

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

ANDRÉ ROMÃO NASCIMENTO

**Estudo de caso: clusterização hierárquica de rugosidade em superfícies
nanotexturizadas em implantes dentários**

São Carlos
2021

ANDRÉ ROMÃO NASCIMENTO

**Estudo de caso: clusterização hierárquica de rugosidade em superfícies
nanotexturizadas em implantes dentários**

Monografia apresentada ao Curso de Engenharia Mecânica, da Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Engenheiro Mecânico.

Orientadora: Profa. Dra. Luciana Montanari

São Carlos
2021

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da EESC/USP com os dados inseridos pelo(a) autor(a).

N244e Nascimento, André Romão
Estudo de caso: clusterização hierárquica de rugosidade em superfícies nanotexturizadas em implantes dentários / André Romão Nascimento; orientador Luciana Montanari. São Carlos, 2021.

Monografia (Graduação em Engenharia Mecânica) -- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2021.

1. Aprendizado de Máquina. 2. Agrupamento Hierárquico. 3. Análise de Componentes Principais. 4. Estruturas Periódicas Induzidas por Laser. I. Título.

FOLHA DE AVALIAÇÃO

Candidato: André Romão Nascimento

Título: Estudo de caso: clusterização hierárquica de rugosidade em superfícies nanotexturizadas em implantes dentários

Trabalho de Conclusão de Curso apresentado à
Escola de Engenharia de São Carlos da
Universidade de São Paulo
Curso de Engenharia Mecânica

BANCA EXAMINADORA

Profa. Dra. Luciana Montanari
(Orientadora)

Nota atribuída: 10 (dez)

Prof. Tit. Jaime Gilberto Duduch
Nota atribuída: 10 (dez)

Eng. Assoc. Alessandro Roger Rodrigues
Nota atribuída: 10 (dez)

Média: 10 (dez)

Resultado: aprovado

Data: 30/4/2021.

Este trabalho tem condições de ser hospedado no Portal Digital da Biblioteca da EESC

SIM x NÃO Visto da orientadora Luciana Montanari

Observação: Após correções indicadas pela banca.

DEDICATÓRIA

*À minha família pela compreensão,
carinho e apoio incansável.*

AGRADECIMENTOS

Aos meus pais, Cláudio e Marinalda, e à minha irmã Fernanda, pelo amor, carinho e apoio incondicionais que me deram forças durante a realização deste trabalho.

Ao Dr.-Ing. David Domingos pela oportunidade de estágio em um dos mais renomados institutos de desenvolvimento de tecnologia do mundo, o Instituto Fraunhofer IPK, em Berlim.

Ao M.Sc. Luiz Guilherme De Souza Schweitzer pela instrução, aconselhamento, compreensão e suporte durante o tempo de estágio e períodos subsequentes.

Ao M.Sc. Peter Schneider pela companhia e conversas durante o período de estágio.

Aos meus colegas de curso, pelo convívio, parceria nos desafios apresentados ao longo dos anos e amizade, que me permitiram crescer não somente enquanto pessoa, mas como profissional.

À professora Luciana Montanari pela atenção, disposição e orientação na elaboração do trabalho.

RESUMO

NASCIMENTO, A. R. Estudo de caso: clusterização hierárquica de rugosidade em superfícies nanotexturizadas em implantes dentários. 2021. 73 f. Monografia (Trabalho de Conclusão de Curso) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2021.

A proposta do trabalho é a aplicação de um algoritmo de aprendizado de máquina não-supervisionado, mais especificamente a clusterização hierárquica (também conhecido como agrupamento hierárquico), em medidas de rugosidade de corpos de prova nanotexturizados através da tecnologia LIPSS (estruturas periódicas induzidas por laser; no inglês, *Laser Induced Periodic Surface Structures*), usados no estudo de impacto na aderência bacteriana e criação de biofilme de implantes dentários. Tais processos biológicos dão início à colonização de bactérias e são a principal fonte de inflamações e infecções que podem levar a falência do implante. A estatística tradicional é o principal método utilizado para embasar as análises quantitativas e qualitativas da metodologia científica. Os algoritmos de aprendizado de máquina oferecem uma nova perspectiva de análise de dados. Enquanto métodos supervisionados buscam uma variável resposta, os não-supervisionados possuem o objetivo de identificar padrões nos dados de forma holística. A análise de componentes principais, um método de redução de dimensionalidade, torna possível a geração de gráficos de dispersão em duas e três dimensões que auxiliam a visualização e análise qualitativa dos dados. Os algoritmos foram implementados em linguagem Python. Conclui-se evidenciando a nova perspectiva que os métodos de aprendizado de máquina trazem ao problema exposto em comparação aos métodos estatísticos tradicionais e, de maneira geral, ressalta-se a importância do desenvolvimento e uso de ferramentas de programação para o engenheiro contemporâneo.

Palavras-chave: Aprendizado de Máquina, Agrupamento Hierárquico, Análise de Componentes Principais, Estruturas Periódicas Induzidas por Laser.

ABSTRACT

NASCIMENTO, A. R. Study case: hierarchical clustering of roughness in nanotextured surfaces in dental implants. 2021. 73 f. Monografia (Trabalho de Conclusão de Curso) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2021.

The projects' purpose is to implement a non-supervised machine learning algorithm, more specifically hierarchical clustering, on the roughness measures of test specimens nanotextured through laser induced periodic surface structures (LIPSS) technology, used to study the impact on bacterial adherence and biofilm formation on the abutment surface of dental implants. These biological processes trigger bacterial colonization and are the main source of inflammation and infection that can lead to the dental implant failure. Traditional statistics' quantitative and qualitative analysis are commonly used to support the scientific method. Machine Learning algorithms offer a new perspective in data analysis. Whereas supervised learning algorithms focus on a finding a response variable, the non-supervised techniques aim at holistically identifying patterns in the data. The principal component analysis, a dimensionality reduction method, enable the generation of scatterplots in two and three dimensions, which assists the visualization and qualitative assessment of data. The algorithms were implemented in Python programming language. This study emphasizes the importance of programming skills development for the modern engineer as an enabler of new perspectives in data analysis with machine learning algorithms and traditional statistics.

Keywords: Machine Learning, Hierarchical Clustering, Principal Component Analysis, Laser Induced Periodic Surface Structures.

LISTA DE ILUSTRAÇÕES

Figura 1 - Partes de um implante dental	21
Figura 2 - Lesões peri-implantares	24
Figura 3 - Tipos de inteligência artificial agrupados por processos	32
Figura 4 - Técnicas de aprendizado de máquina por categorias.....	33
Figura 5 - Exemplificação do conceito de componentes principais	35
Figura 6 - Histograma de R_a nas categorias [nm]	40
Figura 7 - Histogramas possíveis considerando 3 categorias e 6 parâmetros	41
Figura 8 - <i>Boxplots</i> de R_a e R_z nas categorias.....	42
Figura 9 - <i>Boxplots</i> de S_a e S_z nas categorias.....	44
Figura 10 - <i>Boxplots</i> de S_{sk} e S_{ku} nas categorias.....	45
Figura 11 - Percentual de variância explicada por componente principal.....	47
Figura 12 - Gráficos de dispersão das projeções das observações nos 3 primeiros componentes principais	48
Figura 13 - Influência de cada atributo nos componentes principais	49
Figura 14 - Dendrograma do <i>clustering</i> hierárquico com corte em $D = 9$	50
Figura 15 - Clusterização hierárquica - 4 <i>clusters</i>	51
Figura 16 - <i>Boxplots</i> dos 6 parâmetros nos <i>clusters</i> - 4 <i>clusters</i>	52
Figura 17 - Dendrograma do <i>clustering</i> hierárquico com corte em $D = 7,5$	53
Figura 18 - Clusterização hierárquica - 6 <i>clusters</i>	54
Figura 19 - <i>Boxplots</i> dos 6 parâmetros nos <i>clusters</i> - 6 <i>clusters</i>	55

LISTA DE TABELAS

Tabela 1 - Medidas estatísticas para a produção de <i>boxplot</i> de R_a e R_z	42
Tabela 2 - Medidas estatísticas para a produção de <i>boxplot</i> de S_a e S_z	43
Tabela 3 - Medidas estatísticas para a produção de <i>boxplot</i> de S_{sk} e S_{ku}	44
Tabela 4 - Resultado da normalização dos valores dos parâmetros	46
Tabela 5 - Projeção dos valores normalizados nos componentes principais.....	47
Tabela 6 - Influência de cada atributo nos componentes principais.....	49
Tabela 7 - Distribuição das observações nos <i>clusters</i> - 4 <i>clusters</i>	50
Tabela 8 - Distribuição das observações nos <i>clusters</i> - 6 <i>clusters</i>	54

SUMÁRIO

1 INTRODUÇÃO.....	20
1.1 Objetivos.....	20
2 REVISÃO BIBLIOGRÁFICA	21
2.1 Implantes Dentários no Contexto Biológico	21
2.1.1 Implantes Dentários.....	21
2.1.2 Peri-implantite	22
2.1.3 Aderência Bacteriana.....	25
2.2 Processos de Tratamento de Superfícies para Implantes.....	26
2.2.1 Procedimento atual	28
2.2.2 Laser – LIPSS	29
2.3 Análise de Dados	30
2.3.1 Estatística.....	31
2.3.2 Inteligência Artificial.....	32
2.3.3 Aprendizado de Máquina.....	32
2.3.4 Análise de Componentes Principais	34
2.3.5 Clusterização	35
3 MÉTODOS E FERRAMENTAS	37
3.1 Corpos de Prova.....	37
3.2 Sistemas de Laser	37
3.3 Topografia de superfície	37
3.4 Aplicação de Algoritmos	38
4 ANÁLISE DE RESULTADOS E DISCUSSÃO	39
4.1 Histograma.....	39
4.2 <i>Boxplot</i> ou Diagrama de Caixa.....	42
4.3 Análise dos Componentes Principais	45
4.4 Clusterização Hierárquica.....	49
5 CONCLUSÃO.....	57
REFERÊNCIAS	58
Apêndice A – Código Python em Jupyter Notebook.....	61

1 INTRODUÇÃO

O presente trabalho é um estudo de caso com o intuito de explorar as aplicações de métodos de aprendizado de máquinas em estruturas periódicas induzidas por laser para geração de nanoestruturas na superfície do pilar do dente de implantes dentários.

Historicamente os esforços tecnológicos focaram em fazer tal superfície o menos rugosa possível, com o objetivo de minimizar a aderência bacteriana. A nova tecnologia de laser permite explorar a topografia da superfície em uma escala nanométrica, criando-se estruturas organizadas.

Análises de rugosidade de superfícies que utilizam a estatística tradicional dão embasamento ao método científico. A proposta do trabalho é explorar uma nova perspectiva de análise de dados, comparando-se os métodos tradicionais frente à algoritmos de aprendizado de máquina, mais especificamente, clusterização hierárquica e análise de componentes principais.

1.1 Objetivos

Tradicionalmente são feitas análises estatísticas focadas em caracterizar a rugosidade de superfícies, correlacioná-las com os resultados biológicos e tirar conclusões robustamente embasadas, garantindo-se a replicabilidade de resultados. O desafio proposto é o estudo e aplicação de algoritmos de aprendizado de máquina, mais especificamente técnicas de *clustering*, com o objetivo de evidenciar as novas perspectivas de análise de dados possibilitadas pela redução de dimensionalidade dos atributos, que permite geração de visualizações em duas e três dimensões, e pela clusterização hierárquica, que agrupa os corpos de prova levando-se em consideração uma visão holística de seus valores mensurados.

Vale ressaltar que as técnicas de aprendizado de máquina são realizadas paralelamente às análises estatísticas tradicionais e apresentam-se como aliadas, e não substitutas, dos métodos já consolidados.

Conclui-se, evidenciando a versatilidade e abrangência dos modelos de aprendizado de máquina, a importância cada vez maior do desenvolvimento de habilidades de programação para o engenheiro contemporâneo e a importância do conhecimento técnico que torna possível o estudo enquanto método científico, garantindo a confiabilidade de resultados.

Assim, este trabalho tem por objetivo geral apresentar uma aplicação de clusterização hierárquica de rugosidade em superfícies nanotexturizadas em implantes dentários atualmente disponíveis no mercado odontológico.

2 REVISÃO BIBLIOGRÁFICA

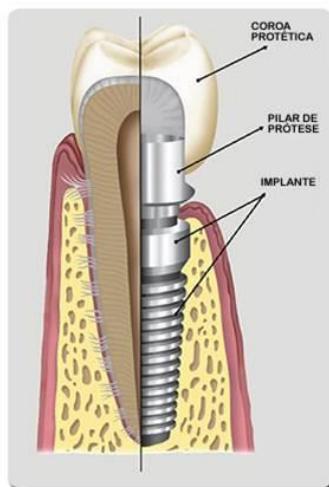
2.1 Implantes Dentários no Contexto Biológico

2.1.1 Implantes Dentários

A perda de elementos dentais constitui um fator de grande impacto na vida de uma pessoa, tanto do ponto de vista estético, quanto em relação às consequências decorrentes da falta de dentes, como problemas na mastigação, aceitação social, autoestima, dentre outros. A utilização de implantes dentais com integração óssea tornou-se um método confiável e previsível de substituir dentes perdidos a fim de melhorar a qualidade de vida dos pacientes. Cada vez mais tem ocorrido concentração de esforços para melhorar ainda mais o conforto do tratamento de implantes do paciente, minimizando o desconforto pré e pós-cirúrgico, maximizando a estética e melhorando o sucesso a longo prazo dos implantes (MENASSA et al., 2014).

São chamados de implantes dentários os elementos aloplásticos (substâncias inertes, estranhas ao organismo humano) alojadas no tecido ósseo completo ou abaixo do periôsteo, para preservar dentes naturais ou substituir dentes faltantes. A técnica cirúrgica para a realização dos implantes sofreu importantes alterações ao longo dos anos, principalmente com o desenvolvimento de tecnologias que permitem a modelagem em 3 dimensões da estrutura óssea, permitindo assim uma menor morbidade e melhor recuperação do paciente, além de aumentar a possibilidade de sucesso do implante (XUEREB et al., 2015). A Figura 1 mostra a estrutura de um implante.

Figura 1 - Partes de um implante dental.



Fonte: RC Odontologia (s/d).

Os implantes dentários produziram resultados satisfatórios em casos nos quais o tratamento por meio da reconstrução completa da boca foi anteriormente difícil. Uma prótese bem elaborada de implantes dentários restaura a função de mastigação, possui biomecânica e estética superiores e facilita cuidados de longa duração eficientes. A boa elaboração de uma prótese de implante dentário baseia-se na colocação apropriada do implante, levando-se em consideração a importância na colocação, ângulo e direção do implante (SIMONIS et al., 2010). Para que estes objetivos sejam alcançados, é preciso que a colocação do implante seja planejada de forma multidisciplinar, de modo que o resultado final seja vislumbrado (BERNARDES et al., 2006).

A descoberta da capacidade de um osso integrar estruturas exógenas, chamada de osseointegração, foi descrita primeiramente pelo professor suíço Per-Ingvar Bränemark e sua equipe, em 1965 (MARTINS, 2011), ao observarem em seus experimentos que o titânio se integrava perfeitamente ao osso de coelhos, sem haver rejeição, e serviu como base para o desenvolvimento dos implantes dentários modernos. A evolução da implantologia oral por meio da pesquisa experimental e clínica possibilitou o desenvolvimento de técnicas cirúrgicas e próteses que melhoraram as expectativas de sucesso entre os pacientes (DOMÍNGUEZ et al., 2013).

A colocação dos implantes simplifica a reabilitação, especialmente nos casos de reabsorção mandibular completa desdentada, tão difícil de resolver por técnicas convencionais. E é graças à osseointegração que os problemas de estética, retenção, suporte e estabilidade da prótese são resolvidos. Esta terceira dentição, como alguns autores denominam, resulta da integração óssea dos implantes e do bom manejo dos tecidos moles (TEIXEIRA, 2010).

Ao longo dos anos os implantes dentários evoluíram, propiciando assim o desenvolvimento de técnicas cirúrgicas avançadas, que possibilitam uma rápida recuperação do paciente. Apesar da biocompatibilidade do titânio, uma modulação positiva dos processos biológicos é de alguma forma limitada porque o titânio por si só é incapaz de induzir a deposição óssea (osteoindução). Portanto, pesquisas recentes têm se concentrado em melhorar os tratamentos de superfície para promover a integração precoce, reduzindo assim o tempo total de tratamento necessário (TEIXEIRA, 2010).

2.1.2 Peri-implantite

Os tecidos peri-implantares saudáveis desempenham um papel importante como barreira biológica para alguns dos possíveis agentes causadores da doença peri-implantar. Se

realizar uma comparação entre o dente e o implante, fica claro o entendimento que existem mecanismos de proteção específicos no dente, como o epitélio juncional, o tecido conjuntivo e os elementos celulares do sistema imunológico. O epitélio e a interface entre o tecido conjuntivo supralveolar e a superfície de titânio de um implante são diferentes da interface homóloga do complexo gengivo-dentário. A união do epitélio com a superfície do implante é do tipo hemidesmossomal, como a do tecido conjuntivo, mas o arranjo de suas fibras é predominantemente longitudinal em relação à superfície do implante e não perpendicular como ocorre no dente natural (SCHWARZ et al., 2018).

A mucosite peri-implantar corresponde em termos básicos à gengivite. Foi definida como uma reação inflamatória reversível na mucosa peri-implantar em torno de um implante dentário osseointegrado. A mucosa peri-implantar é uma barreira biológica de 3 a 4 mm que protege a zona de osseointegração de fatores liberados da placa e da cavidade oral. Sua superfície externa é revestida por epitélio oral estratificado queratinizado contínuo com um epitélio juncional que está preso à superfície do implante por uma lâmina basal e por hemidesmossomos. O epitélio juncional tem 2 mm de comprimento e é separado do osso alveolar por apenas 1 a 2 mm de tecido conjuntivo rico em colágeno semelhante a uma cicatriz, com menos vasos sanguíneos e fibroblastos do que a gengiva ao redor dos dentes. O tecido conjuntivo próximo à superfície do implante constrói a vedação mucosa, onde se forma a resposta inflamatória à colonização microbiológica da superfície do implante (OLIVEIRA et al., 2015).

A peri-implantite corresponde, basicamente, à periodontite do adulto. Foi definida como uma reação inflamatória associada à perda do osso de suporte ao redor do implante dentário. Existem poucos relatos sobre os tecidos peri-implantar de implantes malsucedido em humanos. Em comparação com os tecidos periodontais, o tecido peri-implantar tem uma capacidade limitada para resolver lesões progressivas associadas às plaquetas em modelos animais experimentais. Grandes lesões inflamatórias foram encontradas na mucosa peri-implantar e se estendem para o osso alveolar provocando destruição óssea avançada mediada (NOGUEIRA-FILHO et al., 2011).

As lesões de peri-implantite são frequentemente assintomáticas e geralmente detectadas em consultas de rotina por sangramento à sondagem, que está sempre presente na doença peri-implantar (exceto em alguns fumantes). Outros sinais clínicos da doença incluem supuração, aumento da profundidade de sondagem em relação à linha de base, recessão da mucosa, drenagem dos seios da face e edema da mucosa peri-implantar. Se não for diagnosticado e

administrado de forma eficaz, a doença peri-implantar pode resultar na perda completa da integração e perda do implante (OLIVEIRA et al., 2015).

Figura 2 - Lesões peri-implantares.



Fonte: Cordeiro (2009).

O desenho do implante é um fator importante no início e no desenvolvimento da peri-implantite. Um sistema de implante específico é descrito de acordo com sua morfologia macroscópica, sua microssuperfície e a qualidade do ajuste de seus componentes. A rugosidade da superfície de um implante facilita a aderência da placa bacteriana quando esta é exposta ao meio bucal, embora não haja diferenças quanto ao tipo de superfície e a seleção de espécies bacterianas agressivas colonizadoras (MOMBELLI et al., 2012).

O desencontro entre os componentes que compõem um sistema implante-prótese pode favorecer a retenção da placa bacteriana, além de permitir a passagem de microrganismos para o pilar transepitelial. Isso é possível porque, conforme descrito no estudo de Binon et al. (1992) o erro médio de ajuste entre o *abutment* (também conhecido como pilar do implante, da prótese ou do dente) e o implante mostra discrepâncias entre 20 e 49 micrômetros entre os componentes dos diferentes tipos de implantes comercializados atualmente. Este espaço fornece uma porta de entrada para microrganismos na cavidade oral que têm menos de 10 micrômetros de tamanho.

A morfologia externa do implante de titânio parece ser de menor importância, desde que tenha sido instalado corretamente. Deve-se levar em consideração a influência do desenho macroscópico, no que diz respeito ao padrão de transmissão das forças para o osso, que pode favorecer a sobrecarga mecânica em algum ponto, principalmente na área de união entre o osso e o colar cervical implantar. A perda óssea neste ponto biomecanicamente fraco facilita a

formação de um defeito ósseo neste nível e sua subsequente contaminação (ALGRAFEE et al., 2012).

Os fenômenos de corrosão que podem ocorrer quando uma estrutura de metal não nobre é fixada a um implante de titânio também são apontados como causa de peri-implantite. Nestes casos, verificou-se um aumento do número de macrófagos nos tecidos peri-implantares, o que favoreceria a reabsorção óssea inicial por causas não infecciosas (OLIVEIRA et al., 2015).

2.1.3 Aderência Bacteriana

O biofilme é descrito como uma comunidade microbiana relativamente indefinível, associada à superfície do dente ou a qualquer material rígido que não se solta. Biofilmes são onipresentes e se formam em praticamente todas as superfícies imersas em ambiente aquoso natural, por exemplo, canos de água, tecido vivo, superfície de dente, dispositivos médicos implantados, implantes dentários, etc. Infecções mediadas por adesão de biofilme mais comumente vistas são no coração implantado, válvulas, cateteres venosos, próteses vasculares, dispositivos de fixação de fraturas, implantes mamários, lentes intraoculares e implantes dentários.

Biofilmes consistem em uma ou mais comunidades de microrganismos distribuídos não aleatoriamente em um glicocálice. Esses biofilmes permitem que os microrganismos se colem e se multipliquem nas superfícies. As interações entre as várias espécies bacterianas que residem e crescem no biofilme ocorrem por troca metabólica, contato físico, troca de informações genéticas, sinalização de informações mediadas por moléculas (BUSSHET al., 2010).

O biofilme formado na superfície do dente é denominado placa dentária. As bactérias que proliferam na placa dentária formam os principais fatores etiológicos para a maioria das doenças dentárias, por exemplo, cárie, gengivite, periodontite e peri-implantite. O ataque microbiano foi citado como a principal causa da falha do implante dentário. Os biofilmes são responsáveis pela associação de cerca de 65% das doenças, incluindo peri-implantite e periodontite (SAKKA et al., 2012).

A evidência microbiológica da primeira infecção peri-implantar relacionada ao biofilme humano vem do estudo em amostras de placa coletadas da maior parte apical de 17 implantes doentes. Implantes com bolsas de sondagem mais profundas mostraram uma presença de menor número de cocóides e mais níveis de espiroquetas (RAMS, 1983).

A formação de biofilme em implantes dentários e nos dentes segue um padrão semelhante de colonização microbiana. A formação de biofilme ao redor dos dentes naturais ocorre em minutos e as espécies específicas começam a colonizar em 2 a 6 horas. A razão atribuída possivelmente reside no fato de que as superfícies dentais limpas provavelmente têm restos de microbiota não aderida que podem se multiplicar imediatamente e fornecer uma superfície favorável para a fixação dos colonizadores tardios (LEE; WANG, 2010).

As superfícies imaculadas dos implantes não possuem a microbiota desejada e exigem que os primeiros colonizadores preparem o terreno para o desenvolvimento das comunidades complexas (LI et al., 2004). A película começa a se formar na superfície do implante 30 minutos após a exposição do implante na cavidade oral. A película adquirida nos implantes dentários, devido à sua menor capacidade de absorção de albumina, causa uma baixa formação de placa ao redor dos implantes.

Os primeiros colonizadores são predominantemente cocos Gram-positivos, bastonetes e espécies de actinomices. Os patógenos periodontais que colonizam os estreptococos (*P. gingivalis*, *P. intermedia*, etc) são os microrganismos causadores responsáveis pela peri-implantite e periodontite (FURST et al., 2007).

2.2 Processos de Tratamento de Superfícies para Implantes

Um parâmetro importante para o sucesso clínico dos implantes dentários é a formação do contato direto entre o implante e o osso circundante. A qualidade da interface osso-implante é diretamente influenciada pela rugosidade da superfície do implante que, desde o início da década de 1980, tem sido identificada como um dos seis fatores que são particularmente importantes para a incorporação do implante no osso (NOVAES JR et al., 2010).

Tanto a morfologia quanto a rugosidade da superfície influenciam a proliferação e diferenciação celular, a síntese da matriz extracelular, os fatores de produção locais e até mesmo a forma celular. Além disso, a forma da célula regula seu crescimento, expressão gênica, secreção de proteínas, diferenciação e apoptose. Portanto, a aderência dos osteoblastos nas superfícies dos implantes não é suficiente para a realização da osseointegração, muito menos para melhorá-la, mas é necessária, principalmente, para permitir que a célula receba os sinais para induzir a sua proliferação. Além disso, a rugosidade não só facilita a retenção das células osteogênicas, mas também permite que elas migrem para a superfície do implante por ósseo-conduvidade.

A formação óssea mais rápida e forte proporciona maior estabilidade durante o processo de reparo, permitindo carregamento do implante ainda mais rápido. As superfícies dos implantes orais apresentam estruturas mensuráveis em escala milimétrica, micrométrica e nanométrica. A forma como essas estruturas influenciam o reparo tem sido tema de várias publicações e estudos nos últimos anos (ALLA et al., 2011).

Até agora, as certezas têm se limitado à influência do desenho do implante e da rugosidade da superfície na escala micrométrica. Um desenho em forma de parafuso e uma superfície com rugosidade média (S_a) de 1 a 2 μm apresentaram melhores resultados. Estudos têm demonstrado que implantes de titânio com rugosidade adequada podem melhorar o contato osso-implante, além de aumentar os valores de torque de remoção. Por outro lado, o aumento da rugosidade superficial ao nível de superfícies tratadas com titânio na forma de plasma com S_a acima de 2 μm , provoca uma resposta óssea prejudicada e não reforçada (JEMAT et al., 2015).

Como resultado, nos últimos 20 anos, muitos sistemas de implantes com diferentes topografias de superfície foram introduzidos. Os implantes orais são um exemplo da estreita ligação entre a pesquisa e a indústria, uma vez que as descobertas laboratoriais frequentemente se tornam aplicações clínicas. No entanto, qualquer alteração na morfologia do implante, ou seja, no seu desenho, provoca alterações na topografia ao nível micrométrico e vice-versa. Da mesma forma, mudanças químicas causam mudanças físicas e vice-versa.

Existem muitos tipos de tratamentos de superfície no mercado. Em geral, todos buscam alterar a rugosidade da superfície previamente usinada, elevando-a a níveis considerados ótimos. Uma combinação de jateamento, seguida de condicionamento ácido, tem sido uma técnica comumente usada para tratamento de superfície nos últimos anos.

O principal motivo da combinação de métodos é que, hipoteticamente, o jateamento atinge uma rugosidade e fixação mecânica ideais, enquanto o condicionamento ácido suaviza os picos e pode adicionar um componente de alta frequência à superfície do implante, com potencial importância para a aderência de proteínas, que é considerada importante durante o processo inicial de consolidação óssea (NOVAES JR et al., 2010).

As características superficiais obtidas com a deformação por jateamento dependem do tipo de partícula utilizada, sua dureza, tamanho e velocidade de impacto. O processo geralmente realizado com partículas de óxido de titânio (TiO_2) ou alumina (Al_2O_3) permite um bom controle do tamanho das microcavidades resultantes.

Algumas partículas restantes podem, no entanto, ficar embutidas e contaminar a superfície do implante. O ataque ácido remove algumas camadas atômicas da superfície

deformada, parte da tensão superficial residual, e reduz a possibilidade de contaminação da superfície por partículas que sobraram do processo de jateamento, pois também atua na limpeza da superfície. Esses processos criam microcavidades que se sobrepõem à superfície rugosa pré-jateada.

Cada fabricante tem seu próprio método de condicionamento ácido com relação à temperatura, concentração de ácidos e tempo de exposição. Em geral, ocorre o ataque ácido duplo, que é realizado mergulhando primeiro os implantes em soluções de $\text{HCl} + \text{H}_2\text{SO}_4$, $\text{HNO}_3 + \text{HF}$ ou HNO_3 . Em seguida, o implante é novamente imerso em solução aquosa de HNO_3 para estabilizar a camada de óxido de titânio (BAUER et al., 2013).

A versatilidade desse tipo de tratamento permite uma ampla variação nos procedimentos para se obter a rugosidade desejada. Por outro lado, esta característica pode produzir superfícies significativamente diferentes. Consequentemente, é muito importante caracterizar as superfícies para obter os valores previstos pelos tratamentos propostos. Segundo Wennerberg e Albrektsson (2000), um perfilômetro interferométrico é uma forma segura e eficaz de medir a rugosidade de implantes em formato de parafuso.

2.2.1 Procedimento atual

Diferentes metodologias têm sido aplicadas para o tratamento da superfície de implantes. Difusão plasma de hidroxiapatita e titânio formam uma camada sobre o implante, cuja rugosidade depende do tamanho das partículas. Entretanto proporciona superfícies com rugosidade grande, o que aumenta a possibilidade de contaminação bacteriana (GALLI et al., 2013).

O jateamento por partículas de óxido de alumínio ou titânio resultam impressões irregulares. As partículas não devem aderir ao implante, apenas criar as rugosidades, que dependem do tamanho das partículas, da pressão do disparador e do tempo de disparo. Por ser insolúvel em ácido, a remoção das partículas de óxido de alumínio é difícil, o que pode fazer com que partículas continuem aderidas e interfiram no processo de osseointegração.

Outro fator que interfere na escolha deste tipo de tratamento reside no fato de que ele proporciona uma superfície com características químicas muito heterogêneas, que interferem na resistência da superfície do implante à corrosão (LE GUEHENNEC et al., 2007).

O tratamento por jateamento seguido de tratamento com ácido combina a macrotexturização proporcionada pelo jateamento, com a microtexturização proporcionada pelo ataque ácido. A textura obtida desta forma é bastante homogênea, fator este que auxilia no

processo de integração do implante. O jateamento, neste caso, é feito com areia, seguido por tratamento com ácido. A estabilidade secundária destes implantes se mostrou maior do que em outros tratamentos (GEHRKE et al., 2011).

Feixes de laser podem ser usados para modificar a superfície do implante de forma a produzir erosões e rugosidade. Por não ter interação nem interferência de nenhum material externo, é considerado um tratamento limpo, produzindo rugosidade controlada e alto grau de pureza, que garantem uma boa osseointegração. Outra vantagem deste método é que ele permite a criação de pontos de textura em locais definidos da superfície. Por não envolver nenhum elemento químico, a técnica fica livre da contaminação da camada de óxido de titânio (MALUF et al., 2007).

A nanotexturização consiste na aplicação de uma camada extra de óxido de titânio, obtida pela aplicação de potencial elétrico, usado o implante como um anodo. O campo elétrico guiado aumenta a espessura da camada de óxido de titânio do implante. A adição de outros elementos, como o fosfato, potencializa a osseointegração. A oxidação do implante promove modificações em sua superfície que melhoram a adesão e a orientação das células, resultando em uma osseointegração mais rápida e eficiente (THAKRAL et al., 2014).

2.2.2 Laser – LIPSS

Desde a descoberta de estruturas de superfície periódicas induzidas por laser (LIPSS) por Birnbaum em 1965, este tópico evoluiu para uma perenidade científica. Com a ampla disponibilidade de pulsos de laser ultracurtos e sua capacidade de gerar estruturas de superfície periódicas com dimensões submicrométricas, muitos pesquisadores estudaram seus mecanismos de formação em experimentos com pulsos ultrarrápidos para alcançar períodos cada vez menores (BONSE et al., 2016).

Após a irradiação de sólidos com pulsos de laser ultracurtos polarizados linearmente (durações de pulso de fs a ps) no ar sob incidência normal, geralmente dois tipos distintos de LIPSS são observados, os paralelos e os perpendiculares à orientação de polarização do feixe. Para materiais que absorvem fortemente a radiação laser, na maioria dos casos, os chamados LIPSS de baixa frequência espacial (LSFL) são observados com uma periodicidade (Λ) próxima ao comprimento de onda de irradiação (λ) e uma linha de orientação perpendicular à direção de polarização do feixe. Em materiais transparentes (dielétricos) LIPSS de alta frequência espacial (HSFL) foram relatados com períodos significativamente menores do que o comprimento de onda de irradiação (λ) e com orientações paralelas ou perpendiculares à

polarização do feixe de laser. Atualmente é geralmente aceito que os LSFL são gerados pela interferência do feixe de laser incidente com uma onda eletromagnética de superfície gerada na superfície rugosa, que pode incluir a excitação de polarizações de plasma de superfície (BONSE et al., 2013).

LIPSS podem ser geradas tanto em irradiação "estática" de um único ponto de superfície ou em processamento "dinâmico" (varredura), onde o feixe de laser e a superfície são movidos um em relação ao outro (normalmente a uma velocidade constante v e em forma sinuosa).

A vantagem óbvia da fabricação de LIPSS é a simplicidade e a robustez do processo, ou seja, as nanoestruturas podem ser obtidas de forma confiável em uma única etapa de processo (sem contato) em ambiente de ar sem necessidade de vácuo. (BONSE et al., 2013).

2.3 Análise de Dados

Esse trabalho utiliza algoritmos de aprendizado de máquina para trazer uma nova perspectiva para as análises de dados de rugosidade, diferentemente do que normalmente é adotado no modelo tradicional de análise de dados, com o emprego de tabelas e elaboração de gráficos como o histograma, por exemplo. Dentro desse ambiente de *machine learning* (aprendizado de máquina), os algoritmos aprendem por experiência e incrementam seu desempenho com o passar do tempo. Essa abordagem normalmente é empregada para a detecção de padrões em dados, tanto para a automatização de tarefas complexas ou para efetuar previsões (INAZAWA et al, 2019).

De acordo com Costa Filho et al (2019), o aprendizado de máquina (AM) é um campo da inteligência artificial direcionado para o desenvolvimento de sistemas capazes de instigar hipóteses ou aproximar funções a partir de experiências acumuladas em problemas anteriormente tratados (COSTA FILHO et al, 2019).

Para os autores, as decisões adotadas por algoritmos de AM têm como base o aprendizado indutivo, que pode ser dividido em duas classes: (i) supervisionado, quando o propósito é a resolução de problemas de regressão ou de classificação, e (ii) não supervisionado, quando a tarefa é de agrupamento (*clustering*) ou associação (COSTA FILHO et al, 2019).

Esses e outros conceitos adotados em Estatística são apresentados nos subcapítulos 2.3.1 e 2.3.2.

2.3.1 Estatística

Na forma mais comum e praticada de análise, a organização dos dados é realizada por meio do agrupamento em tabelas de frequência que se mostra uma maneira muito eficaz para realizar a análise de um conjunto de dados. Sob a forma de agrupamentos, os dados também podem ser representados graficamente com a utilização de histogramas e diagramas de caixa (*boxplot*) (DOMINGUES, DOMINGUES, 2010).

Para Behr et al (2008), o histograma é uma ferramenta com uma exterioridade próxima ao diagrama de Pareto, que contempla a medição de dados (tempo, distância, temperatura, velocidade, altura, entre outros), retratando sua distribuição conforme a frequência em que aparecem. Esta técnica mostra-se interessante, pois apresenta de modo visual a concentração dos dados identificados, permitindo a análise de suas variações no decorrer do tempo.

A montagem do histograma é fácil e apoia-se em conceitos estatísticos, como:

- a. conhecer o tamanho da população pesquisada;
- b. definir a amplitude da população, isto é, conhecer a diferença entre os extremos, maior e menor, presente na população;
- c. separar a amplitude em classes ou categorias. Quanto maior o número de classes dos dados, mais precisa torna-se a análise, porém recomenda-se que seja um número inferior a 12 para não dificultar a análise;
- d. estabelecer o limite e o tamanho, com valor inicial e valor determinado para saber como ocorrerá a distribuição da população;
- e. elaborar uma tabela da frequência em que os dados estão presentes;
- f. montar o histograma tendo como base a tabela de frequência.

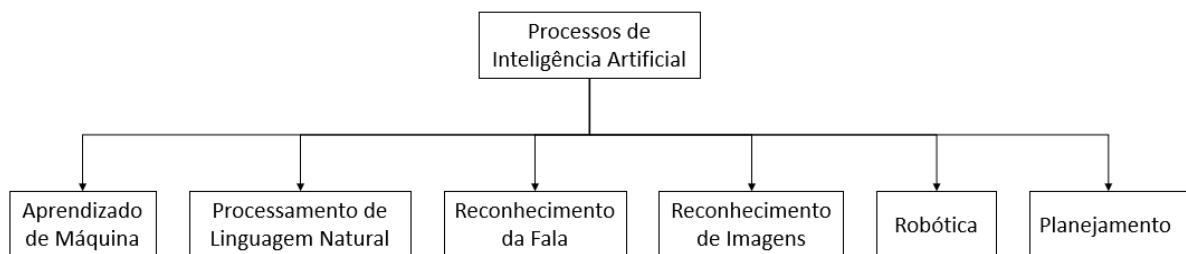
Vale ressaltar que o histograma traz toda a população analisada e não somente sua região perto da média (BERH et al, 2008).

Já o *boxplot*, de acordo com Domingues e Domingues (2010), trata-se de um gráfico que permite reproduzir a distribuição de um conjunto de dados tendo como base alguns de seus parâmetros descritivos, podendo ser a mediana (Q2), o quartil inferior (Q1), o quartil superior (Q3) e o intervalo interquartil ($IQR = Q3 - Q1$). Esse tipo de gráfico é muito útil para a comparação de conjuntos de dados diferentes, sendo necessário para isso que se utilize a mesma escala para ambos conjuntos de dados (DOMINGUES, DOMINGUES, 2010).

2.3.2 Inteligência Artificial

De acordo com Mateus et al (2020), inteligência artificial pode ser definida como a capacidade de os programas computacionais executarem operações de forma que se possa acreditar que estejam imitando os processos de raciocínio humano. Tais processos foram agrupados em seis tipos de IA: (1) Aprendizado de máquina, (2) Processamento de linguagem natural, (3) Reconhecimento de fala, (4) Reconhecimento de imagens, (5) Robótica e (6) Planejamento (MATEUS et al, 2020).

Figura 3 - Tipos de inteligência artificial agrupados por processos



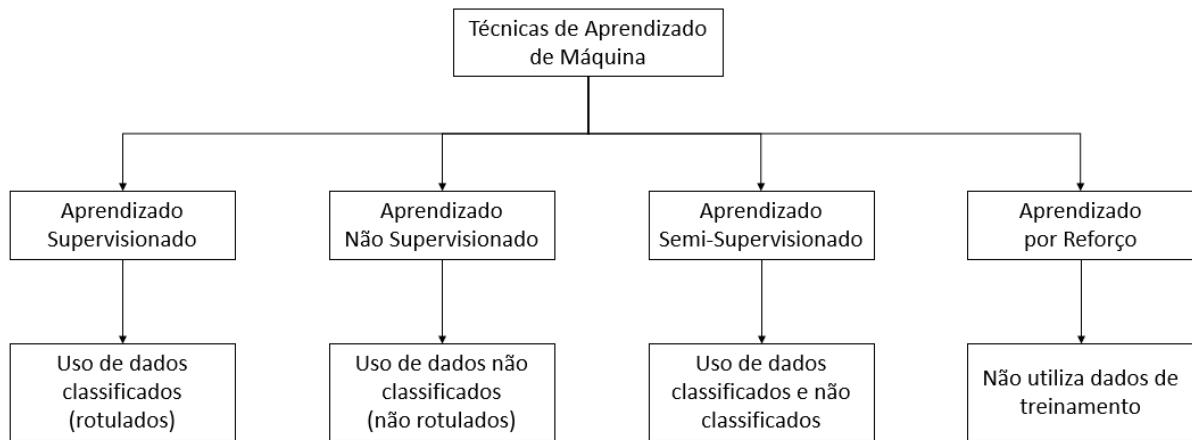
Fonte: Mateus et al (2020).

Com foco no primeiro tipo de IA, o aprendizado de máquina, utilizado neste trabalho, engloba o desenvolvimento de algoritmos que possibilitam o sistema aprender com situações passadas (dados históricos), identificar padrões por meio de métodos estatísticos, efetuar uma determinação ou prognóstico e automaticamente melhorar seu desempenho. No próximo subcapítulo este assunto é tratado com mais detalhes.

2.3.3 Aprendizado de Máquina

Existem quatro técnicas principais de modelagem de aprendizado de máquina: Aprendizado supervisionado, Aprendizado não supervisionado, Aprendizado semi-supervisionado e Aprendizado por reforço. A Figura 4 apresenta as categorias de aprendizado de máquina. (MATEUS et al, 2020)

Figura 4 - Técnicas de aprendizado de máquina por categorias



Fonte: Mateus et al (2020).

De acordo com Mateus et al (2020), a principal diferença entre as técnicas empregadas no aprendizado de máquina, especificamente nos casos enquadrados como supervisionado e não supervisionado, é o fato de que os algoritmos do aprendizado de máquina supervisionados serem praticados em agrupamentos de dados rotulados que direcionam o algoritmo a entender quais recursos são relevantes para a solução do problema em questão. Por sua vez, os algoritmos não supervisionados são preparados com dados não rotulados e devem estabelecer a relevância do recurso por critério próprio, de acordo com os padrões ligados à amostra.

A aprendizagem supervisionada tem como base dados preparados para treinamento quando se conhece o destino de cada registro de um conjunto de dados. Na aprendizagem não supervisionada, os algoritmos procuram padrões em registros com características semelhantes, comparando os valores dos seus atributos. Esse tipo de aprendizagem (não supervisionada) é costumeiramente aplicada em problemas de agrupamento (também conhecidos como clusterização) ou na redução da dimensão de conjuntos de dados multivariados (FERNANDES et al, 2019).

Normalmente o aprendizado supervisionado é aplicado para realizar prognóstico sobre eventos. Já o aprendizado não supervisionado é empregado, via de regra, para a descrição de eventos ainda desconhecidos (MATEUS et al, 2020).

Para a técnica de aprendizagem semi-supervisionada, os dados obtidos apresentam-se como uma composição de dados rotulados e não rotulados. Essa combinação normalmente é empregada para desenvolver um modelo específico de classificação dos dados. O método de aprendizado por reforço visa usar observações reunidas a partir da interação com o meio ambiente para tirar ações que maximizam o ganho ou minimizam o risco. Dessa forma, com o

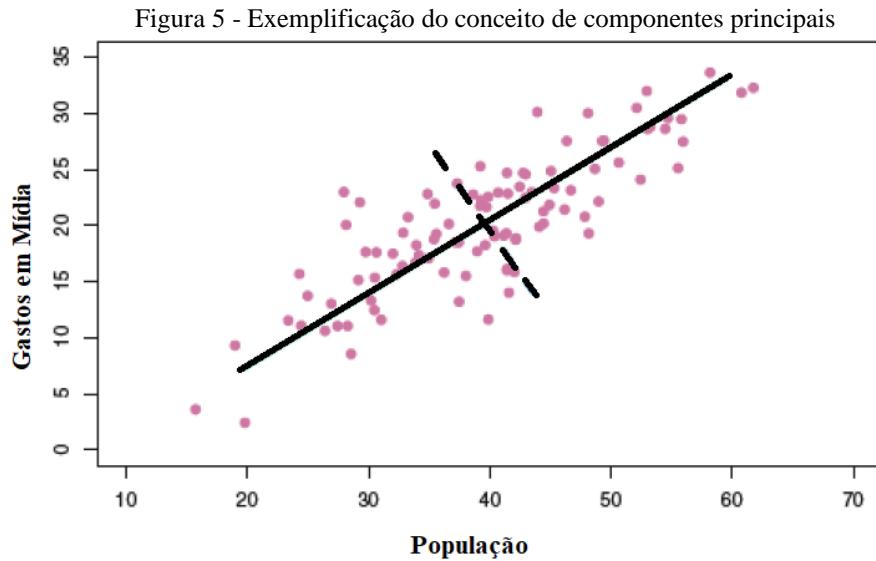
uso deste método, é possível determinar o comportamento ideal que um algoritmo deve apresentar dentro de um contexto específico (MATEUS et al., 2020).

2.3.4 Análise de Componentes Principais

A análise de componentes principais (do inglês, *Principal Component Analysis – PCA*) é um método de redução de dimensionalidade, mais especificamente, de extração de variáveis. Seu objetivo é maximizar a representação da variância dos dados através da geração de M novas direções ($M \leq p$) em que os dados serão projetados. Essas novas direções são chamadas componentes principais (do inglês, *principal components – PC*) e são obtidas através da aplicação dos conceitos de autovetores e autovalores de álgebra linear sobre a matriz de covariância dos dados (GARETH et al., 2013).

Métodos de redução de dimensionalidade possuem como objetivo reduzir o número total de variáveis p (considerando uma matriz $n \times p$) e representar os dados em um conjunto de variáveis principais. São muito utilizados em aprendizado de máquina para reduzir tempos de processamento, economizar espaço de armazenamento e possibilitar visualizações em duas e três dimensões. Eles são divididos em métodos de seleção de variáveis e de extração de variáveis.

O primeiro componente principal (PC1) é a direção que, ao projetar-se os valores sobre ela, maximiza a variância dos dados. O segundo componente principal (PC2) é a direção, ortogonal ao PC1, em que há a segunda maior variância dos dados. A Figura 5 traz visualmente o conceito. É apresentado, como exemplo, um gráfico de dispersão de população por gastos em mídia. A linha cheia representa o PC1, a direção em que há a maior variância de dados, perpendicular ao PC1 tem-se o PC2 em linha pontilhada, que é a direção em que há segunda maior variância de dados. Neste caso dois componentes principais conseguem representar a totalidade da variância dos dados, pois há apenas duas variáveis consideradas.



Fonte: Adaptado de GARETH et al (2013).

2.3.5 Clusterização

Clustering ou métodos de agrupamento são empregados para separar objetos de dados em grupos ou, de outra forma, como uma etapa de pré-processamento para submissão a outros algoritmos. São identificados como aprendizado não supervisionado em razão das informações do rótulo de classe não estarem presentes (SILVA, 2018). Para o autor, trata-se basicamente de colocar em um mesmo grupo objetos similares, de acordo com algum critério, de maneira que as características desses objetos do grupo sejam semelhantes entre si e diferentes dos objetos dos outros grupos.

Os métodos de agrupamento podem ser classificados de várias formas, mas basicamente são adotadas as seguintes categorias de métodos: hierárquicos, particionais e baseados em densidade. Os métodos hierárquicos formam um conjunto de dados em uma estrutura hierárquica de acordo com a proximidade dos elementos. Normalmente, os grupos são representados por uma árvore que separa a base de dados em subconjuntos menores. Nesta representação, um elemento tem a sua representação pela folha da árvore, sendo que a junção de todos os elementos apresenta-se representado pela raiz. No método hierárquico é essencial determinar uma distância de corte para que sejam identificados os grupos formados. Assim, é muito importante ter conhecimento sobre a estrutura dos dados e do objetivo da análise para estabelecer o corte que irá separar os grupos. Já os métodos particionais reúnem apenas grupos de formato circular ou esférico, enquanto os métodos baseados em densidade têm a capacidade

de identificar grupos de formato irregular ou aleatório, além de serem eficientes para encontrar distorções (SILVA, 2018).

3 MÉTODOS E FERRAMENTAS

As informações referentes aos corpos de provas, sistemas de laser e topografia de superfície foram obtidos em (UHLMANN et al., 2019)

3.1 Corpos de Prova

Os corpos de prova são discos de titânio com diâmetro de 10mm e espessura de 2mm que foram fornecidos pelo fabricante de implantes dentários A.K.TEK MEDIZINTECHNIK GMBH, Hagen, Alemanha. Eles foram gerados a partir da liga metálica de Ti-6Al-4V grau 5 para uso biomédico fornecido por HIGH TECH ALLOYS SONDERWERKSTOFFE GMBH, Wuppertal, Alemanha.

3.2 Sistemas de Laser

Foram utilizados dois diferentes sistemas para geração da nanotexturização, ambos com a mesma fonte de pulsos ultracurtos Talisker-Three, COHERENT, Santa Clara, Estados Unidos da América. Os detalhes técnicos das máquinas utilizadas são apresentados:

- Máquina de Laser 1: Modelo LMBS 3W-015-xy300z200-IA de LASERMIKROTECHNOLOGIE DR. KIEBURG GMBH, Berlim, Alemanha. A máquina possui potência média de feixe $P_L = 3\text{W}$ e diâmetro $d_u = 16\mu\text{m}$ na posição de foco. A terceira harmônica $\lambda_{UV} = 355\text{nm}$ foi utilizada.
- Máquina de Laser 2: Modelo MJ-Series de OXFORD LASERS LTD, Didcot-Oxford, Reino Unido. A máquina possui potência média de feixe $P_L = 8\text{W}$ e diâmetro $d_u = 16\mu\text{m}$ na posição de foco. A segunda harmônica $\lambda_{GR} = 532\text{nm}$ foi utilizada.

3.3 Topografia de superfície

A avaliação quantitativa dos parâmetros de superfície foram obtidas através de microscopia de força atômica, utilizando o microscópio NanoWizard II, JPK INSTRUMENTS, Berlim, Alemanha. O software livre Gwyddion® foi utilizado para avaliar o perfil das superfícies.

Para o presente estudo foram considerados os seguintes parâmetros de superfície:

- R_a : Rugosidade aritmética principal de perfil.
- R_z : Rugosidade de altura máxima de perfil.

- S_a : Rugosidade aritmética principal de área.
- S_z : Rugosidade de altura máxima de área.
- S_{sk} : Assimetria de altura da topografia.
- S_{ku} : Curtose de altura da topografia.

3.4 Aplicação de Algoritmos

Todo o código foi escrito em linguagem Python e aplicado utilizando-se o software livre Jupyter Notebooks. As principais bibliotecas utilizadas foram:

- Numpy e Pandas: bibliotecas fundamentais para manipulação geral de dados. Numpy opera dados em forma de arrays enquanto Pandas opera em forma de dataframe.
- Matplotlib e Seaborn: bibliotecas para geração de visualizações (Histogramas, Boxplots e Diagramas de Dispersão).
- Sklearn: contém as funções de normalização e análise de componentes principais.
- Scipy: contém a função de agrupamento hierárquico.

4 ANÁLISE DE RESULTADOS E DISCUSSÃO

A fim de demonstrar a utilidade que técnicas de aprendizado de máquina podem oferecer ao processo de desenvolvimento de tecnologia serão comparadas duas técnicas de estatística descritiva tradicionais, histograma e *boxplot*, com os métodos de análise dos componentes principais e agrupamento hierárquico (clusterização).

Primeiramente apresenta-se como os dados obtidos são organizados. Há 3 categorias de corpos de prova:

- CP (Referência): Gerados com a mesma tecnologia empregada nos atuais implantes dentários;
- UV–LIPSS: Gerados a partir da tecnologia LIPSS utilizando comprimento de onda na faixa dos raios ultravioleta ($\lambda_{UV} = 355$ nm);
- GR–LIPSS: Gerados a partir da tecnologia LIPSS utilizando comprimento de onda na faixa da luz verde ($\lambda_{GR} = 532$ nm).

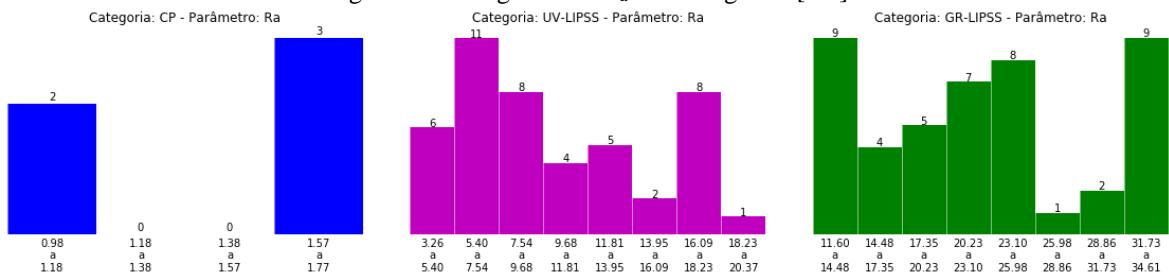
Para CP foram produzidos 5 corpos de prova, para UV-LIPSS e GR-LIPSS foram desenvolvidos 45 corpos de prova em cada categoria, totalizando 95 corpos de prova. Como será demonstrado, os corpos de prova CP, por serem oriundos de técnicas de fabricação já estabelecidas, apresentam grande estabilidade e pouca variação nos valores de mensuração.

4.1 Histograma

O histograma traz graficamente informações sobre a frequência de distribuição das medidas em colunas, que estabelecem o limite inferior e o limite superior considerados nos intervalos que determinam a coluna. Pode-se, assim, observar se existem concentrações em torno de certos valores.

Observa-se já aqui uma limitação de análise devido a característica de poucos corpos de prova disponíveis para medição, especialmente para a categoria de referência (CP). A Figura 6 apresenta o histograma R_a para as três categorias.

Figura 6 - Histograma de R_a nas categorias [nm]



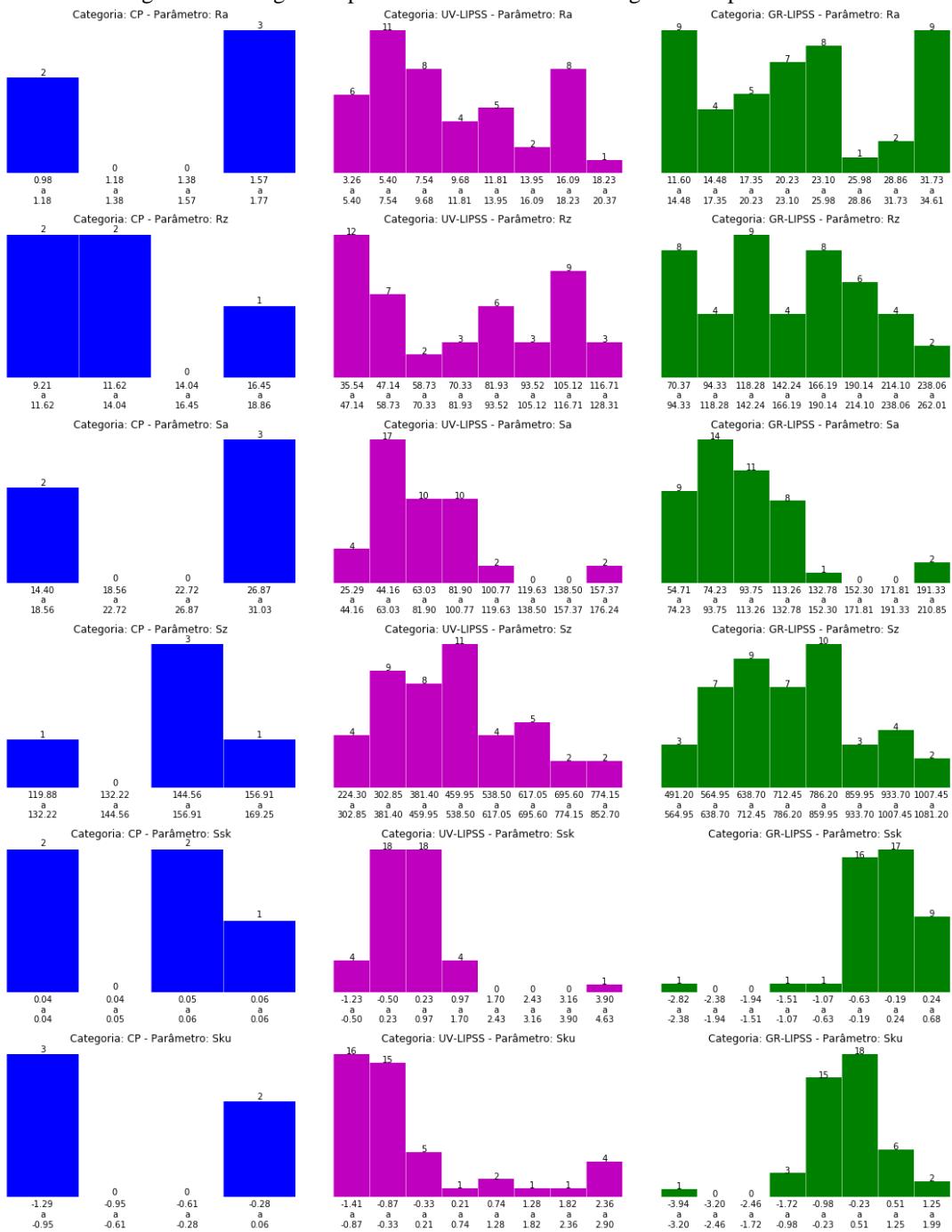
Fonte: Autor (2021).

O histograma de CP apresenta quatro colunas de distribuição, com apenas duas com frequência nos intervalos [0,98 nm, 1,18 nm) e (1,57 nm, 1,77 nm]. No primeiro intervalo preenchido existem duas observações e no segundo, três. Quanto a UV-LIPSS e GR-LIPSS, por existirem 45 corpos de prova em cada, tem-se oito colunas que resumem a distribuição. Em UV pode-se inferir que há dois picos de distribuição e em GR uma ilha.

A forma apresentada de comparação das categorias apresenta duas principais limitações: falta de referência nos eixos para comparação e intervalos de tamanhos diferentes para cada categoria. Em CP, a primeira coluna preenchida possui limite inferior de 0,98 nm e a segunda limite superior de 1,77 nm. Os intervalos possuem tamanho 0,20 nm. Em UV-LIPSS o limite inferior mínimo é 3,26 nm e o superior máximo 20,37 nm, com intervalo de 2,14 nm. Em GR-LIPSS o limite inferior mínimo é 11,60 nm e o superior máximo 34,61 nm, com intervalo de 2,88 nm. Para cada medida em cada categoria os limites e tamanhos dos intervalos se ajustam à distribuição das medidas que contemplam, o que se torna outro obstáculo na obtenção de melhor entendimento e conclusões.

O histograma é, portanto, pouco esclarecedor quanto à forma da distribuição. Além disso, os 3 gráficos resumem a distribuição de apenas uma medida. Considerando 6 medidas e 3 categorias há um total de 18 possíveis gráficos, o que torna a análise onerosa e pouco conclusiva. A Figura 7 apresenta um diagrama dos 18 histogramas possíveis.

Figura 7 - Histogramas possíveis considerando 3 categorias e 6 parâmetros



Fonte: Autor (2021).

Em resumo, considerando o baixo volume de corpos de prova e, portanto, de medidas, característica bastante encontrada no âmbito do desenvolvimento de tecnologia, o histograma se mostra um método pouco esclarecedor quanto a busca por entendimento a partir de comparações. A seguir, são apresentados e analisados os mesmos dados utilizando a visualização de *boxplot*.

4.2 Boxplot ou Diagrama de Caixa

O *boxplot* oferece uma visualização de dispersão, simetria, máximos e mínimos, podendo ainda destacar valores discrepantes em relação aos outros, chamados também de *outliers*.

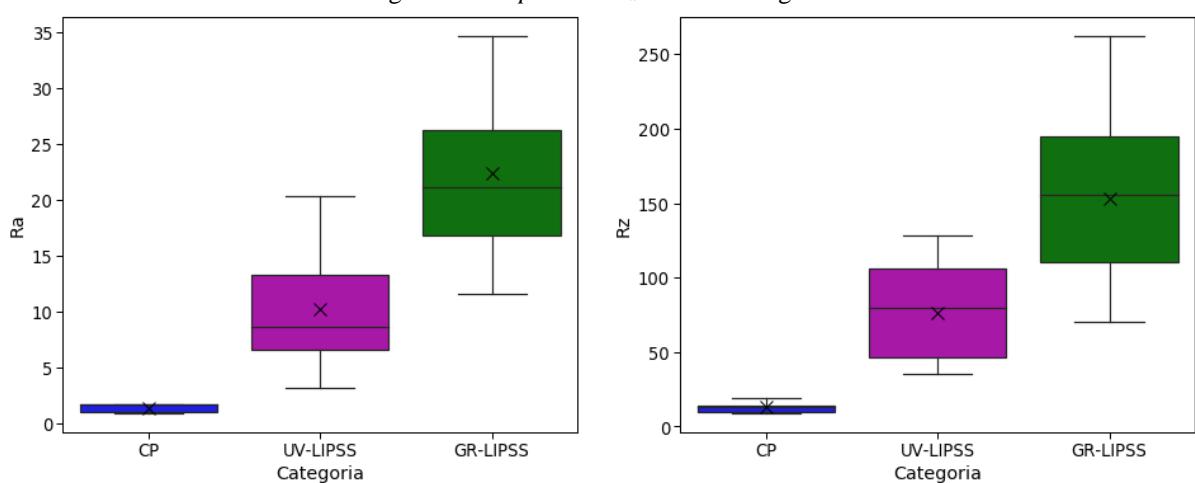
Para montar um *boxplot* são calculadas 5 medidas estatísticas básicas: média, mediana, desvio padrão, valor mínimo e máximo e o 1º e 3º quartis. A Tabela 1 e a Figura 8 demonstram os cálculos e resultado gráfico do *boxplot* para os parâmetros R_a e R_z nas três categorias de corpos de prova. A linha que corta a coluna representa a mediana, os limites da coluna, os quartis, e as extremidades das linhas verticais o valor máximo e mínimo. Um marcador “x” representa a média.

Tabela 1 - Medidas estatísticas para a produção de *boxplot* de R_a e R_z

Medida Estatística	R_a [nm]			R_z [nm]		
	CP	UV-LIPSS	GR-LIPSS	CP	UV-LIPSS	GR-LIPSS
Nº Observações	5	45	45	5	45	45
Média	1,43	10,30	22,36	12,96	75,98	153,43
Desvio Padrão	0,40	4,69	7,37	3,93	29,57	53,34
Valor Mínimo	0,98	3,26	11,60	9,21	35,54	70,37
1º Quartil	1,01	6,56	16,85	9,49	46,36	110,02
Mediana	1,69	8,63	21,10	13,58	79,43	155,64
3º Quartil	1,69	13,25	26,21	13,66	106,10	195,07
Valor Máximo	1,77	20,37	34,61	18,86	128,31	262,01

Fonte: Autor (2021).

Figura 8 - Boxplots de R_a e R_z nas categorias



Fonte: Autor (2021).

Na categoria de referência CP, cada uma das 5 medidas estatísticas consideradas para a construção do *boxplot* está representada por uma das 5 observações existentes. Apesar da

evidente limitação de análise, é didático observar visualmente como se diferenciam os corpos de prova gerados por métodos de fabricação já consolidados e que possuem o objetivo de ser o menos rugosos possível frente às superfícies geradas por LIPSS. Outro ponto a se considerar é que agora há clareza através de comparação visual para comprovar que a categoria de referência cumpre seu propósito de buscar ser o menos rugosa possível. Já as outras categorias permitem uma análise mais completa acerca da distribuição de seus valores.

Considerando R_a , na categoria UV-LIPPS o valor mínimo (3,26 nm) registrado é maior que o máximo registrado para CP. O valor do primeiro quartil (6,56 nm) indica que 25% das observações ficam abaixo desse valor, em contrapartida 25% das observações ficam acima de 13,25 nm, como indicado pelo terceiro quartil, e o valor máximo é 20,37 nm. O valor da é 8,63 nm. O valor médio calculado é 10,30 nm, maior que a mediana. Visualmente, a diferença de proporção em que a mediana corta a coluna indica assimetria na distribuição, resultando em uma curva de distribuição tendendo para a esquerda, como indicado também no histograma na Figura 6.

Verifica-se que GR-LIPSS possui maior variação de dados, bem como valores maiores de mínimo e máximo. No entanto, a distribuição dos valores é mais simétrica que em UV-LIPSS, observando o corte que a mediana faz no intervalo interquartil.

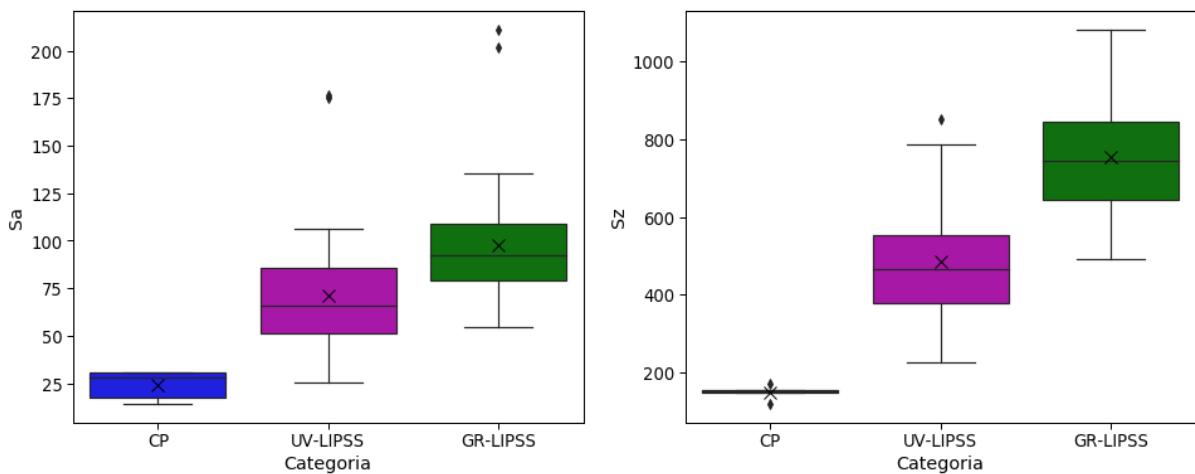
Em R_z , é observado um comportamento muito semelhante ao analisado em R_a . A distribuição de CP demonstra a busca pela superfície menos rugosa possível. UV-LIPSS possui uma variação consideravelmente menor quando comparado a GR-LIPSS, além de uma distribuição em valores menores.

Analizando lado a lado as distribuições de rugosidade média em área S_a e S_z , obtém-se o seguinte resultado, apresentado na Tabela 2 e Figura 9.

Tabela 2 - Medidas estatísticas para a produção de *boxplot* de S_a e S_z

Medida Estatística	S_a [nm]			S_z [nm]		
	CP	UV-LIPSS	GR-LIPSS	CP	UV-LIPSS	GR-LIPSS
Nº Observações	5	45	45	5	45	45
Média	24,21	71,34	97,82	148,49	486,14	754,98
Desvio Padrão	7,84	29,69	31,36	18,05	146,09	140,79
Valor Mínimo	14,40	25,29	54,71	119,88	224,30	491,20
1º Quartil	17,19	50,96	79,01	147,89	378,60	642,50
Mediana	27,80	66,18	92,55	149,70	465,30	745,80
3º Quartil	30,62	85,96	108,71	155,72	553,60	845,90
Valor Máximo	31,03	176,24	210,85	169,25	852,70	1081,20

Fonte: Autor (2021).

Figura 9 - Boxplots de S_a e S_z nas categorias

Fonte: Autor (2021).

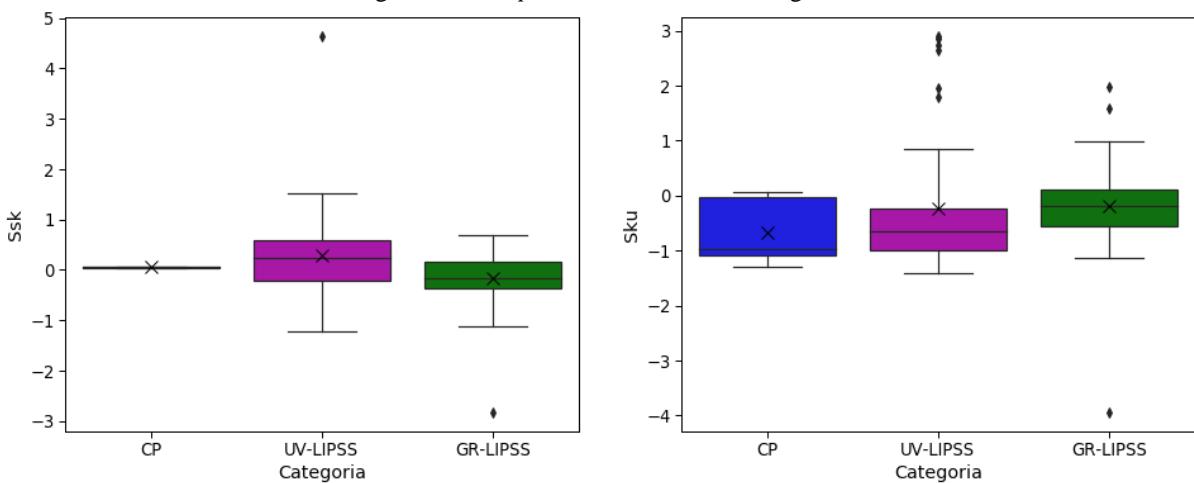
Mais uma vez, ocorre CP seguido de UV-LIPSS e depois GR-LIPSS. As distribuições apresentam simetria. Os pontos fora dos limites das “abas”, representados por pontos, são denominados *outliers*, e indicam valores que superam o valor de 1,5 vez o valor superior determinado pelo terceiro quartil ou 1,5 vez o valor inferior determinado pelo primeiro quartil. É importante a identificação desses valores para consideração de possíveis impactos no estudo a ser feito. São corpos de prova que tiveram valores que se destacam em relação à distribuição, e podem identificar algum erro de fabricação ou inconsistência.

As visualizações de *boxplot* das medidas assimetria estatística (*skewness* – S_{sk}) e curtose (S_{ku}) são apresentadas na Figura 10.

Tabela 3 - Medidas estatísticas para a produção de *boxplot* de S_{sk} e S_{ku}

Medida Estatística	S_{sk}			S_{ku}		
	CP	UV-LIPSS	GR-LIPSS	CP	UV-LIPSS	GR-LIPSS
Nº Observações	5	45	45	5	45	45
Média	0,05	0,28	-0,16	-0,67	-0,23	-0,18
Desvio Padrão	0,01	0,88	0,53	0,63	1,19	0,87
Valor Mínimo	0,04	-1,23	-2,82	-1,29	-1,41	-3,94
1º Quartil	0,04	-0,22	-0,36	-1,08	-1,00	-0,57
Mediana	0,05	0,25	-0,17	-0,98	-0,64	-0,19
3º Quartil	0,05	0,60	0,16	-0,04	-0,24	0,11
Valor Máximo	0,06	4,63	0,68	0,06	2,90	1,99

Fonte: Autor (2021).

Figura 10 - Boxplots de S_{sk} e S_{ku} nas categorias

Fonte: Autor (2021).

Em relação à assimetria, os corpos de prova de CP apresentam maior consistência; UV-LIPSS demonstra uma maior variância de assimetria em relação à GR-LIPSS e em cada uma das categorias há um *outlier*, em sentidos diferentes. Em curtose, todos os valores são menores que 3, classificando os perfis de distribuição de rugosidade como platicúrtica, que apresenta um perfil achatado.

Os métodos de estatística descritiva permitem a definição em medidas estatísticas da distribuição dos corpos de prova. Os histogramas mostram as frequências de distribuição em intervalos e os *boxplots* uma comparação visual de 5 medidas estatísticas básicas, bem como a identificação de *outliers* nos parâmetros específicos.

A seguir, são implementados algoritmos de aprendizado de máquina com o objetivo de clusterizar as observações não-rotuladas em grupos que consideram todos os parâmetros medidos e a descoberta de grupos com características semelhantes. Tal método não leva em consideração a categoria à qual cada corpo de provas pertence, mas apenas os valores dos parâmetros, que são previamente normalizados para correção das diferenças de grandeza.

4.3 Análise dos Componentes Principais

A proposta da clusterização é permitir o agrupamento dos corpos de provas levando-se em conta todas as medidas de todos os atributos de uma vez, consolidando o resultado em uma visualização simples que revela compreensões outrora não visíveis pela complexidade e tamanho dos dados. É uma das técnicas mais extensamente utilizada em *Big Data*. No contexto

de desenvolvimento de tecnologia há um número limitado de observações e parâmetros a se considerar, nesse caso, existe um total de 95 corpos de prova e 6 parâmetros de medida.

Utilizando o *software* Jupyter Notebooks e linguagem Python, implementam-se os algoritmos necessários. Nos próximos parágrafos são descritos os passos para preparar os dados para a clusterização em si, destacando e discutindo os pontos relevantes para entendimento do resultado final.

Após carregamento dos dados em Python a partir de uma planilha Excel, separam-se apenas as 6 colunas com os parâmetros a serem analisados, resultando em uma matriz de tamanho 95 x 6. O primeiro passo de tratamento dos dados é a normalização deles dentro de cada atributo, evitando-se, assim, que haja um enviesamento devido às diferentes grandezas dos atributos. No caso estudado, por exemplo, tem-se que R_a possui valor mínimo de 0,98 nm e máximo de 34,61 nm, enquanto para R_z o mínimo é 9,21 nm e o máximo 262,01 nm. Como o método de clusterização utiliza a distância euclidiana dos pontos para realizar o agrupamento, faz-se necessário a normalização para ajuste da comparação relativa dentro de cada atributo. A descrição da matriz resultante é apresentado na Tabela 4.

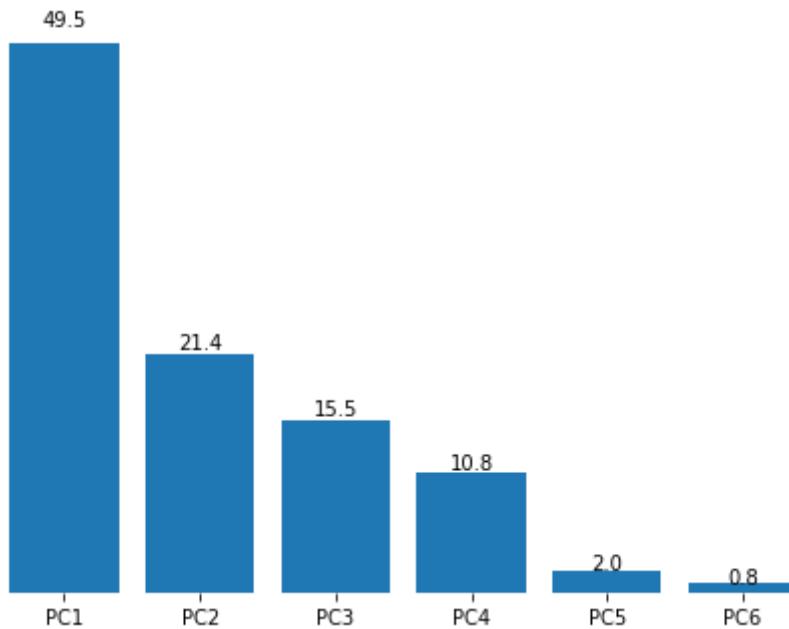
Tabela 4 - Resultado da normalização dos valores dos parâmetros

Medida Estatística	Ra [nm]	Rz [nm]	Sa [nm]	Sz [nm]	Ssk	Sku	Medida Estatística	Ra [nm]	Rz [nm]	Sa [nm]	Sz [nm]	Ssk	Sku
Média	15,55	109,35	81,40	595,72	0,06	-0,23	Média	0,00	0,00	0,00	0,00	0,00	0,00
Desvio Padrão	9,04	60,82	35,03	218,66	0,73	1,02	Desvio Padrão	1,01	1,01	1,01	1,01	1,01	1,01
Valor Mínimo	0,98	9,21	14,40	119,88	-2,82	-3,94	Valor Mínimo	-1,62	-1,66	-1,92	-2,19	-3,95	-3,65
1º Quartil	7,99	60,76	60,51	446,30	-0,24	-0,86	1º Quartil	-0,84	-0,80	-0,60	-0,69	-0,41	-0,62
Mediana	13,58	97,55	79,01	610,32	0,04	-0,40	Mediana	-0,22	-0,20	-0,07	0,07	-0,03	-0,16
3º Quartil	20,76	147,03	97,74	754,00	0,34	0,05	3º Quartil	0,58	0,62	0,47	0,73	0,38	0,28
Valor Máximo	34,61	262,01	210,85	1081,20	4,63	2,90	Valor Máximo	2,12	2,52	3,71	2,23	6,26	3,08

Fonte: Autor (2021).

Uma vez normalizados os atributos, implementa-se a análise de componentes principais (do inglês, *Principal Component Analysis - PCA*). Essa técnica é uma aplicação prática dos conceitos de autovalores e autovetores estudados em álgebra linear. Basicamente, uma matriz com 6 atributos, possui 6 componentes principais. O primeiro componente principal (PC1) é um vetor que se estende na direção em que há a maior variância de dados; o segundo componente principal (PC2) é um vetor que se estende na direção em que há a segunda maior variância de dados, ortogonal (perpendicular, no caso bidimensional) ao PC1; o terceiro componente principal (PC3) segue a mesma lógica, mantendo a ortogonalidade dos vetores; e assim por diante até o sexto componente principal. Um dos resultados, apresentado na Figura 11, é a ordenação decrescente do percentual da variância explicada dos dados que cada componente principal representa, uma vez projetados os pontos nas direções definidas.

Figura 11 - Percentual de variância explicada por componente principal



Fonte: Autor (2021).

O primeiro componente principal representa 49,5% da variância total da base de dados. O segundo representa 21,4%, o terceiro 15,5%, o quarto 10,8%, o quinto 2,0%, o sexto e último apenas 0,8%. Logo, os três primeiros componentes principais representam 86,4% da variância total dos dados, sendo que os últimos três componentes principais representam apenas 13,6%.

O principal retorno da aplicação do algoritmo de análise de componentes principais é a projeção dos valores de atributos nos 6 componentes principais gerados, apresentado na Tabela 5.

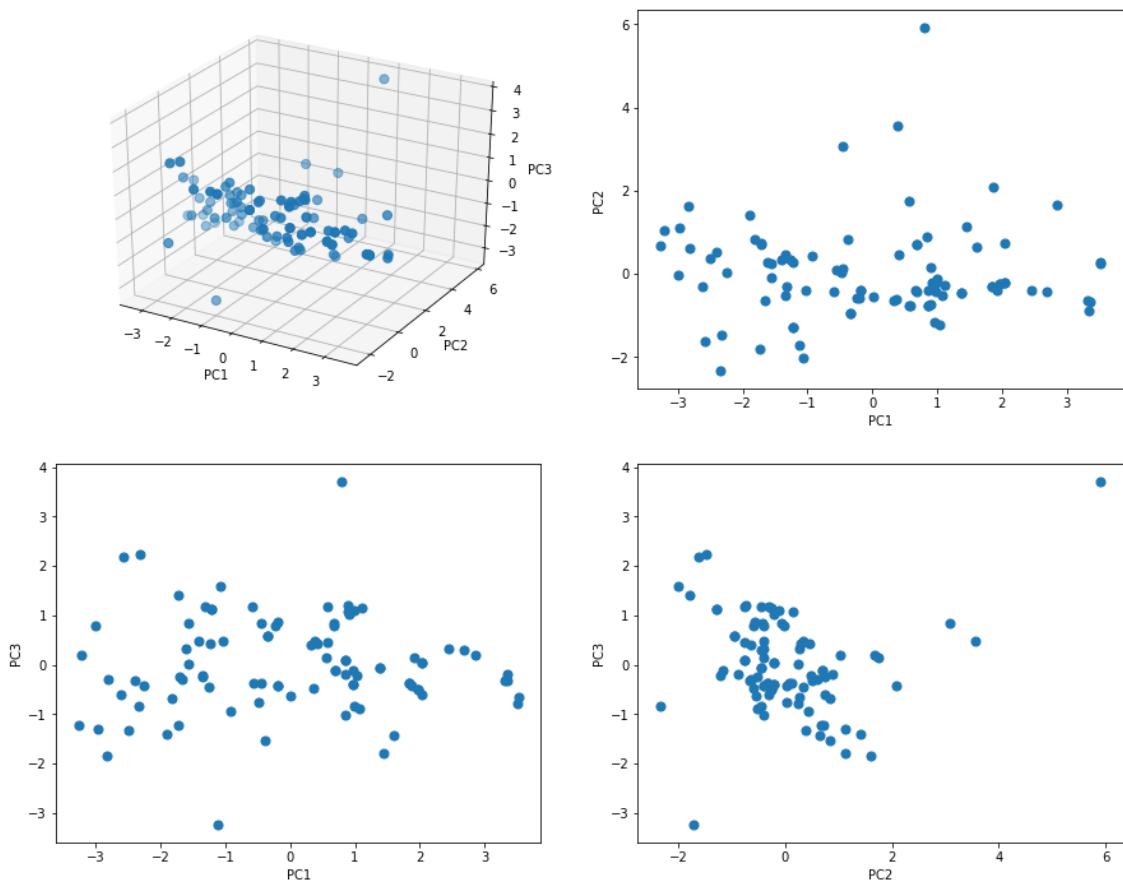
Tabela 5 - Projeção dos valores normalizados nos componentes principais

ID	Ra	Rz	Sa	Sz	Ssk	Sku		ID	PC1	PC2	PC3	PC4	PC5	PC6
CP_1	-1.54	-1.50	-1.92	-2.19	-0.03	0.19	→	CP_1	3.52	0.24	-0.79	0.22	0.14	0.01
CP_2	-1.62	-1.66	-1.45	-1.96	-0.01	-1.04		CP_2	3.35	-0.88	-0.18	0.57	-0.08	-0.03
CP_3	-1.62	-1.65	-1.84	-2.02	0.00	0.28		CP_3	3.52	0.27	-0.66	0.00	0.05	-0.02
UV_1	-0.78	-0.89	-0.19	0.12	-0.41	-0.47		UV_1	0.86	-0.75	0.10	-0.49	-0.53	0.06
UV_2	-0.95	-1.12	-0.44	-0.29	-0.35	-0.11		UV_2	1.37	-0.44	-0.05	-0.58	-0.36	-0.02
UV_3	-1.00	-1.11	-0.93	-1.00	-0.38	-0.01		UV_3	1.97	-0.25	-0.49	-0.24	-0.10	-0.04
GR_1	-0.34	-0.61	-0.60	0.10	0.14	0.76		GR_1	0.69	0.70	-0.12	-0.55	-0.40	-0.09
GR_2	-0.22	-0.43	1.38	1.84	-0.34	-0.81		GR_2	-1.22	-1.29	1.14	-1.07	-0.85	0.05
GR_3	-0.44	-0.64	-0.48	-0.48	-0.38	0.01		GR_3	0.98	-0.20	-0.39	-0.22	-0.09	-0.11

Fonte: Autor (2021).

Os três primeiros componentes principais, por representarem 86,4% da variância dos dados, são escolhidos para serem eixos e projetam-se os valores normalizados das observações nesses eixos. O resultado é um gráfico de dispersão, apresentado na Figura 12.

Figura 12 - Gráficos de dispersão das projeções das observações nos 3 primeiros componentes principais



Fonte: Autor (2021).

Esse é um primeiro resultado muito consistente que o método PCA permite visualizar. A partir de 6 atributos foram separados 3 novos eixos que permitem a geração de visualizações em 2 e 3 dimensões e as descobertas de novos entendimentos.

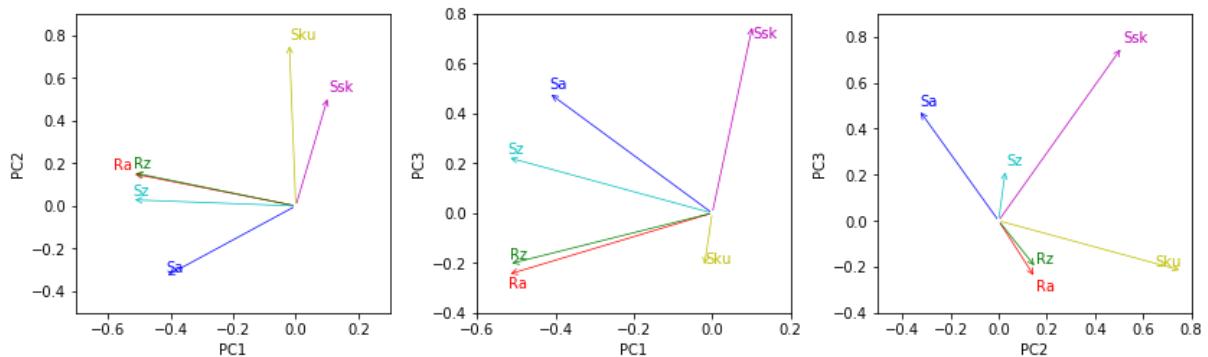
Outro retorno do algoritmo de PCA é a influência que cada atributo inicial teve na geração de cada componente principal. Na Tabela 6, pode-se observar o valor normalizado que apresenta as influências. Por exemplo, o PC1 possui nos parâmetros de rugosidade (R_a , R_z , S_a e S_z) as maiores influências em sua geração, em relação aos parâmetros de distribuição das rugosidades. Em contrapartida, o PC2 possui S_{sk} e S_{ku} como os principais influenciadores. Já o PC3 possui em S_{sk} e S_a os atributos que mais influenciaram. Os valores em módulo são considerados, sendo que o sinal negativo indica apenas um dos sentidos das direções definidas, resquício dos métodos de álgebra linear aplicados. A Figura 13 traz visualmente a informação dessa influência. Com os componentes principais como eixos, são plotados vetores com os valores de cada atributo e tem-se uma representação visual que deixa claro o fator de influência de cada atributo na geração dos componentes principais.

Tabela 6 - Influência de cada atributo nos componentes principais

Componente Principal	Ra	Rz	Sa	Sz	Ssk	Sku
PC1	-0,52	-0,52	-0,42	-0,52	0,10	-0,02
PC2	0,15	0,16	-0,33	0,03	0,51	0,76
PC3	-0,25	-0,20	0,48	0,22	0,75	-0,22
PC4	0,35	0,39	-0,37	-0,34	0,39	-0,56
PC5	-0,03	0,28	0,58	-0,73	-0,02	0,24
PC6	-0,72	0,66	-0,13	0,17	-0,04	-0,03

Fonte: Autor (2021).

Figura 13 - Influência de cada atributo nos componentes principais



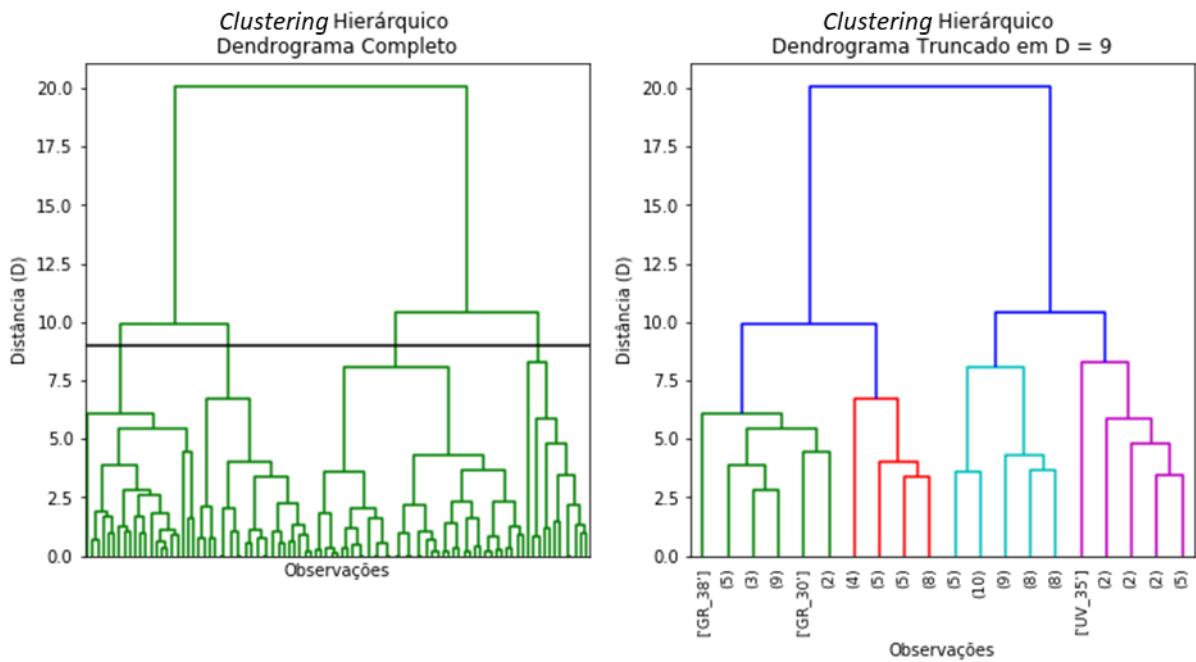
Fonte: Autor (2021).

4.4 Clusterização Hierárquica

Enquanto a análise dos componentes principais objetiva a redução de dimensionalidade para possibilitar a plotagem do gráfico de dispersão em 2 e 3 dimensões, a clusterização é o método que irá efetivamente agrupar as observações, baseado em medidas euclidianas de distância, considerando-se os valores normalizados resultantes do algoritmo de PCA.

Uma forma bastante direta de visualizar como a agregação acontece é por meio do dendrograma. O dendrograma traz uma visualização de “árvore” em que cada nó representa um agrupamento. No limite inferior há todas as “folhas”, que são as observações em si, também chamadas de *clusters* unitários. A partir dos *clusters* unitários são calculados quais estão mais próximos e agrupam-se, definindo um novo *cluster*. Este processo é feito iterativamente.

Figura 14 - Dendrograma do *clustering* hierárquico com corte em $D = 9$



Fonte: Autor (2021).

No limite das iterações forma-se apenas um grande *cluster*. Do dendrograma, pode-se definir uma linha de corte a partir de um valor de distância, representado na Figura 14 pela linha preta horizontal. O número de linhas verticais que são cortadas pela linha preta define quantos *clusters* são obtidos, no caso acima, definindo-se a Distância = 9, resulta 4 *clusters*, destacados à direita. Em parênteses, tem-se quantas observações fazem parte daquela linha e, no caso de apresentar apenas uma observação, leva o índice que a identifica. A Tabela 7 apresenta a distribuição da quantidade de observações nos *clusters*.

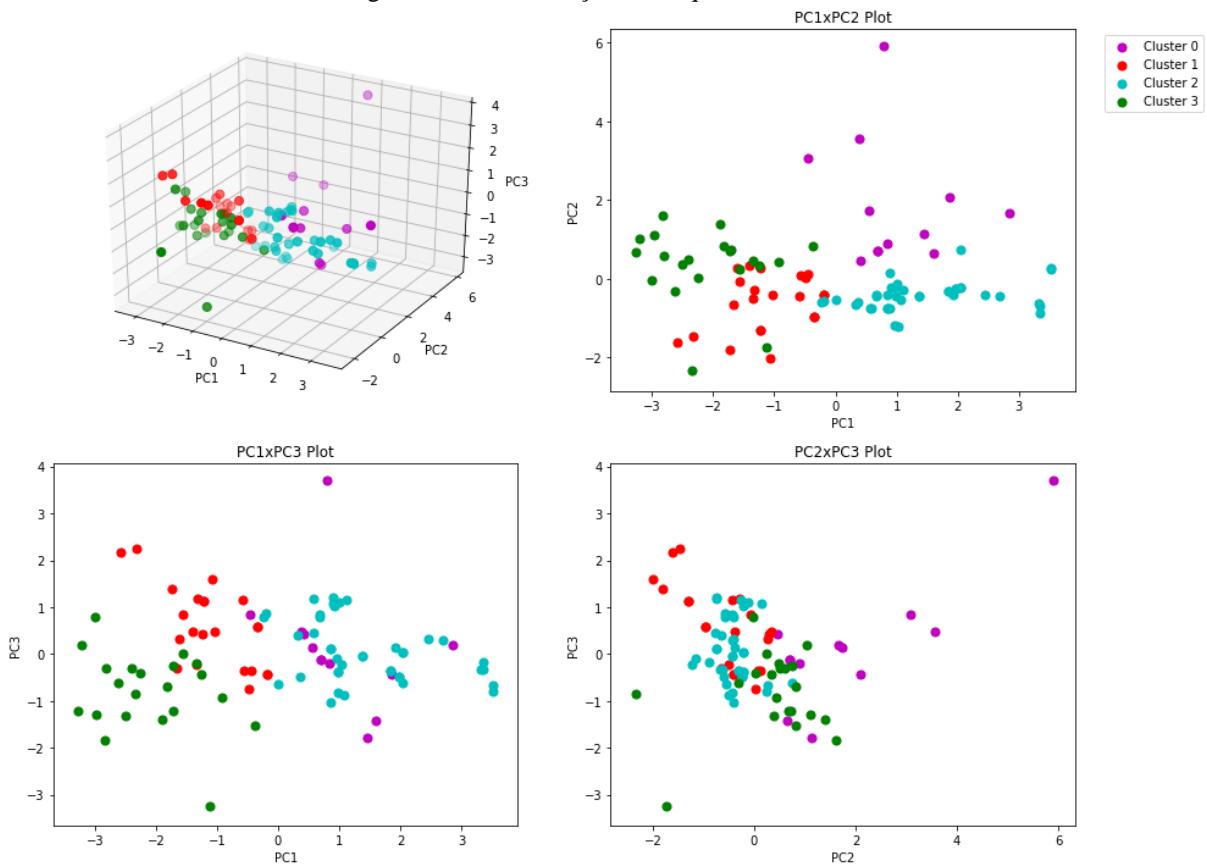
Tabela 7 - Distribuição das observações nos *clusters* - 4 *clusters*

Cluster	CP	GR-LIPSS	UV-LIPSS
Magenta	0	2	10
Vermelho	0	20	2
Ciano	5	2	33
Verde	0	21	0

Fonte: Autor (2021).

Definidos a quais *clusters* as observações pertencem, pode-se plotá-los em 2 e 3 dimensões no gráfico de dispersão obtido com a análise de componentes principais.

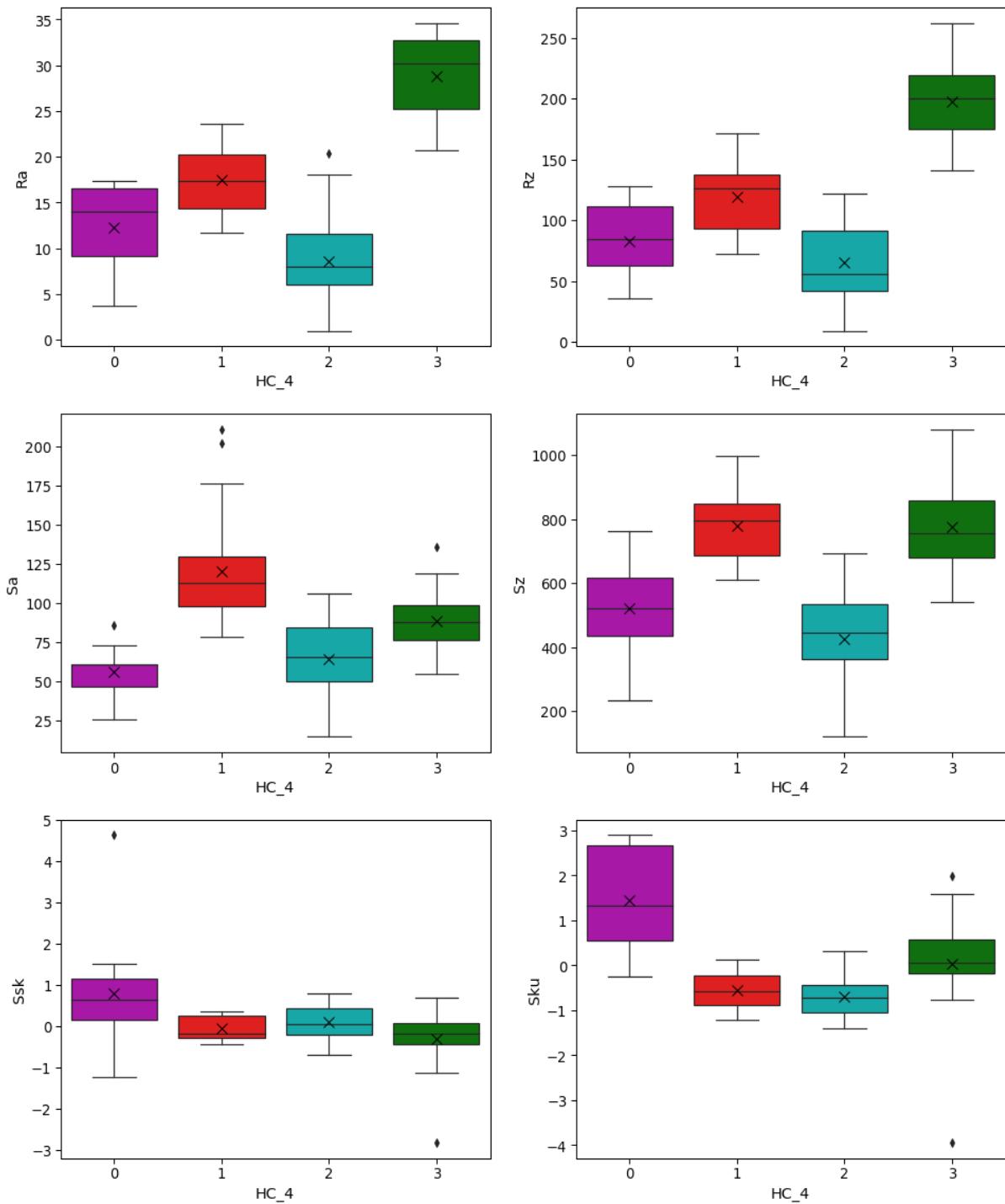
Figura 15 - Clusterização hierárquica - 4 clusters



Fonte: Autor (2021).

Cada observação pertence a um dos 4 *clusters* e a partir das projeções dos valores nos componentes principais destaca-se visualmente a qual *cluster* cada observação pertence. No gráfico de dispersão em duas dimensões que considera o PC1 x PC2, pode-se observar que o primeiro componente principal separa 3 faixas de *clusters*: o verde, o vermelho, e o par ciano/magenta que apresentam valores de PC1 bastante similares. Estes dois últimos são separados pelo segundo componente principal. Sabendo-se que os parâmetros que mais influenciam na geração do PC1 são R_a , R_z , S_a , S_z , conclui-se que o *cluster* verde, o vermelho e o par ciano/magenta possuem distribuições que são claramente diferenciadas em uma combinação dos 4 parâmetros. Já em termos do segundo componente principal, cujos parâmetros S_{sk} e S_{ku} possuem maior influência, o *cluster* magenta se destaca dos demais. Na Figura 16 é apresentado um *boxplot* dos *clusters* encontrados. Aqui pode-se verificar e validar as análises acima feitas.

Figura 16 - Boxplots dos 6 parâmetros nos clusters - 4 clusters



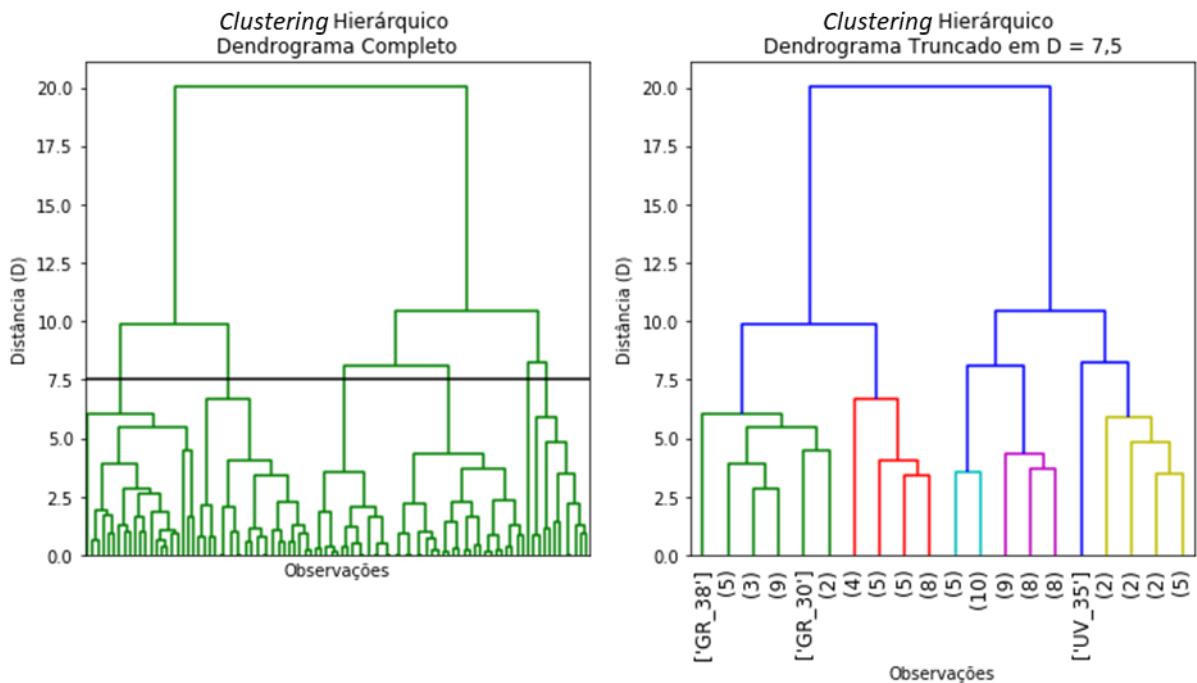
Fonte: Autor (2021).

Considerando a influência de cada atributo na geração dos componentes principais da Figura 13, verifica-se que os atributos que definem o cluster verde, o vermelho e o ciano/magenta em termos PC1 são realmente as medidas de rugosidade R_a , R_z , S_a , S_z . Para R_a e R_z , a distribuição das medidas do cluster vermelho fica entre as distribuições do cluster verde e ciano, enquanto para esses atributos o ciano e o magenta apresentam elevada intersecção. Em

relação a intersecção das distribuições em S_a e S_z do *cluster* verde e do vermelho, pode-se interpretar que representam a parte desses *clusters* que se interseccionam. O que separa o *cluster* magenta dos outros é o PC2, cujas maiores influências são os atributos S_{sk} e S_{ku} . No *boxplot* fica evidente que a distribuição dos valores das observações do *cluster* magenta se destaca para os atributos que definem o segundo componente principal.

Para efeitos de estudo, apresenta-se a clusterização com 6 *clusters*, escolhendo-se um corte em $D = 7,5$ no dendrograma, como apresentado na Figura 17.

Figura 17 - Dendrograma do *clustering* hierárquico com corte em $D = 7,5$



Fonte: Autor (2021).

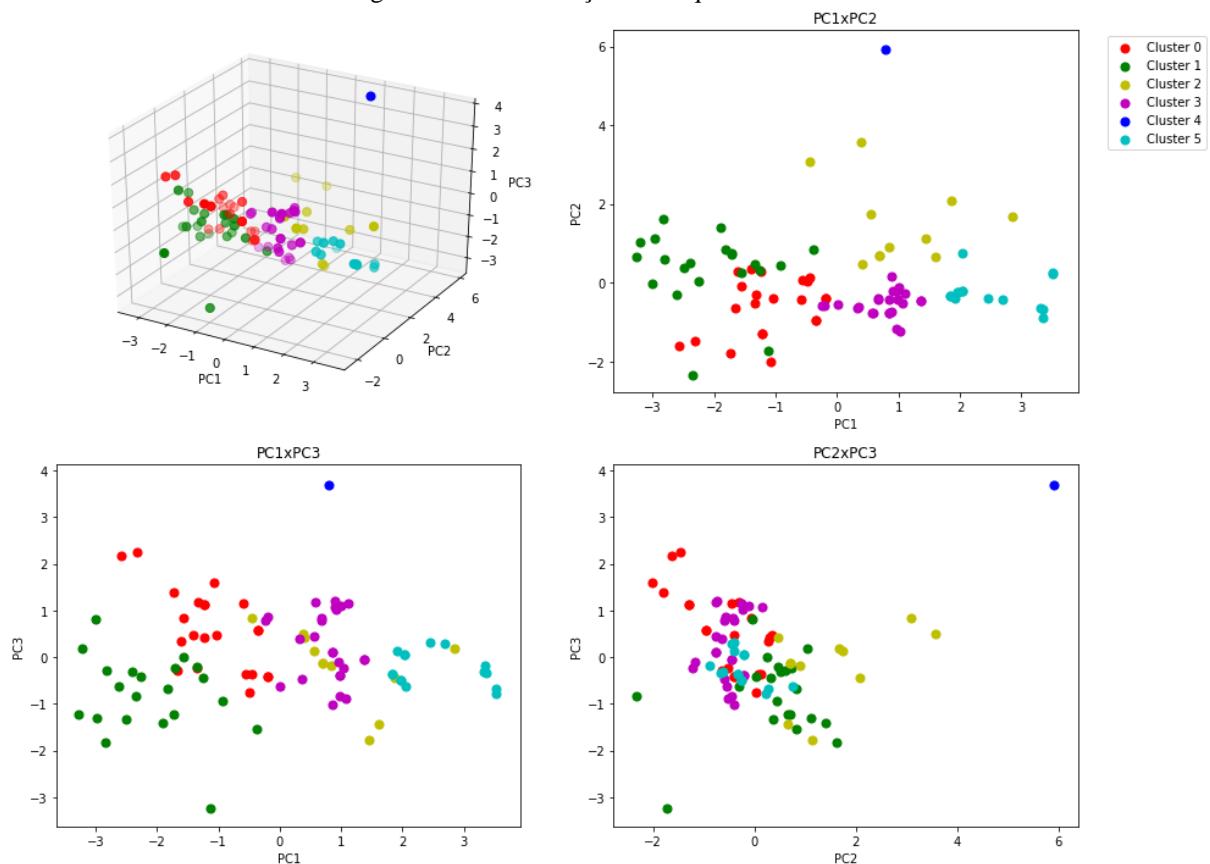
Com o corte em $D = 7,5$ observa-se que não houve nenhuma alteração no *cluster* vermelho nem no verde. O *cluster* magenta e ciano na primeira clusterização foram divididos em 2 cada. A Tabela 8 apresenta a distribuição da quantidade de observações nos *clusters* e o resultado pode ser visto no gráfico de dispersão da Figura 18.

Tabela 8 - Distribuição das observações nos *clusters* - 6 clusters

Cluster	CP	GR-LIPSS	UV-LIPSS
Vermelho	0	20	2
Verde	0	21	0
Amarelo	0	2	9
Magenta	0	2	23
Azul	0	0	1
Ciano	5	0	10

Fonte: Autor (2021).

Figura 18 - Clusterização hierárquica - 6 clusters

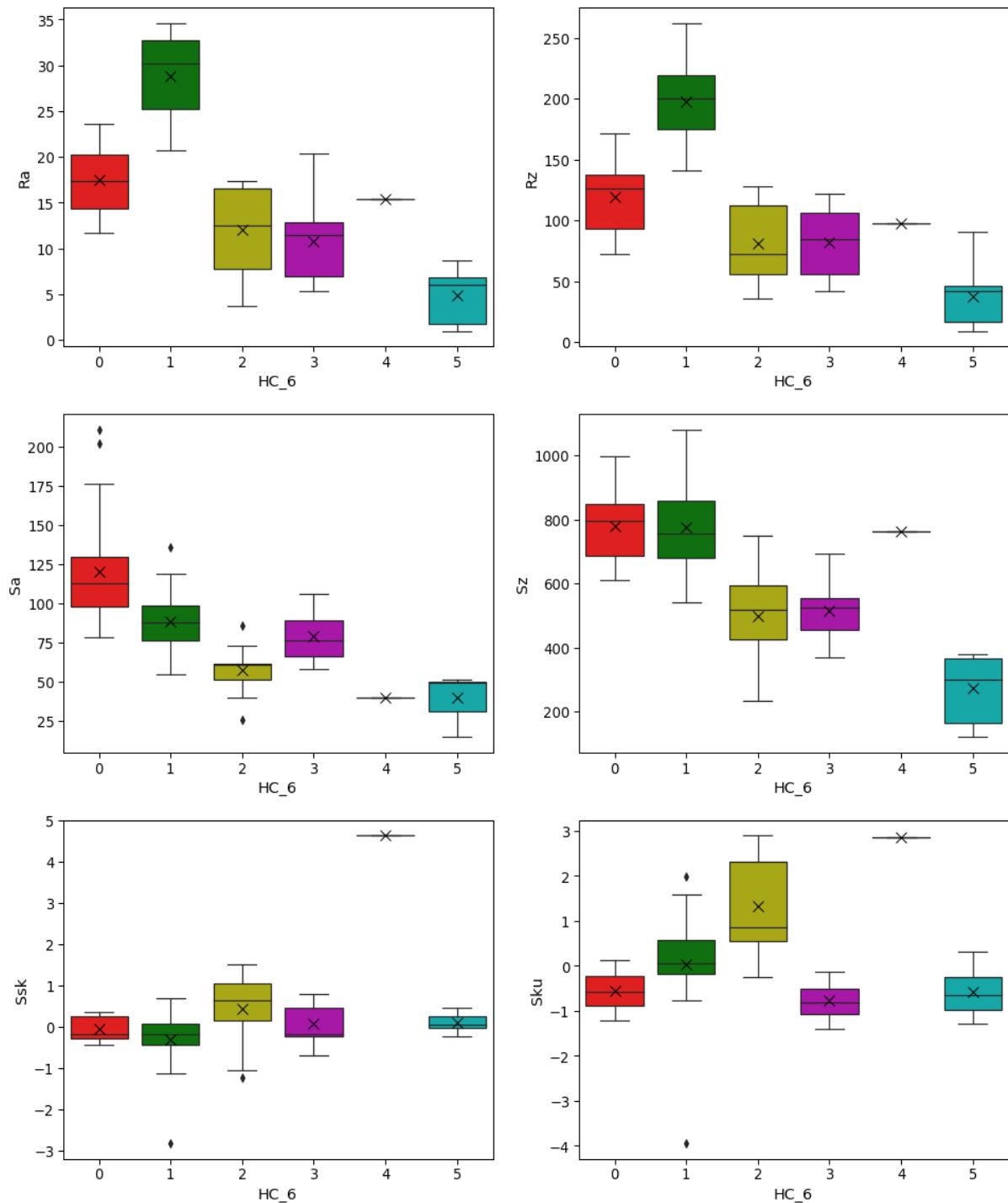


Fonte: Autor (2021).

A clusterização com 6 *clusters* apresenta resultados interessantes. Primeiramente, percebe-se que há um *cluster* unitário, com uma única observação que não se agrupa com nenhuma outra nessa faixa de corte de distância. Esse é um exemplo de um *outlier* que a clusterização permitiu identificar. A partir dessa observação pode-se decidir excluir esse corpo de prova do estudo, dado que potencialmente possa enviesar os resultados da pesquisa. Também observa-se que *cluster* ciano na clusterização com 4 *clusters* se quebrou em dois, um ciano e

um magenta, muito bem definidos, não apresentando nenhuma intersecção em termos de PC1. A Figura 19 apresenta os *boxplots* dos parâmetros para a clusterização com 6 *clusters*.

Figura 19 - Boxplots dos 6 parâmetros nos *clusters* - 6 *clusters*



Fonte: Autor (2021).

Não houve nenhuma alteração no *cluster* verde nem no vermelho considerando a cluterização com 4 *clusters*, portanto a análise é a mesma da apresentada anteriormente. O

cluster ciano e o magenta se diferenciam em termos de PC1, podendo-se observar claramente a distinção nas medidas de rugosidade R_a , R_z , S_a , S_z . A única diferença do cluster amarelo na clusterização com 6 *clusters* para o magenta na clusterização de 4 *clusters* foi o destaque do *outlier* azul. Verifica-se que em ambas clusterizações no PC3 não houve diferenciações relevantes para a análise, concluindo-se que dois componentes principais seriam o bastante para a obtenção da compreensão.

5 CONCLUSÃO

Com a realização deste trabalho foi possível apresentar um estudo de caso de aplicação de algoritmo de clusterização hierárquica em um contexto de desenvolvimento de superfícies nanotexturizadas por laser em implantes dentários, evidenciando uma nova perspectiva de análise quantitativa e qualitativa de medidas de rugosidade, de forma complementar às análises estatísticas tradicionais.

A nova perspectiva rompe com as limitações apresentadas pelos métodos de histograma e *boxplot*, não mais limitando a análise dentro de categorias, mas permitindo uma visão holística baseada puramente nos valores medidos e em suas características estatísticas inerentes.

A análise de componentes principais, enquanto método de redução de dimensionalidade, permite uma análise visual de dados deixando claro as possíveis perdas de informação e influências que cada parâmetro considerado possui na geração de novos eixos.

A clusterização hierárquica, enquanto método não-supervisionado de aprendizado de máquina, cujo objetivo não remete à busca de uma variável resposta, mas à descoberta de padrões de dados anteriormente não identificados e sua representação visual, possibilitando novos esclarecimentos, obtém sucesso e permite um aprofundamento das análises.

Destaca-se a importância do aprendizado e uso das ferramentas de programação, poderosas aliadas do engenheiro contemporâneo, bem como da premissa do conhecimento técnico acerca dos objetos de estudo, nesse caso, medidas de rugosidade de superfície.

Ressalta-se o dominante uso de técnicas de aprendizado de máquina em contextos de *Big Data* e do incipiente, mas promissor, uso dessas técnicas em contextos de desenvolvimento de tecnologia. Apesar da limitação de quantidade de dados a ser considerada, inerente a tal contexto, o estudo e entendimento da matemática por trás dos algoritmos de aprendizado de máquina aliado ao conhecimento técnico do objeto de estudo e ao uso de ferramentas de programação permitiu a execução e análise dos dados com êxito. Encontra-se um terreno fértil de inúmeras possibilidades de uso nos mais diversos contextos.

Em razão do exposto, pode-se afirmar que o objetivo principal deste trabalho foi alcançado, sendo concretizada a apresentação de aplicação de clusterização hierárquica em medidas de rugosidade de superfícies nanotexturizadas em implantes dentários.

Como sugestão para futuro aprofundamento do trabalho recomenda-se a inclusão das medidas biológicas, resumidas em um componente principal, no lugar do terceiro componente principal. Assim, pode-se discriminar os *clusters* de acordo com o resultado biológico de suas observações.

REFERÊNCIAS

- ALGRAFFEE, H.; BORUMANDI, F.; CASCARINI, L. Peri-implantitis. **British Journal of Oral and Maxillofacial Surgery**, v. 50, n. 8, p. 689-694, 2012.
- ALLA, R. K. et al. Surface roughness of implants: a review. **Trends in Biomaterials and Artificial Organs**, v. 25, n. 3, p. 112-118, 2011.
- BAUER, S. et al. Engineering biocompatible implant surfaces: Part I: Materials and surfaces. **Progress in Materials Science**, v. 58, n. 3, p. 261-326, 2013.
- BEHR, A. et al. Gestão da biblioteca escolar: metodologias, enfoques e aplicação de ferramentas de gestão e serviços de biblioteca. **Ciência da Informação**. vol. 37 nº 2, Brasília Mar./Ago. 2008
- BERNARDES, S. R. et al. Análise fotoelástica da união de pilar a implantes de hexágonos externo e interno. **Implant News**, v. 3, n. 4, p. 355-59, 2006.
- BINON, P. et al. Implant component compatibility. **Tissue integration in oral, orthopedic and maxillofacial reconstruction**. Chicago: Quintessence Publishing, p. 218-226, 1992.
- BUSSCHER, H. J. et al. Biofilm formation on dental restorative and implant materials. **Journal of dental research**, v. 89, n. 7, p. 657-665, 2010.
- CORDEIRO, V. A. Peri-implantite: Causas, Consequências e Soluções. Uma Revisão de Literatura. Curso de Especialização em Periodontia. **Escola Bahiana de Medicina e Saúde Pública**, 2009.
- COSTA FILHO, S. V. S. da. et al. Configuração de algoritmos de aprendizado de máquina na modelagem florestal: um estudo de caso na modelagem da relação hipsométrica. **Ciência Florestal**, Universidade Federal de Santa Maria, v. 29, n. 4, p. 1501-1515, out./dez. 2019
- DOMINGUES, M.; DOMINGUES, J. Estatística Exploratória. Universidade Federal Rural de Pernambuco (**UFRPE**), vol.1, 2010.
- DOMÍNGUEZ, J. et al. Estudio de asociación entre enfermedades sistémicas y el fracaso de implantes dentales. **Revista clínica de periodoncia, implantología y rehabilitación oral**, v. 6, n. 1, p. 9-13, 2013.
- FERNANDES, F. T. et al. Perspectivas do uso de mineração de dados e aprendizado de máquina em saúde e segurança no trabalho. **Revista Brasileira de Saúde Ocupacional**, Universidade de São Paulo, São Paulo, 2019.
- FÜRST, M. M. et al. Bacterial colonization immediately after installation on oral titanium implants. **Clinical oral implants research**, v. 18, n. 4, p. 501-508, 2007.
- INAZAWA, P. et al. Projeto Victor - Como o Uso do Aprendizado de Máquina Pode Auxiliar a Mais Alta Corte Brasileira a Aumentar a Eficiência e a Velocidade de Avaliação Judicial dos Processos Julgados. **Sociedade Brasileira de Computação**, 2019.

- JAMES, G. et al. **An Introduction to Statistical Learning**. Nova Iorque: Springer, 2013.
- JEMAT, A. et al. Surface modifications and their effects on titanium dental implants. **BioMed research international**, v. 2015, 2015.
- LEE, A.; WANG, H.-L. Biofilm related to dental implants. **Implant dentistry**, v. 19, n. 5, p. 387-393, 2010.
- LI, J. et al. Identification of early microbial colonizers in human dental biofilm. **Journal of applied microbiology**, v. 97, n. 6, p. 1311-1318, 2004.
- MARTINS, V. et al. Osseointegração: análise de fatores clínicos de sucesso e insucesso. **Revista Odontológica de Araçatuba**, v. 32, n. 1, p. 26-31, 2011.
- MATEUS, F. M. Q. et al. Machine Learning na Melhoria de Processos Internos: Estudos de Caso na Indústria de Varejo Brasileira. Engenharia de Produção da Escola Politécnica, **Universidade Federal do Rio de Janeiro**, Rio de Janeiro, 2020.
- MENASSA, M. et al. Patients' expectations, satisfaction, and quality of life with immediate loading protocol. **Clinical oral implants research**, v. 27, n. 1, p. 83-89, 2016.
- MOMBELLI, A.; MÜLLER, N.; CIONCA, N. The epidemiology of peri-implantitis. **Clinical oral implants research**, v. 23, p. 67-76, 2012.
- NOGUEIRA-FILHO, G.; IACOPINO, A. M.; TENENBAUM, H. C. Prognosis in implant dentistry: a system for classifying the degree of peri-implant mucosal inflammation. **Journal of the Canadian Dental Association**, v. 77, n. 1, p. 25, 2011.
- NOVAES JR, A. Belém et al. Influence of implant surfaces on osseointegration. **Brazilian dental journal**, v. 21, n. 6, p. 471-481, 2010.
- OLIVEIRA, M. C. de et al. Peri-implantite: etiologia e tratamento. **Revista Brasileira de Odontologia**, v. 72, n. 1-2, p. 96-99, 2015.
- RAMS, T. E. Microgibiology of failing dental implants in humans. **J. Oral Implantol.**, v. 11, p. 93-100, 1983.
- RC ODONTOLOGIA. **Implantes Dentários**. Disponível em: <https://www.rcodontologia.com.br/implantodontia.html>. Acesso em 22 nov 2020.
- SAKKA, S.; BAROUDI, K.; NASSANI, M. Z. Factors associated with early and late failure of dental implants. **Journal of investigative and clinical dentistry**, v. 3, n. 4, p. 258-261, 2012.
- SCHWARZ, F. et al. Peri-implantitis. **Journal of clinical periodontology**, v. 45, p. S246-S266, 2018.
- SILVA, D. A. da. Aplicação de técnicas de pré-processamento e agrupamento na base de dados de benefícios previdenciários do Ministério Público do Trabalho. Faculdade de Computação da **Universidade Federal de Uberlândia**, Minas Gerais, 2018.

TEIXEIRA, E. R. Implantes dentarios en rehabilitación oral contemporánea. **Actualidades Medico Odontológicas Latinoamérica (AMOLCA)**, 2010.

UHLMANN, E. et al. Application of laser surface nanotexturing for the reduction of peri-implantitis on biomedical grade 5 Ti-6Al-4V dental abutments. **LASE**, 2019.

WENNERBERG, A.; ALBREKTSSON, T. Suggested guidelines for the topographic evaluation of implant surfaces. **International Journal of Oral & Maxillofacial Implants**, v. 15, n. 3, 2000.

XUEREB, M.; CAMILLERI, J.; ATTARD, N. J. Systematic review of current dental implant coating materials and novel coating techniques. **International Journal of Prosthodontics**, v. 28, n. 1, p. 51-59, 2015.

Apêndice A – Código Python em Jupyter Notebook

Importação de Bibliotecas

In []:

```
# pip install adjustText
# !pip install nbconvert
# !pip install seaborn==0.11.0
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from adjustText import adjust_text
from mpl_toolkits.mplot3d import axes3d
from matplotlib.ticker import AutoMinorLocator
# import matplotlib.colors as mcolors
%matplotlib inline
```

Importação de Planilha Excel

In []:

```
diretorio = "C:\\\\Diretório\\\\"
arquivo = "Excel.xlsx"
df = pd.read_excel(diretorio + arquivo, header=0)
```

Validação

In []:

```
round(df, ndigits=3).head()
# atributos = df.columns[2:]
# atributos
```

Exemplo

In []:

```
df_CP = df[0:5]
df_UV = df[5:50]
df_GR = df[50:]
df_ex = df_CP[0:3].append(df_UV[0:3])
df_ex = df_ex.append(df_GR[0:3])
round(df_ex, ndigits=2)
```

Descrição do Dataset

In []:

```
# dataset
# dataset.shape
# dataset.head(5)
# dataset.tail(5)
# dataset[dataset['Categoria']=='UV-LIPSS'].describe()
# df['Categoria'].groupby('Categoria').describe()

# round(df[df['Categoria']=='UV-LIPSS'].describe(), ndigits = 3)
df_descr = round(df.groupby('Categoria').describe(), ndigits = 3)
# df_descr
df_descr.T
# df_descr
```

Declaração de variáveis

atributos: parâmetros de mensuração

categorias: categorias dos corpos de prova

In []:

```
# atributos = ['Ra', 'Rz', 'Sa', 'Sz', 'Ssk', 'Sku']
atributos = df.columns[2:]
categorias = df['Categoria'].unique()

# atributos
# categorias
```

Histograma

Ra nas 3 Categorias

```
from matplotlib.ticker import AutoMinorLocator

fig = plt.figure(figsize=(16,20), tight_layout=True)

# fig.tight_layout(h_pad=10, w_pad=50)
k=1
colors = ['b', 'm', 'g']
caixas = [2,6,6]
for j in range(0,1):

    for i in range(0,3):
        ax = fig.add_subplot(6,3,k)
        # fig, ax = plt.subplots(2,1,figsize=(12, 6), tight_layout = True)
        n, bins, patches = plt.hist(round(df[df['Categoria']==categorias[i]].loc[:,atributos[j]]), ndigits=2), color = colors[i], bins='rice'#, bins=caixas[i])

        # define minor ticks and draw a grid with them
        minor_locator = AutoMinorLocator(2)
        plt.gca().xaxis.set_minor_locator(minor_locator)
        plt.grid(which='minor', color='white', lw = 0.5)
```

```

# x ticks
xticks = [(bins[idx+1] + value)/2 for idx, value in enumerate(bins[:-1])]
xticks_labels = [ "{:.2f}\n{:2f}".format(value, bins[idx+1]) for idx, value
in enumerate(bins[:-1])]
plt.xticks(xticks, labels = xticks_labels)
# remove major and minor ticks from the x axis, but keep the labels
ax.tick_params(axis='x', which='both', length=0)

# remove y ticks
plt.yticks([])

# Hide the right and top spines
ax.spines['bottom'].set_visible(False)
ax.spines['left'].set_visible(False)
ax.spines['right'].set_visible(False)
ax.spines['top'].set_visible(False)

for idx, value in enumerate(n):
    plt.text(xticks[idx], value+0.05, int(value), ha='center')

plt.title('Categoria: {} - Parâmetro: {}'.format(atributos[j], categorias[i
]), loc='center', fontsize=12)

k=k+1
k=k+1

```

Boxplot

```

my_pal = {"CP": "b", "UV-LIPSS": "m", "GR-LIPSS": "g"}

fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(16, 6))

i=0 # Ra e Rz
# i=2 # Sa e Sz
# i=4 # Ssk e Sku

sns.set_context("paper", font_scale=1.5,
#                 rc={"font.size":9,"axes.titlesize":10,"axes.labelsize":5}
                 )
sns.boxplot(ax = axes[0], x=df["Categoria"], y=df[atributos[i]]
            , palette=my_pal, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":10})

sns.boxplot(ax = axes[1], x=df["Categoria"], y=df[atributos[i+1]]
            , palette=my_pal, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":10})

```

Normalização

Declaração de variável

obs_index: index das observações
x: valores dos parâmetros de mensuração

In []:

```
obs_index_df = df.loc[:, ['ID']]
# obs_index_df
obs_index = pd.DataFrame.to_numpy(obs_index_df, dtype=str)
# obs_index
x = df.loc[:, atributos].values
```

Normalização

In []:

```
from sklearn.preprocessing import StandardScaler

x_norm = StandardScaler().fit_transform(x)

df_xnorm = pd.DataFrame(x_norm, columns = atributos, index = obs_index)

round(df_xnorm, ndigits=3).head()
# round(df_xnorm.describe(), ndigits=2)
```

Exemplo

```
df_xnorm_CP = df_xnorm[0:5]
df_xnorm_UV = df_xnorm[5:50]
df_xnorm_GR = df_xnorm[50:]
df_xnorm_ex = df_xnorm_CP[0:3].append(df_xnorm_UV[0:3])
df_xnorm_ex = df_xnorm_ex.append(df_xnorm_GR[0:3])
round(df_xnorm_ex, ndigits=2)
```

Análise de Componentes Principais (Principal Component Analysis)

Declaração de Variáveis

pca_index: Index dos Principal Components

In []:

```
# n_components = 6
pca_index = ['PC1', 'PC2', 'PC3', 'PC4', 'PC5', 'PC6']
# pca_index[:n_components]
pca_index
```

Aplicação código de Principal Component Analysis

In []:

```
from sklearn.decomposition import PCA

n_components = 6
pca = PCA(n_components)

comp_princ = pca.fit_transform(x_norm)

df_pca = pd.DataFrame(data = comp_princ, columns = pca_index, index = obs_index)

round(df_pca.head(), ndigits=3)
```

Exemplo da Projeção dos valores nos PC's

In []:

```
df_pca_CP = df_pca[0:5]
df_pca_UV = df_pca[5:50]
df_pca_GR = df_pca[50:]
df_pca_ex = df_pca_CP[0:3].append(df_pca_UV[0:3])
df_pca_ex = df_pca_ex.append(df_pca_GR[0:3])
round(df_pca_ex, ndigits=2)
```

Barplot Proporção de Variância Explicado

```
explained_variance = pca.explained_variance_ratio_
cummulative_expl_var = 0
#for i in range(len(explained_variance)):
for i in range(n_components):
    e = explained_variance[i]
    cummulative_expl_var = cummulative_expl_var + e

fig = plt.figure(figsize=(8,5))
ax = fig.add_axes([0,0,1,1])
# ax.plot(pca_index, np.cumsum(pca.explained_variance_ratio_), c = 'r')
barplot = ax.bar(pca_index, pca.explained_variance_ratio_)

# remove y ticks
plt.yticks([])

# Hide the right and top spines
ax.spines['bottom'].set_visible(False)
ax.spines['left'].set_visible(False)
ax.spines['right'].set_visible(False)
ax.spines['top'].set_visible(False)

for idx, rect in enumerate(barplot):
    height = rect.get_height()
    ax.text(rect.get_x() + rect.get_width()/2, 1.03*height
            , np.round(pca.explained_variance_ratio_*100,1)[idx], ha='center')
```

Scatter Plot 3D

In []:

```
fig = plt.figure(figsize=(16,12))
ax = fig.add_subplot(111, projection = '3d')
ax.scatter(comp_princ[:, 0], comp_princ[:, 1], comp_princ[:, 2], s=100)

ax.set_xlabel('PC1')
ax.set_ylabel('PC2')
ax.set_zlabel('PC3')
```

Scatter Plot 3D + 2D Plots

```
fig = plt.figure(figsize=(15,12))
# fig.suptitle('Seu Título Aqui')

ax = fig.add_subplot(221, projection = '3d')
ax.scatter(comp_princ[:, 0], comp_princ[:, 1], comp_princ[:, 2], s=50)
ax.set_xlabel('PC1')
ax.set_ylabel('PC2')
ax.set_zlabel('PC3')

ax = fig.add_subplot(222)
ax.scatter(comp_princ[:, 0], comp_princ[:, 1], s=50)
ax.set_xlabel('PC1')
ax.set_ylabel('PC2')

ax = fig.add_subplot(223)
ax.scatter(comp_princ[:, 0], comp_princ[:, 2], s=50)
ax.set_xlabel('PC1')
ax.set_ylabel('PC3')

ax = fig.add_subplot(224)
ax.scatter(comp_princ[:, 1], comp_princ[:, 2], s=50)
ax.set_xlabel('PC2')
ax.set_ylabel('PC3')
```

Influência de cada atributo nos Componentes Principais

In []:

```

fig, ax = plt.subplots(figsize=(6, 4))

# ax = plt.subplots(311)
offset = 0.1
limites = np.min(pca.components_.T[:,1])-offset, np.max(pca.components_.T[:,1])+offset,
np.min(pca.components_.T[:,2])-offset, np.max(pca.components_.T[:,2]+offset)
limites = np.round(limites, 3)

ax.axis(limites)
colors = ['r', 'g', 'b', 'c', 'm', 'y', 'k']

for i in range (0, n_components):
    ax.annotate('',
        , xy=(pca.components_.T[[i],1],pca.components_.T[[i],2])
        , xytext=(0,0)
        , arrowprops=dict(arrowstyle='->'
                         , linewidth=0.7
                         , color = colors[i]
                         )
    )

texts = [plt.text(pca.components_.T[[i],1]-0.03,pca.components_.T[[i],2], atributos[i],
color=colors[i]) for i in range (0, n_components)]
adjust_text(texts)

ax.set_xlabel('PC2') # Add an x-label to the axes.
ax.set_ylabel('PC3') # Add a y-label to the axes.

```

Clusterização

Dendrograma - 4 e 6 Clusters

```

import scipy.cluster.hierarchy as sch

cut_line = 9 # 4 clusters
# cut_line = 7.5 # 6 clusters

fig = plt.figure(figsize=(12,12))

# ax = fig.add_subplot(221)
# dendrogram = sch.dendrogram(sch.linkage(comp_princ[:,3], method = 'ward'))#, labels =
Labels)

ax = fig.add_subplot(221)
sch.dendrogram(sch.linkage(comp_princ, method = 'ward')
#                 , truncate_mode='lastp'
#                 , p=20
#                 , show_Leaf_counts=False
#                 , show_contracted=True
#                 , Leaf_rotation=90.
#                 , Leaf_font_size=12.
#                 , color_threshold=21
#                 , no_labels=True)
plt.title('Clustering Hierárquico\nDendrograma Completo')

```

```

plt.xlabel('Observações')
plt.ylabel('Distância (D)')
plt.axhline(y=cut_line, color='k', linestyle='--')

ax = fig.add_subplot(222)
sch.dendrogram(sch.linkage(comp_princ, method = 'ward')
               , truncate_mode='lastp'
               , p=20
#               , show_leaf_counts=False
#               , show_contracted=True
               , leaf_rotation=90.
               , leaf_font_size=9
               , color_threshold=cut_line
               , labels = obs_index)
plt.title('Clustering Hierárquico\nDendrograma Truncado em D = {0}'.format(cut_line))
plt.xlabel('Observações', fontsize = 10)
plt.ylabel('Distância (D)')

plt.show()

```

Últimos n agrupamentos

In []:

```

import scipy.cluster.hierarchy as sch
from scipy.spatial.distance import pdist
np.set_printoptions(precision=5, suppress=True)
Z = sch.linkage(comp_princ, method='ward')
n = 20
# c = sch.cophenet(Z, pdist(comp_princ))
# c

Z[-n:, 2]

```

Aplicação de código de clusterização

In []:

```

from sklearn.cluster import AgglomerativeClustering
# n_clusters = 6

hc_4 = AgglomerativeClustering(4, affinity = 'euclidean', linkage = 'ward')
y_hc_4 = hc_4.fit_predict(comp_princ)
y_hc_4
# comp_princ
# df

# hc_6 = AgglomerativeClustering(6, affinity = 'euclidean', linkage = 'ward')
# y_hc_6 = hc_6.fit_predict(comp_princ)
# y_hc_6

```

Scatter Plot com Detalhes dos Clusters (4 Clusters)

In []:

```

fig = plt.figure(figsize=(15,12))
fig.tight_layout(pad=5)

# 4 Clusters
n_clusters = 4
y_hc = y_hc_4
# dic_HC_4 = {0: "m", 1: "r", 2: "c", 3: "g",}
colors_hc_4 = ['m', 'r', 'c', 'g']

# # # 6 Clusters
# n_clusters = 6
# y_hc = y_hc_6
# colors = ['r', 'g', 'y', 'm', 'b', 'c']
# fig.suptitle('Clustering Hierárquico - ' + str(n_clusters) + ' Clusters')

ax = fig.add_subplot(221, projection = '3d')
for i in range(0, n_clusters):
    ax.scatter(comp_princ[y_hc == i, 0]
               , comp_princ[y_hc == i, 1]
               , comp_princ[y_hc == i, 2]
               , s = 50, c = colors_hc_4[i]
               , label = 'Cluster ' + str(i+1))
#ax.set_title('Tridimensional Plot PC1xPC2xPC3')
ax.set_xlabel('PC1')
ax.set_ylabel('PC2')
ax.set_zlabel('PC3')
ax.legend(bbox_to_anchor=(2.25, 1), loc='upper left')

#     for i in range(len(comp_princ)):
#         ax.text(comp_princ[i,0],comp_princ[i,1], comp_princ[i,2]
#                 , '%s' % (str(i+1)), size=10, color='k')

ax = fig.add_subplot(222)
for i in range (0, n_clusters):
    ax.scatter(comp_princ[y_hc==i, 0]
               , comp_princ[y_hc==i, 1]
               , s = 50
               , c = colors_hc_4[i])
ax.set_title('PC1xPC2 Plot')
ax.set_xlabel('PC1') # Add an x-label to the axes.
ax.set_ylabel('PC2') # Add a y-label to the axes.

#     for i in range(len(comp_princ)):
#         ax.text(comp_princ[i,0],comp_princ[i,1], '%s' % (str(i+1))
#                 , size=10, color='k', ha = 'center', va = 'center')

ax = fig.add_subplot(223)
for i in range (0, n_clusters):
    ax.scatter(comp_princ[y_hc==i, 0]
               , comp_princ[y_hc==i, 2]
               , s = 50
               , c = colors_hc_4[i])
ax.set_title('PC1xPC3 Plot')
ax.set_xlabel('PC1') # Add an x-label to the axes.
ax.set_ylabel('PC3') # Add a y-label to the axes.

```

```

#     for i in range(len(comp_princ)):
#         ax.text(comp_princ[i,0],comp_princ[i,2], '%s' % (str(i+1))
#                 , size=10, color='k', ha = 'center', va = 'center')

ax = fig.add_subplot(224)
for i in range (0, n_clusters):
    ax.scatter(comp_princ[y_hc==i, 1]
               , comp_princ[y_hc==i, 2]
               , s = 50
               , c = colors_hc_4[i])
ax.set_title('PC2xPC3 Plot')
ax.set_xlabel('PC2') # Add an x-label to the axes.
ax.set_ylabel('PC3') # Add a y-label to the axes.

#     for i in range(len(comp_princ)):
#         ax.text(comp_princ[i,1],comp_princ[i,2], '%s' % (str(i+1))
#                 , size=8, color='k', ha = 'left', va = 'center')

```

Scatter Plot com Detalhes dos Clusters (6 Clusters)

In []:

```

fig = plt.figure(figsize=(15,12))
fig.tight_layout(pad=5)

# 4 Clusters
n_clusters = 6
y_hc = y_hc_6
# dic_HC_4 = {0: "m", 1: "r", 2: "c", 3: "g",}
colors_hc_6 = ['r', 'g', 'y', 'm', 'b', 'c']

# # # 6 Clusters
# n_clusters = 6
# y_hc = y_hc_6
# colors = ['r', 'g', 'y', 'm', 'b', 'c']
# fig.suptitle('Clustering Hierárquico - ' + str(n_clusters) + ' Clusters')

ax = fig.add_subplot(221, projection = '3d')
for i in range(0, n_clusters):
    ax.scatter(comp_princ[y_hc == i, 0]
               , comp_princ[y_hc == i, 1]
               , comp_princ[y_hc == i, 2]
               , s = 50, c = colors_hc_6[i]
               , label = 'Cluster ' + str(i+1))
#ax.set_title('Tridimensional Plot PC1xPC2xPC3')
ax.set_xlabel('PC1')
ax.set_ylabel('PC2')
ax.set_zlabel('PC3')
ax.legend(bbox_to_anchor=(2.25, 1), loc='upper left')

#     for i in range(len(comp_princ)):
#         ax.text(comp_princ[i,0],comp_princ[i,1], comp_princ[i,2]
#                 , '%s' % (str(i+1)), size=10, color='k')

```

```

ax = fig.add_subplot(222)
for i in range (0, n_clusters):
    ax.scatter(comp_princ[y_hc==i, 0]
               , comp_princ[y_hc==i, 1]
               , s = 50
               , c = colors_hc_6[i])
ax.set_title('PC1xPC2 Plot')
ax.set_xlabel('PC1') # Add an x-Label to the axes.
ax.set_ylabel('PC2') # Add a y-Label to the axes.

#     for i in range(len(comp_princ)):
#         ax.text(comp_princ[i,0],comp_princ[i,1], '%s' % (str(i+1))
#                 , size=10, color='k', ha = 'center', va = 'center')

ax = fig.add_subplot(223)
for i in range (0, n_clusters):
    ax.scatter(comp_princ[y_hc==i, 0]
               , comp_princ[y_hc==i, 2]
               , s = 50
               , c = colors_hc_6[i])
ax.set_title('PC1xPC3 Plot')
ax.set_xlabel('PC1') # Add an x-Label to the axes.
ax.set_ylabel('PC3') # Add a y-Label to the axes.

#     for i in range(len(comp_princ)):
#         ax.text(comp_princ[i,0],comp_princ[i,2], '%s' % (str(i+1))
#                 , size=10, color='k', ha = 'center', va = 'center')

##### Scatter Plot com Detalhes dos Clusters (4 e 6 Clusters)
ax = fig.add_subplot(224)
for i in range (0, n_clusters):
    ax.scatter(comp_princ[y_hc==i, 1]
               , comp_princ[y_hc==i, 2]
               , s = 50
               , c = colors_hc_6[i])
ax.set_title('PC2xPC3 Plot')
ax.set_xlabel('PC2') # Add an x-Label to the axes.
ax.set_ylabel('PC3') # Add a y-Label to the axes.

#     for i in range(len(comp_princ)):
#         ax.text(comp_princ[i,1],comp_princ[i,2], '%s' % (str(i+1))
#                 , size=8, color='k', ha = 'left', va = 'center')

```

In []:

```

# df_clusters = np.c_[df,y_hc]
# pd.DataFrame(df_clusters)

# df_clusters = pd.concat([df, pd.DataFrame(y_hc)], axis=1)
# df_clusters

df_hc_4 = pd.DataFrame(y_hc_4, columns=[ "HC_4"])
df_hc_6 = pd.DataFrame(y_hc_6, columns=[ "HC_6"])
df_clusters = pd.concat([df, df_hc_4, df_hc_6], axis=1)

df_clusters.head()

```

BoxPlot com Detalhes dos Clusters (4 Clusters)

```

dic_HC_4 = {0: "m", 1: "r", 2: "c", 3: "g",}
fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(16, 20))

i=0 # Ra e Rz
# i=2 # Sa e Sz
# i=4 # Ssk e Sku

sns.set_context("paper", font_scale=1.5,
#                 rc={"font.size":9, "axes.titlesize":10, "axes.Labelsize":5}
                )
sns.boxplot(ax = axes[0,0], x=df_clusters["HC_4"], y=df_clustersatributos[i]
            , palette=dic_HC_4, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

sns.boxplot(ax = axes[0,1], x=df_clusters["HC_4"], y=df_clustersatributos[i+1]
            , palette=dic_HC_4, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

sns.boxplot(ax = axes[1,0], x=df_clusters["HC_4"], y=df_clustersatributos[i+2]
            , palette=dic_HC_4, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

sns.boxplot(ax = axes[1,1], x=df_clusters["HC_4"], y=df_clustersatributos[i+3]
            , palette=dic_HC_4, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

sns.boxplot(ax = axes[2,0], x=df_clusters["HC_4"], y=df_clustersatributos[i+4]
            , palette=dic_HC_4, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

sns.boxplot(ax = axes[2,1], x=df_clusters["HC_4"], y=df_clustersatributos[i+5]
            , palette=dic_HC_4, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

dic_HC_6 = {0: "r", 1: "g", 2: "y", 3: "m", 4: "b", 5: "c"}
fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(16, 20))

i=0 # Ra e Rz
# i=2 # Sa e Sz
# i=4 # Ssk e Sku

sns.set_context("paper", font_scale=1.5,
#                 rc={"font.size":9, "axes.titlesize":10, "axes.Labelsize":5}
                )
sns.boxplot(ax = axes[0,0], x=df_clusters["HC_6"], y=df_clustersatributos[i]
            , palette=dic_HC_6, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

sns.boxplot(ax = axes[0,1], x=df_clusters["HC_6"], y=df_clustersatributos[i+1]
            , palette=dic_HC_6, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

sns.boxplot(ax = axes[1,0], x=df_clusters["HC_6"], y=df_clustersatributos[i+2]
            , palette=dic_HC_6, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

sns.boxplot(ax = axes[1,1], x=df_clusters["HC_6"], y=df_clustersatributos[i+3]
            , palette=dic_HC_6, showmeans = True, meanprops={"marker":"x", "markeredgecolor":"black", "markersize":"10"})

```

```

        olor:"black", "markersize":"10"})

sns.boxplot(ax = axes[2,0], x=df_clusters["HC_6"], y=df_clustersatributos[i+4]
            , palette=dic_HC_6, showmeans = True, meanprops={"marker":"x", "markeredgecolor"
        olor:"black", "markersize":"10"})

sns.boxplot(ax = axes[2,1], x=df_clusters["HC_6"], y=df_clustersatributos[i+5]
            , palette=dic_HC_6, showmeans = True, meanprops={"marker":"x", "markeredgecolor
        olor:"black", "markersize":"10"})

```

In []:

```
pd.crosstab(df_clusters["HC_4"], df_clusters["Categoria"])
```

In []:

```
pd.crosstab(df_clusters["HC_6"], df_clusters["Categoria"])
```

In []:

```

def coding(col, codeDict):
    colCoded = pd.Series(col, copy=True)
    for key, value in codeDict.items():
        colCoded.replace(key, value, inplace=True)
    return colCoded

depara_y_hc_4 = coding(y_hc_4, {0:"Cluster 1", 1:"Cluster 2", 2:"Cluster 3", 3:"Cluster
4"})

pd.concat([pd.Series(y_hc_4),depara_y_hc_4], axis=1)

pd.value_counts(y_hc_4)

```