Seu grande potencial pode ser explicado por se tratar de uma programação que atua de maneira muito próxima ao trabalho interno dos *hardwares*, manipulando diretamente os *drivers* dos computadores [18].

Para fazer uso da linguagem C++ o usuário deve utilizar um ambiente de desenvolvimento (união entre um editor e um compilador), existem diversas opções de ambientes disponíveis e cabe ao programador definir aquele de preferência. Os editores são as interfaces nas quais são escritos os códigos da linguagem C. Compiladores são *softwares* que interpretam a linguagem dos códigos e as transformam na linguagem binária (*bytes*), para que sejam entregues ao computador os comandos e, sequencialmente, executados [18].

Diferentes ambientes de desenvolvimento podem apresentar diferenças no uso e visualização da programação em C++, dentre as diversas opções disponíveis, pode-se citar duas que possivelmente estejam dentre as mais conhecidas, a *CodeBlocks* e *DevC++*. Ao acessar estes programas o usuário se deparará com uma tela inicial com diversas opções de atalhos no rodapé superior, ao acessar a função arquivo e então novo arquivo, ficará disponível um quadro branco no qual são escritos os códigos de comando.

3.7 – Matlab

Matlab (sigla para *Matrix Laboratory*) é um *software* desenvolvido pela Mathworks para realização de cálculos matemáticos baseados em matrizes, sendo um sistema pago e *closed source*. O Matlab se destaca como ferramenta por ser um *software* de alto desempenho e pelo seu formato de escrita requerer um nível amigável de abstração, sendo uma linguagem de programação nas quais os *scripts* se assemelham muito à forma a qual se escreve uma expressão algébrica [19].

A linguagem utilizada no Matlab é do tipo interpretada, tornando-a de rápido aprendizado por conta de sua estruturação próxima à uma escrita algébrica. O *software* conta com uma ampla quantidade de funções pré-configuradas, que se encontram agrupadas para uso dentro de *toolboxes*. *Toolboxes* são pacotes de funções ou classes, que entregam ao usuário funcionalidades e ferramentas

dentro do programa, as *toolboxes* ficam disponíveis no *software* agrupadas de acordo com a finalidade que possuem [20].

Ao acessar o Matlab, o usuário se deparará com a interface inicial, que de forma padrão, exibe a *command window*, *current folder*, *workspace* e *command history*. Na *command window* o usuário insere seus comandos ou *scripts* e visualiza os resultados (*outputs*), na aba *current folder* o programa exibe o diretório que está atrelada a sessão de uso e os arquivos disponíveis neste diretório, na aba *workspace* o programa exibe informações estruturais sobre o conjunto de dados inserido, por fim, na *command history* estará disponível o histórico e status dos comandos. Existe ainda o *editor*, aba na qual o usuário pode escrever um código livremente, ou realizar alterações em arquivos de programa Matlab (arquivos .m), é no editor que o usuário gera os *scripts* para depois executar na *command window* de uma única vez.

A capacidade e alto desempenho do Matlab em solucionar cálculos computacionais, tornam-no um *software* de grande valia para lidar com processos de *Data Science*. Existem uma série de *toolboxes* que contém funções e códigos para análise de dados, tais quais o MEDA (*Multivariate Exploratory Data Analysis*). Contudo, o grande diferencial do Matlab para *Data Science*, quando comparado a outros softwares, como o Python ou R, está em seu elevado potencial no processamento de imagem e visualização gráfica. O Matlab apresenta diversas funcionalidades na GUI (*graphical user interface*) do sistema, que potencializam a obtenção de gráficos e visualização dos dados, amparando o usuário na detecção de padrões e anomalias dentro dos *datasets.* O formato no qual os dados e gráficos são mostrados no Matlab torna a compreensão e controle do processo de análise de dados uma tarefa mais simples e intuitiva [19].

3.8 – Octave

O Octave é um *software* gratuito e *open source*, desenvolvido e baseado em uma linguagem de programação interpretada e algebricamente orientada, cuja a finalidade é resolver problemas numéricos. Por contar com um código aberto existem diversos pacotes e sistemas *toolboxes* online (sistemas semelhantes aos *toolboxes* do Matlab), que agregam uma ampla gama de funcionalidades ao *software.* A programação, utilização e resolução de problemas do Octave é muito

correlata àquela encontrada no MatLab, apresentando uma estrutura símil de linguagem algébrica para desenvolvimento dos códigos. Esta semelhança torna a linguagem Octave compatível com o Matlab, assim como torna os *sricpts* de arquivos .m interpretáveis no Octave [21].

Por padrão a GUI do Octave é apresentada com 5 paineis principais, a *command window*, o *file browser*, o *workspace*, o *command history* e o *variable editor*. A *command window* é a aba *core* do sistema, na qual o usuário gerará seus comandos e *scripts*. No *file browser* o usuário é capaz de visualizar o diretório em utilização durante a sessão, assim como poderá acessar os arquivos deste diretório. O *workspace* indica quais as unidades estruturais de dados existentes durante a sessão, trazendo ainda informações relativas à classificação e tamanho das unidades. O *command history* permite visualizar quais os últimos comandos executados e o status de execução. Por fim, o *variable editor* permite que o usuário faça mudanças nas unidades estruturais criadas e disponíveis no *workspace*, ao acessar uma das variáveis presentes no *workspace*, a aba *editor* disponibilizará um *template* amigável para que sejam alteradas as variáveis, de uma forma muito ágil e intuitiva.

O Octave atua com uma linguagem interpretada que classifica a estrutura dos dados em unidades algébricas como vetores, matrizes e semelhantes. O usuário é capaz inserir seus *datasets*, seja através da inserção manual com o diretório do arquivo, ou com a função load(filename.m). O programa interpretará o arquivo inserido e o demonstrará na aba *workspace*, possibilitando que o usuário faça as análises necessárias dentro do *dataset*. Existe uma série de funções e modelos gráficos padrões do sistema, contudo a ampla biblioteca *online* disponível é capaz de expandir significativamente as possibilidades de análise dos dados e modelos para plotagem de gráficos [22].

A visualização de modelos gráficos é uma tarefa pouco complexa no Octave, entregando soluções ricas em modelos e informações. Esta característica da programação do *software* está intimamente ligada ao seu desenvolvimento, uma vez que a sintaxe matematicamente orientada do programa foi criada baseada em ferramentas de plotagem e visualização de modelos 2D e 3D [21].

3.9 – SQL

SQL, ou linguagem de consulta estruturada (*structured query language*), é uma linguagem de programação utilizada para interagir com um banco de dados. Através das *querys* (termo técnico utilizado para se referir as estruturas de linguagens que realizam as consultas) o usuário, ou um *software* terceiro, é capaz de consultar, filtrar, editar, inserir ou remover informações de um banco de dados estruturado [23].

A linguagem SQL é de grande importância para um cientista dos dados. É através dela que o usuário acessará o banco de dados centralizado, a utilização de uma *query* é capaz de retornar um conjunto de dados com filtros e parâmetros determinados pelo programador. A obtenção assertiva de um conjunto de dados proveniente da central de dados é uma etapa determinante para um processo conclusivo de *Data Science* ou *Data Mining*.

Existem uma série de programas utilizados para realização do gerenciamento de um banco de dados, dentre os quais se destacam: *MySQL*, *Oracle*, *SQL server* e *PostgreSQL*. A linguagem SQL pode ser aplicada a todos esses *softwares*, contudo, existem diferenças estruturais da construção das *querys* para atuar com cada um destes gerenciadores, tais diferenças se tornam mais ou menos acentuadas a depender dos fornecedores que se compara. A escolha do gerenciador está intrínseca ao tipo de central de banco de dados utilizado pela empresa ou instituição de pesquisa [23].

O uso de SQL para extração de dados está intimamente relacionado com as mais diversas aplicações do *Data Science*. Para realizar uma análise de dados e alimentar os *softwares* para a análise, como R ou Python, o programador precisará muitas vezes saber aplicar uma *query* precisa para retornar as informações necessárias para seu estudo. Para realizar um estudo de processo KDD e aplicar os conceitos de *Data Mining*, o programador também precisará ser hábil para trazer o conjunto de dados específico que contenha as informações exatas dado os filtros aplicados na *query* [10].

3.10 – Ferramentas do *Data Science* – Convergências e divergências

A tabela 4 ilustra e sumariza as funcionalidades, características e principais vantagens dos diferentes *softwares* e linguagens de programação abordados:

Linguagem /*Software*	Funcionalidade	Gratuito/*Open-source*	Vantagem
R	▪ Importar, estruturar e modelar conjunto de dados. ▪ Ampla gama de funcionalidades para análise de dados e plotagem de gráficos.	Sim/Sim	▪ Imenso arsenal de funções para análise de dados ▪ Grande quantidade de pacotes e fóruns de usuários disponíveis.
Python	▪ Importar, estruturar e modelar conjunto de dados. ▪ Ampla gama de funcionalidades para análise de dados e plotagem de gráficos.	Sim/Sim	▪ Imenso arsenal de funções para análise de dados ▪ Grande quantidade de pacotes e fóruns de usuários disponíveis.
C^{++}	▪ Importar, estruturar e modelar conjunto de dados. ▪ Grande gama de funcionalidades para análise de dados e plotagem de gráficos.	Sim/Sim	▪ Imenso potencial de processamento de dados. Capacidade de processar dados a mais de 1Gb/s
Matlab	▪ Importar, estruturar e modelar conjunto de dados. ▪ Grande gama de funcionalidades para análise de dados. ▪ Ampla gama ferramentas para plotagem de gráficos.	Não/Não	▪ Imensa gama de opções para plotagem de gráficos de forma facilitada.
Octave	▪ Importar, estruturar e modelar conjunto de dados. ▪ Grande gama de funcionalidades para análise de dados. ▪ Ampla gama ferramentas para plotagem de gráficos.	Sim/Sim	▪ Imensa gama de opções para plotagem de gráficos de forma facilitada.
SQL	▪ Realizar consultas ao banco de dados. Permite extrair, inserir, deletar e alterar informações do banco.	Sim/Não	▪ Facilidade em se comunicar com os bancos de dados.

Tabela 4 – Comparação direta entre os *softwares* e linguagens.

3.11 – Ferramentas da qualidade

A garantia e controle da qualidade em indústrias são fundamentos de grande responsabilidade e atenção. Os processos que garantem e supervisionam a conformidade dos produtos, que serão distribuídos e comercializados pelas empresas, são etapas que requerem grande cautela e rigidez. Indústrias químicas, alimentícias e farmacêuticas se situam dentre aquelas que requerem maior seriedade e rigorosas regras de aprovação, isso porque são produtoras de bens consumíveis. Alterações não percebidas em lotes de produtos podem causar problemas muito maiores que a perda de confiança dos clientes. Não obstante, estes setores contam com diversos órgãos regulamentadores como a ANVISA (Agência nacional de vigilância sanitária) ou INMETRO (Instituto nacional de metrologia, qualidade e tecnologia) [24].

Diversos experimentos são realizados em várias indústrias, de uma ampla gama de segmentos, com intuito de melhorar as características e a qualidade dos produtos, otimizar os processos, reduzir tempo de produção, diminuir o número de testes necessários e por fim, minimizar o uso de recursos das empresas. Estes fundamentos do planejamento experimental são muito similares aqueles das ferramentas da qualidade, pois profissionais da qualidade são muitas vezes consultados para que realizem ensaios experimentais para obter informações dos processos e produtos de sua instituição [4]. Os processos de planejamento experimental levam à obtenção de dados, que inevitavelmente precisaram ser analisados para trazer as informações necessárias para os processos de tomada de decisão (SAD). Essa conjuntura de técnicas torna de grande valia para empresas poder contar com um profissional que saiba aplicar as ferramentas da qualidade, realizando o planejamento experimental adequado, e que por fim, saiba extrair as informações e conclusões assertivas dos dados coletados através de um estudo de *Data Science*.

Ao longo de um projeto de implantação de melhorias em uma indústria ou instituição de pesquisa, o *Data Science* como citado pode amparar os estudos realizados em planejamentos experimentais. Porém, o uso dos dados não se limita à essa aplicação, durante um processo de melhorias é de grande importância que o profissional saiba analisar os dados de produção de períodos anteriores e

conheça como extrair destes *insights* (processo KDD). Além disso, o profissional deve ter conhecimento sobre como aplicar técnicas de estatística em processos de CEP (controle estatístico de processos), diagrama Ishikawa e Pareto, entre outras finalidades.

A qualidade é um fundamento central para empresas que querem se destacar, tanto visando receber certificações, quanto para captar e fidelizar clientes [24]. Pensando nisso, existem diversos métodos e práticas que intentam ensinar e capacitar profissionais de diversas áreas, para que possam desenvolver projetos com o objetivo de comandar sua equipe para aumentar a conformidade dos produtos, diminuir o desperdício e encontrar os pontos chaves para padronizar e aperfeiçoar práticas e técnicas produtivas. Dentre os principais métodos e os mais requisitados nas indústrias, encontram-se o Lean 6σ e o 5s.

Diversos fundamentos do *6σ* e do 5s são correlatos às técnicas de *Data Science*, tais semelhanças de método ficarão mais claras quando as práticas dessas ferramentas forem descritas. Entretanto, vale enfatizar dois paradigmas existentes em um projeto de *Data Science* que muito se assemelham àqueles das ferramentas da qualidade [5]:

1. Paradigma dirigido por hipótese: Conhecido um problema, qual tipo de dados precisa-se para apoiar sua solução?
2. Paradigma dirigido por dados: Obtido um conjunto de dados, quais problemas podem ser encontrados ou solucionados?

3.12 – Lean 6Sigma (6σ)

O 6σ foi desenvolvido pela Motorola no ano de 1980, fundamentado pelo método DMAIC (definir, medir, analisar, melhorar e controlar). Seu desenvolvimento tem como base criar estratégias capazes de otimizar processos, atuando no aperfeiçoamento da produção, levando à redução de rejeição de produtos por não conformidade, e diminuindo a variabilidade do processo. O método tem a finalidade de que os produtos não somente apresentem uma alta qualidade, mas que esta também esteja presente em toda a produção, reduzindo a probabilidade de produção de lotes ou peças defeituosas. Esse método traz consigo o ideal de alinhar toda cadeia produtiva na finalidade de agradar o cliente final, de forma que esta cadeia seja capaz de atuar na eliminação das causas de

erros internos, assim como no aumento da confiabilidade e fidelização de seus clientes [25].

O σ (sigma) do método se refere ao desvio padrão (medida da dispersão dos dados em torno da média), sendo o propósito do método alcançar o 6σ, nível no qual não há variação dentro dos produtos e os processos seguem sem falhas. O gráfico da Figura 6 indica que 68% dos dados coletados em um processo estarão a uma distância de ±1σ da média, que 95% das observações estarão a uma distância de ±2σ da média e que 99,9% dos dados estarão a uma distância de ±3σ da média. Em uma aplicação 6σ a proposta é contrária, na qual 99,9% dos dados coletados devem estar na região 6σ (de -3 a 3), e que somente um percentual de 0,1% de erros ocorram [26].

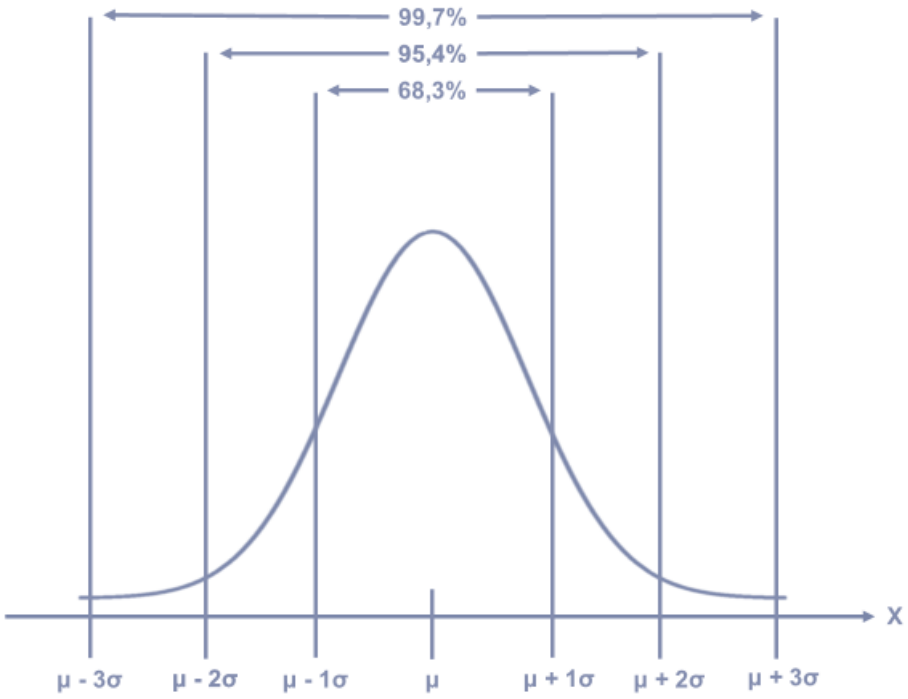

Figura 6 – Distribuição de dispersão 6σ

O método se inicia com a percepção e capacidade de encontrar as falhas, encontrado um problema o projetista passa para as etapas DMAIC, como descritas na Tabela 4.

Etapa	Procedimento	Metodologia
Definir – Encontrar falhas procedurais, elaborar uma solução inicial e montar a equipe de desenvolvimento	▪ Tomando como base os bancos de dados, identificar um problema ou falha do processo. ▪ Definir a equipe responsável.	▪ SIPOC (fornecedor, entradas, processo, saídas e clientes) ▪ *Project Charter* ▪ Crono análise
Medir – Mensurar gravidade e impactos gerados pelas falhas	▪ Avaliar e mensurar a gravidade do problema. ▪ Desenhar um fluxograma do processo de estudo (conhecer todas etapas) ▪ Coletar dados do processo como resultados, gastos, horas dos colaboradores e outros	▪ Diagrama causa e efeito ▪ Fluxograma do processo
Analisar – Estudar todas variáveis e possibilidade da implantação	▪ Baseado nos dados coletados definir quais pontos de dor e defeituosos do processo, definir quais causadores dos defeitos ▪ Estudar o problema e a solução proposta para verificar a viabilidade do projeto.	▪ Diagrama Ishikawa (Espinha de peixe) ▪ 5 porquês? ▪ Diagrama de Pareto ▪ Matriz de priorização ▪ *Brainstorm*
Melhorar – Implantar as soluções projetadas	▪ Estipular projeto ou mecanismo para eliminar ou reduzir a ação dos pontos defeituosos ▪ Implantar os novos mecanismos e parâmetros pré-estabelecidos	▪ Gráficos para acompanhamento (Box-plot, dispersão e outros) ▪ Estratificação
Controle – Garantir bom funcionamento da solução e acompanhar os ganhos reais	▪ Manutenção contínua das soluções implantadas ▪ Verificar constantemente os resultados da operação	▪ POP (procedimento operacional padrão) ▪ CEP (controle estatístico do processo) ▪ Cartas controle

Tabela 5 – Descritivo das etapas DMAIC

O método *Lean manufacturing* é baseado no sistema Toyota de produção, desenvolvido entre os anos 1974 e 1975. A sua finalidade é evitar ao máximo qualquer desperdício, seja este desperdício no uso de recursos, materiais (equipamentos, reagentes, entre outros), horas de empenho do colaborador, processos desnecessários, tempo, recursos financeiros e outros. O Lean tem como proposta analisar todos fluxos e operações de uma empresa, analisando os pontos de falha ou desnecessários, que acabam por diminuir o retorno financeiro e a razão produtividade por tempo. Tal qual o método 6σ, este processo visa prioritariamente gerar a maior satisfação possível ao seu cliente, entregando produtos com a melhor relação preço e qualidade [25].

O Lean 6σ compreende, como o nome sugere, a união e aperfeiçoamento dos métodos Lean e 6σ, fornecendo as ferramentas de diminuição de erros, otimização e diminuição da variabilidade dos processos, unificação de fluxos, baseados no aumento de produtividade do método DMAIC do 6σ. Aliadas às ferramentas da diminuição dos desperdícios, eliminação dos fluxos desnecessários e também aumento da produtividade do *Lean manufacturing*. Essa união de métodos é pensada para atuar em todos setores de uma empresa em busca de falhas, desperdícios, processos improdutivos, entregando as ferramentas necessárias para correção, unificação e implantação de um método ótimo de produção, ou quando possível, eliminando processos [25].

3.13 – *Lean Manufacturing*: Método 5S

Uma das principais técnicas do *Lean manufacturing* é o método 5S, desenvolvido pelo sistema Toyota de produção (TPS), o método 5S é a base para as empresas que buscam atingir a gestão da qualidade total (GQT). O objetivo central do método é de concentrar os esforços de todos colaboradores da equipe em organizar o ambiente, mantendo dentro da empresa somente os processos, instrumentos, maquinários e esforços necessários. Por conta dessa ênfase, é habitual a correlação de que o 5S seja um sistema enviesado em uma grande limpeza (comumente dito como *Housekeeping* no meio corporativo). Entretanto, essa visão física do processo pode deixar grande parte dos valores da técnica passarem desapercebidos, o método 5S vai além de uma interação manual dos

colaboradores, ela representa um conjunto de valores e uma cultura de atuação dentro do ambiente de trabalho [27].

O 5S atua em três diferentes frentes, o *layout* (espaço físico da empresa), o intelectual (compromisso em realizar as tarefas) e o social (conjunto de valores dos relacionamentos e comportamentos cotidianos). A mudança do *layout* é, provavelmente, a mais intuitiva, é logico pensar que modificar o ambiente físico, se desfazer de objetos inúteis e trabalhar para alcançar um ambiente fluído e organizado, deva levar a alcançar ganhos produtivos. As mudanças do intelectual e social são menos intuitivas que o *layout*, tanto em sua assimilação quanto em sua aplicação. Principalmente por que dependem de uma série de mudanças de base e hábitos, sendo necessário um trabalho conjunto de todos os profissionais da instituição. Os funcionários de alta hierarquia possuem especial papel neste processo, uma vez que sua adoção aos novos conceitos culturais da empresa, deve servir de exemplo para os demais, gerando um sistema de reestruturação *top-down* [27].

O 5S é fundamentado em seguir 5 conceitos, derivados de 5 palavras japonesas iniciadas com a letra S, o que origina o nome do método, os 5S são:

1. *SEIRI* – Senso de utilização, arrumação, organização, seleção;

   Consiste em separar as ferramentas, instrumentos e processos úteis daqueles que são inúteis ou pouco proveitosos, eliminando aqueles classificados no segundo grupo.

2. *SEITON* – Senso de ordenação, sistematização, classificação;

   Consiste em classificar e organizar todos componentes que foram mantidos após o primeiro passo, visando os alocar de uma maneira intuitiva e conhecida, facilitando a busca por todos os itens necessários.

3. *SEISO* – Senso de limpeza, zelo;

   Esta etapa visa a manutenção da limpeza e organização constante dentro do espaço de uso comum. Um ambiente limpo e organizado torna mais fácil a percepção de falhas, assim como torna o local mais seguro para utilização.

4. *SEIKETSU* – Senso de asseio, higiene, saúde, integridade;

Esta etapa consiste em orientar e assegurar que todos os colaboradores se empenhem em seguir os passos anteriores, para que sejam alcançadas as metas estabelecidas de limpeza e organização. Neste passo é interessante distribuir tarefas específicas e determinar metas coletivas, para que o procedimento se torne um padrão difundido e comportamental para todos os colaboradores.

5. *SHITSUKE* – Senso de autodisciplina, educação, compromisso.

A etapa da disciplina é um reforço para todas as anteriores, entretanto, isto não a torna menos importante ou relevante, pelo contrário, esta é uma etapa determinante para permanentemente difundir os princípios da organização. O procedimento consiste em criar compromissos, metas e padrões de processos dentro da equipe, assim como orientar todos profissionais, em especial os mais experientes, para que continuem seguindo os passos anteriores, servindo de exemplo para todos os demais e os novos colaboradores, para que tenham consigo esta mentalidade desde início.

4 – Considerações finais

Como evidenciado ao longo do que fora apresentado, o aumento da cadeia produtiva e a obrigatoriedade de certificação dos produtos, a necessidade de buscar meios de otimizar processos e reduzir uso de recursos e gastos, o crescente acúmulo de dados que possuem grande potencial informacional, entre outros diversos fatores da indústria 4.0, tornam imprescindível que exista uma demanda por profissionais alinhados com estas novas vertentes do mercado.

Dessa forma, o estudo do *Data Science* se mostra uma interessante opção e caminho para pesquisadores e cientistas, seja para buscar novas oportunidades ou se manter presente e atualizado no mercado de trabalho. Sobretudo, tendo em vista a amplitude de aplicações desta área, atuando desde a obtenção de informações e conhecimento através do banco de dados, apoiando projetos de tomada de decisão, obtenção de resultados experimentais, projetos de controle da qualidade de processos e experimentos, e muitas outras.

5 – Referencias

[1] Disponível em: https://www.labware.com/pt/about. Acesso em 20 jul. 2022

[2] MONTGOMERY, Douglas C. **Design and Analysis of Experiments**. Arizona, Usa: Ninth Edition, 2012.

[3] LEARDI, Riccardo. **Experimental design in chemistry: A tutorial**. Analytica Chimica Acta, 2009, 652, 161-172.

[4] GALDAMEZ, Edwin V. Cardoza; CARPINETTI, Luiz C. Ribeiro. **Aplicação das técnicas de planejamento e análise de experimentos no processo de injeção plástica**. Gestão & Produção, 2004, v11, nº1 121 - 134.

[5] Gressling, Thorsten. **Data Science in Chemistry**. 1st edn. De Gruyter, 2020.

[6] AMARAL, Fernando. **Introdução à Ciência de Dados**: mineração de dados e big data. Alta Books, 2018.

[7] Cavique, Luís - **Big data e data science**. Boletim da APDIO, 2014, Nº 51, p. 11-14.

[8] WICKHAM, Hadley; GROLEMUND, Garrett. **R para data science**: importe, arrume, transforme, visualize e modele dados. Alta Books, 2019. 528 p.

[9] GOLDSCHMIDT, Ronaldo. **Data Mining**: conceitos, técnicas, algoritmos, orientações e aplicações. 2. ed. Gen Ltc, 2021. 289 p.

[10] Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. ***AI Magazine*****.** *N°* 3 (1996), p. 38-54

[11] FARIA, Pedro Duarte; PARGA, João Pedro Figueira Amorim. **Introdução à Linguagem R**: seus fundamentos e sua prática. 3. ed. Independente, 2022. 686 p.

[12] DEVMEDIA. **Programação Orientada a Objetos e Programação Estruturada**. 2015. Disponível em: https://www.devmedia.com.br/programacao-orientada-a-objetos-e-programacao-estruturada/32813. Acesso em: 29 jun. 2022.

[13] COLLIAU, Taylor. ROGERS, Grace. HUGHES, Z. OZGUR, Ceyhun. "Matlab vs Python vs R", 2017, Business Faculty Publication, 51. Disponível

em:https://scholar.valpo.edu/cba_fac_pub/51?utm_source=scholar.valpo.edu%2Fcba_fac_pub%2F51&utm_medium=PDF&utm_campaign=PDFCoverPages

[14] ALCOFORADO, Luciane Ferreira. **Utilizando a linguagem R**: conceitos, manipulação, visualização, modelagem e elaboração de relatórios. Alta Books, 2021. 384 p.

[15] LOPES, Gesiel Rios; ALMEIDA, Alessandro Wilk Silva; DELBEM, Alexandre C. B.; TOLEDO, Cláudio Fabiano Motta. Introdução à Análise Exploratória de Dados com Python. In: ARAÚJO, Flávio Henrique Duarte de; CARVALHO, Franklhes Santos; VERAS, Jaclason Machado; BALUZ, Rodrigo Augusto Rocha Souza; VERAS, Rodrigo de Melo Souza; SILVA, Romuere Rodrigues Veloso e. **MINICURSOS ERCAS-PI E ENUCOMPI 2019**. Porto Alegre: Sociedade Brasileira de Computação – Sbc, 2019. p. 161-175. Disponível em: https://ercas2019.enucompi.com.br/doc/livro_de_minicursos_ercas_pi_2019.pdf. Acesso em: 22 jun. 2022.

[16] MIRANDA, João Vitor de. **Jupyter Notebook**: exemplos de códigos e como usar. Exemplos de Códigos e Como Usar. 2021. Disponível em: https://www.alura.com.br/artigos/conhecendo-o-jupyter-notebook. Acesso em: 22 jun. 2022.

[17] SANYAL, Sayantani. **HOW ARE C AND C++ PROGRAMS BECOMING HELPFUL IN DATA SCIENCE?** 2021. Disponível em: https://www.analyticsinsight.net/how-are-c-and-c-programs-becoming-helpful-in-data-science/. Acesso em: 22 jun. 2022.

[18] DAVIS, Stephen R. **C++ para leigos**. 7. ed. Alta Books, 2016. 472 p.

[19] SANTOS, Reginaldo J. Introdução ao Matlab. 2009. Disponível em: http://www.fmt.if.usp.br/~luisdias/MiniCurso/Apostilas/matlab_reginaldo.pdf

[20] MARTINEZ, Wendy L. MARTINEZ, Angel R. SOLKA, Jeffrey L. **Exploratory Data Analysis with MATLAB**, 3rd ed. CRC Press, 2017. 616 p.

[21] EATON, John W. GNU OCTAVE. 1998. Disponível em: octave.org. Acesso em: 09 jul. 2022

[22] GEBEL, Łukasz. **Octave — Scientific Programming Language Crash Course**. 2020. Disponível em: https://towardsdatascience.com/octave-scientific-programming-language-crash-course-2ab8d864a01d

[23] NIELD, Thomas. **Getting Started with SQL**: a hands-on approach for beginners. Asfsdf: O'Reilly Media, 2016. 134 p.

[24] BERTI, RC; SANTOS, DC. IMPORTÂNCIA DO CONTROLE DE QUALIDADE NA INDÚSTRIA ALIMENTÍCIA: PROVÁVEIS MEDIDAS PARA EVITAR CONTAMINAÇÃO POR RESÍDUOS DE LIMPEZA EM BEBIDA UHT. **Atas de Ciências da Saúde**, v. 1, n. 4, p. 23-38, jul. 2016. Disponível em: Importância do controle de qualidade na indústria alimentícia: prováveis medidas para evitar contaminação por resíduos de limpeza em bebida UHT | Atas de Ciências da Saúde (ISSN 2448-3753) (fmu.br). Acesso em: 23 jun. 2022.

[25] VENANZI, Délvio; LAPORTA, Bruna Pires. LEAN SIX SIGMA. **South American Development Society Journal**, 2017, v. 1, n. 2, p. 66 – 84. ISSN 2446-5763.

[26] PETENATE, Marcelo. Afinal, de onde vem os 3,4 PPMs do Six Sigma? Disponível em: https://www.escolaedti.com.br/de-onde-vem-os-34-ppms Acesso em: 28 jun. 2022.

[27] CAMPOS, Renato; OLIVEIRA, Luís Carlos Queiroz de; SILVESTRE, Bruno dos Santos; FERREIRA, Ailton da Silva. **A Ferramenta 5S e suas Implicações na Gestão da Qualidade Total**. 2005. 12 f.) - 2005.