UNIVERSIDADE FEEVALE

IGOR VIANA DO AMARAL

APLICAÇÃO DE MACHINE LEARNING PARA BUSCA DE PADRÕES E TENDÊNCIAS NA ÁREA DA SAÚDE

Novo Hamburgo
2019

IGOR VIANA DO AMARAL

APLICAÇÃO DE *MACHINE LEARNING* PARA BUSCA DE PADRÕES E TENDÊNCIAS NA ÁREA DA SAÚDE

Trabalho de Conclusão de Curso, apresentado como requisito parcial à obtenção do grau de Bacharel em Ciência da Computação pela Universidade Feevale.

Orientador: Prof. Dr. Juliano Varella de Carvalho

Novo Hamburgo

2019

RESUMO

Quando um profissional da saúde realiza atendimento a um paciente e o diagnóstico não está evidente, são solicitados exames complementares para a confirmação de hipóteses e tratamento. As informações fornecidas por esses procedimentos são armazenadas, gerando um grande volume de dados. Essa grande massa de dados possibilita aplicar técnicas de *machine learning* (aprendizado de máquina). O aprendizado de máquina é um segmento da inteligência artificial onde os sistemas aprendem com dados, identificam padrões e tomam decisões com o mínimo de intervenção humana. O presente trabalho consiste no estudo de *machine learning*, compreensão de uma base de dados relacionada à área da saúde e em sequência a busca de padrões e tendências sobre essa base. Os dados que foram coletados são referentes a procedimentos médicos e materiais usados nos mesmos, juntamente com dados anonimizados de pacientes e suas doenças pré-existentes, fornecidos pela maior rede de assistência médica do Brasil. Com o resultado proveniente do estudo foi realizado o pré-processamento dos dados, a avaliação da técnica *Random Forest* sobre os dados e a discussão de estratégias que melhor se relacionaram aos objetivos para a busca de padrões e tendências. Por fim, os resultados foram analisados e exibidas as métricas.

Palavras-chave: Aprendizagem de máquina, padrões e tendências, saúde, procedimentos médicos.

ABSTRACT

When a healthcare professional provides care to a patient and the diagnosis is not evident, complementary examinations are required to confirm hypotheses and treatment. The information provided by these procedures is stored, generating a large amount of data. This large mass of data makes it possible to apply machine learning techniques. Machine learning is a segment of artificial intelligence where systems learn from data, identify patterns and make decisions with minimal human intervention. The present work consists in the study of machine learning, understanding of a database related to health area and following the search for patterns and trends on this basis. The data that was collected refers to medical procedures and materials used in them, along with anonymised patient data and their pre-existing illnesses, provided by the largest healthcare network in Brazil. With the result from the study, data were preprocessed, the Random Forest technique was evaluated on the data and the strategies that best related to the objectives for the search of patterns and trends were discussed. Finally, the results were analyzed, and the metrics displayed.

Keywords: Machine learning, patterns and trends, health, medical procedures.

LISTA DE FIGURAS

LISTA DE QUADROS

SUMÁRIO

1 INTRODUÇÃO

Nos dias atuais, o indivíduo ao precisar de auxílio médico ou até mesmo buscando a manutenção de sua saúde, procura um profissional da área da saúde. Quando atendido por esses profissionais é seguida uma série de procedimentos, como a *anamnese* (entrevista clínica) juntamente com o exame físico geral para fornecer uma visão ampla do paciente. Quando o diagnóstico não está claro são solicitados os exames complementares de diagnóstico para a confirmação de hipóteses e possíveis tratamentos. Como exemplo, pode-se citar um exame complementar de diagnóstico: dosagem de glicose, comumente usada como diagnóstico e monitoramento do diabetes mellitus e dos distúrbios da homeostase glicêmica, como também para o rastreamento do diabetes gestacional (PROTOCOLOS EXAMES LABORATORIAIS, 2009).

Essa informação, gerada a partir de exames complementares, produz um grande volume de dados, disponibilizado para consulta e processamento, ocasionando o fenômeno chamado de *Big Data*. Uma das definições mais populares para *Big Data* são os "3 Vs" conforme o pesquisador Laney ensina, sustentando o aumento tridimensional de volume, velocidade e variedade dos dados (LANEY, 2001). Vários anos depois o modelo "3 Vs" foi estendido, adicionando outras características do *Big Data* como veracidade (SCHROECK ET AL., 2012), valor (DIJCKS, 2013) complexidade e desestruturação (INTEL, 2012).

Um médico consegue realizar um diagnóstico após verificar o conjunto de sintomas e resultados de exames clínicos de um paciente, utilizando o conhecimento adquirido e sua experiência. Segundo Faceli (2011) é muito difícil escrever um algoritmo que, dados os sintomas e os resultados clínicos, consiga apresentar um diagnóstico que seja tão bom quanto de um médico experiente. Entretanto, de acordo com Konenko (2001), já existem trabalhos computacionais trazendo resultados promissores na área da saúde, equiparando o resultado de diagnósticos realizados por algoritmos, com porcentagem semelhante ao acerto dos médicos.

As dificuldades em interpretar dados e coletar informações eram tratadas computacionalmente por meio da aquisição de conhecimento de especialistas de um dado domínio, como por exemplo da medicina, que era então codificado, frequentemente, por regras lógicas (CARVALHO, 2011). Entretanto, com o avanço computacional, hoje tendo como maior apoio a inteligência artificial é possível extrair informações, buscar por padrões e informações ocultas em dados. A área de inteligência artificial tem ganho destaque, em conjunto com Aprendizado de Máquina. Ambas áreas estão intimamente relacionadas ao *Big*

Data, usando o grande acúmulo de dados para produzir informação e conhecimento (AMARAL, 2016).

Uma das definições de Aprendizado de Máquina é a capacidade de melhorar o desempenho na realização de alguma tarefa por meio de experiência (MITCHELL, 1977). Segundo Monard (2008), o Aprendizado de Máquina é uma área de inteligência artificial cujo objetivo é o desenvolvimento de técnicas computacionais sobre o aprendizado, bem como a construção de sistemas capazes de adquirir conhecimento de forma automática.

Um exemplo da aplicação do aprendizado de máquina é a classificação de padrões em uma caixa de e-mails. Como por exemplo, para distinguir a diferença entre um e-mail legítimo e um e-mail de *spam*. Sabe-se que um e-mail, na maneira mais simplista, é um arquivo de caracteres, classificados em um cenário onde são considerados *spam* ou não, tendo assim somente duas alternativas, a afirmação ou a negação. Computacionalmente, um algoritmo de classificação de e-mails sem aplicações de técnicas de aprendizado de máquina e ausência de inteligência artificial, não terá um resultado duradouro que possa ser apresentado, pois e-mails de *spam* se alteram de tempos e tempos e de indivíduos para indivíduos, dificultando a classificação pelo algoritmo.

Mesmo com as periódicas modificações de padrões, os e-mails são armazenados, e facilmente são mostradas centenas de exemplos de mensagens que são consideradas *spam*. Assim, quando utilizado o aprendizado de máquina, é realizada a classificação, identificando uma grande massa de dados, aumentando a taxa de acerto e corrigindo classificações errôneas para que o programa reconheça e detecte um *spam* (ALPAYDIN, 2010) Este tipo de classificação será baseada em exemplos, uma prática comum usada no aprendizado de máquina (MONARD, 2008).

No cenário de operadoras de planos de saúde, a Unimed é a maior rede de assistência médica do Brasil, o maior sistema cooperativista no mundo. Em 2018 a cooperativa situada no Vale do Sinos gerou aproximadamente 3.500.000 (três milhões e quinhentos mil) registros referentes aos exames complementares, em conjunto com os materiais utilizados nos mesmos, como remédios controlados, materiais de laboratórios, entre outros. Todos esses registros estão armazenados em uma base de dados relacionando-os com os dados dos pacientes (FILHO, 2018).

Esse trabalho tem como objetivo identificar doenças prévias que estão ocultas ou o paciente desconhece. Possibilitando que seja realizado um tratamento preventivo, melhorando a qualidade de vida e do serviço prestado por uma operadora de saúde.

Após a seleção e coleta dos dados, foram aplicadas diversas técnicas de pré-processamento, preparando os atributos para a indução do modelo de aprendizado de máquina. Com uma grande dimensionalidade dos dados, foi feita a remoção manual de atributos irrelevantes ou ruidosos. Os resultados encontrados foram quantificados e exibidos ao final da análise.

Esse trabalho está dividido em seis Capítulos. Após essa introdução o segundo capítulo aborda os procedimentos e exames médicos realizados por uma operadora de saúde brasileira. O capítulo três aborda alguns trabalhos relacionados, os quais aplicam algoritmos de *Machine Learning* no contexto da área da saúde. No quarto capítulo discute-se a área de Aprendizagem de Máquina, bem como algumas operações de pré-processamento e técnicas. O capítulo cinco é responsável pela análise da aplicação da técnica de *Random Forest* sobre os dados e discussão dos resultados obtidos. No sexto capítulo são feitas algumas conclusões e indicações de trabalhos futuros.

2 PROCEDIMENTOS MÉDICOS EM OPERADORA DE SAÚDE

Este capítulo apresenta os principais procedimentos médicos contidos na base de dados da operadora de saúde. O texto faz uma breve apresentação sobre essa operadora de saúde, bem como detalha de onde os dados são extraídos, como são gerados, além de um detalhamento de seu conteúdo e volume.

2.1 OPERADORA DE SAÚDE

Em 1967 surgiu a primeira cooperativa de trabalho na área de medicina do país e das Américas, a União dos Médicos – ambiente no qual esse trabalho atuará e que doravante será denominado de Cooperativa de Saúde. Em 1º de maio de 1975, a Cooperativa de Saúde começa a ser idealizada dentro da Sociedade de Medicina de Novo Hamburgo, corporação que na época encabeçava as iniciativas científicas, culturais e de trabalho médico da região.

No cenário atual de operadoras de planos de saúde, a Cooperativa de Saúde é a maior rede de assistência médica do Brasil, o maior sistema cooperativista no mundo. Sua filial no Vale do Sinos conta com 511 médicos cooperados, distribuídos entre 42 especialidades reconhecidas pelo Conselho Federal de Medicina, duas unidades de atendimento 24h, dois hospitais-dia 24h, um hospital com duas unidades de internação clínica e cirúrgica, bloco cirúrgico, UTI adulto, UTI neonatal, centro de diagnóstico e muitos outros serviços relacionados a saúde.

2.2 PROCEDIMENTOS MÉDICOS

Quando uma pessoa precisa de auxílio médico, por motivo de indisposição ou buscando manutenção de sua saúde, ela procura um profissional da área da saúde. Quando atendida por esses profissionais é seguida uma série de procedimentos, como a *anamnese* (entrevista clínica), juntamente com o exame físico geral para fornecer uma visão ampla do paciente. Quando o diagnóstico não está claro são solicitados os exames complementares de diagnóstico para a confirmação de hipóteses e escolha do melhor tratamento. Como exemplo, pode-se citar um exame complementar de diagnóstico: a dosagem de glicose, comumente usada como diagnóstico e monitoramento do diabetes mellitus e dos distúrbios da homeostase glicêmica,

como também para o rastreamento do diabetes gestacional (PROTOCOLOS EXAMES LABORATORIAIS, 2009). Outro exemplo de exame é o de creatinina para detecção de lesão renal crônica, quando 50% ou mais dos néfrons estão comprometidos (PROTOCOLOS EXAMES LABORATORIAIS, 2009).

Cada vez que é solicitado um exame complementar de diagnóstico, o médico solicitante envia o pedido para a operadora de saúde, a qual realiza uma avaliação interna para a autorização do mesmo. Uma vez autorizado, é enviada uma mensagem para o paciente se dirigir ao laboratório ou hospital onde será realizada a coleta, exame e/ou procedimento. Após concluído, é informado à operadora de saúde todos os materiais utilizados, medicamentos aplicados no momento da coleta, exame e/ou procedimento, juntamente com seus respectivos valores. Todas essas informações são armazenadas em uma base de dados, onde é registrado o histórico de cada paciente da operadora de saúde.

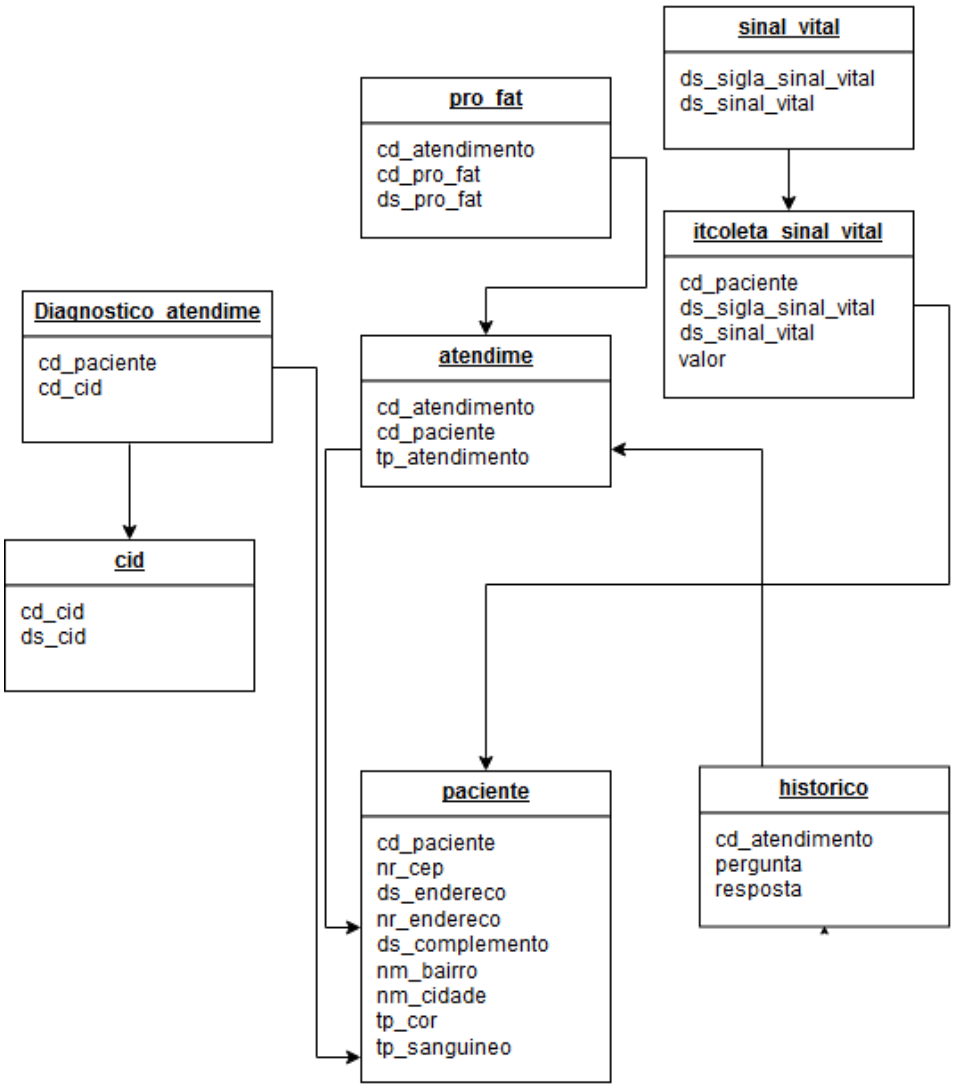
2.3 BASE DE DADOS

Os dados são armazenados em um SGBDR Sistema Gerenciador de Bancos de Dados Relacional, conforme exemplificada na Figura 1.

As informações sobre os pacientes estão localizadas na tabela paciente, assim como as informações de endereço, data de nascimento, sexo, entre outros dados. Vinculada ao paciente, há a tabela diagnostico_atendime, onde estão registrados todas os códigos CID (Classificação Internacional de Doenças) e doenças pré-existentes do beneficiário quando feito o cadastro na operadora de saúde. Essas informações são atreladas a um código de atendimento, presente na tabela atendime, a qual faz o vínculo com a tabela pro_fat, que registra todos os materiais utilizados, procedimentos e medicamentos realizados pelo paciente. A tabela sinal_vital juntamente com a itcoleta_sinal_vital registra os sinais vitais e seus respectivos valores. E por último a tabela historico aplicando o padrão NANDA.

NANDA é uma coleção de diagnósticos em enfermagem, da mesma maneira que o CID descreve as doenças, o padrão descreve as reações dos pacientes sobre as doenças. Em cada tipo de resposta, localiza-se algumas subcategorias (BELTRÃO, 2004).

Figura 1 - Esquema relacional das tabelas utilizadas na coleta de dados.



Fonte: O Autor (2019)

Realizada uma entrevista com um especialista pelo da área pelo Telegram, para orientações de informações relevantes nos registros eletrônicos de saúde, foi questionado quais seriam as características mais comuns no momento de diagnosticar ou solicitar um exame complementar. Os atributos respondidos foram Obesidade, Sedentarismo, tabagismo, histórico familiar e outras doenças já diagnosticadas.

Com um total de 602 campos nas oito tabelas, e aproximadamente 26.000.000.000 (vinte e seis bilhões)[1] de registros, não seria relevante trazer todas as informações e explanar

[1]Visualizado em 12/10/2019

cada campo. Portanto, será realizado um filtro das informações mais relevantes, discriminadas e exemplificadas com alguns registros, demonstrados na Tabela 1 até a Tabela 16.

Tabela 1 – Descrição dos campos da tabela paciente.

Campo	Descrição
CD_PACIENTE	Código do beneficiário, código único para possível rastreio posteriormente.
DT_NASC	Data de nascimento do beneficiário.
SEXO	Sexo do beneficiário.
NM_BAIRRO	Bairro de residência do beneficiário.
NM_CIDADE	Município do beneficiário.
DESCRICAO_COR_PELE	Descrição da cor da pele do beneficiário.
TP_SANGUINEO	Descrição do tipo sanguíneo do beneficiário.

Fonte: O Autor (2019)

Tabela 2 – Exemplificação dos registros da tabela paciente.

CD_PACIENTE	DT_NASC	SEXO	NM_BAIRRO	NM_CIDADE	DESCRICAO_COR_PELE	TP_SANGUINEO
1188975	20131101	2	CONCORDIA	IVOTI	BRANCA	O+
1188974	20050220	1	VICENTINA	SAO LEOPOLDO	BRANCA	A+
1188973	20190405	1	CAMPO GRANDE	ESTANCIA VELHA	BRANCA	O+
1188972	19850602	1	TATUQUARA	CURITIBA	BRANCA	A+
1188969	19830203	2	CENTRO	NOVO HAMBURGO	SEM INFORMACAO	A+

Fonte: O Autor (2019)

Tabela 3 – Descrição da tabela atendime.

Campo	Descrição
CD_ATENDIMENTO	Código do atendimento, código único para possível rastreio posteriormente.
CD_PACIENTE	Código do beneficiário, código único para possível rastreio posteriormente.
TP_ATENDIMENTO	tipo de atendimento, podendo ser hospitalar ou ambulatorial.

Fonte: O Autor (2019)

Tabela 4 – Exemplificação dos registros da tabela atendime.

CD_ATENDIMENTO	CD_PACIENTE	TP_ATENDIMENTO
1188965	1188975	I
1188955	1188974	I
1188945	1188973	H
1188935	1188972	H
1188925	1188969	H

Fonte: O Autor (2019)

Tabela 5 – Descrição da tabela diagnostico_atendime.

Campo	Descrição
CD_PACIENTE	Código do beneficiário, código único para possível rastreio posteriormente.
CD_CID	Código do CID.

Fonte: O Autor (2019)

Tabela 6 – Exemplificação dos registros da tabela diagnostico_atendime.

CD_PACIENTE	CD_CID
1188965	J180
1188955	T780
1188945	Z000
1188935	O820
1188925	C670

Fonte: O Autor (2019)

Tabela 7 – Descrição da tabela cid.

Campo	Descrição
CD_PACIENTE	Código do beneficiário, código único para possível rastreio posteriormente.
CD_CID	Código do CID.

Fonte: O Autor (2019)

Tabela 8 – Exemplificação dos registros da tabela cid.

CD_CID	DS_CID
J180	BRONCOPNEUMONIA NAO ESPECIFICADA
T780	CHOQUE ANAFILATICO DEVIDO A INTOLERANCIA ALIMENTAR
Z000	EXAME MEDICO GERAL
O820	PARTO POR CESARIANA ELETIVA
C670	NEOPLASIA MALIGNA DO TRIGONO DA BEXIGA

Fonte: O Autor (2019)

Tabela 9 – Descrição da tabela sinal_vital.

Campo	Descrição
DS_SIGLA_SINAL_VITAL	Sigla do Sinal Vital.
DS_SINAL_VITAL	Descrição do Sinal vital.

Fonte: O Autor (2019)

Tabela 10 – Exemplificação dos registros da tabela sinal_vital.

DS_SIGLA_SINAL_VITAL	DS_SINAL_VITAL
TEMP	TEMPERATURA(C).
FC	FREQUENCIA CARDIACA
FR	FREQUENCIA RESPIRATORIA
RD	RÉGUA DA DOR
PAS	PRESSÃO ARTERIAL SISTÓLICA

Fonte: O Autor (2019)

Tabela 11 – Descrição da tabela itcoleta_sinal_vital.

Campo	Descrição
CD_PACIENTE	Código do beneficiário, código único para possível rastreio posteriormente.
DS_SIGLA_SINAL_VITAL	Sigla do Sinal Vital.
DS_SINAL_VITAL	Descrição do Sinal vital.
VALOR	Valor do Sinal vital.

Fonte: O Autor (2019)

Tabela 12 – Exemplificação dos registros da tabela sinal_vital

CD_PACIENTE	DS_SIGLA_SINAL_VITAL	DS_SINAL_VITAL	VALOR
1188965	TEMP	TEMPERATURA(C).	34,5
1188955	FC	FREQUENCIA CARDIACA	51
1188945	FR	FREQUENCIA RESPIRATORIA	20
1188935	RD	RÉGUA DA DOR	10
1188925	PAS	PRESSÃO ARTERIAL SISTÓLICA	99

Fonte: O Autor (2019)

Tabela 13 – Descrição da tabela historico.

Campo	Descrição
CD_ATENDIMENTO	Código do atendimento, código único para possível rastreio posteriormente.
DS_GRUPO	Descrição do grupo de perguntas, seguindo o padrão NANDA
CD_PERGUNTA	Código da pergunta
PERGUNTA	Pergunta realizada ao paciente.
RESPOSTA	Resposta do paciente.

Fonte: O Autor (2019)

Tabela 14 – Exemplificação dos registros da tabela historico.

CD_ATENDIMENTO	DS_GRUPO	CD_PERGUNTA	PERGUNTA	RESPOSTA
1188965	19	57	Hábitos	Não se aplica
1188955	11001771	56	Faz uso de alguma medicação?	Não
1188945	1188973	24	Avaliação geral	Íntegra
1188935	1188972	25	Avaliação geral	Eupneico
1188925	1188969	19	Doenças prévias?	não

Fonte: O Autor (2019)

Tabela 15 – Descrição da tabela pro_fat.

Campo	Descrição
CD_ATENDIMENTO	Código do atendimento, código único para possível rastreio posteriormente.
CD_PRO_FAT	Código do procedimento.
DS_PRO_FAT	Descrição do procedimento.
DS_GRU_FAT	Descrição do grupo de faturamento.

Fonte: O Autor (2019)

Tabela 16 – Exemplificação dos registros da tabela pro_fat.

CD_ATENDIMENTO	CD_PRO_FAT	DS_PRO_FAT	DS_GRU_FAT
1188965	11001771	FLAGASS GTS 10ML FR ACHE	MEDICAMENTOS
1188955	11001771	CATETER OCULOS NASAL ADULTO MARKMED	MATERIAIS
1188945	1188973	EQUIPO MACROGOTAS SIMPLES HARTMANN	MATERIAIS
1188935	1188972	OBSTETRICA: PERFIL BIOFISICO FETAL	EXAMES E DIAGNOSTICOS
1188925	1188969	PATOLOGIA OSTEOMIOARTICULAR EM DOIS OU MAIS MEMBROS	EXAMES E DIAGNOSTICOS

Fonte: O Autor (2019)

Os dados foram filtrados para possibilitar a realização da aplicação de um modelo de aprendizado de máquina (*machine learning*) onde será avaliado os procedimentos de um beneficiário e suas doenças pré-existentes, para mensurar a probabilidade de um outro beneficiário desenvolver as mesmas doenças, baseado no seu histórico médico.

3 TRABALHOS RELACIONADOS

Este capítulo busca realizar um estudo sobre trabalhos relacionados na área médica com uso de aprendizado de máquina, visando maior embasamento e *insights* para a aplicação no cenário atual.

3.1 METODOLOGIA DE PESQUISA

Nesta investigação científica, não será usada uma Revisão Sistemática (RS) estrita, porém serão usados alguns conceitos de RS para nortear a pesquisa. A revisão sistemática é um método utilizado para responder a uma pergunta específica sobre um problema específico, comumente aplicada na área da saúde (ERCOLE, 2014). Uma revisão sistemática envolve várias atividades distintas, segundo Kitchenham (2004), a revisão pode ser separada em três fases principais: O planejamento da revisão, a realização da revisão e a explanação da revisão. Os estágios associados ao planejamento da revisão são a identificação da necessidade de uma revisão e o desenvolvimento de um protocolo de revisão. As etapas associadas à realização da revisão se baseiam na identificação da pesquisa, seleção de estudos primários, avaliação da qualidade do estudo, extração e monitoramento de dados e dedução dos dados. A explanação da revisão é uma fase de estágio único (KITCHENHAM, 2004).

Será realizada a busca na base de dados da MEDLINE, um motor de busca de livre acesso contendo mais de 29 milhões de citações na biomedicina, revista de ciências e livros on-line (PUBMED, 2019). O objetivo da busca, se concentra em localizar artigos e/ou publicações em revistas relacionadas com registros eletrônicos de saúde, aplicações de algoritmos de aprendizado de máquina, excluindo os resultados que contenham o uso de imagens.

Como parâmetro para a MEDLINE, foram definidas palavras chaves, a fim de compor *strings* de buscas. A primeira *string* definida foi *(((machine learning) AND electronic health records) AND Temporal Pattern) AND Longitudinal patient electronic health records,* porém, com parâmetros muito específicos, foi obtido somente 1 resultado. Em uma segunda busca, foi removido o parâmetro *Longitudinal patient* para uma maior abrangência nos resultados, gerando a *string (((machine learning OR deep learning)) AND electronic health records) AND NOT image*, a qual resultou em 22 registros. Não sendo suficiente para a pesquisa desejada, foi

removido o termo *deep learning* gerando a última *string* de busca *((machine learning) AND electronic health records) NOT image*, o qual resultou em 708 registros.

Depois de selecionado os termos de busca, foi utilizada a funcionalidade de classificação do PubMed chamada *Best Match*. O *Best Match* é um algoritmo de ordenação por relevância, baseado em um algoritmo de frequência de termo. Essa abordagem calcula a frequência com que os termos aparecem nos registros do PubMed. Essas frequências são então aplicadas de forma ponderada para retornar uma lista ordenada de citações do PubMed que correspondem aos termos da consulta.

Esse algoritmo de relevância inclui aprendizado de máquina para re-classificar os principais artigos retornados e combina mais de 150 parâmetros úteis para encontrar os melhores resultados correspondentes. Para análise dos trabalhos relacionados no resultado da pesquisa, foram considerados e analisados os 50 primeiros resultados, dos quais foram lidos os títulos e resumos. Desses cinquentas, cinco artigos que mais se assemelham no caso de estudo deste trabalho foram isolados para explanação.

3.2 *Early Detection of Hearth Failure Using Eletronic Health Records, Pratical Implications for Time Before Diagnosis, Data Diversity, Data Quantity and Data Density* (NG, 2016)

O artigo aborda a diferença entre indivíduos com e sem uma doença específica. Tais diferenças podem apontar fatores de risco, quando combinado com um modelo quantitativo, e permitem predizer quem está em alto risco de desenvolvimento da doença para direcionar ao tratamento mais adequado.

O modelo gerado detecta insuficiência cardíaca usando registros eletrônicos de saúde. Sabe-se que aos 40 anos de idade, a chance de desenvolver a doença é de 20% ao longo da vida. O diagnóstico está associado a um alto nível de incapacidade, custos de assistência médica e mortalidade (50% em 5 anos após o diagnóstico). A detecção precoce de insuficiência cardíaca abre a possibilidade de testar meios para preservar a função cardíaca e mudar a história natural do início da doença.

O artigo também aborda que a diversidade, quantidade e densidade dos dados determinará fortemente a utilidade do modelo. É importante observar também a janela de previsão, a qual significa o tempo desde o início da doença até o seu diagnóstico. Uma janela

de previsão mais longa provavelmente aumentaria o potencial de prevenção bem-sucedida, mas a precisão tende a diminuir. O desempenho do modelo melhorará até certo ponto, à medida que a janela de observação aumentar. Porém, a cobertura do paciente diminui devido ao requisito de dados mais rigoroso. O desempenho provavelmente melhorará à medida que mais domínios de dados diferentes forem usados. No entanto, à medida que o número de domínios de dados necessários aumenta, torna-se mais desafiador para o sistema de saúde fazer uso de modelos preditivos desenvolvidos. Além disso, alguns domínios de dados (por exemplo, códigos de diagnóstico) podem ser mais úteis do que outros.

Finalmente, o desempenho provavelmente dependerá da quantidade total (ou seja, do número de casos e pacientes de controle) e da densidade dos dados disponíveis. Pouco se sabe sobre os *trade-offs* entre esses fatores. Para esse fim, foram usados dados de um grande sistema de saúde para examinar o desempenho das ferramentas de aprendizado de máquina em relação à duração da janela de previsão, à duração da janela de observação e à diversidade, quantidade e densidade dos dados eletrônicos de saúde disponíveis.

Os dados foram selecionados com um mínimo de três encontros clínicos dentro de 12 meses, limitados entre pacientes com 50 a 84 anos de idade, que tiveram uma ocorrência de um diagnóstico de código HF ICD-9 (Insuficiência Cardíaca) sem uma indicação de que a insuficiência cardíaca havia sido previamente diagnosticada ou tratada. O número médio de pacientes primários foi de 240.000, aplicando os critérios anteriores foram localizados 1684 casos de insuficiência cardíaca no período entre 2003 e 2010, dos quais 28% tinham entre 50 e 84 anos de idade. As idades foram separadas em intervalos de 5 anos juntamente com o sexo. Os dados coletados para a análise estão classificados na Figura 2.

Dados de encontros clínicos foram extraídos para todos os casos e controles de cada um dos 8 domínios do tipo de dados. Os encontros incluíram visitas ao paciente, telefone e consultas de internamento. O número de variáveis exclusivas em cada tipo de dado é substancialmente variado, como representado na Figura 2. Para o tipo de dados de diagnósticos, os códigos da CID-9 de consultas ambulatoriais e listas de problemas são tratados separadamente. Isso significa que o mesmo código ICD-9 associado aos diferentes eventos foram tratados como variáveis separadas. Para o tipo de dados de medicamentos, os nomes de medicamentos normalizados foram usados. Para o tipo de dados de hospitalização, foram usados os códigos primários e secundários da CID-9 associados à admissão hospitalar. Para os diagnósticos, internações e tipos de dados de medicação, diferentes níveis de representações foram explorados. Para os tipos de dados diagnósticos e hospitalizações, a representação não

agrupada foi o código da doença CID-9, enquanto a representação agrupada foi a categoria de condição hierárquica dos Centros de Serviços *Medicare* e *Medicaid*. Para o tipo de dados sobre medicamentos, a representação não agrupada foi o nome normal do medicamento, enquanto a representação agrupada foi a subclasse farmacêutica. O agrupamento reduz o número de recursos disponíveis e cria recursos menos esparso. Como mostrado na Figura 2, o número de características únicas cai de 12 616 para 189 para o tipo de dados diagnósticos, de 7071 para 189 para o tipo de dados de internações, e de 3952 para 631 para o tipo de dados de medicamentos.

Figura 2 - Dados coletados para o modelo de predição de insuficiência cardíaca.

Data Type	Examples	No. of Unique Variables	No. of Grouped Variables
Diagnoses*	ICD-9 codes (from outpatient visits and problem lists treated separately)	12616	189
Medications†	β-blockers, loop diuretics, etc	3952	631
Laboratories‡	Cholesterol, glucose, eGFR, etc	2336	…
Hospitalization§	ICD-9 codes associated with admission	7071	189
Demographics and health behaviors	Age, sex, race/ethnicity, smoking, and alcohol use	5	…
Cardiovascular imaging ordersǁ	2D echo, transesophageal echo, stress echo, etc	18	…
Vitals‡	Pulse, systolic blood pressure, diastolic blood pressure, height, weight, and temperature	6	…
Framingham HF signs and symptoms	Positive and negative mentions of acute pulmonary edema, ankle edema, dyspnea on ordinary exertion, paroxysmal nocturnal dyspnea, etc	28	…

eGFR indicates estimated glomerular filtration rate; HF, heart failure; and ICD-9, *International Classification of Diseases*-Ninth Revision.
*ICD-9 codes from outpatient visits and problem lists are treated separately.
†Normalized drug names, ignoring the dosing information, were used.
‡The laboratories and vitals data types contain information about the test order and the numeric result of the test.
§Primary and secondary ICD-9 codes associated with the hospital admission were used.
ǁThe imaging order type, associated ICD-9 codes, and any left ventricular ejection fraction values were used.

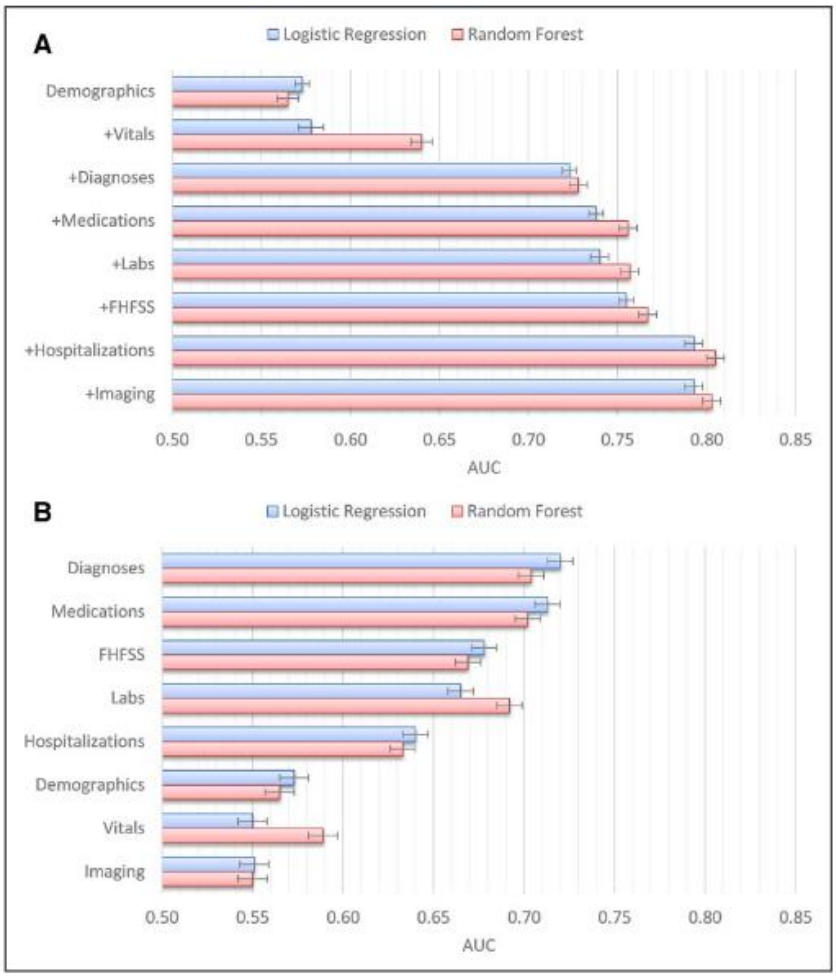
Fonte: (NG, 2016)

Os modelos de classificação de aprendizado de máquina foram criados utilizando as técnicas de *Random Forest*, SVMs (*Support Vector Machines* - Máquina de Vetores de Suporte), K-NN (K-Nearest Neighbors - K vizinhos mais próximos), *Decision Tree* (árvore de decisão), *Logistic Regression* (regressão logística) e *L1-regularized logistic regression* (Regressão Logística Regularizada L1). Os modelos de Regressão Logística, Regressão Logística Regularizada L1 e *Random Forest* tiveram um desempenho preditivo superior e melhor eficiência computacional.

Os resultados tiveram uma avaliação em dois modelos, agrupados e não agrupados, tendo o modelo agrupado um resultado tão bom quanto ou melhor que o modelo não-agrupado, removendo recursos correlacionados e redundantes como um meio de aprimorar o desempenho do modelo. O desempenho da previsão AUC (Área sob a curva ROC) foi de 0,785, mas o autor destaca a presença de muitos ruídos nos registros eletrônicos de saúde que podem alterar

consideravelmente o desempenho do modelo. O desempenho do algoritmo pode ser visualizado na Figura 3.

Figura 3 - Representação do desempenho dos algoritmos de classificação com a base de dados agrupada em cada grupo de itens extraídos, sendo os dados individuais (A) e os agrupados (B). Os valores estão representados pelo AUC.



FONTE: (NG, 2016)

3.3 *Using Machine Learning to Identify Health Outcomes from Electronic Health Record Data* (WONG, 2018)

O artigo aborda o uso *Machine Learning* aplicado em EHR (*Electronic Health Record* - registros eletrônicos de saúde) e foca em alguns dos desafios encontrados pelos pesquisadores devido ao volume e variedade de dados nos sistemas de EHR, em parte devido aos 3Vs do *Big Data*. Ele atribui que a criação de algoritmos processáveis por computador para identificar

indivíduos com condições específicas de saúde, doenças ou eventos clínicos, a partir de dados eletrônicos de saúde, é referida como fenotipagem computacional.

Muitos dos dados são armazenados em uma variedade de formatos não estruturados (texto livre, imagens) que não podem ser consultados além da inspeção manual, que muitas vezes consome tempo e exige trabalho. Uma vez extraído os dados, a tarefa de fenotipagem é especialmente desafiadora para condições específicas de saúde com critérios que considerem uma infinidade de fatores com definições frequentemente vagas e interpretações subjetivas. Nesses casos pode ser difícil conceber um algoritmo preciso, pois o diagnóstico que um profissional da área faz possui elementos abstratos de seu julgamento, como conhecimento implícito e experiência. Com base nestas informações, o artigo apresenta alguns cenários comuns que possam ser úteis para pesquisadores pensar mais criticamente sobre quando e para quais tarefas a aprendizagem de máquina pode ser útil na identificação dos diagnósticos de sáude dos dados eletrônicos e as técnicas mais utilizadas.

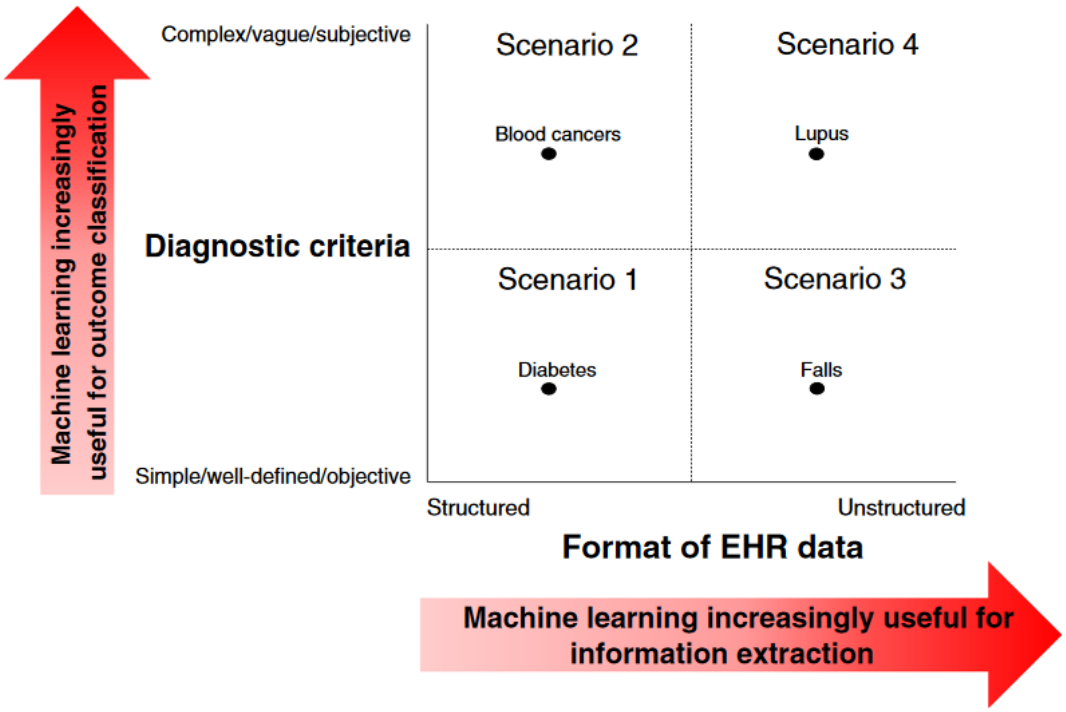
Aplicações de aprendizado de máquina na área da saúde geralmente são supervisionadas ou não supervisionadas. Grande parte das aplicações existentes, incluindo aquelas que criam algoritmos de fenotipagem eletrônica usam o aprendizado supervisionado. As técnicas como *Support Vector Machines*, *Random Forest*, bem como *Neural Networks* e modelos de *Deep Learning* são comumente usadas em registros eletrônicos de saúde.

Para estabelecer a apresentação dos quatro cenários, primeiro foram consideradas duas dimensões de um resultado de saúde, identificando condições em cada dimensão em que o aprendizado de máquina pode ser mais útil. Na primeira dimensão, são consideradas as características dos critérios de diagnósticos utilizados para definir o resultado (Figura 4 - Eixo Y). Na segunda dimensão, foi considerado o formato no qual as informações de diagnóstico sobre o resultado são geralmente armazenadas em sistemas de registros eletrônicos de saúde. (Figura 4 - Eixo X).

O primeiro cenário (Figura 4 - quadrante inferior esquerdo) inclui resultados de saúde que possuem critérios de diagnósticos simples, bem definidos e objetivos, com base nos resultados de testes, diagnósticos e procedimentos armazenados em campos estruturados. Em um sistema de registros eletrônicos de saúde com captura de dados completa e precisa, o aprendizado de máquina geralmente nãos seria necessário para identificar estes pacientes, pois seria possível extrair facilmente as informações clínicas relevantes do sistema e usá-las para classificar os pacientes. A obesidade, hipertensão e diabetes são todos exemplos de diagnósticos nesse cenário. Como todas as medidas clínicas necessárias para diagnosticar

pacientes com essas condições (altura, peso, sinais vitais e resultados de exames laboratoriais) são capturados em campos estruturados dentro de uma base de dados, não haveria razão para usar o aprendizado de máquina para identificar esses dados.

Figura 4 - Quatro cenários comuns de resultados de saúde e sua relação com a utilidade da aprendizagem de máquina para medir os resultados de saúde a partir dos dados do prontuário eletrônico de saúde (EHR).



Fonte: (WONG, 2018)

O segundo cenário (Figura 4 - quadrante superior esquerdo) inclui resultados de saúde que são diagnosticados com base nos resultados de testes e procedimentos de diagnóstico múltiplo, tipicamente armazenados em campos estruturados. No entanto, a interpretação exata de cada resultado e a interação entre eles na identificação de pacientes com o resultado da saúde não é facilmente explicável, muitas vezes exigindo que o especialista da área faça o diagnóstico final. Para esses resultados, o aprendizado de máquina pode ser útil para modelar o complexo diagnóstico, classificando os resultados. Por outro lado, o aprendizado de máquina não seria necessário para extrair esses resultados de testes de um sistema ideal, pois eles já existem em um formato legível. Um exemplo disso são as doenças hematológicas, são diagnosticadas em grande parte com base nos resultados dos exames de sangue laboratoriais.

Para realizar o diagnóstico, o médico deve ser capaz de reconhecer padrões de doenças entre grandes quantidades de resultados de exames de sangue para chegar ao diagnóstico mais provável. Dado esse processo complexo de tomada de decisão, o aprendizado de máquina pode ser útil para detectar sistematicamente padrões específicos de doenças relativas aos exames de sangue, para melhorar a detecção de doenças hematológicas, especialmente as doenças que carecem de critérios amplamente aceitos, como a *basophilic leukemia*. Usando *Random Forest* para diagnosticar diferencialmente as doenças hematológicas com base em uma série de resultados de testes sanguíneos estruturados (181 atributos de 179 diferentes exames de sangue), dentre 43 possíveis categorias de doenças hematológicas, foi identificado corretamente com uma precisão geral de 0,60, semelhante a média de seis especialistas em hematologia (0,62) e melhor que a precisão média de oito especialistas em medicina interna de não hematologia (0,26).

O terceiro cenário (Figura 4 - quadrante inferior direito) inclui resultados de saúde que tem critérios claros e objetivos, baseados em apenas algumas medições clínicas, mas essas medidas são tipicamente capturadas em formatos não estruturados nos sistemas. Para tais dados, a aprendizagem de máquina pode ser útil para facilitar a extração de informações clínicas úteis de dados não estruturados, usando NPL (*Natural Language Processing* - processamento de linguagem natural). No entanto, uma vez que as medidas clínicas necessárias foram extraídas, o aprendizado de máquina geralmente não seria necessário, por que seu papel no processo de diagnóstico já está claramente definido.

Para a tarefa de NPL, foi realizada a técnica clássica de *bag of words* para transformar o documento em um formato estruturado legível, que foi então usado como entrada para um modelo usando técnicas de *Logistic Regression Model* ou *Support Vector Machine*. Em testes realizados em quatro locais médicos diferentes, foi descoberto que a técnica de *Support Vector Machine* alcançou consistentemente a melhor discriminação, variando de 0,95 a 0,98 (AUC) em todos os quatro locais. A SVM também apresentou excelente sensibilidade, variando 0,89 a 0,93 e especificidade, variando de 0,88 a 0,94.

O quarto cenário (Figura 4 - quadrante superior direito) inclui resultados de saúde que têm critérios de diagnósticos mais vagos e subjetivos, baseados em uma variedade de medidas clínicas amplamente captadas em dados não estruturados. Estes são indiscutivelmente os mais difíceis de identificar dos registros eletrônicos de saúde, porque a informação diagnóstica útil sobre o resultado não pode ser facilmente extraída do sistema e o papel de cada medida clínica no processo de tomada de decisão não é bem descrito. Nesses casos, o aprendizado de máquina

pode ser útil tanto para extração de informações quanto para tarefas de classificação de resultados.

O Lúpus Eritematoso Sistêmico (LES) é uma doença autoimune cujo diagnóstico depende amplamente do julgamento clínico. os pacientes com LES exibem uma ampla gama de sinais e sintomas que provavelmente só seriam documentados em anotações clínicas, como erupções malares ou discóides, fotossensibilidade, úlceras orais entre outras. Turner usaram um sistema de NPL que envolvia aprendizado de máquina para identificar pacientes com LES a partir de anotações clínicas ambulatoriais. Foram extraídos conceitos médicos que serviram de entrada para quatro classificadores, *Neural Network*, *Random Forest*, *Naïve Bayes* e *Support Vector Machine*. Os pesquisadores extraíram os termos médicos usando o *clinical Text Analysis and Knowledge Extraction System* (cTakes). A *Random Forest*, *Neural Network* e *Support Vector Machine* tiveram excelente desempenho com precisão variando de 0,91 a 0,95 e AUC variando de 0,97 a 0,99 quando comparadas às classificações determinadas por reumatologistas treinados.

3.4 *A machine learning-based framework to identify type 2 diabetes through electronic health records* (ZHENG, 2017)

O artigo apresenta um estudo sobre *Type 2 diabetes mellitus* (T2DM - diabetes mellitus tipo 2), doença que é uma das principais causas de morbilidade e mortalidade e contribui para o aumento dos riscos de doença cardíaca de 2 a 4 vezes. Uma abordagem amplamente adotada para identificar indivíduos com e sem T2DM é fazer com que especialistas médicos projetem algoritmos manualmente com base em sua experiência e nos dados eletrônicos de saúde. No entanto, tais estratégias se mostram cada vez mais limitadas e não escalonáveis, devido ao árduo processo de intervenção humana e a capacidade de abstração das regras dos especialistas. Além disso, os algoritmos especialistas são frequentemente projetados com uma estratégia de identificação conservadora, portanto podem não identificar indivíduos complexos e perder um número significativo de casos e controles potenciais de T2DM.

O objetivo deste trabalho foi desenvolver um *framework* semi-automatizado baseado no aprendizado de máquina para a identificação de indivíduos com e sem T2DM. Os dados baseiam-se em três anos (variando de 2012 a 2014) de uma grande rede distribuída de registros eletrônicos de saúde composta por vários centros médicos e hospitais chineses em Xangai, China. A escolha do repositório foi motivada pelo fato que os dados chineses de registros

eletrônicos de saúde são frequentemente muito piores que os ocidentais em termos de qualidade de dados.

Os dados consistem em 123,241 pacientes no total de 3 anos de investigação. A estratégia de filtragem para pré-selecionar pacientes foi definida: os dados eletrônicos da saúde deveriam satisfazer pelo menos um dos três critérios, diagnóstico relacionado ao diabetes, medicação relacionada ao diabetes e teste laboratorial diabético. Através deste processo, foram obtidas 23,281 amostras de pacientes com informações relacionadas ao diabetes.

A estrutura de aprendizado de máquina é baseada em aprendizado supervisionado, que requer amostras de treinamento rotuladas. Assim, foram convidados dois especialistas clínicos com experiência em diabetes para avaliar os dados de amostras e rotular essas amostras em três categorias, caso, controle e não confirmado. Devido à enorme quantidade de esforço manual no processo de revisão de especialistas, foram selecionados aleatoriamente 300 amostras de 23,281. Amostras de T2DM confirmadas pelos dois especialistas foram consideradas casos, com duas confirmações de não-DMT2 foram consideradas como controle. As outras amostras com rótulos conflitantes ou duas confirmações não determinadas pelos dois clínicos serão indicadas como não confirmadas.

Por meio destas avaliações, foram obtidos 161 casos, 60 controles e 79 amostras não confirmadas. Para uma verificação dupla, foi notado que das 79 amostras não confirmadas, a maioria (78,3%) estava gravemente incompleta em sua documentação de registros eletrônicos de saúde, que não são adequadas para a fenotipagem. Para redução das influências negativas nos registros eletrônicos de saúde incompletos no modelo de classificação, foi descartado as 79 amostras não confirmadas.

A construção de bons recursos dos dados eletrônicos de saúde é muitas vezes uma obrigação para garantir um bom desempenho de previsão para algoritmos especialistas ou modelos baseados em aprendizado de máquina. Isso ocorre porque os dados brutos são frequentemente ruidosos, esparsos e contém informações não estruturadas como por exemplo, campos de texto livre que não são diretamente computáveis. Pesquisas tradicionais sobre a identificação de sujeitos com e sem T2DM usavam estratégias de seleção baseadas em três características, diagnóstico diabético, exames laboratoriais diabéticos e medicações diabéticas extraídas dos registros eletrônicos de amostras investigadas. Tais pesquisas são limitadas devido às suas altas taxas de falta na identificação de casos e controles. Isto porque estas extrações aplicaram um critério de seleção conservador em casos de controle, como por exemplo, satisfazer duas das três características mencionadas.

Para tornar a identificação de caso/controle mais precisa, foi incorporado mais recursos do que os usados tradicionalmente. Foi construído recursos adicionais de T2DM, como sintomas relacionados a diabetes, complicações diabéticas e assim por diante, na esperança de melhor identificar amostras limítrofes ou ambíguas. No total foram derivadas 110 características de sete fontes, denotados como recurso de primeiro nível que são: informações demográficas, relatório de comunicação, relatório de diagnóstico ambulatorial, relatório de diagnóstico de internação, resumo de alta hospitalar, relatório de prescrição e relatório de teste de laboratório.

Para a classificação foram usados os modelos amplamente utilizados, como *k-Nearest-Neighbors* (kNN), *Naïve Bayes* (NB), *Decision Tree* (J48), *Random Forest* (RF), *Support Vector Machine* (SVM) and *Logistic Regression* (LR). Esses modelos de classificação são frequentemente utilizados com uma ampla gama de campos e são reconhecidos como escolhas populares para tarefas de classificação.

O J48, RF, SVM tiveram alto desempenho de previsão em várias métricas, produzindo mais de 0,95 em precisão, sensibilidade, especificidade e AUC em todos as três técnicas de recursos. Para comparação, o algoritmo especialista de última geração leva a um desempenho de 0,84 na precisão, 0,78 na sensibilidade e 1,00, especialidade e 0,71 na AUC. Isso indica que os recursos construídos em todas as três técnicas podem identificar assuntos de T2DM muito melhor o algoritmo especialista popular. O algoritmo especialista de última geração executa um pouco melhor e quase perfeitamente em termos de especificidade e precisão. Isto parece mais provável devido às suas condições rigorosas na seleção de casos. No geral, modelos como RF, J48 e SVM são mais estáveis do que os outros três classificadores. Isso pode ter sido causado porque LR kNN e NB são mais influenciados por ruídos e esparsidade.

3.5 Predicting the risk of emergency admission with machine learning: Development and validation using linked electronic health records. (RAHIMIAN, 2018)

O artigo aborda as admissões hospitalares de emergência e seu alto custo na área da saúde. No Reino Unido houve mais de 5,9 milhões de admissões hospitalares de emergência registradas em 2017, um aumento de 2,6% em comparação com o ano anterior. Com a intenção de evitar que essas ocorrências alcancem grandes proporções, tem havido interesse de pesquisa. Para orientar a tomada de decisão, modelos de previsão de risco têm sido usados, no entanto, em média os modelos tendem a ter uma fraca capacidade de discriminação de risco.

Com o aumento e disponibilidade de conjuntos de dados clínicos, tais como registros eletrônicos de saúde, juntamente com o avanço do aprendizado de máquina, surgem novas possibilidades para o desenvolvimento de novos modelos de previsão de risco. Tais modelos têm se mostrados com uma performance maior que modelos estatísticos, particularmente em ambientes onde os dados clínicos foram mais ricos. Com o objetivo de avaliar se duas técnicas de aprendizado de máquina padrões poderiam aumentar a previsão de internações hospitalares de emergências na população em geral em comparação com um modelo de alto desempenho *Cox Proportional Hazards* (CPH) que também usa registros eletrônicos de saúde.

O estudo foi conduzido usando dados eletrônicos de saúde ligados a *Clinical Practice Research Datalink* (CPRD) desde a sua criação em 01 de janeiro de 1985 a 30 de setembro de 2015. Os dados de pacientes da CPRD são pseudo-anônimos e representam aproximadamente 7% da atual população do Reino Unido, contendo informações sobre sexo, idade e etnia. Ele liga registros de cuidados primários com diagnósticos de alta do *Hospital Episode Statistics* e dados de mortalidade do *National Death Registries* com um sistema de códigos equivalente ao *World Health Organization International Classification of Diseases*.

Foram considerados todos os pacientes entre 18 e 100 anos, como no mínimo um ano de registro e excluídos todos sem um número válido no *National Health Service* (NHS) ou informações faltantes no *Index of Multiple Deprivation* (IMD), uma área baseada em indicadores socioeconômicos. Semelhante ao modelo CPH (*QAdmissions*), a data de entrada para cada paciente foi definida como a data de seu aniversário de 18 anos ou o primeiro registro com mais de um ano, desde que essa data seja anterior a linha de base, definida como 1º de janeiro de 2010. Do total de 7,612,760 pacientes, 4,637,297 pacientes preencheram os critérios de seleção.

Foram usados 3 conjuntos de preditores, variáveis utilizadas nos modelos de previsão. No primeiro conjunto, referido como QA, foi incluído 43 variáveis referidas ao modelo estabelecido *QAdmissions*, cobrindo dados demográficos do paciente, fatores de estilo de vida como *status* socioeconômico, índice de massa corporal, tabagismo e consumo de álcool. histórico familiar de doenças, vários testes de laboratórios, 16 comorbidades, 6 medicações prescritas e de admissões de emergências anteriores. No segundo conjunto de preditores, referido como QA+, foram adicionados 13 novos preditores, incluindo estado civil, 11 novos comorbidades e o número de visitas no ano. No terceiro conjunto, referido com T, foram modificados alguns preditores do QA+ para armazenar informações temporais.

Primeiramente foi replicado o modelo CPH para modelo de *benchmark*. Esse modelo foi baseado nas mesmas variáveis de preditores. Como *QAdmissions* demonstra os resultados separadamente para homens e mulheres, os resultados também foram separados por sexo. Foi comparado os resultados com dois modelos de aprendizado de máquina, *Gradient Boosting Classifier* (GBC) e *Random Forest* (RF). Ambos os modelos são baseados em árvores de decisão.

Usando preditores de QA no modelo CPH, a AUC foi de 0,740 (0,741 para os homens e 0,739 para mulheres). Aplicado RF e GBC aos mesmo preditores, teve um aumento a AUC de 0,752 e 0,799 respectivamente. Usando os preditores QA+, todos os modelos mostraram um AUC ligeiramente superior, com o maior aumento observado para o modelo RF. Quando foram utilizados os preditores T, todos os modelos apresentaram maiores valores AUC, mas modelo GBC apresentou o melhor desempenho com 0,848, o modelo RF apresentou 0,825 e o modelo CPH apresentou 0,805.

3.6 Temporal Pattern and Association Discovery of Diagnosis Codes using Deep Learning (MEHRABI, 2015)

O último artigo aborda os registros eletrônicos de saúde longitudinais, que acompanham as características de doenças, seu tratamento e seus resultados. A análise dessas informações fornece informações valiosas para a tomada de decisões clínicas que incluem a fenotipagem e o diagnóstico precoce, também facilita a descoberta de novos padrões na progressão da doença com base no conhecimento adquirido de pacientes similares.

A abstração temporal é uma das abordagens mais comuns no estudo do padrão temporal de observações clínicas amostradas com temporização desigual. A abstração temporal transfere eventos clínicos com registro de data e hora para a representação baseada no intervalo, de modo que os métodos de mineração de dados temporais possam ser aplicados. O *KarmaLego* é um algoritmo para mineração rápida de padrões de intervalos temporais, baseia-se nas sete relações de Allen com adição de um valor épsilon a todas as sete relações. *Chrono Miner* é um sistema de mineração temporal conduzido por analogia que extrai dinamicamente a associação temporal em vários níveis hierárquicos.

Neste artigo foi adotado a ideia de representar o registro longitudinal de cada paciente como uma matriz de diagnósticos, onde o eixo Y corresponde a características clínicas, como

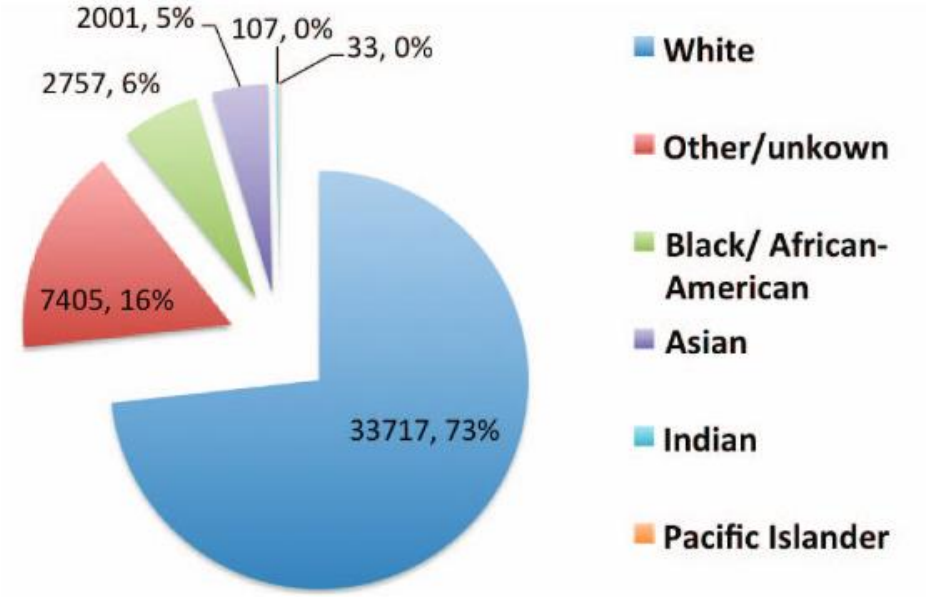
sintomas, valores laboratoriais e radiológicos, etc, e o eixo X corresponde ao tempo em que foram registrados em prontuários longitudinais. Aplicando algoritmos de *Deep Learning*, é explorado a descoberta de padrões temporais usando os códigos ICD-9 e HCUP CSS, construindo duas matrizes de diagnóstico independentes para cada paciente.

O CD9-CM consiste em mais de 14.000 códigos de diagnóstico com granularidade e detalhes finos. A *Agency for Healthcare Research and Quality* (AHRQ, Agência de Pesquisa e Qualidade em Saúde) desenvolveu uma coleção de bancos de dados e ferramentas de software relacionadas por meio do *HealthcareCust and Utilization Project* (HCUP) que possibilitou a pesquisa em uma ampla gama de tópicos, incluindo custo e qualidade de serviços de saúde, resultados de tratamentos e padrões de práticas médicas. O *Clinical Classification Software* (CCS) no HCUP classifica os códigos de diagnóstico ICD9-CM e os códigos CPT (Current Procedural Terminology) em categorias mais gerenciáveis e clinicamente significativas. Neste estudo, é explorado a técnica de aprendizagem profunda para descobrir padrões temporais entre os códigos de diagnóstico.

O *Rochester Epidemiology Project* (REP) é uma infra-estrutura de pesquisa, que reúne os registros médicos dos residentes do condado de Olmsted, Minnesota, e apoiou vários estudos analíticos de base populacional sobre doenças e seus resultados. O REP administra um grupo dinâmico de 502.820 pacientes únicos que viveram no condado de Olmsted em algum momento entre 1966 e 2010 e receberam assistência médica de um dos 50 profissionais de saúde participantes. O REP relaciona os registros médicos longitudinais de pacientes que contribuíram com um total de 6.239.353 pessoas-anos de acompanhamento. O REP fornece índices para todos os prontuários eletrônicos baseados em papel e para cada paciente, contendo informações como características demográficas, códigos de diagnóstico médico, códigos de procedimentos cirúrgicos e informações sobre a morte, incluindo sua causa.

Os dados do REP utilizados neste projeto consistiram em um código de paciente (*patientID*), sexo, raça, data de nascimento, códigos diagnósticos ICD9-CM e HCUP CSS, contagens de códigos de diagnóstico em cada visita, duração da estadia e datas de visitação. A análise foi direcionada em pacientes pediátricos e adolescentes, isto é, com 18 anos de idade ou menos, assumindo que os pacientes têm menos diversidade fenotípica com padrões temporais claros. Um grupo de 46.020 pacientes, 23.128 femininos e 22.892 masculinos com 271 códigos únicos HCUP CSS e 6.902 códigos ICD9 exclusivos durante 6 anos, de 2004 a 2009, foi construída.

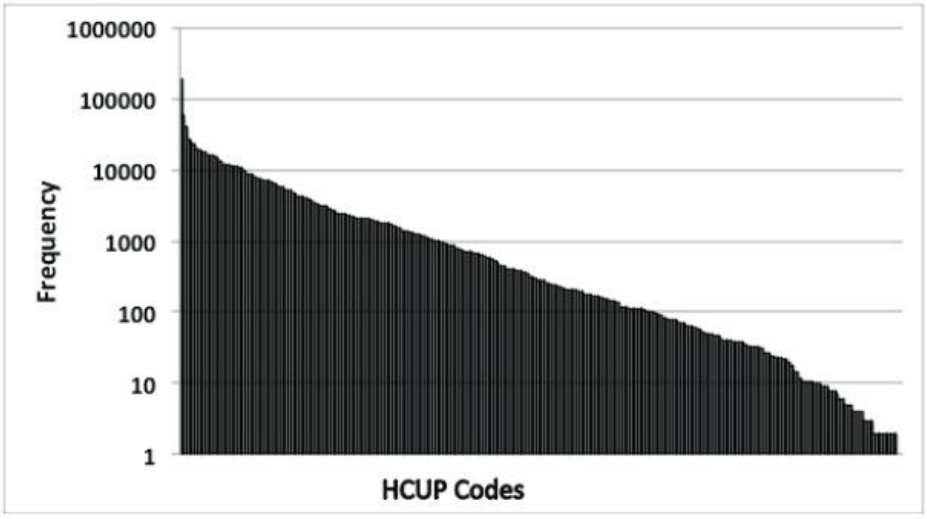
Figura 5 - Populações de pacientes, estratificados por raça / etnia, com número de pacientes e sua porcentagem correspondente separada por vírgula.



Fonte: (MEHRABI, 2015)

Para modelar os registros clínicos de cada paciente como uma matriz de diagnóstico, uma classe de pacientes com atributos como ID do paciente, raça, gênero e duas matrizes de diagnóstico foram construídas. Para cada paciente, uma instância da classe do paciente foi criada pela atualização de seus atributos. Como mencionado anteriormente, os códigos ICD9 e HCUP foram considerados neste estudo, portanto, duas matrizes foram construídas representando cada código de diagnóstico separadamente. Em cada matriz, a linha e a coluna representam o código de diagnóstico e o ano de diagnóstico, respectivamente. A fim de criar uma matriz de códigos de diagnóstico e ano de diagnóstico com tamanho gerenciável, foi limitado as granulometrias da data do diagnóstico ao ano de diagnóstico para reduzir o número de datas de visitas possíveis e correspondentemente o número de colunas na matriz. Para reduzir o número de linhas, 6.902 na matriz ICD9 e 271 na matriz CSS do HCUP, foi selecionado os códigos CSS mais usados em ICD9 e HCUP no grupo.

Figura 6 - Representação de códigos CSS ICD9 atribuídos a cada paciente no grupo com o número de pacientes na escala logarítmica.



Fonte: (MEHRABI, 2015)

Figura 7 - Representação de códigos HCUP atribuídos a cada paciente no grupo com o número de pacientes na escala logarítmica.

Fonte: (MEHRABI, 2015)

A *Restricted Boltzmann Machine* (RBM) é um dos blocos de construção populares do *Deep Learning* que é usada neste estudo. O RBM é um modelo gráfico não-dirigido de duas camadas que, ao contrário do Boltzmann *machine algorithm*, não tem conexões *hidden to hidden* e *visible to visible* . Ele usa os pontos de conexão W entre as unidades visíveis v e as

unidades ocultas h para definir a probabilidade conjunta das duas camadas P (v, h) com uma função de energia E.

Vários RBMs podem ser treinados usando cada ocultador como dados de treinamento para a próxima camada de nível superior. Essa quantidade de RBMs pode ser vista como um único modelo probabilístico *Deep Belief Network* (DBN). Salakhutdinov e Hinton introduziram a *Deep Boltzmann Machines* (DBM) que também compôs de múltiplas camadas de RBM com uma pequena modificação ao algoritmo DBN. Eles usaram a aproximação experimental para estimar as expectativas dependentes dos dados e a cadeia de Markov persistente para estimar a expectativa dependente da data. Estas duas técnicas de estimativa tornam prático aprender máquinas Boltzmann com camadas ocultas múltiplas e milhões de parâmetros.

Pylearn2 foi usado para implementar o DBM com três camadas ocultas e o algoritmo de aprendizado de PCD. *Pylearn2* é uma biblioteca construída sobre Theano e escrita em *Python* com ênfase em flexibilidade e extensibilidade.

Um conjunto de 45.627 matrizes CSS do HCUP com 286 dimensões foram usadas como entradas para uma a rede DBM. A Figura 8 mostra o valor dos pesos que conectam as unidades visíveis à primeira camada oculta na rede. O valor desses pesos mostra a força ou importância da contribuição dos nós visíveis para os nós da camada oculta. Valores mais altos de pesos são mostrados pela cor vermelha versus a cor azul que representa valores mais baixos nas figuras de *heatmaps*.

Após realizado o estudo de trabalhos relacionados, organizou-se as técnicas de classificação mais usadas em diferentes cenários, tratamentos e agrupamentos de dados, tanto estruturados como não estruturados. Buscando sintetizar as informações coletadas nos artigos apresentados, foi construída uma tabela para quantificar as técnicas de classificação apresentadas. Como exemplificado no Tabela 7, a técnica com maior ocorrência foi o *Random Forest* (RF) sendo aplicada em 4 dos 5 artigos. Em segundo lugar no número de ocorrências está a *Support Vector Machines* (SVM). Estas duas técnicas de classificação serão aprofundadas no próximo capítulo juntamente com as definições de *Machine Learning* para, posteriormente aplicar-se na construção de uma aplicação prática.

Figura 8 - Valor dos pesos que conectam as unidades visíveis à primeira camada oculta na rede, representados em mapa de calor.

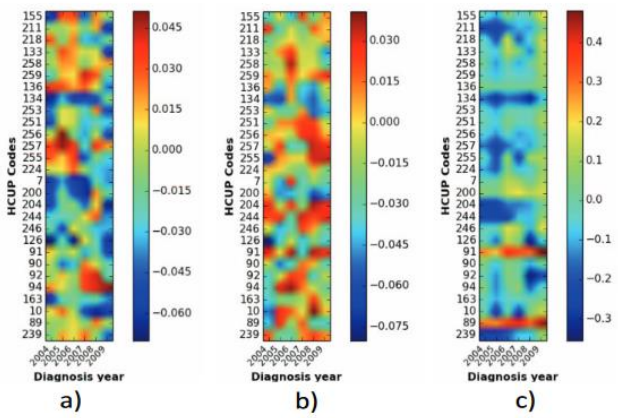
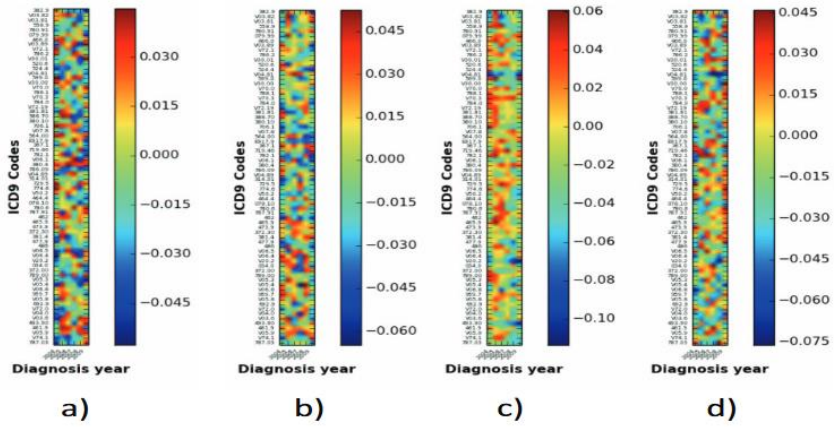


Figure 6. Heatmaps of first hidden layer weights



Fonte: (MEHRABI, 2015)

3.7 - CONSIDERAÇÕES FINAIS

Buscando sintetizar as informações coletadas nos artigos apresentados, foi construída uma tabela para quantificar as técnicas de classificação apresentadas. Como exemplificado no Tabela 17, a técnica com maior ocorrência foi o *Random Forest* (RF) sendo aplicada em 4 dos 5 artigos. Em segundo lugar no número de ocorrências está a *Support Vector Machines* (SVM). Estas duas técnicas de classificação serão aprofundadas no próximo capítulo juntamente com as definições de *Machine Learning* para, posteriormente aplicar-se na construção de uma aplicação prática.

Tabela 17 - Quantificação das ocorrências de técnicas de classificação nos artigos estudados e seus totais. Legendas: RF (*Random Forest*), SVM (*Support Vector Machines*), K-NN (*K-Nearest Neighbors*), J-48 (*Decision Tree*), LR (*Linear Regression*), L1-RLR (*L1-regularized logistic regression*), NB (*Naïve Bayes*), GBC (*Gradient Boosting Classifier), DBM* (*Deep Boltzmann Machines*).

Artigos	**Técnicas**								
	RF	SVM	K-NN	J-48	LR	L1-RLR	NB	GBC	DBM
(NG, 2016)	X	X	X	X	X	X			
(WONG, 2018)	X	X							
(ZHENG, 2017)	X	X	X	X	X		X		
(RAHIMIAN, 2018)	X							X	
(MEHRABI, 2015)									X
Total	4	3	2	2	2	1	1	1	1

Fonte: O Autor (2019)

4. APRENDIZADO DE MÁQUINA

Nos dias atuais, a computação tem a capacidade de guardar e processar grandes quantidades de dados. Por exemplo, uma rede de supermercados com centenas de lojas interligadas, vendendo produtos a milhões de consumidores. Para cada transação é registrado cada produto, data, identificação do consumidor e valor gasto. Essa quantidade de informações geralmente equivale a *Gigabytes* de dados todos os dias. Essas informações se tornarão úteis quando analisadas e transformadas em informação que possam ser usadas, como por exemplo, para realizar predições (ALPAYDIN, 2010).

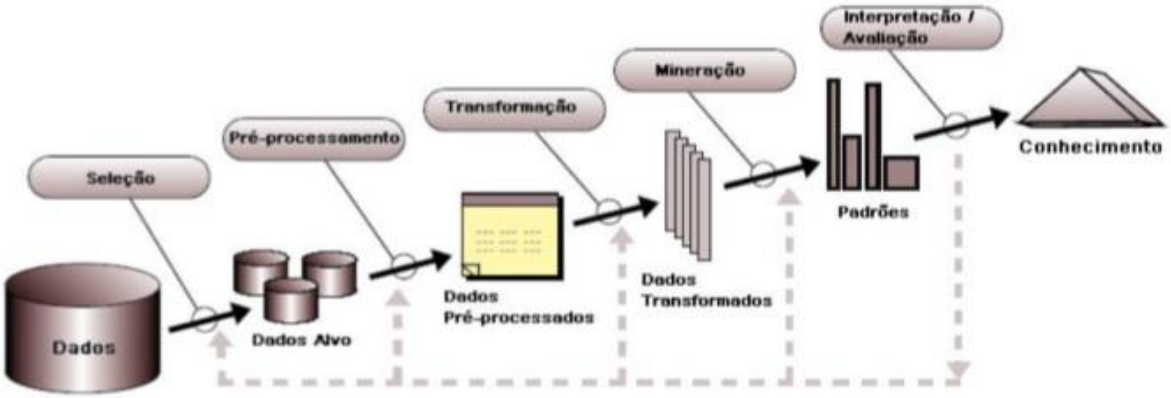
Segundo Alpaydin (2010) existe um processo para explicar os dados que são observados, apesar de não saber os detalhes do processo adjacente. Como exemplo do comportamento de um cliente, sabe-se que as pessoas não vão ao mercado comprar coisas totalmente aleatórias, ou seja, quando se compra cerveja, se compra salgadinhos, quando é verão se compra sorvete. Existe um certo padrão nos dados.

Não é possível identificar o processo completamente, mas pode ser construído uma aproximação útil. Essa aproximação pode não conseguir explicar tudo, mas pode explicar parte dos dados. Essa construção e análise, a partir de dados existentes, é chamada de *Machine Learning* (Aprendizado de Máquina) (ALPAYDIN, 2010).

Segundo Mitchell (1997) *Machine Learning* é a capacidade de melhorar o desempenho na realização de alguma tarefa por meio de experiência (MITCHELL, 1997). O processo de mineração desses dados (experiências pregressas) deve ser automático ou mais comumente semiautomático. Os padrões descobertos devem ser significativos, pois levam geralmente a uma vantagem competitiva (WITTEN, 2015).

O processo deste trabalho pode ser dividido em três grandes etapas: Pré-processamento, Classificação e Pós-processamento. A Figura 9 ilustra as etapas separadamente, as quais serão introduzidas a seguir.

Figura 9 - Processo de Mineração de Dados



Fonte: Processo KDD (Adaptado FAYYAD et al., 1996)

4.1 PRÉ-PROCESSAMENTO

O pré-processamento é uma fase abrangente e consiste de um número de diferentes estratégias a serem aplicadas sobre os dados. Ele tem como objetivo tornar os dados mais apropriados para a extração de conhecimento (TAN, 2009). Ainda que os algoritmos de *machine learning* extraiam conhecimento de dados, seu desempenho é correntemente afetado pelo estado dos dados (FACELI, 2011). Os valores de um determinado conjunto de dados podem conter ruídos e imperfeições, com valores incorretos, inconsistentes, duplicados ou ausentes (FACELI, 2011).

Quando um atributo não apresenta relevância em um objetivo específico, dentro de um conjunto de dados, é aplicado a eliminação manual. Como por exemplo, em uma base de dados hospitalar, não é relevante usar os valores dos atributos “nome” e “id” para o diagnóstico. Segundo Faceli (2011), quando um atributo claramente não contribui para a estimativa do valor do atributo alvo, ele é considerado irrelevante. Outra maneira de detectar que um atributo é irrelevante, é quando um atributo possui o mesmo valor para todos os objetos, não tendo relevância na distinção dos objetos.

4.2.1 Amostragem de Dados

Alguns algoritmos de aprendizado de máquina podem ter dificuldades em lidar com um número grande de objetos, como pode ser observado no *k*-NN (*k-Nearest Neighbours*), que pode apresentar problemas de saturação de memória, quando o conjunto de dados tem grande número de exemplos (FACELI, 2011). Segundo Tan (2009), em alguns casos, usar um algoritmo de amostragem pode reduzir o tamanho dos dados e propiciar que um algoritmo melhor, porém mais custoso, possa ser usado.

Precisa-se observar que uma amostra pequena pode não representar bem o problema que se deseja modelar. A amostra deve ser representativa do conjunto de dados original (FACELI, 2011). Devido a amostragem ser um processo estatístico, a representatividade de qualquer amostra pode variar. O mais indicado a ser realizado é escolher uma amostragem que garanta uma alta probabilidade de se obter uma amostra representativa (TAN, 2009). Existem muitas técnicas de amostragem, apenas algumas das mais básicas e suas variações serão cobertas.

4.2.1.1 Amostragem Aleatória Simples

O tipo mais simples de amostragem é a denominada aleatória simples. Ela possui duas variações, a amostragem simples sem reposição de exemplos, em que exemplos são extraídos do conjunto original para a amostra a ser utilizada. Cada exemplo não pode ser selecionado mais de uma vez (FACELI, 2011). A segunda variação, a amostragem com substituição, onde os objetos não são removidos do conjunto original quando selecionados para a amostra. Nesse método cada objeto pode ser selecionado mais de uma vez (TAN, 2009).

As amostras produzidas pelos dois métodos são semelhantes quando as amostras são relativamente pequenas se comparadas com o conjunto de dados. A amostragem com substituição é mais simples de analisar, por que a probabilidade de se selecionar qualquer objeto se mantém constante (FACELI, 2011).

4.2.1.2 Amostragem Estratificada

Quando o conjunto de dados consiste de tipos de dados muito diferentes, com números de objetos diferentes, a amostragem simples pode falhar em representar todos os tipos de objetos que sejam menos frequentes (TAN, 2009). Um cuidado que deve ser tomado na amostragem, tem relação a distribuição de dados nas diferentes classes (FACELI, 2011). Deste modo, é necessário a aplicação da amostragem estratificada, a qual acomoda frequências diferentes para os objetos. Na sua versão mais simples, números iguais de objetos são trazidos de cada grupo, mesmo quando os grupos possuem tamanhos diferentes.

4.2.1.3 Amostragem Progressiva

Pode ser encontrado dificuldade no momento de determinar o tamanho apropriado da amostra (TAN, 2009). A amostragem progressiva inicia com uma amostra de tamanho pequeno, e aumenta progressivamente o tamanho da amostra extraída, enquanto a acurácia

preditiva continuar a melhorar. Dessa forma, é possível definir a menor quantidade de dados necessária, reduzindo ou eliminando a perda de acurácia (FACELI, 2011).

4.2.2 Dados Desbalanceados

Conjuntos de dados com classes desbalanceadas são aqueles que possuem uma grande diferença de ocorrências entre os valores de um atributo de classe (BATISTA, 2003). A maioria dos algoritmos de aprendizado de máquina tem dificuldades em criar um modelo que classifique com precisão os exemplos da classe minoritária. Uma forma de solucionar esse problema é redefinir o tamanho do conjunto de dados ou utilizar diferentes custos de classificação para as diferentes classes (FACELI, 2011).

4.2.3 Limpeza de Dados

A Limpeza de dados é aplicada quando o algoritmo de aprendizado de máquina encontra dificuldades relacionadas a qualidade dos dados. Exemplos mais frequentes destas dificuldades são dados ruidosos (que possuem erros ou valores que são diferentes do esperado), inconsistentes, redundantes ou incompletos. A principal dificuldade se encontra na detecção de dados ruidosos (FACELI, 2011).

Diversas técnicas podem ser utilizadas para reduzir o ruído em um atributo. Elas podem ser resumidas em cinco grupos, sendo eles a técnica de encestamento, técnicas baseadas em agrupamento de dados, técnicas baseadas em distância e a técnica baseada em regressão ou classificação (FACELI, 2011).

4.2.4 Transformação de Dados

Muitas técnicas de aprendizado de máquina estão limitadas a manipular valores de determinados tipos, por exemplo somente valores numéricos ou apenas valores simbólicos (FACELI, 2011). O principal objetivo é realizar a transformação dos dados para superar as limitações existentes nos algoritmos que serão usados para a extração de padrões. A decisão de quais transformações são necessárias depende do algoritmo que será utilizado. (BATISTA, 2003).

4.2.5 Redução de Dimensionalidade

O conjunto de dados pode ter um grande número de atributos e poucas técnicas de aprendizado de máquina podem lidar com muitos atributos. A consequência de um número muito grande de atributos em algoritmos de aprendizado de máquina é descrita pelo problema

da maldição da dimensionalidade (FACELI, 2011). Para que os dados dos atributos possam ser utilizados, a quantidade de atributos precisa ser reduzida.

A redução pode ainda melhorar o desempenho do modelo induzido, reduzir seu custo computacional e tornar os resultados obtidos mais compreensíveis. Diferentes técnicas de áreas de pesquisa como Reconhecimento de Padrões, Estatística e Teoria da Informação podem ser utilizadas para a redução do número de atributos. As técnicas podem ser divididas em duas abordagens: Agregação e Seleção de Atributos (FACELI, 2011).

4.2.5.1 Agregação

Uma das abordagens mais comuns para a redução de dimensionalidade especialmente para dados contínuos, usam técnicas de álgebra linear ou não linear para projetar dados de um espaço de alta dimensionalidade para uma dimensionalidade menor (TAN, 2009). Uma das técnicas é chamada de Análise de Componentes Principais (PCA, *Principal Component Analysis*). A técnica relaciona estatisticamente os exemplos, reduzindo a dimensionalidade do conjunto de dados original pela eliminação de redundâncias.

As técnicas de agregação, ao combinar os atributos podem levar a perda dos valores originais. Em áreas como a medicina ou monitoramento ambiental, geralmente é importante preservar os valores dos atributos para que o resultado obtido possa ser interpretado, associando o resultado aos valores dos atributos. Por essa questão, nessas áreas é mais frequente a redução do número de atributos pelo uso de técnicas de seleção (FACELI, 2009).

4.2.5.2 Seleção de Atributos

Segundo Faceli (2009), a seleção de atributos permite identificar atributos importantes, melhorar o desempenho de várias técnicas de aprendizado de máquina, reduzir a necessidade de memória e tempo de processamento, eliminar atributos irrelevantes e reduzir ruído, lidar com a maldição de dimensionalidade, simplificar o modelo gerado e tornar mais fácil sua compreensão, facilitar a visualização dos dados, reduzir o custo de coleta de dados e com isso aumentar o acesso a novas tecnologias. Alguns atributos são facilmente identificados como irrelevantes ou redundantes, podendo ser eliminados manualmente. Por outro lado, vários atributos passíveis de eliminação não são facilmente identificados, o que torna pouco eficiente o uso de técnicas visuais.

Para tratar casos onde a eliminação visual não é efetiva, têm sido propostas técnicas automáticas para a seleção de atributos. Uma delas tem relação com a avaliação do conjunto

de atributos selecionados. Nesse cenário, as técnicas existentes podem estar incorporadas a um algoritmo de indução ou serem independentes do algoritmo. Para qualificar a qualidade ou desempenho de um subconjunto de atributos, três abordagens são utilizadas (FACELI, 2009).

4.2.5.2.1 Embutida

A seleção do subconjunto é embutida ou integrada no próprio algoritmo do aprendizado e trabalha com seleção de subconjuntos. As árvores de decisão, por exemplo, realizam esse tipo de seleção interna de atributos (FACELI, 2009). No geral as técnicas embutidas fazem melhor uso dos dados que as baseadas em *wrapper*. Além disso por não precisar retreinar o algoritmo para cada novo conjunto de atributos, geralmente são mais rápidas (FACELI, 2009).

4.2.5.2.2 Baseada em filtro

A abordagem baseada em filtro, na etapa de pré-processamento, aplica um filtro sobre o conjunto de atributos originais que seleciona um subconjunto de atributo, sem levar em consideração o algoritmo de aprendizado que utilizará esse subconjunto. As técnicas que seguem a abordagem baseada em filtro verificam a correlação baseada entre atributos e são geralmente mais rápidas que as baseadas em *wrapper*. Uma característica considerada negativa dos filtros refere-se a sua independência em relação ao algoritmo, por outro lado, pode ser considerado uma vantagem, pois pode ser utilizado por diferentes técnicas (FACELI, 2009).

4.2.5.2.3 Baseada em *Wrapper*

A abordagem baseada em *wrapper* utiliza o próprio o próprio algoritmo de aprendizado como uma caixa-preta para a seleção. Normalmente utilizada com uma técnica de amostragem, para cada subconjunto, o algoritmo é consultado e o subconjunto que apresentar a melhor proporção entre redução da taxa de erro e redução do número de atributos é em geral selecionado. Essa técnica representa uma alternativa simples e desenvolvida para a seleção de atributos. Embora a técnica seja criticada pela utilização de força bruta com custo computacional elevado, estratégias de busca eficientes têm sido utilizadas por algumas dessas técnicas (FACELI, 2009).

4.2 CLASSIFICAÇÃO

Classificação é a tarefa de organizar objetos em uma entre diversas categorias pré-definidas, é um problema universal que engloba muitas aplicações diferentes. Os dados de entrada da tarefa de classificação são um conjunto de registros. Cada registro, também conhecido como uma instância ou exemplo, é caracterizado por uma tupla (x, y), onde x é o conjunto de atributos e y é o atributo especial, designado como rótulo de classe (atributo alvo). Os atributos podem ser apresentados com características discretas ou contínuas. O rótulo de classe, por outro lado, deve ser um atributo discreto. Esta é a característica chave que diferencia a classificação da tarefa de regressão (TAN, 2009).

Um modelo de classificação pode ser utilizado em diferentes propósitos: descritivo ou preditivo. O modelo descritivo pode adequar-se como ferramenta explicativa para distinguir entre objetos e classes diferentes. Como por exemplo, na classificação de vertebrados, representado na Figura 10.

Figura 10 - Classificação de vertebrados.

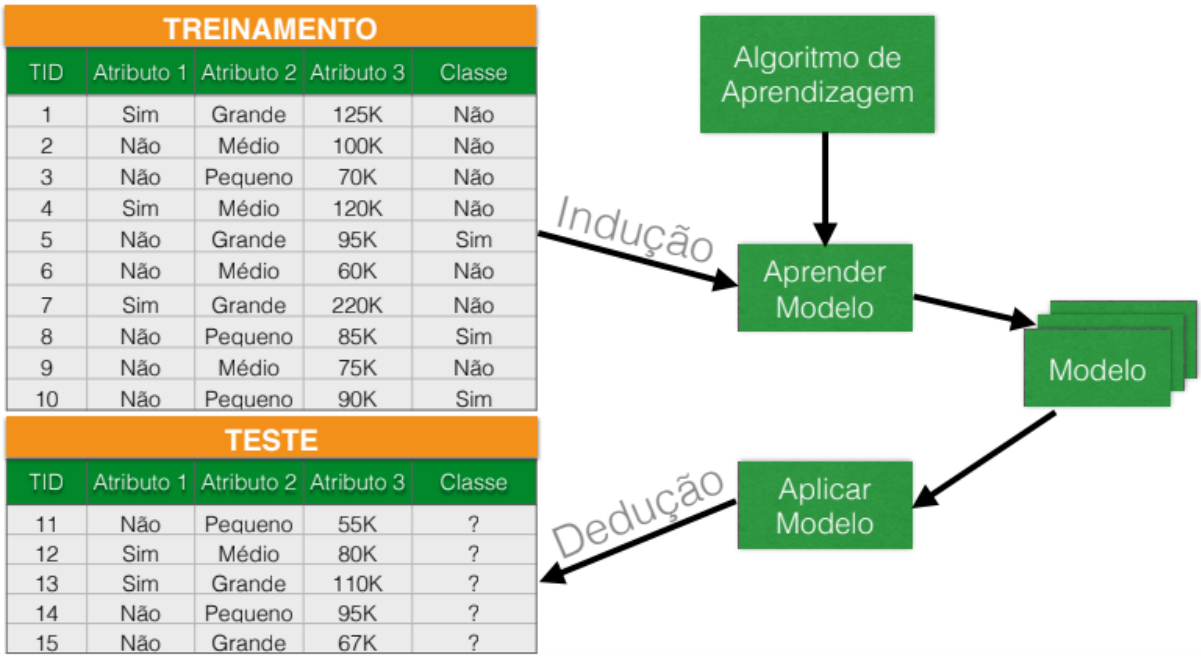
Nome	Temperatura Corporal	Cobertura de Pele	Dá cria	Ser Aquático	Ser Aéreo	Possui Pernas	Hiberna	Classe
Humano	Sangue Quente	Cabelo	Sim	Não	Não	Sim	Não	Mamífero
Piton	Sangue Frio	Escamas	Não	Não	Não	Não	Sim	Réptil
Salmão	Sangue Frio	Escamas	Não	Sim	Não	Não	Não	Peixe
Baleia	Sangue Quente	Cabelo	Sim	Sim	Não	Não	Não	Mamífero
Sapo	Sangue Frio	Nenhuma	Não	Sim	Não	Sim	Sim	Anfíbio
Dragão de Komodo	Sangue Frio	Escamas	Não	Não	Não	Sim	Não	Réptil
Morcego	Sangue Quente	Cabelo	Sim	Não	Sim	Sim	Sim	Mamífero
Pomba	Sangue Quente	Penas	Não	Não	Sim	Sim	Não	Ave

Fonte: Han (2011)

Com esses dados o modelo resumiria e destacaria quais características definem um vertebrado como mamífero, réptil, ave ou anfíbio (TAN, 2009). Como o rótulo de classe é fornecido, essa técnica também é conhecida como aprendizado supervisionado (HAN, 2006).

O modelo de classificação também pode ser usado para prever o rótulo da classe de registros não conhecidos, esse modelo é chamado de modelagem preditiva. O modelo pode ser tratado como uma caixa preta que atribui automaticamente um rótulo de classe quando recebe o conjunto de atributos de um registro desconhecido (TAN, 2009).

Figura 11 - Exemplificação do processo de Indução e Dedução.



Fonte: Han (2011)

A tarefa de classificação é um comportamento sistemático para a construção de modelos, a partir de um conjunto de dados de entrada. Modelos incluem técnicas de *Random Forest*, SVMs, Redes Neurais, *Decision Tree*, *Logistic Regression*, dentre outras. Cada técnica utiliza um algoritmo de aprendizagem para identificar um modelo que seja mais adequado para o relacionamento entre o conjunto de atributos e rótulo da classe dos dados de entrada (TAN, 2009). De acordo com os resultados encontrados no capítulo 3 deste trabalho, será seguido o enfoque nas técnicas *Decision Trees* e *Random Forest*.

4.2.1 Decision Trees

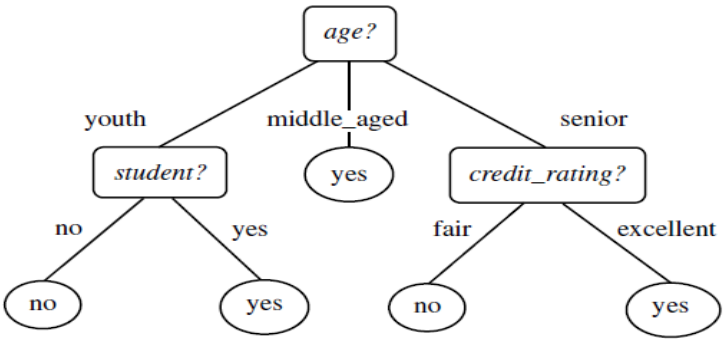
Segundo Faceli (2009) uma árvore de decisão usa a estratégia dividir para conquistar para resolver um problema de decisão. Um problema complexo é fracionado em problemas mais simples, os quais, recursivamente, usam a mesma estratégia. A resolução destes subproblemas pode ser combinada na estrutura de uma árvore, para gerar a solução de um problema complexo.

A árvore de decisão realiza seu aprendizado a partir de atributos de treinamento rotulados por classe. Ela apresenta uma estrutura do tipo fluxograma, em que cada nó interno

denota um teste em um atributo. Cada ramo representa um resultado do teste e cada folha contém um rótulo de classe (HAN, 2006).

Uma vantagem da árvore de decisão é sua fácil interpretação. Por esse motivo, a técnica se tornou muito popular e, às vezes, preferida sobre outras técnicas mais precisas, porém mais difíceis de interpretar (ALPADYN, 2004). A Figura 12 representa uma árvore de decisão para predizer se um cliente de uma loja compraria um computador baseado nas informações de idade, estudo e crédito. Os retângulos representam os nós internos enquanto as folhas são representadas pelos ovais.

Figura 12 - Representação de uma árvore de decisão



Fonte: HAN (2006)

4.3.2 *Random Forest*

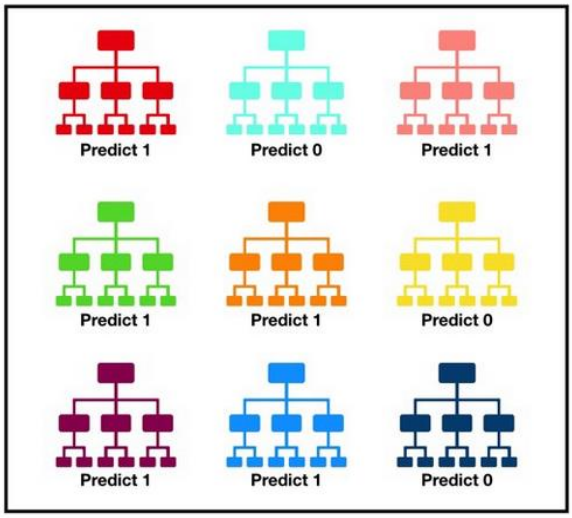
Nos últimos anos, o classificador *Random Forest* vem ganhando grande atenção devido aos seus excelentes resultados e a velocidade de processamento. O classificador produz suas predições a partir de um conjunto de árvores de decisão (BELGIU, 2016).

Breiman desenvolveu a metodologia de *Random Forest* como uma extensão das árvores de decisão usando o método de *bagging*. A técnica possui as características adicionais da seleção aleatória de recursos em cada nó e nenhuma regra de remoção ou parada (ARCHER, 2008).

Segundo Breiman (2001), melhorias significativas na precisão da classificação resultaram no crescimento de um conjunto de árvores e do seu voto na classe mais popular. Para desenvolver esses conjuntos, muitas vezes são gerados vetores aleatórios que governam o crescimento de cada árvore do conjunto.

A técnica de *bagging* é usada em conjunto com a seleção de características aleatórias. Cada novo conjunto de treinamento é desenhado com substituição, do conjunto de treinamento original. Em seguida, uma árvore é desenvolvida no novo conjunto de treinamento usando a seleção aleatória de recursos. O *bagging* pode ser usado para fornecer estimativas contínuas do erro de generalização do conjunto combinado de árvores, bem como estimativas para a força e a correlação. Estas estimativas são feitas *out-of-bag* (BREIMAN, 2001).

Figura 13 – Representação de árvores de decisão geradas pelo Random Fore



Fonte: (YAN, 2019)

A Figura 13 detalha a representação de uma aplicação de *Random Forest*. Foram geradas 9 árvores de decisões, usando atributos aleatórios de um mesmo conjunto de dados. Cada árvore não tem correlação com a outra, resultando em um rótulo de classe para cada uma. Ao final, o rótulo mais recorrente na votação é atribuído a instância, como podemos ver na Figura 9, nesse caso, o modelo atribuído à instância é a “predição 1”.

Neste capítulo foram investigadas estratégias de pré-processamento de dados, bem como algumas técnicas de classificação. Selecionou-se abordar Árvores de Decisão e *Random Forest*, em virtude dos trabalhos relacionados que foram discutidos no Capítulo 3. O próximo capítulo aplicará *Random Forest* sobre a base de dados de registros eletrônicos de saúde.

5. APLICAÇÃO DE *RANDOM FOREST* EM UM *DATASET*

Neste capítulo temos como propósito a utilização de uma técnica de *machine learning*, aplicando a técnica escolhida no capítulo 3, *Random Forest*. O objetivo deste trabalho é encontrar um modelo de classificação que possa predizer doenças prévias de pacientes, com base nos registros eletrônicos de uma cooperativa de saúde. Nesses registros serão utilizados, como atributos descritivos, exames e procedimentos realizados por esses pacientes ao longo do tempo, além de atributos como sexo, idade, cid, cidade, dentre outros.

Para atingir esse objetivo, foi realizado, inicialmente, o pré-processamento dos dados. Essa etapa ocupou a maior parte do tempo, se comparada ao tempo da aplicação da técnica. Na sequência serão exibidos os resultados do modelo de classificação e explicadas as métricas aplicadas.

5.1 SELEÇÃO DE DADOS

Inicialmente, foi realizada a seleção dos dados e a verificação da quantidade de registros em um período de um ano de exames e procedimentos. Foi contabilizado 26.013.528.427 (vinte e seis bilhões e treze milhões e quinhentos e vinte e oito mil e quatrocentos e vinte e sete) de registros. Com essa grande quantidade de dados e um grande custo de processamento necessário, não tendo recursos computacionais suficientes, foi decidido utilizar um critério para a redução desse volume de dados.

Para uma análise inicial, foi realizada uma consulta de uma pequena porção dos dados, 300.740 registros, equivalente a dois dias. Primeiramente foram categorizados os registros da tabela “pro_fat” (que registra todos os materiais utilizados, procedimentos e medicamentos realizados pelo paciente), separados em 9 tipos, como é exemplificado na Tabela 18.

Foram mantidos somente os dados do tipo “EXAMES E DIAGNOSTICOS”, os quais tem maior influência para o objetivo da pesquisa. Os dados da tabela “sinal_vital” e “itcoleta_sinal_vital” também foram removidos, pois apresentavam grande quantidade de dados pela frequência que são coletados. Devida a ausência de um profissional da área mais próximo do estudo, não foi possível definir quais sinais vitais teriam maior importância e poderiam ser mantidos. Assim, removendo os dados citados acima e realizando a busca novamente com o período de 01/01/2018 a 31/12/2018, foi retornado o total de 2.868.460 (dois milhões oitocentos e sessenta e oito mil e quatrocentos e sessenta) registros e 32 colunas.

Tabela 18 - Categorização dos registros da tabela pro_fat.

Tipo	Quantidade de registros
MATERIAIS	126666
MEDICAMENTOS	126000
EXAMES E DIAGNOSTICOS	31610
TAXAS	6601
DIARIAS	5467
HONORARIOS MEDICOS	2891
GASES MEDICINAIS	1036
PACOTES	255
MATERIAIS ESPECIAIS (OPME)	5

Fonte: O Autor (2019)

Para iniciar a etapa de pré-processamento, foi selecionada uma linguagem de programação, a linguagem de programação *Python* na versão 3.6.8. *Python* é uma linguagem de programação de alto nível, projetada para uma fácil leitura e implementação e de código aberto. Utilizou-se também a biblioteca pandas, a qual trabalha com estrutura de dados e ferramentas de análise de dados.

Realizada a coleta os dados foram transformados em um *DataFrame.* Um *DataFrame* é uma estrutura bidimensional de dados, como uma planilha. Entretanto, ao manipular os dados na forma de um *DataFrame*, adquirimos facilidade na visualização dos dados. Com o *DataFrame* começou a etapa de pré-processamento. Foi iniciada a seleção de dados, eliminando atributos irrelevantes e a redução de ruído. Foram removidos do *DataFrame* as colunas que continham códigos do sistema como: CD_ATENDIMENTO, CD_CID, CD_GRUPO, CD_IBGE, CD_LOCALIDADE, CD_PACIENTE, CD_PERGUNTA, CD_PRO_FAT, CD_RESPOSTA e COD_HISTORICO. A coluna TP_SANGUINEO e COR_PELE tinham 98% e 90% respectivamente em quantidade de registros nulos, sendo assim também removidas.

Devido às aplicações onde são inseridos estes dados, uma grande quantidade de registros não apresentava a pergunta “Doenças Prévias?”, foco desta pesquisa. Isto ocorre em função de as aplicações permitirem às enfermeiras que insiram as perguntas que desejam e preencham manualmente as respostas dadas pelos pacientes.

Logo, para um melhor desempenho da ferramenta, foram removidos todos os registros em que a pergunta não fosse “Doenças Prévias?”. Foram selecionados, portanto, os registros com a pergunta “Doenças Prévias?”, e foi criada uma coluna denominada DOENCA_PREVIA, cujos valores possíveis, são as doenças indicadas pelos pacientes ou sua negação (o paciente não tem doença prévia). Ao final o *DataFrame* totalizou 165.812 (cento e sessenta e cinco mil oitocentos e doze) registros e 7 colunas, como exemplificado na Figura 14.

Figura 14 - Registros e colunas dentro de um DataFrame

	DOENCA_PREVIA	DS_CID	DS_PRO_FAT	DT_NASCIMENTO	NM_BAIRRO	NM_CIDADE	TP_SEXO
0	DOENÇAS NEUROLÓGICAS	ANOREXIA	SODIO	1926-12-20 02:00:00	Feitoria	Sao Leopoldo	F
1	HIPERTENSÃO	ANOREXIA	SODIO	1926-12-20 02:00:00	Feitoria	Sao Leopoldo	F
2	DOENÇAS CARDIOVASCULARES	ANOREXIA	SODIO	1926-12-20 02:00:00	Feitoria	Sao Leopoldo	F
3	DOENÇAS NEUROLÓGICAS	ANOREXIA	POTASSIO	1926-12-20 02:00:00	Feitoria	Sao Leopoldo	F
4	HIPERTENSÃO	ANOREXIA	POTASSIO	1926-12-20 02:00:00	Feitoria	Sao Leopoldo	F

Fonte: O Autor (2019)

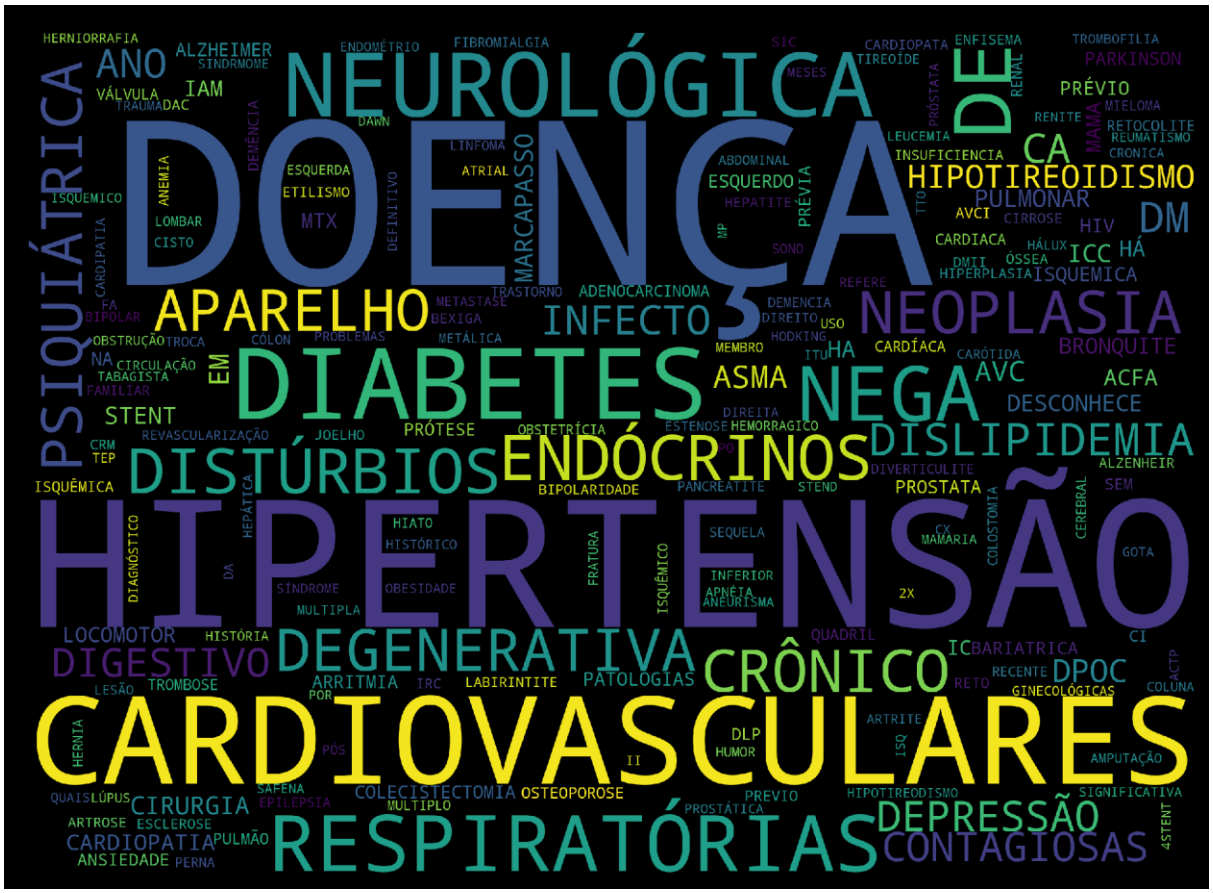
As colunas foram selecionadas seguindo o critério informado por um especialista da área. De acordo com o especialista informações importantes nesse contexto são as seguintes: “DOENCA_PREVIA”, a qual será usada como rótulo de classe da desta pesquisa; “DS_CID”, o qual indica para qual doença o paciente está sendo tratado no momento do atendimento; “DS_PRO_FAT”, que representa os exames e procedimentos realizados pelo paciente; “NM_BAIRRO”, o qual indica o bairro onde reside o paciente; “NM_CIDADE”, indicando a sua cidade de residência; “TP_SEXO”, indicando o sexo entre Masculino e Feminino e “DT_NASCIMENTO”, o qual é o registro da data de nascimento do paciente.

5.2 TRANSFORMAÇÃO DOS DADOS

Aplicada a transformação de dados, foram calculadas as idades, utilizando como referência a data de 25/09/2019, transformando a coluna “DT_NASCIMENTO” do tipo date para um valor inteiro, calculado em anos e renomeado para “AGE”. Logo depois foi aplicada uma segunda transformação, discretizando a coluna “AGE” em faixas, conforme a tabela de faixas de idade no site da Agência Nacional de Saúde Suplementar. A coluna “TP_SEXO” foi transformada em registros numéricos, onde 0 representa o sexo masculino e 1 o sexo feminino.

Verificando os valores preenchidos na coluna “DOENCA_PREVIA”, foram contabilizados 994 tipos diferentes de rótulos de classe, como por exemplo: HIV, Alzheimer, Diabetes, etc. Para uma melhor representação, foi utilizada a biblioteca *wordCloud*, a qual constrói uma nuvem de palavras. A representatividade (tamanho da palavra) de cada rótulo de classe é denotada pelo número de vezes em que ela aparece no *DataFrame*, como mostrado na Figura 15.

Figura 15 - Nuvem de Palavras geradas a partir dos registros de doenças prévias.



Fonte: O Autor (2019)

Foram encontradas 994 doenças prévias diferentes. Logo, com essa grande quantidade de rótulos de classe, os algoritmos de classificação escolhidos tendem a apresentar problemas para generalização do modelo, o que dificultaria a visualização dos dados e comprometeria o desempenho de suas classificações. Em uma primeira abordagem, foi reduzida a quantidade de rótulos da classe “DOENCA_PREVIA”. Foram selecionadas as doenças crônicas e grupos de doenças com maior ocorrência do que as outras na base de dados. Os registros de

“DOENCA_PREVIA” com negações, como por exemplo: não se aplica, não apresenta, nenhuma, etc, foram alterados para o valor “NEGA”.

5.3 ESTRATÉGIAS UTILIZADAS PARA APLICAÇÃO DO *RANDOM FOREST*.

Foram elaboradas três estratégias diferentes para a aplicação do classificador *Random Forest* na base de dados, que serão explicadas a seguir. Todas as estratégias utilizaram a biblioteca *scikit-learn*, uma biblioteca de aprendizado de máquina de código aberto para a linguagem de programação *Python*, ela inclui vários algoritmos de classificação, regressão e agrupamento.

Outro método utilizado em comum nas três estratégias foi o *one-hot-encoder*, que transforma registros do tipo *object* em colunas, convertendo os valores para uma representação numérica. Contendo o valor 0 onde ela não está presente e 1 onde existem ocorrências da mesma. Para melhor exemplificar a funcionalidade do método, são usados como exemplos os dias da semana dentro de um *DataFrame*, como pode ser observado nas Figura 16 e 17.

Figura 16 - Exemplificação dos dias da semana dentro de um *DataFrame*.

	Dias da Semana
0	Domingo
1	Sgunda
2	Terça
3	Quarta
4	Quinta
5	Sexta
6	Sabado

Fonte: O Autor (2019)

Para as pessoas em geral, dias da semana são informações intuitivas. Geralmente é sabido que Domingo é o primeiro dia da semana, mas o algoritmo não tem o conhecimento intuitivo, o que ele sabe é valor numérico correspondente ao dia da semana, por isso é realizada

essa transformação. Poderia ser feito o mapeamento dos dias da semana em valores de 1 a 7, porém, isso poderia fazer com que o algoritmo colocasse maior importância no Sábado, por que teria o maior valor numérico. Por esse motivo é realizada a transformação da coluna “dias da semana” em sete colunas com dados binários, como exemplificado na Figura 17.

Figura 17 - Exemplificação da técnica *one-hot-encoder*.

	Dias da Semana_Domingo	Dias da Semana_Quarta	Dias da Semana_Quinta	Dias da Semana_Sabado	Dias da Semana_Sexta	Dias da Semana_Sgunda	Dias da Semana_Terça
0	1	0	0	0	0	0	0
1	0	0	0	0	0	1	0
2	0	0	0	0	0	0	1
3	0	1	0	0	0	0	0
4	0	0	1	0	0	0	0
5	0	0	0	0	1	0	0
6	0	0	0	1	0	0	0

Fonte: O Autor (2019)

5.3.1 Estratégia inicial: configurando parâmetros manualmente

Inicialmente o *DataFrame* foi reduzido para registros que continham as doenças prévias mais frequentes. Utilizou-se as doenças com uma frequência maior ou igual a 1000 registros, totalizando quinze doenças prévias, exemplificadas na Tabela 9. O *DataFrame* ficou com 165.812 registros, mantendo as 7 colunas iniciais.

Tabela 19 - Quinze maiores ocorrências no *DataFrame*.

Doenças Prévias		
HIPERTENSÃO	DOENÇAS RESPIRATÓRIAS	DOENÇAS INFECTO-CONTAGIOSAS
DOENÇAS CARDIOVASCULARES	DISTÚRBIOS ENDÓCRINOS	DOENÇAS DO APARELHO DIGESTIVO
DIABETES	NEOPLASIAS	DISLIPIDEMIA
DOENÇAS NEUROLÓGICAS	DOENÇAS CRÔNICO DEGENERATIVAS	ASMA
NEGA	DOENÇA PSIQUIÁTRICA	BRONQUITE

Fonte: O Autor (2019)

Para realizar a predição, primeiramente é preciso dividir a classe alvo dos atributos descritivos. Usando os recursos da biblioteca pandas, a classe alvo foi separada e recebeu o nome de “y”, e o restante dos atributos recebeu o nome de “X”. A fim de iniciar o treinamento, primeiro foi verificado o tipo dos dados do *DataFrame*. Algoritmos de classificação, como o *Random Forest*, na biblioteca *scikit-learn* trabalham geralmente com números, não sendo possível realizar a classificação com textos, identificados na biblioteca pandas como *object*. Um recurso disponível na biblioteca pandas é o método *info*, o qual descreve o tipo de dado presente em cada atributo do *DataFrame*. Na Figura 18 exemplifica-se a aplicação do método *info*.

Figura 18 - Exemplificação do método info.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 165812 entries, 0 to 165811
Data columns (total 7 columns):
DOENCA_PREVIA    165812 non-null object
DS_CID           130297 non-null object
DS_PRO_FAT       165812 non-null object
NM_BAIRRO        149252 non-null object
NM_CIDADE        165812 non-null object
TP_SEXO          165812 non-null int64
age              165812 non-null float64
dtypes: float64(1), int64(1), object(5)
memory usage: 8.9+ MB
```

Fonte: O Autor (2019)

Observa-se na Figura 18 que muitos valores contidos nas colunas do *DataFrame* são do tipo *object*. Para solucionar esse problema, foi usada a técnica *one-hot-encoder*. Depois de realizada a aplicação da técnica *one-hot-encoder*, o *DataFrame* foi expandido. Ele ficou com um total de 956 atributos descritivos e a classe alvo com 15 colunas. Para o treinamento e teste da técnica *Random Forest*, foi necessária a separação de dados em quatro conjuntos distintos. Foi usado o *train_test_split*, método que recebe os atributos descritivos e a classe alvo, decompondo aleatoriamente em quatro conjuntos. Ele separa o conjunto de treinamento em 75% dos dados e o conjunto de teste com os 25% restantes. A geração desses conjuntos foi realizada utilizando-se o parâmetro *random_state* com o valor de 42 para o *random.seed*, a fim de ser possível reproduzir os mesmo dados posteriormente, caso necessário. Como resultado final, tem-se quatros conjuntos de dados: “X_test, X_train, y_test, y_train”.

Com os dados separados, foi iniciada a etapa de classificação. Foi usada a biblioteca *scikit-learn* que contém os métodos de classificação do *Random Forest*, definidos como *RandomForestClassifier*. Com objetivo de buscar os melhores parâmetros para o modelo, inicialmente foram configurados três classificadores de *Random Forest*, a fim de avaliar qual dele teria a melhor acurácia. Os três classificadores foram configurados com o parâmetro *random_state* igual a 42. No primeiro classificador o parâmetro "número de árvores" foi definido em 500, enquanto no segundo e terceiro classificador, foram usados os valores 10 e 100, respectivamente.

Ao executar o algoritmo, é necessário usar os conjuntos de dados "X_train" e "y_train" como entrada para a etapa de treinamento, a fim de induzir o modelo. Logo em seguida, é enviado ao modelo, o conjunto de dados "X_test", a fim de predizer os valores de doenças prévias da classe alvo. Por fim, é utilizado o conjunto de teste "y_test" para avaliar a acurácia do modelo gerado, a partir da aplicação do método *accuracy_score* da biblioteca *scikit-learn*. Para uma melhor visualização, foi montada a Tabela 20, contendo os resultados dos conjuntos de treinamento e conjuntos de testes.

Tabela 20 - Resultado dos classificadores do método *Random Forest*

	Classificador 1	**Classificador 2**	**Classificador 3**
Número de Árvores	500	10	100
Acurácia de Testes	0.2322	0.2322	0.2341

Fonte: O Autor (2019)

A Tabela 20 denota que nenhum dos classificadores obteve uma acurácia adequada, pois o melhor classificador obteve 23,41% de acurácia. Motivado pela baixa precisão dos resultados e pouco aproveitamento dos recursos da técnica, buscou-se estratégias automatizadas para configuração dos parâmetros do *Random Forest*. Não foram realizadas as verificações de correlação no *DataFrame* pelo motivo que após a expensão das colunas usando a técnica do *one-hot-encoder*, foi encontrada dificuldades de exibir e validar a correlação entre eles.

5.3.2 *Hyper Parameters*

Muitas técnicas de aprendizado supervisionado são baseadas em um único modelo de predição. A ideia principal da técnica de *hyper parameters* é iniciar com um modelo simples,

por exemplo, uma árvore de decisão com apenas algumas divisões, e aumentar sequencialmente seu desempenho, continuando a construir árvores onde cada nova árvore na sequência, tenta consertar onde a anterior cometeu os maiores erros, como exemplificado na Figura 19.

Figura 19 - Representação da técnica *Hyper Parameters*.

Fonte: O Autor (2019)

Para esta técnica ser aplicada, foi utilizado o método *RandomizedSearchCV*, disponível na biblioteca *scikit-learn*. que recebe um dicionário de possíveis parâmetros usados para avaliar o classificador que será avaliado, a métrica de avaliação dos resultados, os atributos descritivos e a classe alvo. Foi utilizado como métrica de desempenho a acurácia.

Os parâmetros definidos foram: *N_estimators*, referente ao número de arvores que serão criadas, quanto maior o número, melhor é o desempenho. É importante ressaltar que quanto maior o valor, maior se torna o custo computacional. *Max_depth*, o valor define a profundidade máxima da árvore e por quantas vezes será feita a divisão. Se não definida, será realizada até que todas as folhas sejam puristas, porém, pode ocasionar em um *overfitting*. *Max_features*, valor que define o número de recursos a serem considerados ao procurar a melhor divisão. Esse recurso melhora o desempenho do modelo, já que cada nó de cada árvore estará considerando um número maior de opções, mas segue o mesmo princípio do *n_estimators*, quanto maior o valor, maior o custo computacional. *Max_leaf_nodes* é responsável por aumentar a árvore da melhor maneira possível, resultando em uma redução relativa na impureza. *Min_sample_split* é o valor que define o número mínimo de amostras que devem estar presentes nos dados para que uma divisão ocorra. Os valores enviados para uma melhor visualização, foram construídos em uma tabela com o valor inicial e final, abrangendo todos os valores dentro do intervalo definido. Os parâmetros foram selecionados de acordo com Koehrsen, O ajuste do hiperparâmetro depende mais de resultados experimentais do que da teoria e, portanto, o

melhor método para determinar as configurações ideais é tentar muitas combinações diferentes para avaliar o desempenho de cada modelo (2018). Tabela 21 exemplifica os valores enviados no dicionário.

Tabela 21 – Parâmetros enviados no dicionário para avaliação do melhor modelo, representando o valor inicial e final, abrangendo todos os possíveis números dentro do intervalo.

Parâmetro	Valor Inicial	Valor Final
n_estimators	10	200
max_depth	3	20
max_features	0.1	1
max_leaf_nodes	10	500
min_samples_split	2	10

Fonte: O Autor (2019)

Aplicada a técnica *one-hot-encoder*, uma alteração na etapa de separação foi realizada antes de iniciar a aplicação da técnica de *Hyper Parameters*. Para uma maior representatividade dos dados, foi utilizada a estratégia de estratificação, mantendo a proporção dos mesmos. Os outros parâmetros se mantiveram iguais. Foi utilizado o parâmetro *random_state* igual a 42, conforme realizado na seção 5.3.1. Após a aplicação do *Hyper Parameters*, os resultados dos melhores parâmetros podem ser observados na Tabela 22.

Tabela 22 – Resultado da técnica de Hyper Parameters.

Parâmetro	Valor
n_estimators	196
max_depth	17
max_features	0.7
max_leaf_nodes	49
min_samples_split	10

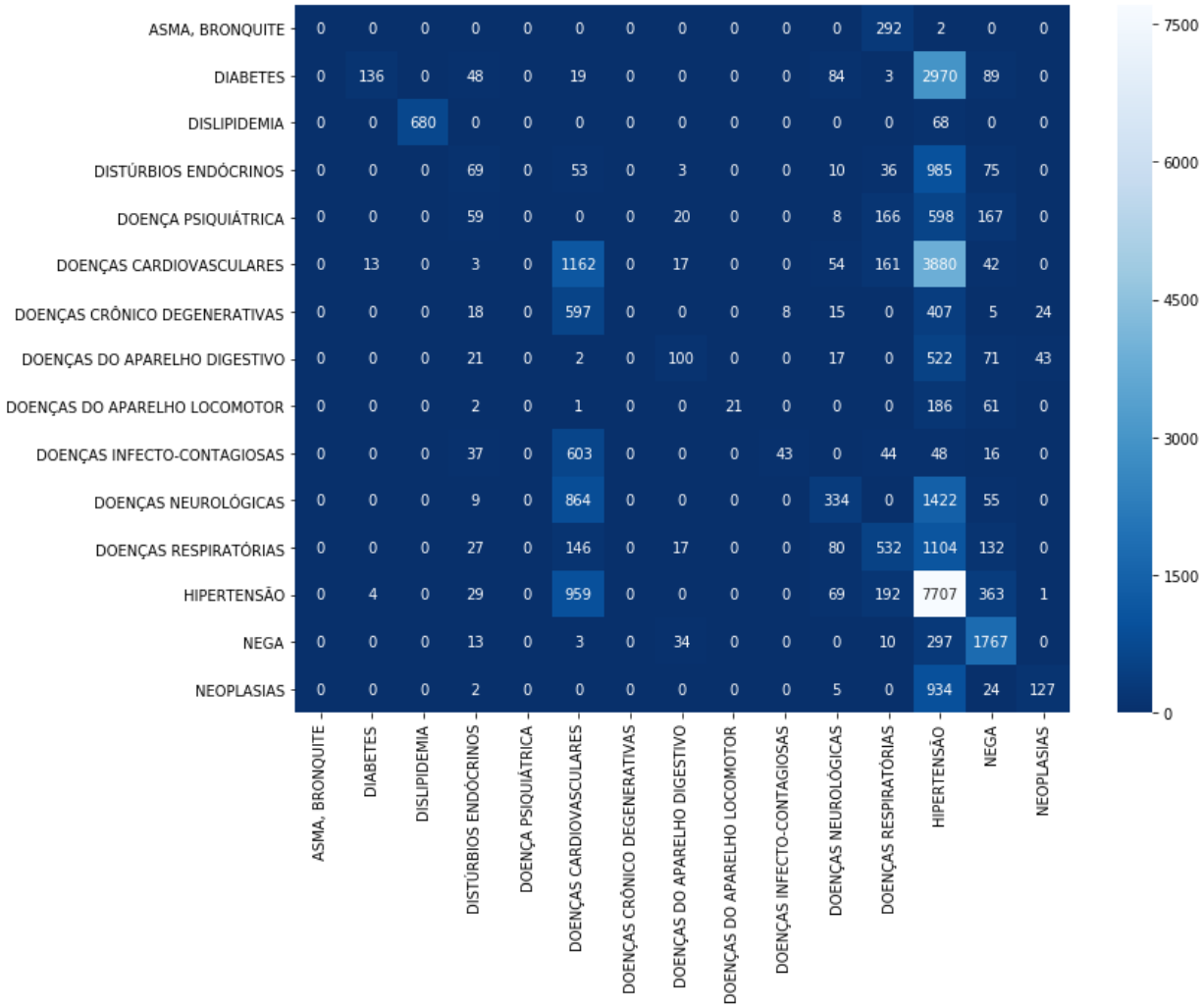
Fonte: O Autor (2019)

Definidos os melhores parâmetros, segundo a técnica de *Hyper Parameters*, foi instanciado o classificador, induzindo o modelo com os valores resultantes da técnica de separação de dados estratificada. Posteriormente foi enviado ao modelo os dados de testes para a dedução da classe alvo. Utilizado novamente o método *accuracy_score* para avaliar a acurácia, o resultado foi de 39.46%, 16.05% maior que o melhor visto na seção 5.3.1, a qual não usa a técnica de *Hyper Parameters*.

Na intenção de visualizar os dados que foram preditos, foi construída uma matriz de confusão, usando o método *confusion_matrix* da biblioteca *scikit-learn*. Uma matriz de

confusão é uma métrica voltada para modelos de classificação e tem como objetivo calcular a quantidade de falso positivo, falso negativo, verdadeiro positivo e verdadeiro negativo, além de fornecer a acurácia, sensibilidade e especificidade. Na esquerda, ficam os valores reais da classe alvo, e embaixo os resultados que foram deduzidos pelo modelo. A Figura 20 exemplifica a matriz de confusão.

Figura 20 – Matriz de confusão criada a partir dos resultados do modelo.



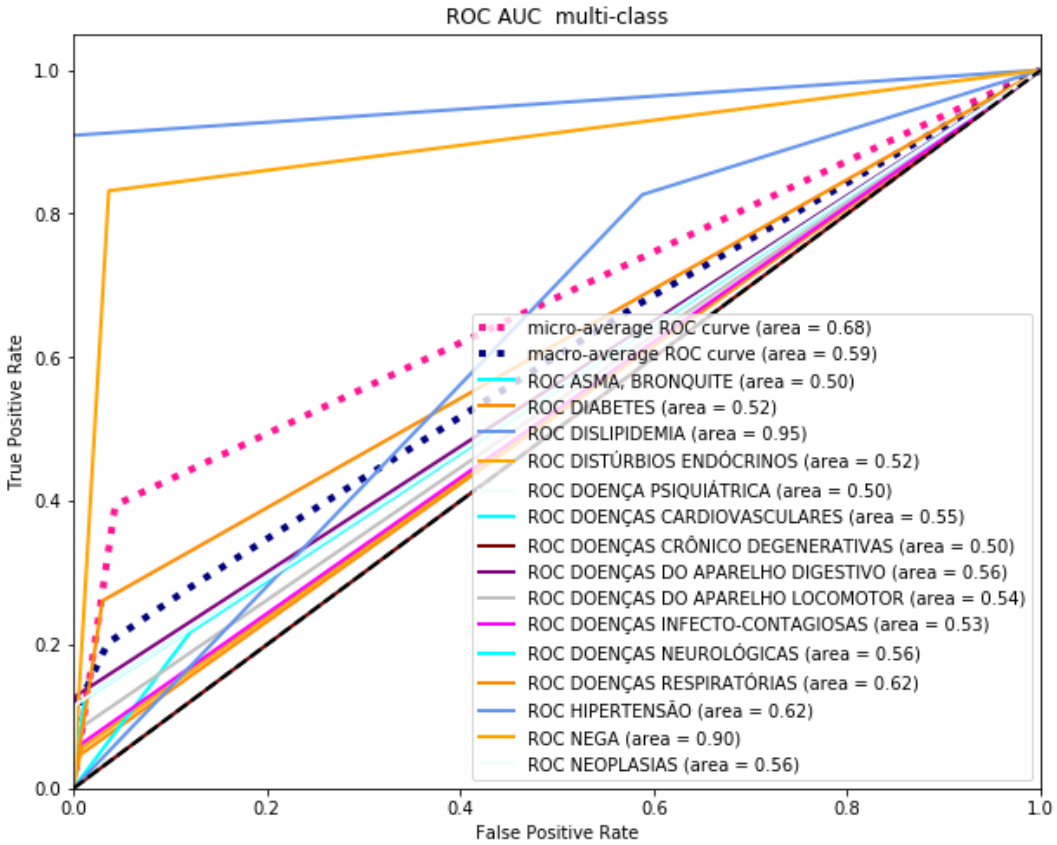
Fonte: O Autor (2019)

Observando a matriz de confusão, é percebido que a grande maioria dos rótulos de classe foram classificados como “HIPERTENSÃO”. Foi utilizado o método *classification_report*, recebendo como parâmetro os mesmos conjuntos de dados enviados para a matriz de confusão. Este método retorna valores que representam as métricas de precisão, sensibilidade e especificidade. A métrica de precisão define o valor de quantas classificações

verdadeiramente positivos foram induzidas no modelo. A sensibilidade é a proporção de verdadeiros positivos, e a especificidade é a proporção de verdadeiros negativos. Foram retornados os valores 0.44 para a precisão, 0.39 referente a sensibilidade e 0.39 para especificidade. Estes valores são providos sobre a média referente a cada rótulo de classe.

Outra métrica avaliada foi a ROC (*Receiver Operation Characteristics*) e AUC (*Area Under the Curve*). São duas métricas muito utilizadas para modelos de classificação. Segundo Prati (2008) a métrica ROC é baseada nas taxas de verdadeiros positivos e falsos positivos. A AUC é a probabilidade de o modelo classificar um exemplo positivo aleatório seja pontuado acima de um exemplo negativo aleatório. Segundo Paulino (2018), matematicamente, seja a função *f* do classificador, que recebe uma entrada "x" e retorna à probabilidade de *f*(x) de "x" ser um caso positivo. Ao ser definido um valor *T* para que o classificador responda sim quando $f(x) > T$ ou não quando $f(x) < T$. A taxa de verdadeiros positivos (TPR) é a razão entre a quantidade de casos positivos para os quais o classificador disse sim e a quantidade de casos positivos. A taxa de falsos positivos (FPR) é a razão entre a quantidade de casos negativos para os quais o classificador disse sim e a quantidade de casos negativos. Assim a curva *roc* é definida pelos pontos FPR e TPR, onde $T \in (-\infty, \infty)$. Foi utilizado o método *roc_curve* para cada rótulo de classe, retornando os valores de falso positivo e verdadeiro positivo. Esses valores foram enviados para o método *auc* a fim de calcular a área sob a curva. A Figura 21 exemplifica o gráfico AUC.

Figura 21 – Gráfico AUC sobre a curva ROC.



Fonte: O Autor (2019)

Como pode ser observado no gráfico, a média macro dos rótulos de classe apresentou o valor de 0.59, enquanto a média micro apresentou o valor de 0.68. A média macro calcula a métrica independentemente para cada classe e, em seguida, obterá a média, tratando todas as classes igualmente. A média micro agrega todas as classes para calcular a métrica. Alguns rótulos de classe apresentaram um valor discrepante, devido à falta de balanceamento dos dados.

Como pode ser visualizado na matriz de confusão, muitos registros classificados como HIPERTENSÃO foram classificados como DOENÇAS CARDIOVASCULARES. Desta forma, foi solicitado a ajuda de um especialista da área para realizar o agrupamento dos rótulos de classe pelo tipo de uma doença específica em seu grupo de doenças. Foram informadas as doenças especificas com maior ocorrência, como DIABETES, HIPERTENSÃO e os grupos de doenças especificas com maior ocorrência, como DOENÇAS CARDIOVASCULARES e DISTÚRBIOS ENDÓCRINOS.

Tabela 23 – Agrupamento dos rótulos de classe exemplificando a classificação realizada pelo profissional da área.

Doenças respiratórias	Doenças cardiovasculares	Distúrbios endócrinos	Doenças do aparelho digestivo	Neoplasias	Doença psiquiátrica	Doenças neurológicas	Doenças do aparelho reprodutor
Dpoc	Hipertensão (has)	Diabetes melitus (1 e 2 - dm)	Retocolite	Ca de mama	Depressão	Alzenheir	Cirurgia de próstata
Asma	Cardiopatia isquêmica (ci)	Dislipidemia	Cirurgia de bariatrica ha 4 anos		Etilismo		
Bronquite	Hipertensão pulmonar	Hipotireoidismo	Colecistectómica há 1 ano				
	Insuficiência cardíaca (ic)	Obesidade	Doenças do aparelho digestivo				
	Arritmias		Retocolite				
	Marcapasso						

Fonte: O Autor (2019)

5.3.3 Agrupamento dos Rótulos de Classe

Com o retorno das informações fornecidas pelo profissional da área, foi possível realizar o agrupamento. Foram classificadas entre as maiores ocorrências dos rótulos de classe específicos, como por exemplo: DIABETES E HIPERTENSÃO, e agrupados nos rótulos de classe, exibido na Tabela 23.

Realizado o agrupamento dos dados seguindo as orientações do especialista da área, foi reduzido 24 rótulos de classe do total de 994 rótulos inicias. Foram selecionados os grupos de doenças o número mínimo de 1000 ocorrências, resultando em 11 rótulos de classe. O *Dataframe* ficou com um total de 136.780 registros, uma redução de 29.032 registros se comparado a quantidade do *DataFrame* anterior, exemplificado na Figura 22 com a sua representatividade.

Figura 22 – Rótulos de classe agrupados e sua representatividade no *DataFrame*.

```
DOENÇAS CARDIOVASCULARES         61044
DISTÚRBIOS ENDÓCRINOS            22509
DOENÇAS NEUROLÓGICAS             11060
DOENÇAS RESPIRATÓRIAS            10625
NEGA                              8770
DOENÇA PSIQUIÁTRICA               5316
NEOPLASIAS                        4901
DOENÇAS CRÔNICO DEGENERATIVAS     4298
DOENÇAS DO APARELHO DIGESTIVO     4007
DOENÇAS INFECTO-CONTAGIOSAS       3166
DOENÇAS DO APARELHO LOCOMOTOR     1084
Name: DOENCA_PREVIA, dtype: int64
```

Fonte: O Autor (2019)

Observando a Figura 22, fica claro o desbalanceamento dos dados, ocasionando em problemas já perceptíveis na seção 5.3.2. Em uma análise mais aprofundada nos dados, foi percebida a existência de dados duplicados, devido à falta de agrupamento no momento da seleção dos mesmos. Para solucionar o problema, foi utilizado o método *duplicate* disponível na biblioteca *scikit-learn*. Este método retorna uma informação booleana para cada registro que já estiver no *DataFrame*, sendo possível removê-los. Executando o método, foram retornados 38.824 registros, reduzindo o *Dataframe* para 97.956 registros. Observando novamente a representatividade dos rótulos de classe, ainda é perceptível o desbalanceamento.

Com a finalidade de evitar o *overfitting*, foi realizado um balanceamento nos dados, criando um novo *DataFrame* com a quantidade de rótulos de classe em igualdade. Foram selecionados os grupos de doenças presentes nos rótulos de classe com mais de 1000 (mil) ocorrências, representados na Figura 23.

Figura 23 – Rótulos de classe agrupados com mais de 1000 ocorrências.

```
DOENÇAS CARDIOVASCULARES         15171
DISTÚRBIOS ENDÓCRINOS             6928
NEGA                              5192
DOENÇAS NEUROLÓGICAS              2906
DOENÇAS RESPIRATÓRIAS             2551
DOENÇA PSIQUIÁTRICA               1834
NEOPLASIAS                        1529
DOENÇAS DO APARELHO DIGESTIVO     1179
```

Fonte: O Autor (2019)

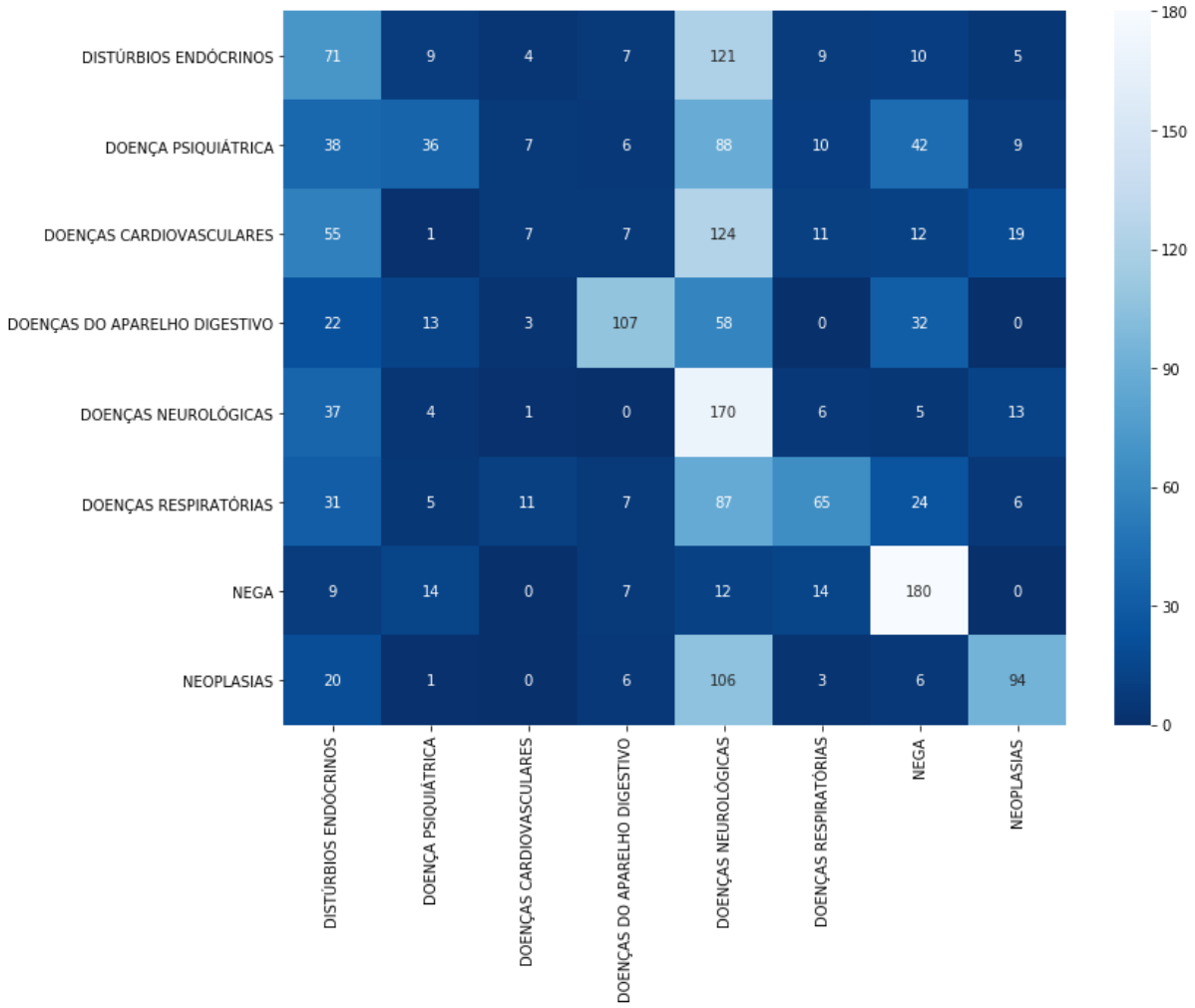
Para realizar o balanceamento, foi criado um novo *DataFrame* contendo no máximo 1179 registros de cada rótulo, valor definido pela quantidade de ocorrências do rótulo de classe "DOENÇAS DO APARELHO DIGESTIVO". Utilizando a técnica *one-hot-encoder*, *train_test_split* estratificado com a divisão de 80% de dados para treinamento e 20% para teste. Realizando a definição do classificador *Random Forest* com os parâmetros resultantes da técnica de *Boosting*, foi induzido o modelo com o *DataFrame* balanceado e sem duplicados.

Enviado ao modelo os dados de testes para a dedução da classe alvo, foi utilizado novamente o método *accuracy_score* para avaliar a acurácia, o resultado foi de 38.68%. Foi retornando os valores 0.45 para a precisão, 0.39 referente a sensibilidade e 0.38 para especificidade. Estes valores são fornecidos a partir da média referente a cada rótulo de classe. A matriz de confusão pode ser observada na Figura 24.

Com uma melhor distribuição nos rótulos de classe não ocorreu *overfitting*, como no *DataFrame* desbalanceado. Ainda que muitos erros tenham sido cometidos, como por exemplo as deduções realizadas para o rótulo de classe "DOENÇAS NEUROLÓGICAS", o modelo apresentou um resultado melhor. O gráfico AUC apresentou a média macro e micro dos rótulos de classe o valor de 0.65, representado pela Figura 25.

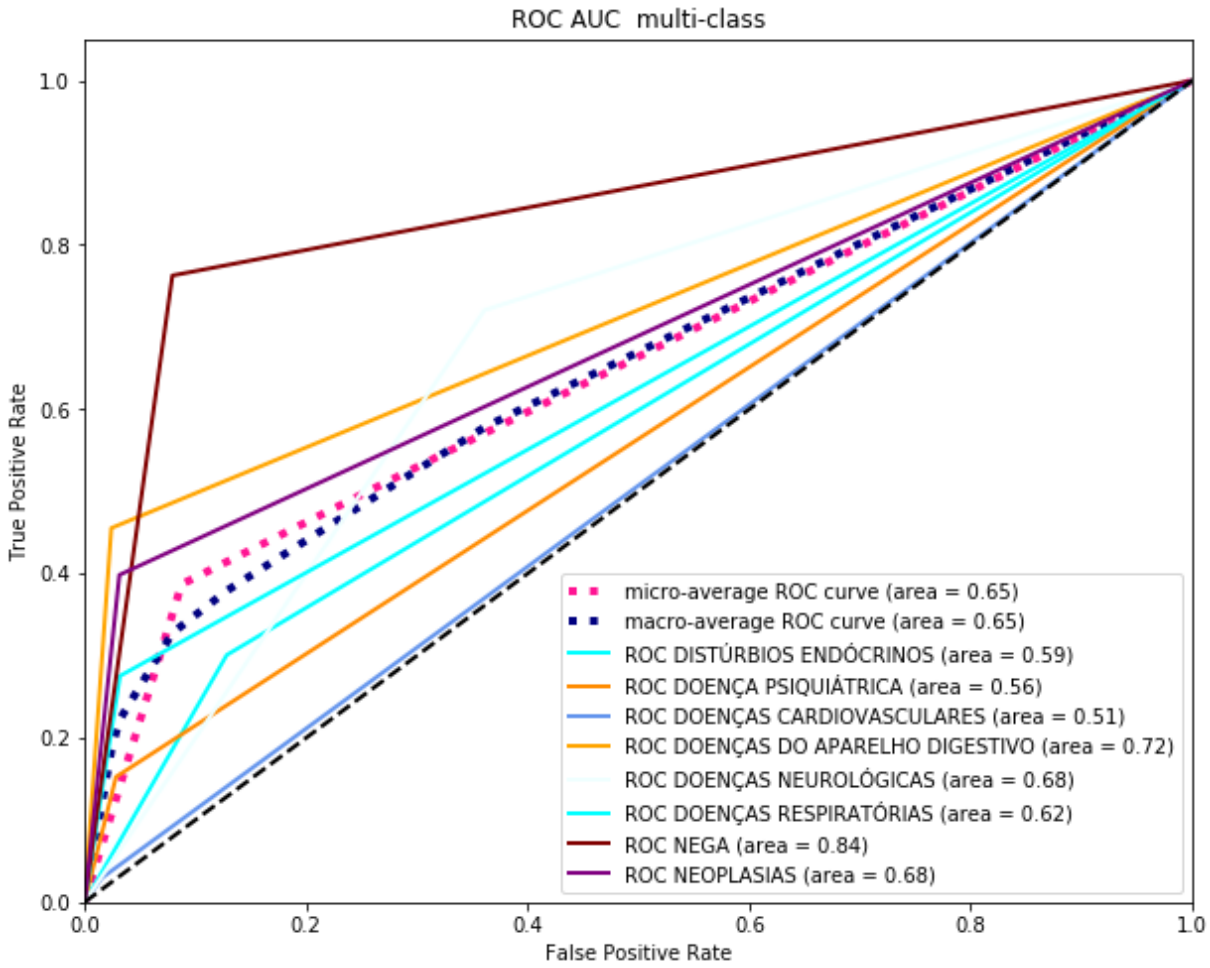
Buscando sintetizar as informações coletadas nas estratégias realizadas neste capítulo, foi construída uma tabela para agrupar os resultados obtidos, Tabela 24.

Figura 24 – Matriz de confusão do modelo de indução com os atributos balanceados e sem registros duplicados.



Fonte: O Autor (2019)

Figura 25 - Gráfico AUC da linha ROC do modelo de indução com os atributos balanceados e sem registros duplicados.



Fonte: O Autor (2019)

Tabela 24 – Quantificação dos resultados nas diferentes estratégias.

	Estratégia 1 capitulo 5.3.1	**Estratégia 2 capitulo 5.3.2**	**Estratégia 3 capitulo 5.3.3**
Quantidade de Atributos	128.581	128.581	38.824
Quantidade de Rótulos de Classe	15	15	8
Divisão dos atributos Estratificada	Não	Sim	Sim
Acurácia	0.2341	0.3946	0.3868
Precisão	Não aplicado	0.44	0.45
Sensibilidade	Não aplicado	0.39	0.39
Especificidade	Não aplicado	0.39	0.38
Micro Roc	Não aplicado	0.68	0.65
Macro Roc	Não aplicado	0.59	0.65

Fonte: O Autor (2019)

6 CONCLUSÃO

Neste trabalho realizou-se uma breve introdução sobre as operadoras de saúde, foi analisado como é realizada a geração dos dados eletrônicos de saúde. Apresentou-se os principais procedimentos médicos, seu conteúdo e volume contidos na base de dados da operadora, que foram selecionados para a utilização de técnicas de aprendizado de máquina.

Com o objetivo de levantar fundamentos e técnicas, foi realizado um estudo de trabalhos relacionados para a identificação de métodos de análise preditiva aplicadas a registros eletrônicos de saúde. Após realizado o estudo de trabalhos relacionados, organizou-se as técnicas de classificação mais usadas em diferentes cenários, tratamentos e agrupamentos de dados, tanto estruturados como não estruturados.

Após quantificar as técnicas de classificação apresentadas, a técnica com maior ocorrência foi a *Random Forest,* sendo aplicada em 4 dos 5 artigos analisados. Foi realizado, portanto, um estudo bibliográfico sobre as estratégias de pré-processamento de dados e as técnicas de análise preditiva *Decision Trees* e *Random Forest* para uma melhor aplicação nos dados eletrônicos de saúde.

A técnica de análise preditiva *Random Forest* foi aplicada nos dados selecionados, utilizando-se diferentes estratégias, visando uma melhor performance. Inicialmente foi aplicada a estratégia para uma melhor parametrização do classificador manualmente. Não tendo sucesso na performance apresentada, foi utilizada a técnica de *Hyper parameters Tunning* na busca dos melhores parâmetros para o modelo selecionado. Com o resultado e análise das métricas apresentadas na segunda estratégia, foi constatada a presença de *overfitting*. Na terceira estratégia, foi realizado o agrupamento dos rótulos de classe com o auxílio de um especialista da área, reduzindo o número de valores distintos no atributo alvo. Também na terceira estratégia, foi percebido uma grande quantidade de registros duplicados. Realizada a aplicação da técnica com dados balanceados para evitar o *overfitting*, o resultado teve uma performance melhor, entretanto continuou não satisfatório.

Durante o desenvolvimento deste trabalho, foram identificadas algumas limitações. A dificuldade de encontrar especialistas na área e a grande diversidade dos dados, causou uma falta de conhecimento nas informações mais relevantes para a pesquisa, como por exemplo os sinais vitais coletados e os procedimentos mais relevantes em cada rótulo de classe. Outro problema é a informação selecionada para o rótulo de classe, a pergunta da doença prévia.

Contendo 994 rótulos de classes diferentes, sem uma padronização presente, não foi possível aproveitar todas as informações coletadas.

Outro problema detectado foi a validação da veracidade das informações que são coletadas dos pacientes pelas enfermeiras. Pode ocasionar que o modelo desenvolvido nesta pesquisa classifique um paciente com uma doença prévia como por exemplo a Diabetes, mas na resposta estar presente a informação de negação. Não foi possível realizar o diagnóstico destes casos, para a validação da técnica.

Para trabalhos futuros, outras técnicas de aprendizado de máquina podem ser utilizadas, a fim de alcançar resultados com melhor acurácia. Em relação ao atributo alvo, um maior envolvimento de especialistas da área será solicitado para a realização da validação dos dados. Será realizada a representação das informações coletadas em formas mais amigáveis, como gráficos. Será realizada uma apresentação dos resultados encontrados na cooperativa de saúde, a fim de envolver os profissionais da área.

REFERÊNCIAS BIBLIOGRÁFICAS

ALPAYDIN, Ethem. Introduction to Machine Learning. 2° ed. Cambridge: The MIT Press, 2010.

AMARAL, Fernando. Introdução à Ciência de Dados: mineração de dados e big data, 2016.

BATISTA, Gustavo Enrique de Almeida Padro Alves, Pré-processamento de Dados em Aprendizado de Máquina Supervisionado, 2003.

BELGIU, Mariana, DRAGUT, Lucian, Random Forest in remote sensing: A review of application and future directions, 2016.

BELTRÃO, Claudio Jose, DIAS João Silva, RIBEIRO Luzia Fátima, Utilização do padrão NANDA e outras funções em um sistema de apoio à enfermagem baseado em protocolos. 2004.

CARVALHO, André, Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina, 2011.

COLLINS, Marie, *Updated Algorithm for the PubMed Best Match Sort Order. NLM Tech Bull*, 2017.

DIJCKS, Jean-Pierre, Oracle: Big data for the enterprise. Oracle White Paper. Redwood Shores, CA: Oracle Corporation, 2013.

ERCOLE, Flávia, MELO, Laís Samara de, ALCOFORADO, Carla Lúcia Goulart Constant, Revisão Integrativa versus revisão sistemática, Revista Mineira de Enfermagem, 2014.

FACELI, Katti; LORENA, Ana Carolina; GAMA, João; CARVALHO, André Carlos Ponce de Leon Ferreira de. Inteligência artificial: uma abordagem de aprendizado de máquina. [S.l: s.n.], 2011.

FILHO, Ubiratan, Análise de viabilidade técnica para coleta de informações de um monitor multiparamétrico, 2018.

INTEL, IT Center, Big Data Analytics. Intel's IT Manager Survey on How Organizations Are Using Big Data. Intel IT Center. Santa Clara, CA: Intel Corporation. 2012.

KITCHENHAN, Barbara, Procedures for Performing Systematic Reviews, 2004

KOEHRSEN, Will, *Hyperparameter Tuning the Random Forest in Python*. Disponivel em: <https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74> Acessado em: dezembro/2019

KONENKO, Igor, Machine learning for medical diagnosis: History, state of the art and perspective, 2001.

LANEY, Doug. 3-D Data Management:Controlling Data Volume, Velocity and Variety. META Group Research Note, 2001.

MAURO, Andrea De MARCO. Greco, GRIMALDI, Michele, What is Big Data? A Consensual Definition and a Review of Key Research Topics, AIP Conf. Proc., vol. 1644, 2014.

MEHRABI, Saeed, SOHN, Sunghwan., LI, Dingcheng, PANKRATZ, Joshua J., THERNEAU, Terry, SAUVER, Jennifer L. S., PALAKAL, Mathew, *Temporal Pattern and Association Discovery of Diagnosis Codes Using Deep Learning.* 2015 *International Conference on Healthcare Informatics.* doi:10.1109/ichi.2015.58

MONARD, Maria (2008). 4 Conceitos sobre Aprendizado de Máquina. Disponivel em <http://dcm.ffclrp.usp.br/~augusto/publications/2003-sistemas-inteligentes-cap4.pdf> acessado em: abril/2019

National Center for Biotechnology Information, Disponivel em <https://www.ncbi.nlm.nih.gov/pubmed/>, 2019. acessado em Maio/2019.

NG, Kenney., STEINHUBL, Steven. R., DEFILIPPI, Christopher., DEY, Sanjoy, STEWART, Walter F., *Early Detection of Heart Failure Using Electronic Health Records. Circulation: Cardiovascular Quality and Outcomes,* 2016.

RESENDE. Leticia. (2009). Protocolos Exames Laboratoriais. Universidade Federal de Minas Gerais. Disponível em: <http://www.uberaba.mg.gov.br/portal/acervo/saude/arquivos/oficina_10/protocolos_exames_laboratoriais.pdf>. Acesso em: abril/2019.

RAHIMIAN, Fatemeh, SALIMI-KHORSHIDI, Gholamreza, PAYBERAH, Amir, TRAN, Jenny, SOLARES, Roberto Ayala, RAIMONDI, Francesca, NAZARZADEH, Milad, CANOY, Dexter, RAHIMI, Kazem, Predicting the risk of emergency admission with machine learning: Development and validation using linked electronic health records, 2018

SCHROECK, Michael., SHOCKLEY, Rebecca, SMART, Janet, ROMERO-MORALES, Dolores, TUFANO, Peter, *Analytics: The real-world use of big data. New York, NY: IBM Institute for Business Value*, Said Business School, 2012.

WONG, Jenna, MURRAY, Mara M., ZHOU, Li, TOH, Sengwee, *Using Machine Learning to Identify Health Outcomes from Electronic Health Record Data. Current Epidemiology Reports.* doi:10.1007/s40471-018-0165-9, 2018.

WITTEN, Ian H, FRANK, Eibe, *Data Mining Pratical Machine Learning Tools and Techniques, Second Edition,* 2015.

YUI, Tony, Understanding Random Forest, 2019

https://towardsdatascience.com/understanding-random-forest-58381e0602d2

ZHENG, Tao., XIE, Wei, XU, Liling, HE, Xiaoying, ZHANG, Ya, YOU, Mingrong, CHEN, Yang, *A machine learning-based framework to identify type 2 diabetes through electronic health records. International Journal of Medical Informatics*, 97, 120–127, doi:10.1016/j.ijmedinf.2016.09.014, 2017.