UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO DE ENGENHARIA QUÍMICA
ENGENHARIA QUÍMICA

GABRIELA FERNANDA FANTINATTI

COMPARAÇÃO DE ALGORITMOS PREDITIVOS PARA INCÊNDIOS EM CANAVIAIS

PONTA GROSSA
2021

GABRIELA FERNANDA FANTINATTI

COMPARAÇÃO DE ALGORITMOS PREDITIVOS PARA INCÊNDIOS EM CANAVIAIS

Comparação de algoritmos preditivos para incêndios em canaviais

Trabalho de conclusão de curso apresentada como requisito para a obtenção do título de Bacharel em Engenharia Química, do Departamento de Engenharia Química da Universidade Tecnológica Federal do Paraná (UTFPR).
Orientadora: Profª. Drª. Maria Regina Parise
Coorientadora: Cristiane Yoko Takahashi Carpanezi

PONTA GROSSA

2021

GABRIELA FERNANDA FANTINATTI

COMPARAÇÃO DE ALGORITMOS PREDITIVOS PARA INCÊNDIOS EM CANAVIAIS

Trabalho de conclusão de curso apresentada como requisito para a obtenção do título de Bacharel em Engenharia Química, do Departamento de Engenharia Química da Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 02/Dezembro/2021

_____

Everton Moraes Matos
Doutor em Engenharia Química
Universidade Tecnológica Federal do Paraná

_____

Regina Negri Pagani
Doutora em Engenharia de Produção
Universidade Tecnológica Federal do Paraná

_____

Maria Regina Parise
Doutora em Engenharia Química
Universidade Tecnológica Federal do Paraná

PONTA GROSSA

2021

AGRADECIMENTOS

Agradeço inicialmente aos meus pais Elisangela e José Henrique, por terem cultivado a curiosidade daquela garotinha, instruído, ensinado, educado, por terem apoiado a aquela adolescente que tomou a decisão de ir estudar em outro estado, por terem amparado e confiado durante todos estes anos e por terem incentivado esta mulher a seguir em frente e chegar até este momento tão esperado.

Agradeço a Profª. Drª. Maria Regina Parise, esta mulher de coração enorme, que me acolheu como orientanda da vida, que se preocupou, me amparou, torceu por mim e não me abandonou durante todo este período e que foi meu ponto de apoio nos últimos anos de graduação.

Agradeço ao meu namorado Vinicius pela parceria, por ter acreditado em mim e por me segurar todas as vezes em que ia cair.

Agradeço aos meus colegas de estágio que contribuíram com o desenvolvimento deste trabalho, como também aos tutores que me deram base para que o mesmo fosse construído, além de todo conhecimentos e ensinamentos que me foram passados.

Agradeço também a Cristiane Yoko Takahashi Carpanezi pela coorientação, toda a disponibilidade oferecida a mim neste período e pela oportunidade de fazer parte da sua equipe.

RESUMO

A ocorrência de incêndio em canaviais é um problema que há muito assola diversas regiões do país. Devido à preocupação em relação a isso e suas trágicas consequências, faz -se necessário a tomada de ações a fim de evitar ou atenuar esse problema. Com o desenvolvimento da Inteligência Artificial (IA) e a crescente utilização de *Machine Learning* (ML), encontra-se uma oportunidade de utilizar a tecnologia a favor da antecipação quanto a um iminente incêndio. Neste contexto, o objetivo deste trabalho foi implementar algoritmos preditivos utilizando a linguagem Python para comparar e definir o que melhor se aplica a predição de incêndio em canaviais. Os dados necessários para essa finalidade foram fornecidos por uma empresa sucroenergética localizada no interior de São Paulo. Dentre os quatro algoritmos analisados, sendo estes *Supporte Vector Machine (SVM), Naive Bayes, Random Forest e XGBoost*, o modelo SVM mostrou um melhor desempenho frente as métricas de desempenho utilizadas.

Palavras-chave: Canavial. Incêndio. *Machine Learning*. Algoritmo. *Python*

ABSTRACT

The occurrence of fire in sugarcane fields is a problem that has plagued several regions of the country for a long time. Due to the concern about this and its tragic consequences, it is necessary to take actions in order to avoid or alleviate this problem. With the development of Artificial Intelligence (AI) and the growing use of Machine Learning (ML), there is an opportunity to use technology in favor of anticipating an imminent fire. In this context, the objective of this work was to implement predictive algorithms using the Python language to compare and define what best applies to fire prediction in sugarcane fields. The data necessary for this purpose were provided by a sugarcane company located in the interior of São Paulo. Among the four analyzed algorithms, these being the Support Vector Machine (SVM), Naive Bayes, Random Forest and XGBoost, the SVM model showed a better performance against the performance metrics used.

Keywords: Sugarcane. Fire. Machine Learning. Algorithm. Python

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE SIGLAS

AM	Aprendizado de Máquina
IA	Inteligência Artificial
KDD	Knowledge Discovery in Databases
ML	Machine Learning
NDVI	Normalized Difference Vegetation Index
RF	Random Forest
SMOE-NC	Synthetic Minority Oversampling Technique - Nominal Continuous
SVM	Support Vector Machine
XGBoost	XGBoost Extreme Gradient Boosting

SUMÁRIO

1 INTRODUÇÃO

A queima da palha da cana-de-açúcar foi um método amplamente utilizado para a limpeza prévia de canaviais, com o objetivo de aumentar a produtividade. Seu uso se tornou mais intenso a partir da década de 70, com a iniciativa do Proálcool. Quando o fogo é ateado à cana, a maior parte das folhas verdes e secas são eliminadas, o que facilita o trabalho do operador, diminuindo os riscos de acidentes. Além disso o uso do fogo elimina animais peçonhentos que oferecem perigos aos trabalhadores, facilita o corte manual e a queima favorece também o corte mecanizado, diminuindo, assim a quantidade de matéria orgânica recolhida pela colhedora. A queima permite com que o campo de visão do operador aumente, evitando acidentes com o maquinário.

Com as mudanças da sociedade e a preocupação com o meio ambiente, nos últimos anos, a prática da queima da cana passou a ser questionada (TORQUATO, MARTINS E RAMOS, 2009). Diversas leis municipais foram criadas com o intuito de acabar com essa prática e a longo prazo tornar a colheita mecanizada.

O Protocolo Agroambiental, criado pelo Governo do Estado de São Paulo em 2007, tem o intuito de que suas signatárias façam o uso de boas práticas para estimular a sustentabilidade na produção do etanol, açúcar e bioenergia. As ações deste protocolo visam a antecipação da eliminação do uso da queima da cana-de-açúcar, a restauração de matas ciliares e nascentes próximas as áreas de cultivo. Juntamente com este protocolo, a fim de reforçar essas ações, no ano de 2017, foi criado o Protocolo Etanol Mais Verde, visando a diminuição e reuso de água, a prevenção de combate a incêndio florestais, entre outras medidas (GOVERNO DO ESTADO DE SÃO PAULO, 2020; UNICA, 2020).

A não utilização da queima prévia inviabiliza o corte manual da cana, o que trouxe diversas discussões e preocupações com relação ao desemprego e a falta de qualificação para a colheita mecanizada. Inicialmente, pode se observar através da literatura (SILVA e GARCIA, 2009) que o corte da cana sem queima prévia oferecia mais pontos negativos do que positivos, pois as colhedoras não estavam preparadas para o corte da cana crua (ou seja, quando não está queimada), sendo necessário investimentos tecnológicos e qualificação para adaptação ao novo cenário. Com as frequentes mudanças e uso de novas tecnologias, a queima da cana está praticamente extinta, sendo necessária autorização para seu uso.

Como a queima em canaviais não é mais utilizada como forma de manejo, a ocorrência de incêndios é decorrente de causas naturais, como, por exemplo, raios. A ação humana também é uma das causas de ocorrência de incêndios, podendo ser criminosa ou acidental, exemplificada pela deposição de lixo em áreas próximas aos canaviais e bitucas de cigarro. Como a palha da cana é altamente inflamável, o fogo pode se alastrar rapidamente pelo canavial, gerando grandes impactos.

Os impactos provocados pela queima da cana são diversos, podendo ser classificados como ambientais, sociais e econômicos. Quando a palha da cana é queimada, diversos gases são liberados como metano, percursores de ozônio, óxido de nitrogênio, além de materiais particulados, que geram grandes discussões de seus riscos e efeitos sobre a saúde humana e poluição atmosférica. A fauna e flora são diretamente afetadas com as queimadas, pois muitos mamíferos, insetos e aves utilizam os canaviais como refúgio, podendo ser dizimados pelo fogo. Quando ocorre um incêndio, nem sempre este pode ser controlado, podendo atingir áreas próximas aos canaviais, muitas destas de preservação permanente.

Com o avanço da tecnologia, surge a oportunidade deste na previsibilidade da ocorrência de incêndios. A predição de incêndios em canaviais é de extrema importância, pois os riscos supracitados podem ser mitigados. Conhecendo a probabilidade de uma área ser incendiada, medidas podem ser tomadas à fim de diminuir ou até mesmo evitar o incêndio iminente.

Com a aplicação de *Machine Learning* (ML), que corresponde ao aprendizado de máquina, consegue-se utilizar algoritmos que sejam capazes de predizer a ocorrência de incêndios com base em aprendizado de dados passados.

1.1 Delimitação do tema

Este trabalhou delimitou-se a aplicação de *Machine Learning* para definição de modelos capazes de realizar predições de incêndios em canaviais. Através de aprendizado de máquina supervisionado, utilizando-se de algoritmos preditivos, aplicados e posteriormente comparados, dispondo-se de dados fornecidos por uma empresa sucroenergética localizada no interior do Estado de São Paulo.

1.2 Objetivo geral

O presente trabalho tem como objetivo definir o melhor algoritmo capaz de predizer a ocorrência de incêndio em canaviais.

1.3 Objetivos específicos

Os objetivos específicos desse trabalho são:

- Caracterizar os incêndios em canaviais;
- Fazer levantamento dos algoritmos preditivos;
- Coletar dados em campo;
- Testar os algoritmos a partir dos dados coletados.

1.4 JUSTIFICATIVA

Com a grande recorrência de incêndios em canaviais, diversos são os impactos tanto para a organização responsável, quanto para a sociedade. Um incêndio em grandes proporções, além da geração de fumaça, possível perda de flora e fauna, acarreta prejuízos financeiros para a organização, seja através de multas ambientais ou de perda de qualidade do açúcar. Faz-se, então, necessário o uso da tecnologia para quantificar essas ocorrências e identificar os eventos em comum que as envolvem. Com o sistema de probabilidade de ocorrência de incêndios em canaviais, o portador da informação poderá se antecipar e criar ações a fim de evitar que o foco do incêndio se inicie ou que este se espalhe, mitigando os terríveis impactos causados pelo fogo. No entanto, com a quantidade de algoritmos preditivos existentes, é primordial um comparativo e definição do modelo de melhor se aplica aos dados disponíveis.

2 REFERENCIAL TEÓRICO

Nesta seção aborda-se de forma sucinta sobre o cultivo de cana-de-açúcar e aprofunda-se no universo de *Machine Learning*. Conceitos de aprendizado de máquina, modelos preditivos e seus funcionamentos são explanados assim como a forma de avaliação de desempenho destes.

2.1 Cana-de-açúcar, um breve histórico e cenário atual

A cana-de-açúcar é uma planta perene originária de climas tropicais e subtropicais, pertencente ao gênero *Saccharum* e à família Poaceae. A hipótese mais difundida é de que a cultura provém do norte da Índia. O país foi invadido pelos árabes que passaram a produzir o açúcar, sendo também difundido pelo Egito e Mesopotâmia após essa invasão. Sua propagação pela Europa se deu através das cruzadas (século XI) em que a Espanha passou a cultivar a cana-de-açúcar nas Ilhas Canárias e Portugal em Cabo Verde e na Ilha da Madeira, tornando-se assim grandes distribuidores europeus de açúcar (SATO, 2012;SILVA et al., 2015).

No início do século XVI, trazidas da Ilha da Madeira por Martin Afonso, as primeiras mudas de cana-de-açúcar chegaram ao Brasil. O primeiro engenho foi fundado em São Vicente por Martin, iniciando o cultivo da cana no Brasil colônia, que encontrou em terras brasileiras, o solo massapé, ideal para seu desenvolvimento. (ALMEIDA, 2016). O cultivo da cana-de-açúcar se manteve como base econômica brasileira por mais de um século, período em que foi detentor do monopólio mundial de produção, correspondendo ao primeiro ciclo econômico do país (MIRANDA, 2020).

De acordo com Miranda (2020), a produção da cana-de-açúcar alcançou um novo nível quando o etanol passou a ser uma de suas finalidades, onde o primeiro contato com o produto no Brasil, foi realizado em 1953. No entanto, o impulso à produção do etanol se deu através da criação do Programa Nacional do Álcool, conhecido como Proálcool, em 1975.

Atualmente, o Brasil é o maior produtor mundial de cana-de-açúcar. O setor sucroenergético nacional atua com uma postura positiva e sustentável, tendo um grande potencial para a produção de etanol e seus subprodutos. A estimativa para a safra 2019/20 correspondeu a uma área de 8.382,2 mil hectares de cana-de-açúcar colhida, em que o Estado de São Paulo é o maior produtor nacional. A produção

estimada para esta safra girou em torno de 615.978,9 mil toneladas de cana-de-açúcar (CONAB, 2019).

2.2 Colheita da cana-de-açúcar e o uso da queima

Segundo Silva e Garcia (2009), a operação de colheita da cana-de-açúcar pode ser classificada em manual, semimecanizada e mecanizada, e o sistema como um todo envolve o corte, carregamento, transporte e recepção da cana. Até meados dos anos 2000, a mão de obra para a colheita da cana-de-açúcar era exclusivamente manual. O processo de colheita manual, consistia na entrada do trabalhador no canavial que utilizava um facão para o corte da cana. Na operação semimecanizada, fazia-se uso de guindastes e outros maquinários para auxiliar o colaborador na colheita, que não mais necessitava transportar a cana colhida até o caminhão (MORENO, 2011).

O uso do fogo, segundo Moreno (2011), foi inserido à colheita, por volta das décadas de 60 e 70. Segundo Avolio (2002), o uso da queima como forma de manejo foi inicialmente utilizada para o controle de alguns tipos de vegetação, facilitando o plantio e a renovação de áreas de pastagem. A prática é antiga, sendo empregada na retirada de áreas florestais para o desenvolvimento das cidades, campos para a agricultura e pecuária. O uso da queima em canaviais teve início no período da Segunda Guerra Mundial, na Austrália e no Havaí, sendo uma alternativa à escassez de mão de obra.

Além do uso do fogo para despalha, este também trazia como benefícios o controle de pragas e vegetações daninhas que atrapalhavam o desenvolvimento da cana e a destruição de animais peçonhentos que ofereciam riscos aos trabalhadores durante o corte da cana-de-açúcar (SANTOS, 2012).

De acordo com Torquato, Martins e Ramos (2009), apenas com a pressão social e a mudança de visão mundial em relação aos problemas climáticos, pois o meio ambiente não era considerado na produção de cana-de-açúcar. Foi este o ponto de partida o para a modernização do processo produtivo. Com a necessidade de adequação às exigências ambientais e a busca de uma produção mais sustentável, a indústria canavieira passou a expandir a mecanização do plantio e colheita da cana-de-açúcar.

2.3 Impactos negativos da queima da cana

Os meses entre abril e novembro correspondem ao período mais seco do ano no Estado de São Paulo e onde ocorrem a maior parte das queimadas. O clima nesta época do ano é propício para a propagação de materiais particulados através do ar. De acordo com Borges et al. (2020), durante a queima há grande emissão de fumaça e fuligem, sendo que a fuligem possui 95 tipos de partículas finas e ultrafinas. Sendo assim, estes autores desenvolveram um trabalho com o intuito de analisar os impactos causados pela queima da cana-de-açúcar em uma usina no Estado do Paraná. Os pesquisadores concluíram que a queima está relacionada à emissão de gases poluentes como monóxido de nitrogênio (NO), dióxido de nitrogênio (NO_2), amônia (NH_3), dióxido de carbono (CO_2), entre outros, além de estarem associados a problemas respiratórios.

A folha da cana-de-açúcar possui nitrogênio em sua estrutura. Quando esta é queimada, libera nitrogênio ativo, assim como, a conversão do nitrogênio do ar realizada pelo calor. O nitrogênio ativo pode ser transportado pela atmosfera entrando em contato com a água. Em rios e lagos existem plantas e algas que na presença deste nitrogênio ativo desenvolvem-se com maior intensidade (processo de eutrofização). Muitas dessas algas podem ser tóxicas, gerando assim a morte de peixes, causando um desequilíbrio no ecossistema. O mesmo ocorre com o transporte de nitrogênio pelo solo, que beneficiam algumas espécies de vegetais, e estas acabam se desenvolvendo em excesso, o que afeta a biodiversidade da região. (CARDOSO, MACHADO, PEREIRA, 2008).

De acordo com impactos levantados por Ferreira, Siqueira e Bergonso, (2009), reações alérgicas e inflamatórias estão associadas às partículas provenientes da fumaça da queima da cana-de-açúcar. Estas partículas, por serem ultrafinas, podem penetram no sistema respiratório, e também percorrer a corrente sanguínea ocasionando complicações em alguns órgãos.

A prática de queima da cana, segundo Ronquim (2010), ocasiona a perda da fauna da região. Insetos pequenos podem ser totalmente incinerados, já os mamíferos e pássaros acabam não tendo tempo de realizar a fuga quando a propagação do fogo se inicia, podendo assim morrer por queimaduras ou asfixiados pela fumaça, além de perder seus ninhos. O fogo pode atingir a vegetação limítrofe aos canaviais, sendo estas, muitas vezes, áreas de preservação permanente. As áreas afetadas podem ser

próximas a rios e lagos e com a diminuição da vegetação adjacente às águas, o processo de infiltração da água da chuva acaba sendo menor, o que possibilita a ocorrência de erosão do solo e arraste de materiais sólidos para os rios e lagos.

Tosto, Paiva e Andrade (2010) realizaram um trabalho comparativo da taxa de erosão entre os dois manejos de cultivo de cana-de-açúcar, com queima e mecanizado. Verificou-se que a perda de nutrientes do solo foi superior com uso da queima palha da cana-de-açúcar, gerando um custo de 3,8 vezes maior para a reposição desses nutrientes, em relação à colheita mecanizada.

2.4 Lei de proibição da queima

Segundo Ronquim (2010), a primeira legislação ambiental para o setor canavieiro foi o Decreto n° 42.056 de 06 de agosto de 1997, para o Estado de São Paulo. Esta legislação declarou que as queimadas deveriam ser evitadas, sendo utilizadas apenas com autorização da Secretaria da Agricultura e Abastecimento.

A Lei N° 11.241, de 19 de setembro de 2002, instaurada no Estado de São Paulo, corresponde à eliminação do uso da queima como forma de manejo do corte da cana-de-açúcar. Essa eliminação, ocorre de forma gradativa, conforme demonstrado no Quadro 1 (São Paulo, 2002).

Quadro 1 - Eliminação do uso da queima em canaviais

Ano	**Área mecanizável onde não se pode efetuar a queima**	**Percentagem de eliminação da queima**
1° ano (2002)	20% da área cortada	20% da queima eliminada
5° ano (2006)	30% da área cortada	30% da queima eliminada
10° ano (2011)	50% da área cortada	50% da queima eliminada
15° ano (2016)	80% da área cortada	80% da queima eliminada
20° ano (2021)	100% da área cortada	Eliminação total da queima
Ano	**Área não mecanizável com declividade superior a 12% e/ou menor de 150h á (cento e cinquenta hectares), onde não se pode efetuar a queima**	**Percentagem de eliminação da queima**
10° ano (2011)	10% da área cortada	10% da queima eliminada
15° ano (2016)	20% da área cortada	20% da queima eliminada
20° ano (2021)	30% da área cortada	30% da queima eliminada

25° ano (2026)	50% da área cortada	50% da queima eliminada
30° ano (2031)	100% da área cortada	100% da queima eliminada

Fonte: São Paulo (2002)

Segundo previsto por lei, até o ano de 2031, todas as áreas, sendo estas mecanizáveis ou não, deverão atingir 100% de eliminação da queima. De acordo com a Lei N° 11.241, área mecanizável corresponde a terrenos com mais de 150 hectares, contendo plantações e que possuem declividade igual ou inferior à 12%.

Se um incêndio iniciado no canavial atingir uma área de vegetação nativa, como por exemplo uma área de prevenção permanente, a organização ou o responsável pelo canavial pode receber uma multa que varia de R$500,00 até R$50.000.000,00 por hectare atingido, de acordo com o tipo de vegetação prejudicada (SENAR, 2018).

O Protocolo Agroambiental consiste em um acordo feito entre o governo do Estado de São Paulo, usinas e fornecedores de cana-de-açúcar que foi assinado voluntariamente em 2007. Esse protocolo consiste na antecipação dos prazos para eliminação do uso da queima em canaviais até 2014 para áreas mecanizáveis e até 2017 para áreas não mecanizáveis (IEA, 2014) além de medidas visando a restauração de matas ciliares e nascentes próximas às áreas de cultivo no estado, além de medidas de conservação (UNICA,2020).

O Protocolo Etanol Mais Verde foi definido em junho de 2017 para dar continuidade às boas práticas definidas pelo Protocolo Agroambiental. Este protocolo abrange algumas diretrizes como: eliminação da queima, adequação à Lei 12651/2012 (Código Florestal), conservação do solo, reuso e conservação da água, aproveitamento dos subprodutos da cana-de-açúcar, responsabilidade socioambiental, prevenção de combate a incêndios florestais, entre outras (GOVERNO DO ESTADO DE SÃO PAULO, 2020).

Segundo UNICA (2020), desde a assinatura do Protocolo Agroambiental, houve uma queda de 9,3 milhões de toneladas de CO_2 emitidos e 56 milhões de toneladas de poluentes como monóxido de carbono, material particulado e hidrocarbonetos. Em relação a matas ciliares, 200 mil hectares foram protegidos e recuperados como também 8230 nascentes. No Estado de São Paulo, não há uso da prática de queima em 98% das áreas de cana-de-açúcar.

2.5 Sistemas de predição de incêndios

No presente trabalho, foram realizadas pesquisas a fim de levantar trabalhos que envolvem sistemas de predição de incêndio, e observar quais são os dados e as ferramentas mais utilizadas. Até o momento do desenvolvimento deste trabalho, não foram encontradas informações disponíveis na literatura a fim de predizer incêndios em canaviais.

Prata (2019) desenvolveu um trabalho de mapeamento de probabilidade de incêndios florestais, tendo como área de estudo plantios de eucalipto no Estado da Bahia. Os dados utilizados no sistema foram: dados de ocorrência de incêndio; variáveis climáticas como precipitação, temperatura média e umidade relativa do ar; dados sociais como distância de municípios, assentamentos, estradas, densidade demográfica, população residente em municípios e zonas rurais. Os modelos utilizados foram:

- Regressão Logística, ferramenta estatística para realização de predições.
- Random Forest, modelo de classificação que utiliza-se de árvore de decisão.
- SVM - *Support Vector Machine,* modelo que pode ser utilizado tanto para classificação, quanto para regressão.

Massulo (2018) desenvolveu uma pesquisa a fim de propor um modelo preditivo da ocorrência de queimadas no bioma amazônico. Inicialmente foram feitas pesquisas a fim de levantar informações das ocorrências dos focos de calor, levantando dados dos anos 1997 a 2017. Para a realização da predição, o autor fez o uso de Co-krigagem, que através de vários atributos estima-se um atributo específico, utilizando-se de ferramentas geoestatísticas, onde atributos são um conjunto de dados, podendo ser variáveis quantitativas ou qualitativas.

Oliveira et al. (2017) realizaram um estudo de padrões espaciais de riscos de incêndios em uma cidade situada no Estado da Paraíba, utilizando a metodologia *fuzzy*. Esta baseia-se no conceito de pertinência, em que determina se a variável em questão pertence ou não ao conjunto analisado, utilizando o *software* MATLAB® para cálculo do valor *fuzzy*. Segundo os autores, áreas com grande circulação de pessoas, presença de vegetação e declividades, possuem maior risco de incêndio.

2.6 Machine learning

Machine Learning (ML) é um conceito que surgiu na década de 60 e faz parte da Inteligência Artificial (IA). Este pode ser definido como um conjunto de métodos capazes de detectar padrões em dados e utilizá-los para efetuar previsões de novos dados como também realizar tomadas de decisão. (MURPY, 2012; IZBICK e SANTOS, 2020).

Segundo Silva (2021), *Machine Learning* (ML) ou Aprendizado de Máquina (AM) permite que computadores aprendam a partir de dados, de forma automática, extraiam conclusões, convertendo dados de entrada em informações úteis. Esta aprendizagem é feita após o modelo ser treinado a reconhecer padrões.

O aprendizado de máquina pode ser dividido em 2 subgrupos: aprendizado não supervisionado e aprendizado supervisionado.

2.6.1 Aprendizado não supervisionado

O aprendizado não supervisionado, também chamado de modelagem descritiva (ROZA, 2016), faz uso de dados não rotulados, ou seja, não há dados de saídas desejadas. Neste subgrupo, durante o seu aprendizado, o algoritmo não recebe os resultados esperados, então este deve encontrar relações entre os dados de entrada. De acordo com as similaridades encontradas pelo algoritmo, este organiza ou agrupa os dados. A partir da capacidade de encontrar estas similaridades o algoritmo é capaz de classificar e agrupar novos conjuntos de dados (ESCOVEDO, 2020).

Este tipo de aprendizado pode ser utilizado na detecção de *outliers*, que podem ser dados discrepantes, anomalias, valores atípicos de um de conjunto de dados, entre outros. (FUCKS, 2017).

2.6.2 Aprendizado supervisionado

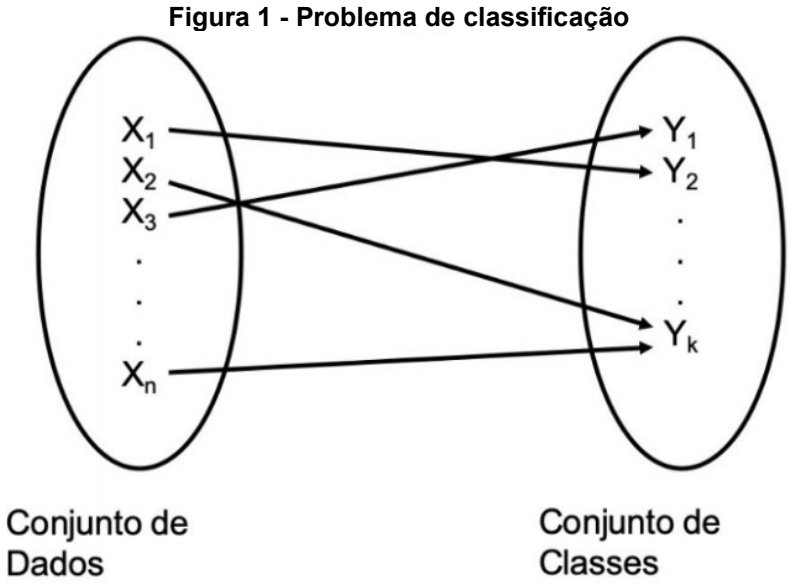
No aprendizado supervisionado, diferentemente do não supervisionado, o algoritmo recebe tanto informações de entrada quanto de saída, que são dados rotulados, pois para cada entrada tem uma saída esperada. Então o algoritmo aprende com exemplos de entradas e saídas desejadas (ESCOVEDO, 2020). Segundo Murphy (2012), os métodos de aprendizado supervisionados são os mais utilizados na prática.

Dentro deste subgrupo de aprendizado, há dois tipos de problemas de modelagem, o de classificação e o de regressão, e serão descritos a seguir.

a) Classificação

Problemas de classificação são os mais populares e importantes de *Machine Learning* (ML). Neste tipo de problema, o algoritmo aprende uma regra geral responsável por mapear os dados de entrada nos dados de saída de forma correta, ou seja, busca uma função matemática capaz de associar as entradas Xi para uma classe Yi, em que Y é uma classe categórica que se deseja prever. Com essa regra geral (função) aprendida, o algoritmo ou método como também é denominado, consegue ser aplicado a novos dados para a predição de suas respectivas classes conforme ilustrado na Figura 1 (ESCOVEDO, 2020).

Figura 1 - Problema de classificação



Fonte: ESCOVEDO (2020)

b) Regressão

O problema de regressão tem por objetivo estimar o valor esperado para a variável de saída (resposta) a partir de um novo padrão de entrada, onde o modelo aprende padrões, compostos por variáveis independentes e uma variável resposta que é dependente (ESCOVEDO, 2020).

2.6.3 Data Mining

Data Mining, em português, Mineração de Dados, é uma etapa que faz parte do KDD (*Knowledge Discovery in Databases*), Descoberta de Conhecimento em Banco de Dados. O KDD consiste em uma sequência de etapas para a extração de conhecimento através de dados (GOLDSCHMIDT E PASSOS, 2005). Segundo Silva (2019), o objetivo da mineração de dados é extrair informações relevantes dos dados e transformá-las em conhecimento.

A mineração de dados aplica o processo de extração e exploração de dados utilizando de ferramentas adequadas após o pré-processamento dos mesmos. Este processo é composto pelas seguintes tarefas (CASTRO E FERRARI, 2016):

- Análise descritiva: responsável por visualizar, medir, explorar e compreender os dados;
- Predição: uso de ferramentas para classificar um conjunto de dados ou estimar o valor de variáveis;
- Análise de grupos: particionar ou segmentar um conjunto de dados em grupos;
- Associação: encontrar relação entre os dados;
- Detecção de Anomalias: encontrar dados discrepantes que não seguem o comportamento dos demais.

Segundo Goldschmidt e Passos (2005), o KDD da qual a mineração de dados faz parte, originou-se do Aprendizado de Máquina (AM). *Machine Learning* e a Mineração de Dados são campos complementares. Sendo assim *Machine Larning* utiliza-se de técnicas de Mineração de Dados, como os tipos de aprendizado (SILVA, 2019).

2.6.4 Big Data

Big Data pode ser definido como uma grande quantidade de dados. Devido ao tamanho e complexidade destes dados, estes não podem ser gerenciados por técnicas tradicionais (ALARU, et. al, 2012). Segundo Silva (2019), esse grande conjunto de dados é proveniente de fontes diferentes, assim como seu formato.

De acordo com Russom (2011), além do tamanho massa de dados, o *Big Data* possui três importante características, denominadas Os Três Vs: Volume, Velocidade e Variedade. O volume corresponde não apenas ao tamanho (*terabytes* por exemplo), mas também a quantidade de registros, tabelas, arquivos e tempo de armazenamento (histórico). A velocidade corresponde a frequência que esses dados são gerados, podendo até ser em tempo real. A variedade vem das diferentes fontes de dados, essas podendo ser dados geospaciais, dados da web, de aplicativos, entre outras fontes.

Devido à complexidade e o tamanho desta massa de dados utilizasse de técnicas de machine learning e mineração de dados para a extração para tratar este conjunto de dados (SILVA, 2019).

2.6.5 Modelos preditivos

Segundo Andrade e Brilhante (2018), os modelos de *Machine Learning* (ML) podem ter dois objetivos: um de Inferência onde, a partir dos dados, encontra-se padrões não observados pela visão humana. E outro de predição, onde prevê-se um resultado futuro com base em experiência (dados passados).

De acordo com Guazzelli (2012), modelo preditivo é a associação de dados com técnicas de modelagem preditiva. Neste contexto, ocorre o desenvolvimento de uma função, no momento de aprendizagem, que é capaz de mapear um conjunto de dados de entrada em uma variável de saída.

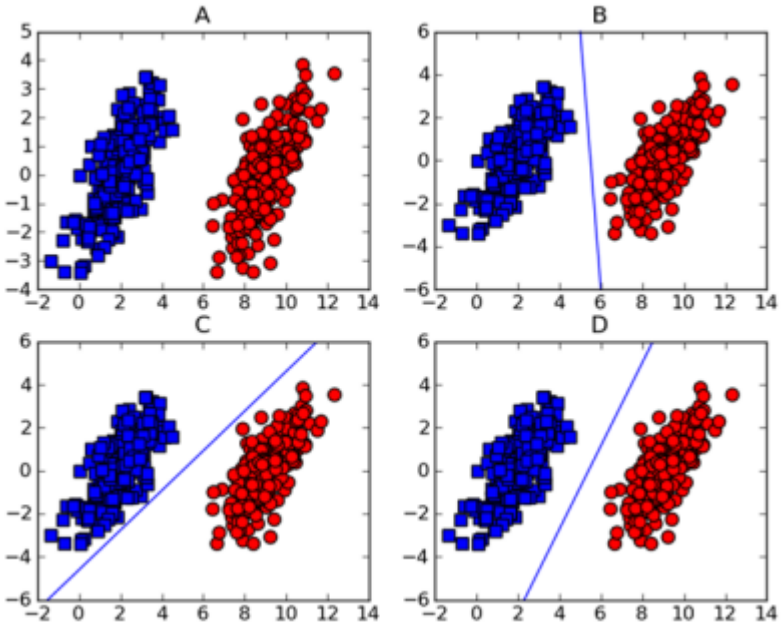
Na literatura diversos são os algoritmos preditivos (como por exemplo Redes Neurais Artificiais, Árvores de Decisão, Regressão Linear, Regressão Logística e *k-Nearest Neighbors*). No presente trabalho será realizada a modelagem e comparação de quatro modelos que são: *Suppot Vector Machine, Naive Bayes, Random Forest e XGBoost*. Esses modelos serão apresentados a seguir. Salientando que a escolha desses modelos foi sugerida pela equipe de IA da empresa estudada.

2.6.5.1 Support Vector Machine

Support Vector Machine (SVM) ou Máquina de Vetor de Suporte, em português, é um algoritmo de aprendizado supervisionado que pode ser empregado tanto em problemas de regressão quanto de classificação. Seu princípio baseia-se na separação dos dados em classes por uma reta. Essa reta ou linha de separação é

denominada hiperplano que é o limite de decisão do modelo, conforme apresentado na Figura 2. Esta figura tem como exemplo um conjunto de varáveis linearmente separáveis no quadro A e diferentes possibilidades de linhas de separação nos quadros B, C e D (HARRINGTON, 2012).

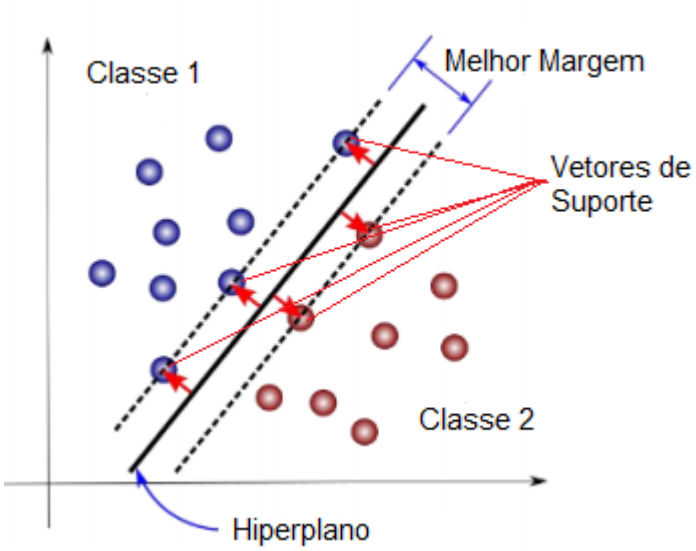
Figura 2 - Separação de variáveis



Fonte: HARRINGTON, 2012

De acordo com Harrington (2012) na Figura 2, tem-se várias possibilidades de linhas que separam as classes de dados, o melhor hiperplano para a separação das variáveis é aquele que possui a maior margem possível, ou seja, a maior distância possível até o ponto mais próximo de cada uma das classes. Estes pontos são denominados vetores de suporte conforme demonstrado na Figura 3.

Figura 3. Hiperplano e vetores de suporte



Fonte: Adaptado de PEREIRA ET. AL, (2020).

De acordo com Santos (2018), nem sempre o conjunto de dados a ser classificado através de fronteira de decisão é linear. Para isso o algoritmo possui uma extensão de classificação. Nesta extensão um conjunto de treinamento, onde o espaço é não linear, é mapeado em um espaço de maior dimensão linear. Para realizar este mapeamento utiliza-se a função Kernel.

2.6.5.2 Naive Bayes

Naive Bayes é uma técnica de classificação que faz parte das Redes Bayesianas, também denominadas redes casuais ou rede de crença. Redes Bayesianas são capazes de realizar predições de acordo com relação probabilística das variáveis, também predizer valores de variáveis desconhecidas a partir de valores de variáveis conhecidas. Esta predição é denominada inferência probabilística (LIRAet.al, 2019).

O algoritmo leva *naive* em seu nome (ingênuo em português), pois ele não considera a correlação entre as *features* durante a classificação. Este trata cada uma das variáveis de forma independente. As *features* são os dados de entrada inseridos no modelo. *Features* é uma outra forma de denominar os atributos, mencionados na Seção 2.5 deste trabalho.

Para a classificação o modelo Bayesiano utiliza-se do Teorema de *Bayes* para o cálculo das probabilidades, em problemas de classificação, conforme Equação 1.

$$P(classe|A) = \frac{P(A|Classe)x(P(classe)}{P(A)} \quad (1)$$

Onde A corresponde às novas instâncias a serem classificadas e A= a1, a2,...,a_n.

Durante a aplicação do teorema então, calcula-se a probabilidade de cada *feature* pertencer a cada uma das classes. Esta é classificada de acordo com o maior valor de $P(classe|A)$, ou seja, para que $P(classe|A)$ seja maior, devemos aumentar o numerador $P(A|Classe)x(P(classe)$ e diminuir o denominador $P(A)$. Desta afirmação obtém-se a Equação 2 (PARDO E NUNES, 2002):

$$argmaxP(classe|a_1 \dots a_{n)} = argmax \prod_i P(a_i|classe)xP(classe) \quad (2)$$

Onde:

$P(classe)$ = número casos pertencentes a classe analisada dividido pelo total de casos (Equações 3 e 4).

$$P(ocorrência\ incêndio) = \frac{n°\ de\ blocos\ incendiados}{total\ de\ blocos} \quad (3)$$

$$P(não\ ocorrência\ incêndio) = \frac{n°\ de\ blocos\ não\ incendiados}{total\ de\ blocos} \quad (4)$$

P(ai|classe) = número de casos pertencentes a classe analisada de acordo com a varável a_i dividido pelo total de casos (Equações 5 e 6). Considerando-se a análise de uma variável temperatura alta:

$$P(temperatura\ alta|ocorrência\ incêndio) = \frac{n°\ de\ \ blocos\ incendiados\ temperatura\ alta}{total\ de\ blocos} \quad (5)$$

$$P(temperatura\ alta|não\ ocorrência\ incêndio) = \frac{n°\ de\ blocos\ não\ \ incendiados\ temperatura\ alta}{total\ de\ blocos} \quad (6)$$

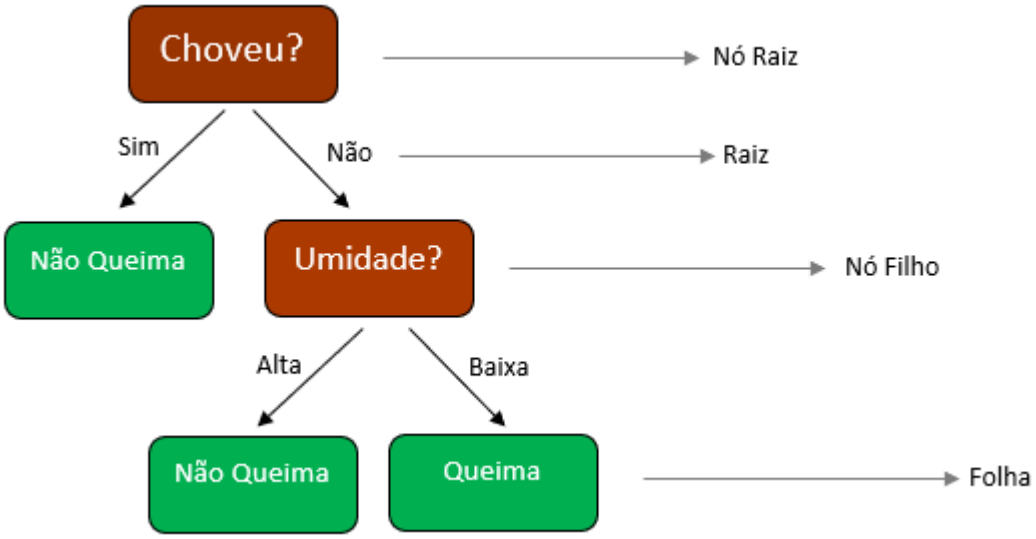
Este processo mostrado nos exemplos é aplicado para todas as varáveis existentes em cada uma das classes. As probabilidades obtidas então são inseridas na Equação 2, e multiplicada pelas classes analisadas.

2.6.5.3 Random Forest

De acordo com Breiman et al (2001), *Random Forests* (Floresta Aleatória) é um algoritmo composto por uma combinação de preditores, árvores de decisão, onde cada uma das árvores geradas depende de um vetor aleatório.

Árvore de Decisão (*decision tree*) é uma estrutura classificadora em formato de árvore, que possui um nó raiz, o ponto de partida, sendo uma das variáveis analisadas. Deste nó partem os ramos que são os valores desse atributo, que dão origem aos nós filhos, que são novos atributos. Estes atributos geram sucessivamente novos ramos e novos nós filhos que por fim chega ao nó folha, que é o valor de saída, a variável resposta. Esta estrutura é demostrada na Figura 4. Uma árvore de decisão é responsável por classificar um novo objeto a partir dos valores de seus atributos (CASTRO E FERRARI, 2016).

Figura 4 - Estrutura Árvore de Decisão



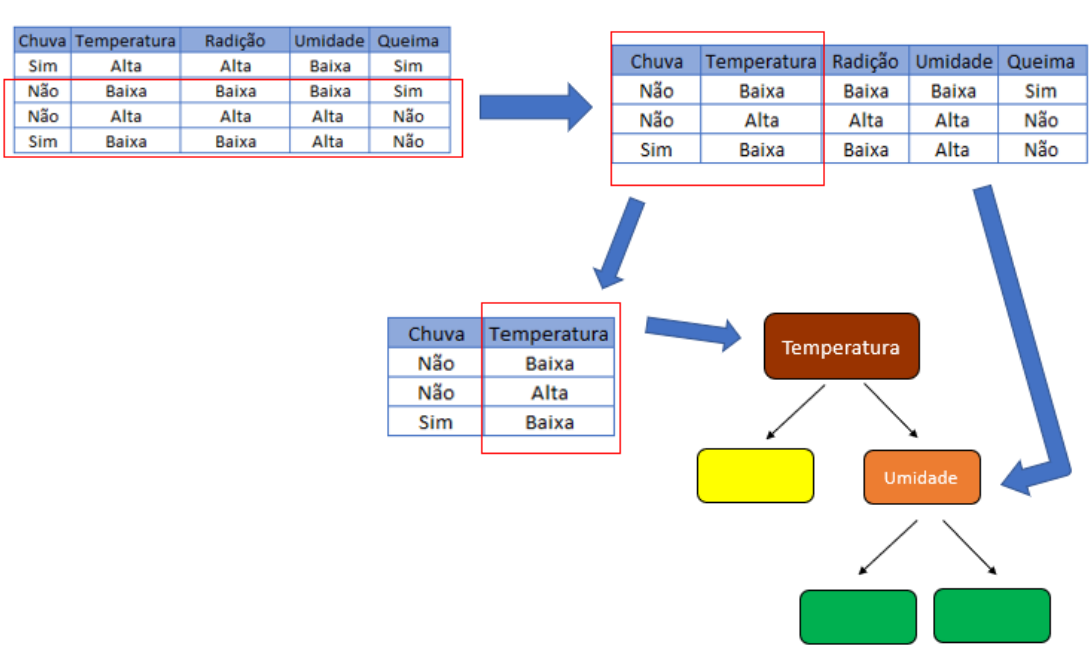
Fonte: Autoria Própria (2021).

Então o modelo *Random Forest* utiliza em sua fase de treino um conjunto de árvores de decisão, que são aplicadas em amostras do *dataset* completo e a partir dos resultados que mais aparecem, este realiza as predições. Em seu algoritmo é aplicado uma técnica chamada *bagging* ou *bootstrap aggregating*, em que o *dataset*

é separado em várias partes, sendo uma para cada árvore. Para que o número de amostras não seja insuficiente para o treinamento das árvores, novas variáveis são criadas a partir das iniciais (técnica de *booststrap sample*) (MATSUMOTO E FERNANDES, 2019).

Além da técnica de *bagging,* o modelo *Random Forest* possui mais uma técnica que garante a aleatoriedade das árvores, como também de suas *features*, a fim de evitar a correlação entre as próprias variáveis contidas nas estruturas classificadoras. Por fim, após cada treinamento com "novo" *dataset*, os resultados são combinados para que o modelo realize a predição com base nas respostas mais "votadas" pelas árvores (MATSUMOTO E FERNANDES, 2019). A Figura 5 traz um exemplo de criação de uma árvore aleatória.

Figura 5 - Formação árvore aleatória



Fonte: Autoria Própria (2021).

2.6.5.4 XGBoost

O Algoritmo *XGBoost – Extreme Gradient Boosting,* é um algoritmo, assim como o *Random Forest*, baseado em árvores de decisão. Sua diferença está no aumento de gradiente, que amplia a velocidade e o desempenho do modelo. O *Gradient Boosting,* aumento de gradiente, em português, é responsável por criar

árvores capazes de prever os ruídos que são levados e somados a predição final. (BROWNLEE, 2016).

Este algoritmo se diferencia do algoritmo *Random Forest* durante a criação das árvores que fazem parte da floresta, que não trazem apenas a classe preditora em sua folha, mas também uma pontuação para cada uma das categorias em relação a resposta real. Através de cálculos matemáticos de similaridade e ganhos, este irá montar as folhas das árvores de forma com que o erro seja o menor possível (DUARTE, 2020). A técnica de *Gradiente Boosting* corrige os erros obtidos nas predições anteriores, e o gradiente de perda que faz parte de sua modelagem matemática é minimizado (BROWNLEE, 2020).

2.6.6 Avaliação de desempenho de algoritmos

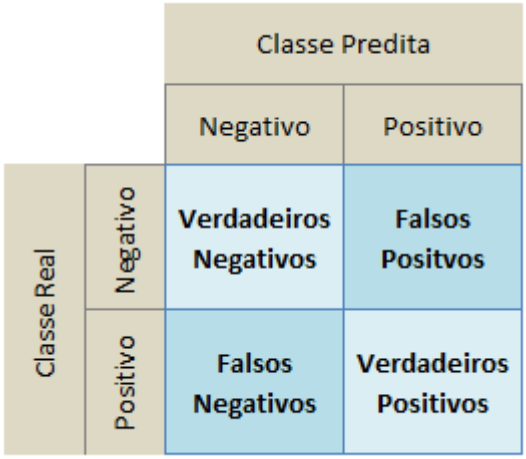
Em problemas de classificação, a avaliação de desempenho dos algoritmos baseia-se em cálculos de taxas de acertos e erros do conjunto de dados. Esses dados são utilizados no momento de teste do modelo, indicando um valor quantitativo de sua qualidade quanto as classificações.

A classificação possui duas classes a serem preditas, a classe alvo, ou positiva, e a classe negativa, em que cada uma delas possui medidas específicas (CASTRO E FERRARI, 2016), que aqui serão exemplificadas com o problema em questão, a ocorrência ou não de um incêndio:

- Verdadeiros Positivos (VP): ocorrência positiva, classificada pelo algoritmo como positiva. Exemplo: ocorreu um incêndio no bloco (uma subdivisão do canavial) e o modelo o classificou como ocorrência de incêndio;
- Falsos Positivos (FP): ocorrência negativa, classificada pelo algoritmo como positiva. Exemplo: o bloco não sofreu um incêndio e o algoritmo classificou como ocorrência de incêndio;
- Verdadeiros Negativos (VN): ocorrência negativa, classificada como negativa. Exemplo: o bloco não sofreu incêndio e algoritmo classificou como sem ocorrência de incêndio;
- Falsos Negativos (FN): ocorrência positiva, classificada como negativa. Exemplo: o bloco sofreu um incêndio e algoritmo classificou como sem ocorrência de incêndio.

Essas classes podem ser observadas através de uma matriz, denominada onfusão, contingência, ou zero erros (CASTRO E FERRARI, 2016). É uma matriz de fácil visualização, ilustrada na Figura 6, que auxilia na avaliação do desempenho, principalmente em casos em que se deseja maximizar alguma das medidas específicas, como por exemplo os verdadeiros positivos, ou verdadeiros negativos.

Figura 6 - Construção Matriz Confusão



Fonte: Adaptado de Castro e Ferrari (2016)

Com base nas classes é possível, então, calcular as métricas de avaliação de desempenhos do modelo que são: acurácia, precisão, *recall* (sensibilidade), especificidade e f1- *score*.

2.6.6.1 Acurácia

A acurácia corresponde a taxa global de sucesso do algoritmo sendo uma métrica simples, que calcula o percentual de acertos realizados pelo algoritmo, ou seja, a razão de total das classes verdadeiras pelo total de classes (CASTRO E FERRARI, 2016), sendo apresentada na Equação (6):

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN} \quad (6)$$

Onde:

- Verdadeiros Positivos (VP): ocorrência positiva, classificada pelo algoritmo como positiva;
- Falsos Positivos (FP): ocorrência negativa, classificada pelo algoritmo como positiva;

- Verdadeiros Negativos (VN): ocorrência negativa, classificada como negativa;
- Falsos Negativos (FN): ocorrência positiva, classificada como negativa.

Apesar de ser um classificador simples, a acurácia não considera pesos diferentes para as classes. Se por exemplo o verdadeiro positivo for a taxa de maior importância para o estudo, avaliar apenas por esta métrica não é o mais indicado.

2.6.6.2 Precisão

A precisão é uma métrica que relaciona os verdadeiros positivos e todos os valores positivos preditos, obtida mediante Equação (7):

$$Precisão = \frac{VP}{VP + FP} \quad (7)$$

A precisão indica então o percentual de positivos reais, dentro dos positivos preditos. Esta métrica é indicada para um modelo em que se deseja minimizar os falsos positivos (MARIANO,2021).

2.6.6.3 Recall

Esta métrica também é denominada sensibilidade ou revogação e corresponde a taxa de Verdadeiros Positivos, sobre o total de classe real positiva, como mostra a Equação (8):

$$Recall = \frac{VP}{VP + FN} \ (8)$$

O *recall* indica o percentual de positivos preditos de forma correta pelo algoritmo, ou seja, a capacidade do algoritmo de detectar resultados positivos (MARIANO,2021).

2.6.6.4 Especificidade

A especificidade corresponde a taxa de Verdadeiros Negativos sobre o total real da classe negativa, obtida na equação (9):

$$Especificidade = \frac{VN}{VN + FP} \ (9)$$

A métrica demostra percentual de negativos preditos de forma correta pelo algoritmo, ou seja, a capacidade do algoritmo de detectar resultados negativos (MARIANO,2021).

2.6.6.5 F1 – Score

A métrica f1- *score* é uma visualização das duas métricas: precisão e *recall*. É uma métrica indicada para quando o peso de falsos positivos e falsos negativos são diferentes. Seu cálculo é feito a partir da média harmônica entre a precisão e o recall (SCHADE, 2019), e pode ser determinado na Equação (10):

$$F1 - score = \frac{2 * Precisão * Recall}{Precisão + Recall} \quad (10)$$

Esta seção foi responsável por apresentar os problemas que permeiam incêndios em canaviais e demostrar que tem-se tecnologias disponíveis para antever este problema. Na literatura existem sistemas preditivos para diversos conjuntos de dados e varáveis de saída, mas até o momento da execução do presente trabalho, não foi encontrado um que se aproximasse do objetivo deste, reforçando-se assim a necessidade da modelagem e comparativo de algoritmos capazes de realizar predições para ocorrência de incêndios em canaviais.

3 PROCEDIMENTOS METODOLÓGICOS

Esta seção possui em seu conteúdo toda a metodologia aplicada durante o desenvolvimento deste trabalho, que inicia-se pelo entendimento do problema e percorrendo todas as etapas e sub etapas até o momento da comparação e definição do melhor modelo preditivo.

3.1 Levantamento das ferramentas utilizadas e dados

Como objeto deste estudo foi utilizado o banco de dados de um grupo sucroenergético localizado no interior do Estado de São Paulo, contendo diferentes níveis de informação como clima, histórico de operações e áreas afetadas por incêndios.

Os dados utilizados provêm de três safras: 2017/18, 2018/2019 e 2019/2020, correspondendo ao período de abril a novembro de cada ano. Estas safras dizem respeito a períodos passados, analisados como histórico. Para o presente trabalho não foram inseridos dados em tempo real para a implementação dos modelos, apenas dados de históricos. Esses dados já estavam disponíveis no banco de dados e foram coletados até meados do mês e outubro de 2020, antes deste trabalho ter se iniciado.

O Quadro 2 mostra os dados levantados inicialmente e suas respectivas descrições.

Quadro 2 - Dados coletados

Dado	Descrição
Ambiente	Conjunto de informações relacionadas a estrutura do canavial, como tipo de solo, capacidade de retenção de água e produtividade. Esse é um dado divididos em 5 categorias, onde A é o melhor ambiente e a qualidade vai diminuindo até chagar em E.
Maturação	Corresponde à época em que a cana é colhida, sendo: Precoce: colhida no início da safra Mediana: colhida no meio da safra Tardia: colhida no final da safra
Corte	Corresponde ao número de vezes que a cana foi colhida, onde inicialmente é considerada como cana soca (cana de foi plantada) e conforme os ciclos do canavial avançam, a cana é cortada, brota e é cortada novamente.

NDVI - *Normalized Difference Vegetation Index*	Índice Vegetativo, com um valor entre -1 e 1, onde quanto mais próximo de 1, mais ver de viva se encontra a cana-de-açúcar.
HHI	Índice Vegetativo, com um valor entre -1 e 1, onde quanto mais próximo de 1, mais ver de viva se encontra a cana-de-açúcar.
Temperatura	Temperatura dada em °C, coletadas diariamente por estações meteorológicas. Sendo medidas de temperatura máxima, mínima e média do dia.
Umidade do solo	Medição diária da umidade do solo a uma profundidade de 5cm.
Pluviosidade	Coletada diariamente por estações meteorológicas espalhas pelos canaviais, dada em milímetros.
Radiação Solar	Coletada diariamente, assim como a temperatura.
Informação do Centroide dos blocos analisados	Dado obtido por satélite onde tem - se a localização geográfica do centroide dos blocos.
Localização das Rodovias da Região	Informação de latitude e longitude das rodovias da região analisada.
Queima	Corresponde à forma como a cana foi colhida. Se colhida após uma ocorrência, leva o valor 1, se foi colhida in natura, leva o valor 0. Essa é a variável de saída, denominada *target.*

Fonte: Autoria Própria (2018)

O acesso aos dados foi feito com uso do *software* DBeaver integrado software PostgreSQL©. Com a utilização desses softwares foi possível realizar uma mineração das informações obtidas, trazendo apenas as que eram relevantes para o projeto. Para a utilização do *software* foi necessário a aplicação de linguagem SQL.

Para a extração dos dados no *sofware* DBeaver, utlizou-se de "*selects*" que são funções da linguagem SQL. Com eles foi possível selecionar as colunas necessárias, vindas das *tables* fontes. Essas *tables* estavam disponíveis no banco de dados da empresa. Os dados utilizados se encontravam em diferentes *tables,* portanto nos "*selects"* realizados utilizou-se das funções "*left join"* para a união destas informações. A função "*left join"* é responsável por unir duas tabelas diferentes por meio de informações comuns.

Além dos dados utilizados como variáveis do modelo, temos neste primeiro momento outras informações que servem como ponte na união dos dados, que são as safras (anos), o número dos blocos e a data em que a cana foi colhida. Essas

informações não são utilizadas como entradas, apenas fazem parte da construção do *dataset.*

Como resultado da extração inicial obteve-se 5 consultas (montadas através de *selects* e *letf joins*):

i. queima, ambiente e maturação
ii. centroide
iii. NDVI e HHI
iv. temperatura e pluviosidade
v. radiação solar e umidade

A informação de localização das rodovias foi extraída da página do Ministério da Infraestrutura, onde o arquivo em formato “shp”, foi extraído no formato de *geodaframe* através no ambiente Jupyter através da função “gpd.read_file” disponível na biblioteca geopandas.

Para o desenvolvimento e comparação dos algoritmos foi utilizada a linguagem Python™ no ambiente Jupyter, obtido através da plataforma Anaconda©. A forma como estes dados foram conectados ao Jupyter será explicado na seção a seguir.

O Quadro 3 ilustra as bibliotecas utilizadas no ambiente Jupyter:

Quadro 3 - Bibliotecas

Biblioteca	Finalidade
sqlalchemy	Conexão com o banco de dados
geopandas	Manipulação de dados de localização
pandas	Montagem do *dataset*
numpy	Cálculos de correlação
sns	Formatação da Matriz Confusão
sklearn.metric	Métricas de avaliação
xgboost	Implementação do modelo *XGBoost*
sklearn.preprocessing	Normalização de dados
sklearn	Implementação modelo SVM
sklearn.ensemble	Implementação modelo *Random Forest*
sklearn.naive_bayes	Implementação modelo *Naive Bayes*
imblearn.over_sampling	Balanceamento dos dados

Fonte: Autoria Própria (2018)

3.2 Pré-processamento dos dados

Como a quantidade de dados escolhida no primeiro momento foi grande e com informações de diferentes tipos e níveis, o *Big Data*, como foi apresentado na Seção 2.6.4, faz-se necessário o pré-processamento dos dados. O pré-processamento evita valores falantes, inconsistências e formatos de dados não reconhecidos pelos modelos. Segundo Sivakumar e Gunasundari (2017), o pré-processamento possui quatro etapas: limpeza dos dados, integração, transformação e redução dos dados.

Neste estudo o pré-processamento descrito pelos autores foram aplicados, porém em uma ordem diferente, descritos a seguir:

c) Integração

Nesta etapa do pré-processamento foi realizada a integração das diversas tabelas de dados, que correspondem as consultas descritas anteriormente, que inicialmente foram manipuladas via SQL no *software* DBeaver.

Na interface Jupyter após a conexão com o banco de dados da empresa, onde utiliza-se das credenciais cadastradas no DBeaver para realizar a conexão, iniciou-se a extração dos “selects” ou consultadas escritas inicialmente. Para trazer as informações do banco de dados para o ambiente Jupyter utilizou-se da função *“get”,* aplicada a cada uma das consultas criadas anteriormente. Cada uma das consultas retorna em formato de *data frame,* que corresponde a uma planilha que pode ser acessada pelo Jupyter e manipulada através de linguagem Python.

Após a integração, fez-se necessário a união dos cinco data *frames,* para isso utilizou-se da função *“merge”*, uma função capaz de unir duas fontes de dados a partir de colunas em comum, como nesse caso, bloco e data. Este processo é demostrado na Figura 7 para a união do *data frame* dado pelo *“select”* i. e iii., mencionados na Seção 3.1

Figura 7 - Merge

```
########################################## Merges ##########################################

def merge_base_ndvi(df_bloco,df_ndvi):

    df_mv_ndvi = pd.merge(df_bloco, df_ndvi, right_on=["data_queima","bloco"], left_on=["dt_queima","bloco"], how='inner')

    return df_mv_ndvi
```

Fonte: Autoria Própria (2021)

Na Figura 7, “df_bloco” e df_ndvi” correspondem as duas consultas que serão unidas. A função para que ocorra a união é dada pelo “pd.merge”, que recebe o nome “df_mv_ndvi” (nome de escolha do autor). Os parâmetros desta função são dados pelas colunas “data_queima” e” bloco” do *data frame* da direita (df_ndvi) e pelas colunas “dt_queima” e “bloco” do *data frame* da esquerda (df_bloco). O parâmetro *how=’inner’* corresponde ao tipo de “*merge*”, em que só trará as linhas em que “df_bloco” tenha correspondência no *data frame* “df_ndvi”

O *“merge”* resultante “df_mv_ndvi”, é um novo *data frame* (tabela/planilha) que será agrupado ao *data frame* gerado pelo *“select”* ii. (mencionado na Seção 3.1) e assim sucessivamente, até que obtém-se um *data frame* único com todas as variáveis.

d) Limpeza Dos Dados

De acordo com Sivakumar e Gunasundari (2017), os dados que formam um *dataset* podem ser incompletos, inconsistentes e ruidosos. A correção desses dados, também chamada de rotinas de limpeza, preenchem os valores ausentes, corrige inconsistências e diminui o ruído gerado. Ruídos, segundo Libralon (2017), são dados dentro de um conjunto que não seguem o mesmo padrão dos demais e que quando presentes no *dataset* podem atrapalhar o desempenho do modelo.

Realizou-se então a substituição de valores ausentes pela respectiva média da variável em questão. Esta é uma etapa necessária, pois tem-se de diferentes naturezas, com períodos de coleta diferentes. Os dados de temperatura, por exemplo, possuem coletas diárias, já o *NDVI,* coleta em um intervalo de três dias. Então, quando as informações são unidas, estas podem ser realizadas entre o intervalo de coleta, o que traria uma informação nula, por isso a média é aplicada.

Realizou-se a seleção da última ocorrência de queima para blocos que tiverem mais de um talhão (subdivisão do bloco) queimado, a fim de evitar duplicidade de informação para um mesmo bloco. Para garantir que o processo de limpeza foi realizado da forma correta, através de linhas de código, conferiu-se se o *dataset* possuía valores nulos ou repetidos antes de dar continuidade aos passos.

e) Transformação dos dados e criação de variáveis

Na etapa de transformação, os dados podem ser modificados ou agregados para que se adequem aos modelos utilizados. Uma técnica que pode ser utilizada é a

de normalização onde os dados são inseridos em um intervalo de -1 a 1 ou de 0 a 1. Tem-se também as técnicas de clusterização, de agregação onde os dados podem ser armazenados em intervalos de tempo, como por exemplo um total semanal, mensal, anual... e a categorização dos dados (SIVAKUMAR e GUNASUNDARI, 2017).

Para os dados de chuva, radiação, temperatura e umidade, que são dados coletados diariamente, fez-se cálculos médios de diferentes intervalos de tempo: 40 dias, 30 dias, 15 dias, 8 dias e 4 dias para posterior escolha do intervalo de melhor influência no modelo. Esses intervalos diferentes foram escolhidos após consultar os profissionais que trabalham com a cultura de cana-de-açúcar. Esses relataram que as variáveis relacionadas ao clima possuem influência na ocorrência de um incêndio não apenas no dia do evento, como também em períodos precedentes, e essas devem ser analisadas em diferentes intervalos de dias.

Além dos tratamentos descritos anteriormente, fez-se necessário também o cálculo da distância do bloco até as rodovias da região. Estes cálculos foram feitos a partir das coordenadas do bloco, mais especificamente do centroide e as informações de latitude e longitude das rodovias. Para a realização dos cálculos utilizou-se a biblioteca de código aberto chamada GeoPandas ©, que facilita o uso de dados geoespaciais em *python* ™.

Após os métodos descritos nesta seção, categorizou-se os dados para melhor desempenho do modelo no momento de teste, como por exemplo, as distâncias dos blocos até as rodovias: distâncias, $\leq$ 5km receberam o valor 1, entre 5 e 10 km receberam o valor 2, entre 10 e 15 km o valor 3, assim por diante e acima de 30 km recebeu o valor 7. O mesmo tipo de categorização aplicou-se as médias de chuva, umidade do solo, temperatura e radiação. Esta categorização foi definida de forma visual, onde plotou-se um gráfico de colunas e a partir das faixas com maior número de dados, foram definidas as categorias.

A última técnica utilizada foi a normalização dos dados, realizada com uso da função *"MInMaxScaler"* disponível na biblioteca *sklearn*. Esta técnica dimensiona as variáveis no intervalo 0 e 1, deixando as entradas em um intervalo padrão, o que melhora o desempenho dos modelos de *Machine Lerning.* (BROWNLEE, 2020).

f) Redução dos dados

Um elevado número de variáveis de entrada pode retardar o desempenho do modelo, além do risco de haver variáveis de pouca influência no resultado Características muito próximas de outras variáveis ou até mesmo da variável resposta, podem deixar o modelo enviesado. Após o levantamento de todas as varáveis de interesse coletadas, tratadas e unidas no *dataset*, atingiu-se um conjunto de dados com 30 variáveis diferentes, considerando também os intervalos, aplicou-se a análise de correlação para a escolha das melhores *features*.

A Matriz Correlação consiste em uma tabela que mostra o resultado da correlação das variáveis contidas no *dataset*. As correlações de interesse são entre cada uma das 30 variáveis de entradas (*features*) e a variável de saída (target) que no trabalho em questão é a ocorrência ou não da queima. Então a função “corr” disponível na biblioteca Pandas foi aplicada. Para facilitar a visualização da tabela, esta foi exportada em um documento do tipo “xlsx”, facilitando a exploração e escolha das varáveis. Com esta etapa definiu-se então as seguintes vaiáveis para compor o modelo:

- maturação;
- ambiente;
- corte;
- NDVI médio de 30 dias;
- HHI médio de 30 dias,
- umidade do solo;
- média de chuvas em 40 dias;
- temperatura máxima média de 40 dias;
- radiação solar média de 40 dias;
- distância do bloco até a rodovia.

No presente trabalho, as variáveis não serão apresentadas de forma direta a fim de manter o sigilo da empresa, pois estas são de propriedade da mesma.

3.3 Os modelos

Com o *dataset* pronto, o próximo passo foi o teste dos modelos e comparativo dos resultados com base nas métricas descritas na Seção 2.6.6. Os modelos foram escolhidos a partir da sugestão dos profissionais da área de Inteligência Analítica da empresa que disponibilizou os dados. A escolha baseia-se na diferença como estes modelos realizam a predição e na robustez, em que inicia-se com modelos mais simples que são o SVM e *Naive Bayes* e parte-se para os modelos mais robustos que são o *Random Forest* e *XGBoost*. O processo de tratativas e testes serão descritas nesta seção.

3.3.1 Separação entre treino e teste

Como mencionado no início deste trabalho o objetivo de *Machine Learning* é que os computadores aprendam a partir de dados. Para que esse aprendizado ocorra, o algoritmo deve ser treinado com dados passados. No entanto a fim de evitar que o modelo se depare com alguma informação não conhecida antes, realiza-se uma divisão entre treino e teste. A partir dessa divisão o modelo aprende com uma determinada parcela dados e depois aplica na segunda parcela para assim ser avaliado (GROOTENDORST,2019). Os dados são divididos em 70% para treino 30% para teste. Neste momento como o modelo está aprendendo, este aprendizado é feito apenas com dados de histórico. A divisão dos dados de treino e teste é feita a partir do *dataset* formado com dados das safras 2017/18, 2018/19 e 2019/20.

Além da separação entre treino e teste dentro do *dataset,* tem-se dois tipos de dados, as denominadas *features* ou variáveis de entrada, que são aquelas que o modelo observa para encontrar um padrão e separá-las em uma classe, e a target que é a variável resposta ou de saída, que é a informação se o bloco queimou ou não. Então dentro do conjunto de treino há uma separação entre variáveis X (*features*) e Y (*target*) em que tem-se “X_train” e “Y_train” para o momento de treino “x-test”, o “y-test” é utilizado para comparação dos resultados obtidos pelo teste.

3.3.2 Reamostragem

O canavial é uma extensa área dividida em diversos blocos. Durante as três safras avaliadas. O número de blocos que tiveram sua cultura colhida sem ocorrência de incêndio foi bem maior do que os que tiveram sua cultura queimada. Com a

separação entre treino e teste, dos 70% dos dados de treinamento, 416 eram de blocos em que ocorrem queima contra 3256 blocos sem ocorrência de queima. Como os modelos estão sendo treinados para que aprendam a identificar se o bloco irá ou não se incendiar, necessita-se fornecer o número igual das duas ocorrências, para que os modelos não fiquem enviesados e capazes apenas de identificar com confiabilidade apenas características de não queima. Então, se os modelos fossem treinados com o conjunto de dados neste formato, estes teriam um mau desempenho em relação a classe de queima, por ser minoritária. Por isso a parcela destinada ao treino deve ser reamostrada, para que o número de dados relacionados a queima seja igual aos de não queima.

Para o balanceamento utilizou-se a técnica SMOTE - NC*, Synthetic Minority Over-Sampling Technique for Nominal and Continuous Features*, ou em português, técnica de sobreamostagem de minoria sintética para recursos nominais e contínuos. Esta técnica utiliza a criação de uma nova amostra baseada em vizinhos mais próximos. Então pontos são criados em um segmento de linha, selecionados aleatoriamente, assim como seus vizinhos mais próximos. Esta nova variável então, receberá o valor mais frequente dos vizinhos mais próximos (AGUIAR, 2019).

3.3.3 Treinamento dos modelos

Após o *dataset* preparado, separado entre treinamento e teste e com as classes balanceadas, foram realizados os treinos dos modelos e posterior avaliação. Todo o processo de construção, desde o preparo das variáveis até a obtenção dos resultados foi realizado no ambiente *python™*.

Os modelos preditivos utilizados neste trabalho foram o *Support Vector Machine* (SVM), *Naive Bayes*, *Random Forest* e *XGBosst.* Como a finalidade deste trabalho é a comparação e definição do melhor algoritmo, os parâmetros padrões dos modelos não foram alterados.

Para instanciar o modelo SVM utilizou-se a função *“modelo_svm = svm.SVC()”* da biblioteca sklearn.sv, seguida da função “*modelo_svm.fit(X_trein, Y_trein)*”. Com essas duas funções treina-se o modelo com os 70% dos dados do *dataset*. Para o modelo *Naive Bayes* utilizou-se a função “*modelo_nb = MultinomialNB()”* da biblioteca sklearn.naive_bayes, seguido da função *“modelo_nb.fit(X_trein, Y_trein)”.*

Os dois modelos seguintes seguiram a mesma lógica de treino, inserindo-se a função para “trazer” o modelo, seguida da função em que passe-se os dados para esses modelos treinem. Para o modelo *Random Forest* utilizou-se *“modelo_rf = RandomForestClassifier”* da biblioteca sklearn.ensemble, seguido da função “*modelo_rf.fit(X_trein, Y_trein)”.* E para o modelo *XGBoost* utilizou-se a função *“modelo_xg = XGBClassifier”* da biblioteca xgboost e a função “*modelo_xg.fit(X_trein, Y_trein)”.*

Após chamada cada função, os parâmetros de treino e teste são passados nas funções. Os algoritmos ou modelos tiveram seu funcionamento explicados na revisão bibliográfica. Código a código referente a cada algoritmo foi inserido no ambiente. Um exemplo do processo de treino é mostrado na Figura 8.

Figura 8 - Treinamento modelo *Random Forest*

```
from sklearn.ensemble import RandomForestClassifier


# instanciando o modelo de Random Forest
modelo_rf = RandomForestClassifier()

# treinando o modelo
modelo_rf.fit(X_trein, Y_trein)
```

Fonte: Autoria Própria (2021)

3.3.4 Teste

O momento de teste ocorre na sequência do treinamento, onde utiliza-se funções para inserir as variáveis no treinamento, conforme mencionado na Seção 3.3.1. Para o modelo SVM a função utilizada é a *“pred = modelo_svm.predict(X-test)”,* para *Naive Bayes* a *“pred = modelo_nb.predict(X-test)”,* para o *Random Forest “pred = modelo_rd.predict(X-test)”* e para o *XGBoost* a função *“pred = modelo_xg.predict(X-test)”.* Este processo é mostrado na Figura 9 para o modelo *Random Forest.*

Figura 9 - Teste modelo *Random Forest*

```
# realizar as previsões no dataset de teste
y_pred = modelo_rf.predict(X_test)
```

Fonte: Autoria Própria (2021)

Neste momento pode-se ou não verificar as predições fornecidas pelo algoritmos, em um formato de tabela, porém como é uma fase de aprendizado do modelo, o objeto de interesse corresponde a todas as saídas fornecidas pelas predições, denominado de "y_pred" (saída predita), que será utilizado na fase de avaliação de desempenho dos modelos.

3.3.5 Desempenho dos modelos

Como apresentado na Seção 2.6.6, diversas são as formas de avaliar o desempenho de modelos preditivos. As métricas de avaliação utilizadas neste trabalho foram: a acurácia obtida pela função *"accuracy_score"*, *recall*, precisão e f-1 *score* obtidas pela função *"classification_report",* duas funções disponíveis na biblioteca *sklearn,* e utilizou-se também a Matriz Confusão para a verificação do número de verdadeiros positivos preditos pelo modelo, aplicando a função "*confusion_matrix"* também disponível na biblioteca *sklearn*.

3.3.6 Definição do melhor modelo

Para a definição do melhor modelo para o problema em questão, foi avaliado o algoritmo que possui a melhor performance na predição das ocorrências de incêndios. Entende-se que o modelo deve ser capaz de realizar predições tanto para uma possível ocorrência de queima, quanto para a de não queima, porém a resposta com um maior peso é em relação a ocorrência do incêndio. Esta diferença de pesos se dá, pois, um bloco que sofre um incêndio e não é predito pelo algoritmo gera um impacto negativo para a organização e a sociedade. Entretanto um bloco que é predito como caso de incêndio, porém este não ocorre, gera um impacto bem menor.

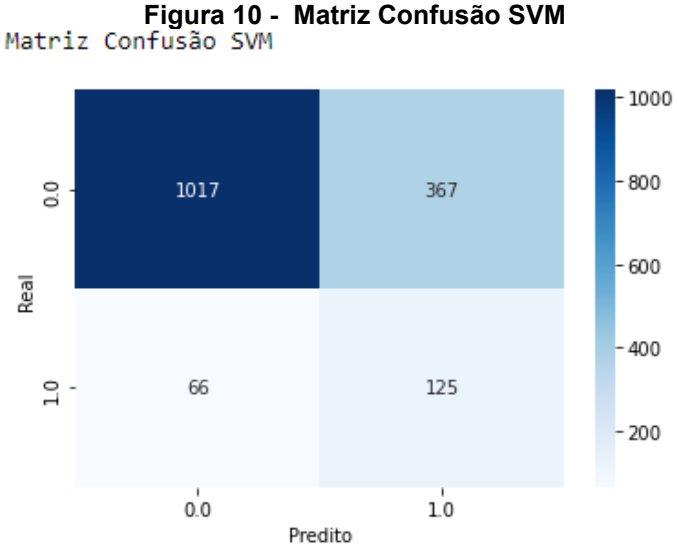
4 RESULTADOS E DISCUSSÃO

Após todas as etapas descritas nas seções anteriores, cada um dos modelos foi implementado e as métricas de avaliação de desempenho foram exibidas Todo o processo de tratamento, implementação e cálculos das métricas de avaliação foram realizadas no mesmo ambiente do Jupyter. O tempo para processamento, foi de aproximadamente 30 minutos. Esse tempo é variável, dependente principalmente do momento da realização dos *"merges"*. Quanto maior for a quantidade de linhas presentes nas tabelas, maior será o tempo de processamento. Os resultados serão discutidos nesta seção.

4.1 Desempenho SVM – Suporte Vector Machine

Inicialmente observou-se a Matriz Confusão, Figura 10, atribuída ao modelo, comparando o resultado predito pelo modelo e o valor real da variável saída, a ocorrência ou não de queima.

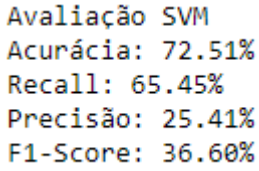
Figura 10 - Matriz Confusão SVM



Fonte: Autoria Própria (2021)

O modelo foi capaz de predizer 125 das 191 ocorrências de incêndio e 1017 não ocorrência de um volume de 1384. A Figura 11 ilustra os valores obtidos para as métricas de desempenho.

Figura 11 - Métricas de desempenho SVM



Fonte: Autoria Própria (2021)

Na Figura 11, pode se observar que o modelo alcançou um valor de 72,51%, ou seja, o algoritmo acertou 72,51% das predições realizadas, que é considerado um valor alto. A definição do melhor modelo não será apenas avaliando esta métrica, pois como mencionado anteriormente, a acurácia não considera a classe de maior importância. Sendo essa para o presente trabalho, a de ocorrência de incêndio, representada pelo valor 1.0 na Matriz Confusão.

O *recall* alcançado foi de 65,45%, significando que o modelo foi capaz de predizer 65,45% das ocorrências de queima. Essa é a métrica mais importante para objetivo deste trabalho, pois, a partir dela é que será definido o melhor modelo capaz de realizar predições de ocorrências de incêndios com base no conjunto de dados definido.

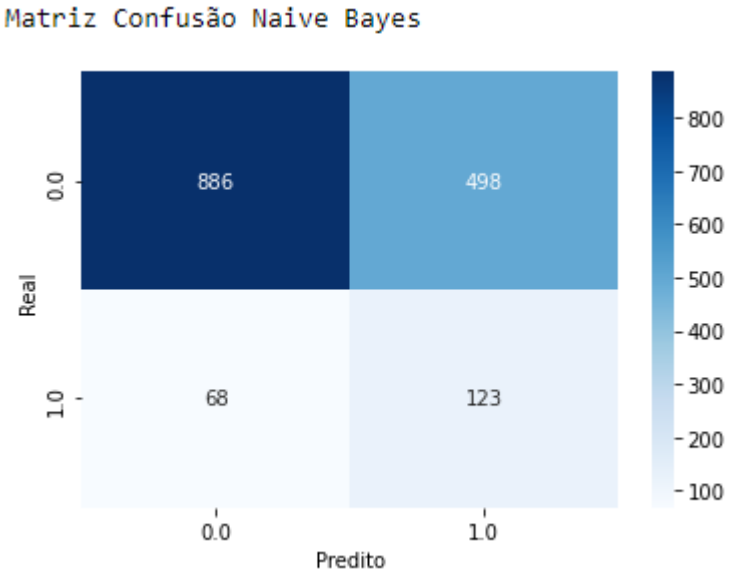
A precisão alcançada foi de 25,41%, que corresponde ao percentual de acerto das predições dadas como queima pelo algoritmo. Isso significa que o modelo predisse que muitos blocos iriam incendiar, mais precisamente 492, porém apenas 125 se incendiaram. Unindo a métrica *recall* e precisão chega-se ao valor de 36,60% de f1-*score*, um valor condizente visto que se obteve um baixo valor de precisão.

Pode-se observar que o modelo SVM, obteve bom desempenho de acurácia, um recall satisfatório e uma baixa precisão.

4.2 Desempenho Naive Bayes

Assim como para o algoritmo SVM, a Matriz Confusão, apresentada na Figura 12, foi observada a fim de verificar de forma visual a quantidade de acertos que o modelo obteve em cada classe predita.

Figura 12 - Matriz Confusão *Naive Bayes*



Fonte: Autoria Própria (2021)

O 4° quadrante da Matriz presente na Figura 12, representa o número de ocorrências de incêndios que o modelo foi capaz de predizer. O 3° quadrante representa o número de ocorrências de incêndio que o modelo não foi capaz de predizer, estes foram considerados como não queima. Então somando os valores contidos nos dois quadrantes, tem-se um total de 191 ocorrências de incêndio inseridas no modelo, e desta 123 foram preditas corretamente. Os quadrantes superiores correspondem ao número de blocos não queimados, um total de 1384, em que o modelo foi capaz de predizer a não ocorrência de forma correta em 886 blocos, presente no 2° quadrante.

A Figura 13 apresenta os valores obtidos para as métricas de desempenho.

Figura 13 - Métricas de desempenho *Naive Bayes*

```
Avaliação Naive Bayes
Acurácia: 64.06%
Recall: 64.40%
Precisão: 19.81%
F1-Score: 30.30%
```

Fonte: Autoria Própria (2021)

Na Figura 13, observa-se que o modelo alcançou um valor de 64,06%, ou seja, percentual de acertos pelo algoritmo das predições realizadas. Na literatura não se encontra um valor ótimo para as métricas analisadas, pois depende da complexidade, do tipo de comparação e do tipo de dados. Como neste trabalho

deseja-se implementar um algoritmo capaz de predizer as ocorrências de incêndio nos canaviais, uma porcentagem de acertos de 64,06% não é atrativa, pois tem-se mais de 35% de blocos podendo ser classificados de forma errônea.

O *recall* alcançado foi de 64,40%, significando que o modelo foi capaz de predizer 64,40% das ocorrências de queima. Como a classe de importância deste trabalho é a de ocorrência de incêndio, necessita-se que o modelo tenha o maior valor de *recall* possível e em comparação com o modelo SVM demostrado na seção anterior, o modelo *Naive Bayes* obteve desempenho inferior.

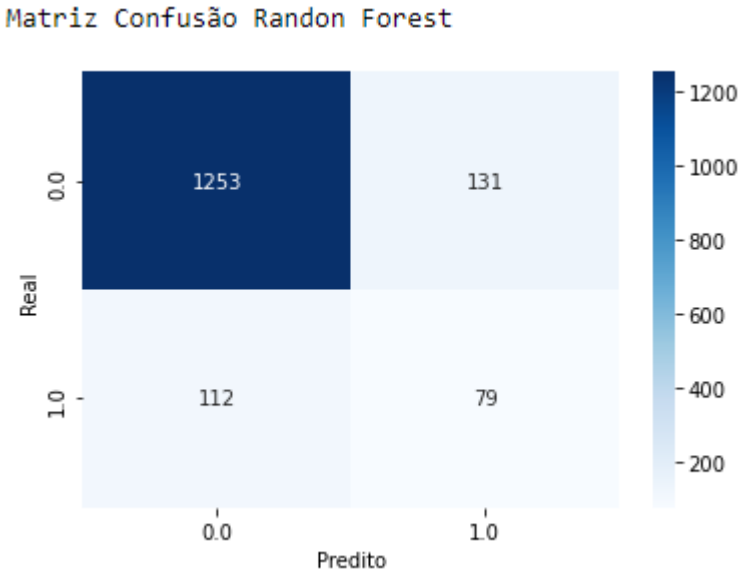
Já a precisão alcançada foi de 19,81%, correspondendo ao percentual de acerto das predições dadas como queima pelo algoritmo. Este valor abaixo de 20%, mostra que o modelo está tendenciado a predições de ocorrências de queima, então este pode não ser ideal para realizar predições precisas com o conjunto de varáveis disponíveis. Unindo a métrica *recall* e precisão chegamos ao valor de 30,3% de *f1-score*, um valor condizente visto que se obteve um baixo valor de precisão.

O modelo *Naive Bayes* então, obteve bom desempenho de acurácia, um *recall* satisfatório considerando que este é capaz de predizer mais de 50% das ocorrências e uma baixa precisão. Faz-se necessário então o comparativo com os demais modelos.

4.3 Desempenho Random Forest

Observou-se também a Matriz Confusão, Figura 14, composta pelos acertos e erros obtidos pelo modelo no momento de teste.

Figura 14 - Matriz Confusão *Random Forest*



Fonte: Autoria Própria (2021)

O modelo foi capaz de predizer 79 das 191 ocorrências de incêndio, mostrando que nesta classe o número de erros foi maior do que os de acertos, e 1283 não ocorrência de um volume de 1384. Os valores de desempenho estão demostrados na Figura 15:

Figura 15 - Métricas de desempenho *Random Forest*

```
Avaliação Random Forest
Acurácia: 84.57%
Recall: 41.36%
Precisão: 37.62%
F1-Score: 39.40%
```

Fonte: Autoria Própria (2021)

Observa-se na Figura 15 que o modelo realizou 84,57% predições de forma correta, um valor atrativo comparado as acurácias dos algoritmos anteriores, mas como mencionado anteriormente, essa métrica analisada isoladamente não define o melhor modelo no estudo em questão.

O *recall* alcançado foi de 41,50%, significando que o modelo foi capaz de predizer de forma correta menos de 50% dos incêndios ocorridos, um percentual inferior aos modelos analisados anteriormente, e que distancia este de ser o melhor modelo para predizer a as ocorrências de incêndio.

Já a precisão alcançada foi de 37,62%, ou seja, dos 219 blocos classificados pelo algoritmo como queima, 79 estavam corretos. Visto que o número real de

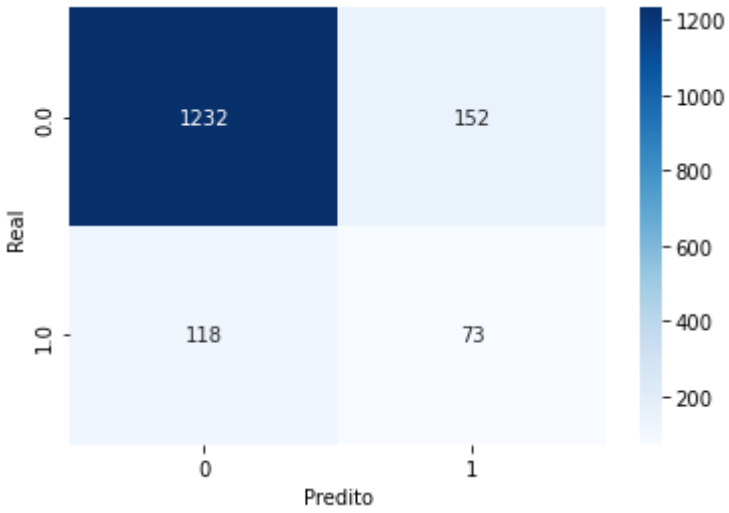
ocorrências de queima foi de 191, e mesmo o modelo classificando 219 ocorrências de queima, foi de maneira errônea, mostrando não estar encontrando uma forma de classificar precisamente as ocorrências de incêndio com base nas variáveis disponíveis. O *f1-score* obtido foi de 39%, como essa métrica é calculada através da média harmônica do *recall* e da precisão, e nos dois casos foram valor próximos a 40%. O desempenho médio do modelo considerando que as ocorrências de incêndio foram preditas de forma correta e que o algoritmo conseguiu encontrar padrões para classificar incêndio, foi de próximo a 40%

Apesar da excelente acurácia em relação aos demais algoritmos, o modelo *Random Forest* não se mostrou eficiente no momento de realizar predições para a classe de incêndio. Pode-se observar na Matriz Confusão (Figura 14) um baixo número de verdadeiros positivos, como também o baixo valor de *recall* apresentado, comparado aos modelos analisados anteriormente.

4.4 Desempenho XGBoost

O último modelo a ter seu desempenho avaliado foi o *XGBoost*. Como nos demais, observou-se a Matriz Confusão (Figura 16) com o comparativo de suas predições e os valores reais de ocorrência ou não de incêndios.

Figura 16 - Matriz Confusão *XGBoost*

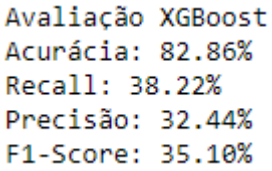


Fonte: Autoria Própria (2021)

O algoritmo foi capaz de predizer apenas 73 das 191 ocorrências de incêndio, mostrando que nesta classe o número de erros foi maior do que o de acertos, e

predisse 1232 não ocorrências de incêndio de um volume de 1384. Os valores de desempenho estão demostrados na Figura 17.

Figura 17 - Métricas de desempenho *XGBoost*



Fonte: Autoria Própria (2021)

Conforme observado na Figura 17, o modelo alcançou uma acurácia 82,86%, um valor próximo ao algoritmo *Random Forest*, assim como mencionado na seção anterior, esta métrica engloba todos as predições corretas realizadas pelo algoritmo, o que pode significar que os 17,14% possam ser incêndios não preditos, sendo necessária a avaliação das demais métricas.

O *recall* alcançado foi de 38,22%, mostrando que este algoritmo, apesar de robusto, não está sendo capaz de predizer a classe de incêndio de uma forma assertiva, visto que resultados melhores já foram alcançados pelos outros algoritmos avaliados.

Já a precisão alcançada foi de 32,44%, correspondendo porcentagem de acertos das predições dadas como queima pelo algoritmo, mostrando que das 191 predições feitas para queima (soma do 3º e 4° quadrante da figura 16), 32,44% foram corretas. O *f1-score* obtido pelas duas métricas anteriores foi de 35,10%, desempenho relacionado a falsos positivos, ou seja, os blocos classificados como queima, realmente queimaram.

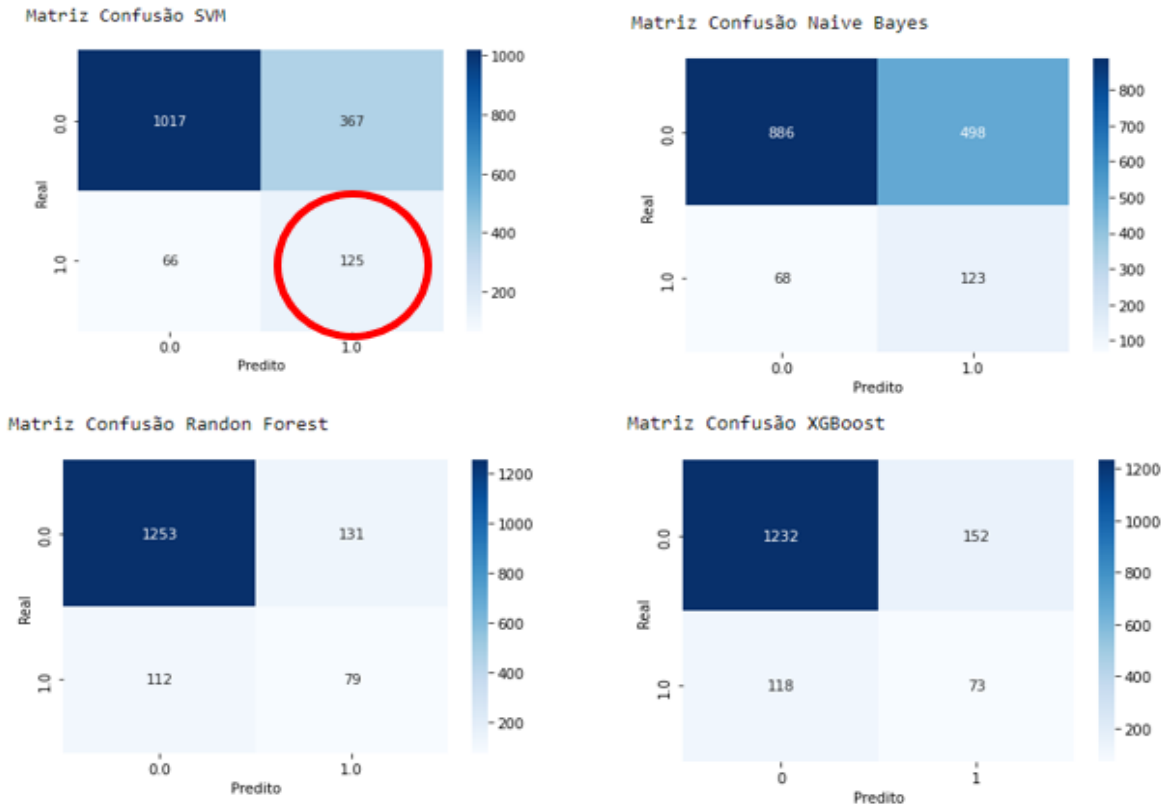
O modelo *XGBoost* mostrou-se com baixo desempenhos nas predições de incêndios, comparado ao outros modelos, apesar de uma boa taxa de acertos nas predições de blocos não queimados, o que caracteriza que os dados não estão sendo suficientes para o modelo identificar uma ocorrência de incêndio.

4.5 Definição do melhor modelo

Como mencionado na Seção 3.3.6, o modelo a ser escolhido como melhor preditor de incêndio será avaliado com base na métrica *recall*. O *recall* corresponde à capacidade do algoritmo de predizer classes positivas. Neste caso a ocorrência de incêndios para o bloco, ou a queima como denominada a *target* do *dataset.*

Inicialmente, como a Matriz Confusão é uma ferramenta em que facilmente se observa os resultados verdadeiros e falsos obtidos por cada algoritmo em sua etapa de teste, foram analisadas as matrizes dos quatro modelos em conjunto, conforme Figura 18, a fim de verificar qual o foi algoritmo capaz de predizer de forma correta o número de incêndios,

Figura 18 - Comparativo Matriz Confusão



Fonte: Autoria Própria (2021)

O 4° quadrante de cada matriz, referente ao número de verdadeiros positivos correspondente as ocorrências de incêndios preditas de corretamente. Observando esse quadrante na Figura 18, verifica-se que os modelos *Random Forest* e *XGBoost* obtiveram valores inferiores e próximos entre si, em que maior foi o número de erros do que acertos para a classe preditora queima, representada pelo valor 1. Por outro os modelos SVM e *Naive Bayes* possuem os valores mais altos de ocorrência de incêndios preditos corretamente. Desta análise inicial, o modelo SVM mostra-se mais eficiente em predizer incêndios, conforme destacado em cor vermelha na Figura 18, seguido do modelo *Naive Bayes*, com uma diferença de apenas dois blocos preditos corretamente para classe de queima.

O modelo *Naive Bayes* também se mostrou com bom desempenho em relação aos verdadeiros positivos, mas comparando este com o modelo SVM é possível observar que o algoritmo não teve um desempenho tão elevado em relação à classe não queima, apresentada no 2° quadrante das matrizes contidas na Figura 18.

O Quadro 4 mostra uma comparação das principais métricas para os quatro modelos avaliados no presente trabalho.

Quadro 4 - Comparativo métricas de desempenho

Métrica/ Modelo	SVM	Naive Bayes	Random Forest	XGBoost
Acurácia	72,51%	64,06%	84,57%	82,86%
Recall	65,45%	64,40%	41,36%	38,22%
Precisão	25,41%	19,81%	37,62%	32,44%

Fonte: Autoria Própria (2021)

Os modelos que obtiveram maiores valores de acurácia foram o *Random Forest* e o *XGboos*t, conforme apresentado Quadro 2. Esse parâmetro mede a taxa total de predições realizadas corretamente pelo algoritmo. Conforme observa-se na Figura 18, a adição do 3° e 4° quadrante corresponde aos valores reais de ocorrência de incêndios (191 no total), que é inferior a soma do 1° e 2° quadrante (138 no total), que representa a não ocorrência de incêndio. Essa análise implica que mesmo os modelos não sendo capazes de predizer incêndios, estes podem apresentar um bom desempenho avaliando apenas acurácia. Isso ocorre devido ao maior número de blocos não queimados nos canaviais. Avaliando então o *recall* dos mesmos, nota-se que estes foram capazes de predizer menos da metade das ocorrências de incêndio, sendo essa a métrica com maior peso. A precisão nestes dois casos foi superior aos demais, porém como poucos foram as predições realizadas para incêndio. Essa métrica não é suficiente para dizer que os modelos obtiveram desempenho superior. Devido as considerações feitas a partir do *recall*, necessita-se avaliar e comparar os demais modelos, pois o *Random Forest* e *XGBoost* não são os modelos mais indicados para predizer ocorrências de incêndio.

Os modelos SVM e *Naive Bayes* obtiveram o maior desempenho em relação ao *recall*. Este comportamento mostra que esses modelos resultaram em uma melhor predição das ocorrências de incêndio dentro do conjunto de teste. Partindo desta análise estes são os modelos mais indicados no primeiro momento para o objetivo

deste trabalho, entretanto, comparando as demais métricas, o modelo *Naive Bayes* teve um desempenho inferior, tendo uma precisão menor que 20% e a menor taxa de acertos comparado aos demais algoritmos.

De maneira geral, os melhores resultados foram obtidos utilizando o algoritmo SVM, pois seu *recall* atingiu o valor mais alto, sendo a métrica de maior importância, seguida de uma precisão mais elevada que a do modelo *Naive Bayes* que foi o segundo melhor modelo capaz de predizer as ocorrências de incêndio e também uma acurácia superior a 75%.

5 CONCLUSÃO

O presente trabalho teve como finalidade definir o melhor algoritmo capaz de predizer a ocorrência de incêndio em canaviais. Para o objeto de estudo, foi utilizado o banco de dados de uma empresa sucroenergética localizada no interior do Estado de São Paulo. Inicialmente foi necessário o entendimento da problemática que envolve os incêndios, embasando-se em pesquisas e consultando pessoas experientes que trabalham na empresa em questão.

A imersão neste universo de *Machine Learning* se deu através também de pesquisas e consultando os membros da equipe de Inteligência Analítica (IA), da empresa fornecedora dos dados. Para que os modelos escolhidos fossem aplicados necessitou-se da conexão com os dados da empresa feitas através de banco de dados relacional *PostgresSQL©*, através do *software DBeaver.* Neste *software* foram realizadas as primeiras manipulações dos dados em questão, onde manipulou, filtrou, transformou e uniu os dados iniciais, que correspondem a varáveis características dos blocos próprios da empresa, condições climáticas e localização.

Os modelos escolhidos para as análises, comparativo e definição do que melhor se aplica a predição de incêndio foram algoritmos classificadores, sendo estes: *Support Vector Machine* (SVM), (algoritmo que realiza a classifica através da separação das varáveis através de um hiperplano), *Naive Bayes* (classifica as variáveis através de probabilidades), *Random Forest* (utiliza de um conjunto de árvores aleatórias de decisão) e o *XGBoos*t (algoritmo também baseado em árvores decisão com aumento de gradiente). O ambiente utilizado para as implementações foi o *Jupyter Notebook*, utilizando-se de diversas bibliotecas como pandas, *sklearn* e geopandas.

Durante o processo de execução deste trabalho até o momento da definição do melhor modelo, passou-se pelas seguintes fases: levantamento de ferramentas e dados utilizados, pré-processamento de dados e execução dos modelos.

A partir da avalição das métricas e visualização dos resultados através da Matriz Confusão, onde a métrica de *recall* foi a de maior importância, já que o objetivo deste projeto foi definir o modelo que melhor prediz a ocorrência de incêndio, definiu-se o algoritmo SVM como o melhor em realizar as predições positivas para ocorrência de queima nos blocos dos canaviais.

Para este trabalho as limitações encontradas foram em relação ao número de varáveis disponíveis, pois existem outras variáveis, como por exemplo: umidade do ar, velocidade do vento, direção do vento e alguns índices vegetativos que influenciam na ocorrência ou não de um incêndio. Essas variáveis são limitadas pois passaram a ser coletadas recentemente pela empresa e não possuem histórico.

Partindo das limitações encontradas para a execução deste trabalho, uma das sugestões para trabalhos futuros é a busca por novas variáveis que possam ter influência sobre o incêndio. Seja a espera por uma quantidade significante de dados que sirvam para o aprendizado do modelo, seja pela busca de dados externos ao banco da empresa. Uma segunda sugestão para trabalhos futuros é o estudo e testes com os diferentes parâmetros internos do modelo, a fim de melhorar seu poder de predição incêndios.

Este trabalho contribui para a empresa fornecedora dos dados, pois esta pode se antever a uma ocorrência de incêndio evitando ou limitando seus impactos, além de preservar as matas ciliares ao redor dos canaviais e prevenir a geração de multas ambientais. O presente trabalho contribui também com a sociedade, pois esses incêndios podem ser evitados, evitando a emissão de gases e partículas finas, poluição visual e perda de fauna e flora.

Conclui-se com este trabalho então o objetivo de definição do algoritmo capaz de predizer a ocorrência de incêndio em canaviais, com base em dados históricos de informações sobre os blocos de cana-de-açúcar, dados climáticos e de localização.

REFERÊNCIAS

AGUIAR, F. ***Synthetic Minority Over-sampling Technique for Nominal and Continuous*****.** Outubro, 2019. Disponível em:< https://medium.com/analytics-vidhya/smote-nc-in-ml-categorization-models-fo-imbalanced-datasets-8adbdcf08c25> Acesso em: 31 de outubro de 2021.

ALARAU, E. G, et al, **Perspectives on Big Data and Big Data Analytics,** Database Systems Journal, Vo. 3, no 4, 2012.

ALMEIDA, M. R., **Uso da tecnologia na produção canavieira**, Trabalho de conclusão de curso – Universidade Federal do Paraná, Curitiba, p. 48, 2016.

ANDRADE, P. I de, BRILHANTE, L. V. T. **Predizendo epidemias de Dengue, no Distrito Federal, utilizando algoritmos de Regressão**, Trabalho de Conclusão de Curso – Universidade de Brasília – UnB Faculdade UnB Gama – FGA, Brasília, DF, , 2018.

AVOLIO, E. B., **Da (i)licitude das queimadas da palha da cana-de-açúcar**. Dissertação Mestrado – Escola de Engenharia de São Calos – Universidade de São Paulo, São Carlos, p. 231, 2002.

BREIMAN, L., ***Random Forests***, *Statistics Department, University of California, Berkele*y, 2001. Disponível em: <https://link.springer.com/article/10.1023%2FA%3A1010933404324#article-info> Acesso em: 25 de outubro de 2020.

BORGES, L. S. F, et al., **Impactos ambientais e sociais causados pela queima da cana-de-açúcar**. Monumenta, Paraíso do Norte, PR, v. 1, n. 1, p. 73-83, maio 2020.

BROWNLEE, J., ***A Gentle Introduction to XGBoost for Applied Machine Learning***. Agosto, 2016. Disponível em: <https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/> Acesso em: 02 de Novembro de 2011.

BROWNLEE, J., ***How to Use StandardScaler and MinMaxScaler Transforms in Python*****.** Junho, 2020. Disponível em: < https://machinelearningmastery.com/standardscaler-and-minmaxscaler-transforms-in-python/> Acesso em: 24 de outubro de 2021.

BROWNLEE, J., ***Extreme Gradient Boosting (XGBoost) Ensemble in Python*****.** Novembro, 2020. Disponível em: < https://machinelearningmastery.com/extreme-gradient-boosting-ensemble-in-python/> Acesso em: 24 de outubro de 2021.

CARDOSO, A. A.; MACHADO, C. M. D.; PEREIRA, E. A. **Combustível, o mito do combustível limpo**. Química Nova Na Escola, n. 28, p. 9 -14, maio 2008.

CASTRO, L., N. DE, FERRARI, D. G. **Introdução a Mineração de Dados**, Conceitos Básicos, algoritmos e aplicações, Saraiva, 1ª ed., p. 324 – 332, São Paulo, 2016.

CONAB, Companhia Nacional de Abastecimento. **Acompanhamento da safra brasileira – Cana-de-açúcar**. Safra 2019/20, Brasília, v.6, n.1, p. 1 – 58, maio 2019

DUARTE, G. ***Gradient Boostings* Parte 2: *XGBoost.*** Junho, 2020. Disponível em: <https://datarisk.io/gradient-boostings-parte-2-xgboost/?cn-reloaded=1>. Acesso em 24 de outubro de 2021.

ESCOVEDO, T., ***Machine Learning:* Conceitos e Modelos — Parte I: Aprendizado Supervisionado.** Junho, 2020. Disponível em:<https://tatianaesc.medium.com/machine-learning-conceitos-e-modelos-parte-ii-aprendizado-n%C3%A3o-supervisionado-fb6d83e4a520 >Acesso em: 24 de outubro de 2021.

ESCOVEDO, T., ***Machine Learning:* Conceitos e Modelos — Parte II: Aprendizado Não-Supervisionado.** Junho, 2020 Disponível em:<https://tatianaesc.medium.com/machine-learning-conceitos-e-modelos-parte-ii-aprendizado-n%C3%A3o-supervisionado-fb6d83e4a520 >Acesso em: 23 de outubro de 2021.

FERREIRA, J. C., SIQUEIRA, S. S., BERGONSO, V. R., **Impactos causados pela fuligem da cana-de-açúcar**, Lins – SP, 2009.

FUCKS, K., ***Machine Learning: Classification Model***, Disponível em:< https://medium.com/fuzz/machine-learning-classification-models-3040f71e2529> Acesso em: 23 de outubro de 2021.

GOVERNO DO ESTADO DE SÃO PAULO, Secretaria de Infraestrutura e Meio Ambiente, **Etanol Mais Verde**, São Paulo, 2020. Disponível em:< https://www.infraestruturameioambiente.sp.gov.br/etanolverde/> Acesso em: 30 de setembro de 2020.

GOLDSCHMIDT, R., PASSOS, E., **Data Mining, Um Guia Prático. Conceitos, técnicas, ferramentas, orientações e aplicações.** Elsevie, 4ª Reimpressão, Rio de janeiro, 2005.

GROOTENDORST, M. ***Validating your Machine Learning Model***. Setembro, 2019. Disponível em: <https://towardsdatascience.com/validating-your-machine-learning-model-25b4c8643fb7> Acesso em: 25 de outubro de 2021.

GUAZELLI, A. **Técnicas de Modelagem Preditiva**. Disponível em: < https://developer.ibm.com/br/articles/ba-predictive-analytics2/ > Acesso em 05 de dezembro de 2021.

HARRINGTON, P., ***Machine Learning in Action***. *Manning Publications* Co. p. 101-125., 2012 Disponível em: < http://www2.ift.ulaval.ca/~chaib/IFT-4102-7025/public_html/Fichiers/Machine_Learning_in_Action.pdf> Acesso em: 24 de outubro de 2021

IEA, Instituto de Economia Agrícola, **Protocolo Agroambiental do Setor Sucroenergético Paulista: dados consolidados das safras 2007/08 a 2013/14**, Dezembro de 2014. Disponível em: < http://www.iea.sp.gov.br/Relat%C3%B3rioConsolidado1512.pdf > Acesso em: 16 de Novembro de 2021.

IZBICKI, R. e Santos, T. M. dos, **Aprendizado de máquina: uma abordagem estatística.** [livro eletrônico], São Carlos, SP, 2020. Disponível em: < http://www.rizbicki.ufscar.br/AME.pdf> Acesso em: 17 de outubro de 2021.

LIBRALON, G., L., **Investigação de Combinações de Técnicas de Detecção de Ruído para Dados de Expressão Gênica**, Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação – IC MC – USP, São Carlos, 2017.

LIRA, L. D.De B., et al**.. Análise do algoritmo *naive bayes* na classificação de amostras do banco de dados hepatite**. Anais IV CONAPESC, Realize Editora, Campina Grande, 2019. Disponível em: <https://www.editorarealize.com.br/index.php/artigo/visualizar/56474>. Acesso em: 27 de outubro de 2021.

MARIANO, D., **Métricas de avaliação em *machine learning*. Junho, 2021.** Disponível em: <https://bioinfo.com.br/metricas-de-avaliacao-em-machine-learning-acuracia-sensibilidade-precisao-especificidade-e-f-score/ > Acesso em: 31 de outubro de 2021.

MASSULO, Y. A G., **Análise preditiva de ocorrências de incêndios no bioma amazônico do Maranhão**. GeoTextos, vol. 14 n. 2, p.185-211, dezembro 2018.

MATSUMOTO, F., FERNANDES. G. **Modelos de Predição | *Random Forest.*** Setembro de 2019. Disponível em: <https://medium.com/turing-talks/turing-talks-18-modelos-de-predi%C3%A7%C3%A3o-random-forest-cfc91cd8e524> Acesso em: 31 de Outubro de 2021

MIRANDA, R. A. de. **Breve história da agropecuária brasileira**. In: LANDAU, E. C.; SILVA, G. A. da; MOURA, L.; HIRSCH, A.; GUIMARAES, D. P. (Ed.). Dinâmica da produção agropecuária e da paisagem natural no Brasil nas últimas décadas: cenário histórico, divisão política, características demográficas, socioeconômicas e ambientais. Brasília, DF: Embrapa, v. 1, cap. 2, p. 31-57, 2020.

MURPHY, K. p., ***Machine Learning: A Probabilistic Perspective*.** *Massachusetts Institute of Technology*, p. 1 -25, 2012.

OLIVEIRA, A. L. S., **Comparação e validação da modelagem espacial de riscos de incêndios considerando diferentes métodos de predição**. Bulletin of Geodetic Sciences, Articles Section, Curitiba, v. 23, no4, p.556 - 577, Oct - Dec, 2017.

PARDO, T. A S., NUNES, M DAS G. V., **Aprendizado Bayesiano Aplicado ao Processo de Línguas Naturais**, Série de Relatórios do Núcleo Interinstitucional de Linguística Computacional, São Carlos, 2002.

PEREIRA, A. L., et al**,** Diagnóstico de arritmias cardíacas aplicando técnicas de aprendizado de máquina**. Impactos das tecnologias na engenharia biomédica.** a [recurso eletrônico] / Organizador Fabrício Loreni da Silva Cerutti. – Ponta Grossa, PR: Atena Editora, p.34, 2020. Disponível em: < https://www.researchgate.net/publication/338661978_ANALISE_EM_MULTIRRESOLUCAO_DO_SINAL_DE_ELETROCARDIOGRAMA_PARA_DETECCAO_DE_CARDIOPATIAS/link/5f7f3b0ba6fdccfd7b4fd4fb/download> Acesso em: 24 de outubro de 2021.

PLEC, O.; et al., **Mecanização do corte da cana-de-açúcar como fator de sustentabilidade ambiental no paraná: uma análise de cenário**. Rev. Ciên. Empresariais da UNIPAR, Umuarama, v. 8, n. 1 e 2, p. 53-72, jan./dez. 2007.

PRATA, G. A., **Mapeamento da probabilidade de incêndio e cicatrizes de dano como suporte ao manejo florestal**. Tese Doutorado. USP – Escola Superior de Agricultura “Luiz de Queiroz”, Piracicaba, 2019.

RONQUIM, C. C. **Queimadas na colheita da cana-de-açúcar: impactos ambientais, sociais e econômicos. Campinas**, SP: Embrapa Monitoramento por Satélite. Documentos, 77, p.45, dez 2010.

ROZA, F. S. da. **Aprendizagem de máquina para apoio à tomada de decisão em vendas do varejo utilizando registros de vendas**. Monografia - Curso de Engenharia de Controle e Automação. Florianópolis, setembro de 2016.

RUSSON, P, **Big Data Analytics**, TDWI (The Data Warehousing Institute^{TM}), 2011,. Disponível em:< https://vivomente.com/wp-content/uploads/2016/04/big-data-analytics-white-paper.pdf> Acesso em: 05 de dezembro de 2021.

SANTOS, A. C., **Colheita Mecanizada de cana-de-açúcar (*Saccharum spp.*) sem queima prévia: análise de parâmetros de desempenho efetivo**. Dissertação de Mestrado – Escola Superior de Agricultura “Luiz de Queiroz”, Piracicaba, 2012.

SANTOS, H. G. dos**, Comparação da performance de algoritmos de *machnie leraning* para a análise preditiva em saúde pública e medicina**. Tese Programa de Pós-Graduação, Faculdade de Saúde Pública da Universidade de São Paulo, São Paulo, 2018.

SCHADE, G., **Machine Learning: métricas para Modelos de Classificação.** Abril, 2019. Disponível em: <https://imasters.com.br/desenvolvimento/machine-learning-metricas-para-modelos-de-classificacao> Acesso em 31 de outubro de 2021.

SENAR, **Serviço Nacional de Aprendizagem Rural. Fogo: prevenção e controle no meio rural**. Brasília, Coleção SENAR, 227, p.88, 2018.

SILVA, A. B DA, **Big Data, Mineração de Dados e Aprendizagem de Máquina: Formas de extrair informação em grandes volumes de dados**, Revista Dimensão Acadêmica, v.4, n.2, jul-dez,2019

SILVA, F. B. F. DA, **Pré-processamento de Dados e Comparação entre Algoritmos de *Machine Learning* para a Análise Preditiva de Falhas em Linhas de Produção para o Controlo.** Dissertação para obtenção do Grau de Mestre em Engenharia Informática**,** Politécnico do Porto**,** Porto, junho 2021.

SILVA, F. I. C., GARCIA, A. **Colheita mecânica e manual da cana-de-açúcar: histórico e análise**, Nucleus, Ituverava, v.6, n.1, p. 233- 248, abr. 2009

SILVA, F., et al., **Avaliação da produtividade agrícola da cana-planta e cana-soca sob diferentes espaçamentos entre plantas para produção de açúcar e etanol**. Campinas, SP: Boletim de pesquisa e desenvolvimento / Embrapa Informática Agropecuária, ISSN 1677-9266, 40, p.86, dez 2015.

SIVAKUMAR, A., GUNASUNDARI, R. ***A Survey on Data Preprocessing Techniques for Bioinformatics and Web Usage Mining***, *Department of Computer Science, Karpagam University,* ***International Journal of Pure and Applied Mathematics***, v.117, n.20, p 785-794, 2017.

SÃO PAULO**. Lei n. 11.241, DE 19 DE SETEMBRO DE 2002**. Dispõe sobre a eliminação gradativa da queima da palha da cana de açúcar. Disponível em:< https://www.al.sp.gov.br/repositorio/legislacao/lei/2002/lei-11241-19.09.2002.html> Acesso em: 29 de setembro de 2020.

SOUZA, Z. M. de; PRADO, R.M.; PAIXAO, A.C.S.; CESARIN, L.G. **Sistemas de colheita e manejo da palha da de cana-de-açúcar**. Pesquisa Agropecuária Brasileira, v.40, p.271-278, 2005b.

TORQUATO, S.A. MARTINS, R. e RAMOS, de F. **Cana-de-açúcar no Estado de São Paulo: eficiência econômica das regionais novas e tradicionais de produção**. Informações Econômicas, SP, v.39, n. 5, maio de 2009.

TOSTO, S. G.; PAIVA SOBRINHO, R.; ANDRADE, D. C. **Valoração ambiental da perda de solo na cultura da cana-de-açúcar sob colheita queimada e mecanizada no município de Araras**, SP. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE ECONOMIA, ADMINISTRAÇÃO E SOCIOLOGIA RURAL - SOBER, 48., 2010, Campo Grande, MS. Anais... Campo Grande, MS: ANPPAS/IRIS, 2010.

UNICA, União da Agroindústria Canavieira do Estado de São Paulo, **Protocolo Agroambiental**. Disponível em: < https://unica.com.br/iniciativas/protocolo-agroambiental/> Acesso em: 30 de setembro de 2020.