INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
***CAMPUS* CAJAZEIRAS**
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

ANÁLISE PREDITIVA E EXPLORATÓRIA DE DADOS PARA AUXILIAR NO COMBATE À EVASÃO ESTUDANTIL NOS CURSOS SUPERIORES DO IFPB

FRANCISCO MATHEUS VALENÇA TRAJANO

Cajazeiras
2023

FRANCISCO MATHEUS VALENÇA TRAJANO

ANÁLISE PREDITIVA E EXPLORATÓRIA DE DADOS PARA AUXILIAR NO COMBATE À EVASÃO ESTUDANTIL NOS CURSOS SUPERIORES DO IFPB

Trabalho de Conclusão de Curso apresentado junto ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba - *Campus* Cajazeiras, como requisito à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador

Prof. MSc. Francisco Paulo de Freitas Neto.

Cajazeiras
2023

IFPB / Campus Cajazeiras
Coordenação de Biblioteca
Biblioteca Prof. Ribamar da Silva
Catalogação na fonte: Cícero Luciano Félix CRB-15/750

T768a Trajano, Francisco Matheus Valença.
Análise preditiva e exploratória de dados para auxiliar no combate à evasão estudantil nos cursos superiores do IFPB / Francisco Matheus Valença Trajano.– 2023.

57f. : il.

Trabalho de Conclusão de Curso (Tecnólogo em Análise e Desenvolvimento de Sistemas) - Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Cajazeiras, 2023.

Orientador(a): Profª. Me. Francisco Paulo de Freitas Neto.

1. Evasão escolar. 2. Análise de dados. 3. Análise preditiva. 4. Desenvolvimento de sistemas. I. Instituto Federal de Educação, Ciência e Tecnologia da Paraíba. II. Título.

IFPB/CZ CDU: 004.4:37(043.2)

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

FRANCISCO MATHEUS VALENÇA TRAJANO

ANÁLISE PREDITIVA E EXPLORATÓRIA DE DADOS PARA AUXILIAR NO COMBATE À EVASÃO ESTUDANTIL NOS CURSOS SUPERIORES DO IFPB

Trabalho de Conclusão de Curso apresentado junto ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba - Campus Cajazeiras, como requisito à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador

Prof. MSc. Francisco Paulo de Freitas Neto.

Aprovada em: **29 de Agosto de 2023.**

Prof. MSc. Francisco Paulo de Freitas Neto - Orientador

Prof. Dr. Fábio Gomes de Andrade - Avaliador

IFPB - Campus Cajazeiras

Prof. Dr. Leandro Luttiane da Silva Linhares - Avaliador

IFPB - Campus Cajazeiras

Documento assinado eletronicamente por:

- **Francisco Paulo de Freitas Neto**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 30/08/2023 16:36:28.
- **Leandro Luttiane da Silva Linhares**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 30/08/2023 21:45:32.
- **Fabio Gomes de Andrade**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 04/09/2023 08:22:48.

Este documento foi emitido pelo SUAP em 30/08/2023. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse https://suap.ifpb.edu.br/autenticar-documento/ e forneça os dados abaixo:

Código 468439
Verificador: cbde1dbc3a
Código de Autenticação:

Rua José Antônio da Silva, 300, Jardim Oásis, CAJAZEIRAS / PB, CEP 58.900-000
http://ifpb.edu.br - (83) 3532-4100

AGRADECIMENTOS

Agradeço a minha mãe Marlene e ao meu pai de consideração Erinelsom, por todo apoio e esforço em permitir que eu tivesse tempo em focar nos meus estudos, contribuindo para realização deste trabalho. Agradeço também ao meu pai Francisco por me ter concedido um computador mais adequado que ajudou a produzir este TCC. Emanuele, obrigado por dividir este fardo comigo e pela compreensão em minha ausência diária. Agradeço à Deus, por não ter me deixado faltar saúde e determinação. Agradeço ao professor Paulo pela orientação e ajuda no desenvolvimento do trabalho. E agradeço as minhas irmãs, Mariana e Eloisa, e a todos aqueles que contribuíram, de alguma forma, para a realização deste trabalho.

RESUMO

A evasão estudantil é um problema complexo no âmbito educacional, podendo estar ligada a diversos fatores. Identificar variáveis que inferem na ocorrência de um fenômeno e até mesmo prevê-lo é uma tarefa possível com a aplicação de *Machine Learning* em análises preditivas. O propósito deste trabalho é obter um conhecimento que auxilie no sentido preditivo, ao encontrar padrões em comum, sobre os alunos que evadem de cursos superiores no Instituto Federal de Educação, Ciência e Tecnologia da Paraíba por meio de uma análise preditiva e exploratória de dados da Plataforma Nilo Peçanha (PNP), assim facilitando a previsão do abandono dos estudos por parte dos alunos. Foram utilizados os algoritmos *Support Vector Machine*, *Decision Tree*, *K-Nearest Neighbors* e *Logistic Regression* em uma tarefa de classificação, onde o melhor modelo foi selecionado e implantado em uma aplicação *web*, bem como foram retirados *insights* importantes sobre a evasão por meio de uma análise exploratória dos dados estudantis da rede federal de educação extraídos da PNP.

Palavras-chave: Machine Learning. Evasão. Análise preditiva. Plataforma Nilo Peçanha.

ABSTRACT

Student dropout is a complex problem in the educational field and may be linked to several factors. Identifying variables that infer the occurrence of a phenomenon and even predicting it is a possible task with the application of Machine Learning in predictive analyzes. The purpose of this work is to obtain knowledge that helps in the predictive sense, by finding common patterns, about the students who drop out of higher education courses at the Federal Institute of Education, Science and Technology of Paraíba through a predictive and exploratory analysis of data from the Nilo Peçanha Platform (PNP), thus facilitating the prediction of dropout by students. The Support Vector Machine, Decision Tree, K-Nearest Neighbors and Logistic Regression algorithms were used in a classification task, where the best model was selected and deployed in a web application, as well as important insights into dropout were drawn through an exploratory analysis of student data from the federal education network extracted from the PNP.

Keywords: Machine Learning. Dropout. Predictive analysis. Nilo Peçanha Platform.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE QUADROS

LISTA DE CÓDIGOS

LISTA DE ABREVIATURAS E SIGLAS

ADS	Análise e Desenvolvimento de Sistemas
AS	Aprendizado Supervisionado
BD	Banco de Dados
DM	*Data Mining*
DW	*Data Warehouse*
ETL	*Extract, Transform and Load*
EaD	Ensino à Distância
IFPB	Instituto Federal de Educação, Ciência e Tecnologia da Paraíba
KDD	*Knowledge Discovery in Databases*
KNN	*K-Nearest Neighbors*
RL	Regressão Logística
ML	*Machine Learning*
PNP	Plataforma Nilo Peçanha
SVM	*Support Vector Machine*
SEMESP	Secretaria de Modalidades Especializadas de Educação
TCC	Trabalho de Conclusão do Curso
UNICEF	O Fundo das Nações Unidas para a Infância

SUMÁRIO

1 INTRODUÇÃO

O fenômeno em que variados tipos de dados são gerados e armazenados por uma grande quantidade de diferentes dispositivos é conhecido como *Big Data* (AMARAL, 2016). Com o avanço da tecnologia e sua necessidade de uso diário, mais sistemas com potencial para guardar e gerar mais dados surgem e são disseminados exponencialmente por todo o planeta. É um evento que pode ser facilmente entendido ao observarmos, por exemplo, a quantidade de dados de milhares de usuários que existe apenas em redes sociais como *Instagram*, *Facebook* e *Twitter*.

Devido a massiva quantidade de dados disponíveis, fez-se necessário encontrar processos automatizados para obter informações úteis a partir desses (ESCOVEDO; KOSHIYAMA, 2020). Tal necessidade foi gerada pelo acontecimento do *Big Data*, que por sua vez, desencadeou a necessidade de produção destes métodos. Entretanto, para melhor compreensão, faz-se necessário analisar o que é dado, o que é informação e como a partir disso pode-se gerar um conhecimento, pois se tratando de tecnologia, são conceitos que diferem.

"Dados são fatos coletados e normalmente armazenados. Informação é o dado analisado e com algum significado. O conhecimento é a informação interpretada, entendida e aplicada para um fim"(AMARAL, 2016, p.3). Um dado é somente um valor bruto que sozinho não é capaz de fornecer um sentido, a menos que se estude um conjunto desses e suas eventuais correlações e padrões.

Dessa forma, para se extrair conhecimento de uma base de dados, antes é preciso obter uma informação, que no que lhe concerne só é produzida através de uma tarefa de exploração dos mesmos. Para que esse processo seja possível é que existem as técnicas conhecidas como *Machine Learning* (Aprendizado de Máquina em português), que segundo (ALPAYDIN, 2020) é a aplicabilidade de algoritmos capazes de fazer um computador aprender por meio da identificação de padrões entre dados, tendo como resultado modelos capazes de prever eventos ou apenas fornecer conhecimento para uma dada circunstância.

O aprendizado por padrões também existe de maneira evidente no ser humano. Sabe-se o que é um pássaro, porque todo ele possui bico, penas, asas e a capacidade de voar, que evidencia um padrão em comum entre um pássaro e outro, permitindo o ato de classificá-lo como tal. Além disso, o cérebro humano é capaz de reconhecer padrões em linguagens, músicas, rostos de pessoas, entre outros domínios. O mesmo

ocorre em processo similar com o *Machine Learning* (ML), com a diferença de que lida-se com algoritmos que possuem um alicerce matemático, fundamentalmente estatístico, aptos ao aprendizado por meio de uma imensa quantidade de dados.

Grandes organizações recorrem ao ML para obter informações úteis a serem utilizadas na melhoria de seus serviços. É possível imaginar como seria para um usuário do serviço de *streaming Netflix* ter que encontrar um filme de seu gosto entre os milhares presentes na plataforma. Seria uma tarefa cansativa que faria com que a maioria dos usuários abandonassem o serviço. Mas essa tarefa não é necessária, pois a *Netflix* desenvolveu um sistema de recomendação fundamentado no ML, que por si só consegue recomendar filmes com base nas preferências dos assinantes (GOMEZ-URIBE; HUNT, 2015).

Todo esse processo de geração e aplicação de um conhecimento obtido através do *Machine Learning* é chamado de *Data Mining* (Mineração de Dados em português)(ESCOVEDO; KOSHIYAMA, 2020). Essa tarefa de *Data Mining* (DM) não é exclusivamente isolada, ela pode fazer parte de um processo maior conhecido como *Knowledge Discovery in Databases* (KDD) que possui o mesmo objetivo do DM (encontrar padrões para conhecer um fenômeno) (FAYYAD et al., 1996). Na seção 2.2 deste documento, esse conceito é melhor abordado.

> A analogia é que um grande volume de terra e matéria-prima mineral é extraído de uma mina, que ao ser processado leva a uma pequena quantidade de material muito precioso; Da mesma forma, na mineração de dados, um grande volume de dados é processado para construir um modelo simples com uso valioso, por exemplo, tendo alta exatidão preditiva(ALPAYDIN, 2020, p.2).

A geração de modelos preditivos faz parte de um conceito conhecido como Análise preditiva, que é um conjunto de métodos utilizados com o objetivo de prever eventos e comportamentos futuros baseados na identificação de padrões em grandes volumes de dados (ECKERSON, 2007). Um destes métodos é exatamente o *Machine Learning*. Com o ML é possível fazer um sistema trabalhar e aprender de maneira autônoma sem explicitamente ter que programá-lo e por meio deste, realizar análises preditivas.

Neste trabalho, o termo "modelo" será muito recorrente e quando faz-se menção a ele isso significa que estamos falando de uma função matemática que representa a relação entre diferentes variáveis, que é justamente o que um modelo de ML é em sua essência (PARSONS, 2021). Pode-se dizer que um modelo é uma função obtida

por meio de um algoritmo, onde tal função é a representação matemática dos padrões existentes nos dados utilizados.

Análise Exploratória de Dados (AED) também é parte essencial de um projeto de ML. A AED é um trabalho de detetive que visa investigar os dados numericamente e graficamente em busca de evidências sobre um determinado problema (TUKEY et al., 1977), ou nas palavras de (LOPES et al., 2019, p.1), "podemos descrever a AED como um conjunto de métodos adequados para a coleta, a exploração e descrição e interpretação de conjunto de dados numéricos". Dessa forma, uma análise de dados nos permite extrair informações que nos diga as raízes do problema, como ele ocorre, onde ocorre e muitos outros *insights* (percepções) sobre o problema em questão.

No sentido preditivo, estudos relacionados ao ML no ambiente educacional têm crescido pela existência de um potencial de uma contribuição efetiva no combate à evasão estudantil. Em (BIANCHI, 2017) foram utilizados dados de aprovação em disciplinas de alunos de cursos superiores da Universidade Federal de Santa Catarina que resultaram em um modelo com taxas de desempenho acima de 70%, com números de acertos animadores.

No trabalho de (PRIMÃO et al., 2022) foi proposto um modelo de ML para previsão da evasão escolar no Instituto Federal de Santa Catarina, onde tal modelo foi construído com dados estudantis relacionados a disciplinas concluídas, formas de ingresso no curso, média geral de um aluno e demais outros atributos que permitiram bons números no modelo final obtido, assim como a possibilidade do instituto identificar o perfil de alunos com potencial de evasão.

Em (VIANA et al., 2022) diferentes modelos de classificação foram propostos para identificação de Evadidos e Graduados de cursos da Universidade Federal do Piauí por meio de uma abordagem de janela temporal, onde tinha-se um modelo treinado para cada período de um curso, chegando em acurácias entre 85% a 96%.

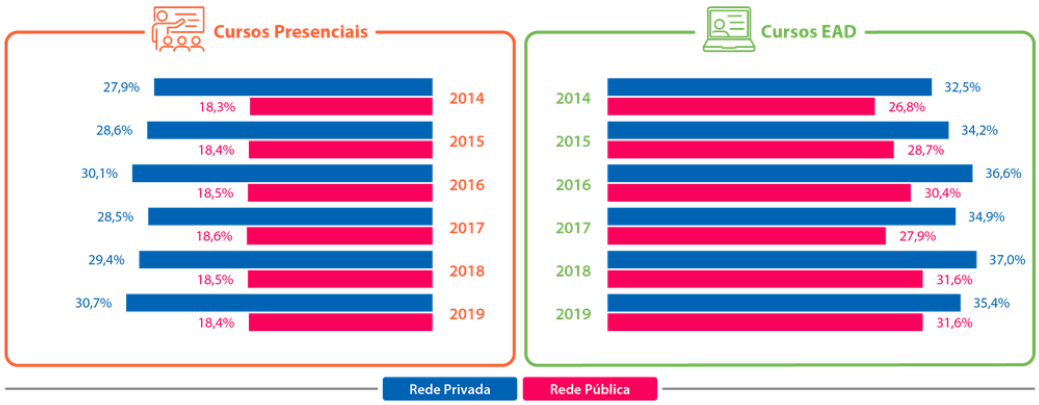
O diferencial do corrente trabalho, é a utilização de dados abertos da rede federal de educação, disponibilizados pela Plataforma Nilo Peçanha (abordada no Capítulo 2), ou seja, o trabalho visa construir uma solução para o problema da evasão estudantil por meio de dados públicos.

A evasão estudantil é um empecilho respectivo a todas as instituições de ensino que gera prejuízos sociais e econômicos, impactando diretamente em nossa sociedade. "As perdas de estudantes que iniciam, mas não terminam seus cursos são desperdícios sociais, acadêmicos e econômicos"(FILHO et al., 2007, p.642). Segundo dados do

UNICEF - O Fundo das Nações Unidas para a Infância, em uma pesquisa realizada em 2022, pelo menos dois milhões de adolescentes entre 11 e 19 anos estavam fora da escola no Brasil (UNICEF, 2022).

A Secretaria de Modalidades Especializadas de Educação (SEMESP) mostra um crescimento na taxa de evasão no ensino superior para os cursos presenciais e de Ensino à Distância (EaD) nas instituições públicas e privadas entre os anos de 2014 a 2019 (SEMESP, 2021). Os dados podem ser verificados na Figura 1.

Figura 1 – Taxa de evasão no ensino superior - Brasil



Fonte: (SEMESP, 2021)

Investimentos sem o devido retorno afetam tanto o setor público quanto o privado (FILHO et al., 2007). É um panorama que necessita de um estudo sistemático para criar medidas de retenção capazes de diminuir as taxas de evasão, tendo em vista as consequências socioeconômicas por ela provocadas.

1.1 PROBLEMATIZAÇÃO

1.1.1 Formulação do Problema

A educação é um dos pilares que sustenta a sociedade e requer cuidados para garantir sua qualidade movendo-a sempre adiante. Contudo, ainda existem índices consideráveis de evasão estudantil que afetam o sistema educacional, trazendo desperdício na aplicação de recursos pelas instituições, como também afetando no futuro de cada estudante evadido. Os evadidos de cursos superiores sofrem o risco de não estarem preparados para as demandas sociais que atualmente exigem cada vez mais qualificações, capacidade de adaptação e um posicionamento competitivo (DUARTE, 2010). A evasão no ensino superior pode limitar a capacidade de um indivíduo de lidar com as exigências da sociedade, dificultando sua inserção no mercado de trabalho.

Segundo dados da Plataforma Nilo Peçanha (PNP)[1], que será melhor abordada no Capítulo 2, os números relacionados á evasão são praticamente os mesmos desde 2018 para os *Campi* do IFPB em que há cursos superiores. Vê-se na Tabela 1 que o percentual médio de evasão e número de evadidos entre esses *Campi* é constante durante os anos de 2018 a 2022, além disso o ano de 2021 apresentou um número muito alto de evasões.

Tabela 1 – Evasão nos *Campi* com curso superior

Ano	Pecentual médio de evasão	Número de evadidos
2018	16,11%	3245
2019	17,26%	3585
2020	8,02%	2320
2021	23,83%	11116
2022	16,43%	3283

Fonte: Adaptado pelo autor de (PNP, 2022)

O único ano em que tem-se um queda no percentual de evasão é no ano de 2020, onde a modalidade de ensino passou a ser EaD (Ensino à Distância), devido à pandemia. A princípio, tal formato pode ter influenciado na diminuição da taxa de evasão no referido ano, pois existem fatores que contribuem para esse cenário, como ser mais fácil para um aluno conseguir conciliar estudo com trabalho ou para discentes que residem fora da cidade dos *Campi* em que estudam terem a possibilidade de dispensar esforços com locomoção, havendo também a abstenção de eventuais gastos com apartamentos e (ou) casas e alimentação, no caso dos que possuem residência na mesma cidade em que está localizado o *Campus*.

Com base nos dados da Tabela 1 é razoável considerar que existe a possibilidade de um movimento ascendente nas estatísticas de evasão ou mesmo de que se mantenha, uma vez que observamos que de 2018 até 2022 os números não apresentaram queda e se mantiveram regulares causando interrupção no ciclo de estudos de ainda mais alunos dos *Campi* com ensino superior, que são eles: *Campus* Sousa, *Campus* Cajazeiras, *Campus* João Pessoa, *Campus* Campina Grande, *Campus* Cabedelo, *Campus* Monteiro, *Campus* Patos, *Campus* Picuí, *Campus* Guarabira e *Campus* Princesa Isabel. A investida contra a evasão no ensino superior no IFPB, diante do exposto, torna-se um objeto de estudo importante, onde através de métodos de ML e AED é possível, com base em dados estudantis, elaborar medidas preventivas contra este problema.

[1] https://www.gov.br/mec/pt-br/pnp

1.1.2 Solução proposta

A necessidade de melhorar a eficiência acadêmica, bem como contribuir para melhoria nas tomadas de decisão e uma melhor gestão dos administradores do IFPB somadas a importância do tema motivaram o desenvolvimento deste trabalho. A Coordenação Pedagógica (COPED) e a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) podem utilizar o conhecimento extraído por este trabalho para direcionar suporte e recursos aos estudantes inseridos em grupos com padrões semelhantes aos identificados na análise preditiva e exploratória de dados realizada.

O presente estudo busca realizar uma análise preditiva e exploratória para identificar padrões entre as características dos alunos que evadem, facilitando a tomada de ações preventivas com o objetivo de evitar que a evasão ocorra em um ponto futuro com outro aluno. Tais medidas são de grande importância para a instituições, dada a possibilidade de intervir sobre alunos em situação de risco de evasão, oferecendo, por exemplo, suporte didático pedagógico, aconselhamento e demais intervenções adotadas de acordo com a necessidade de cada aluno.

Com a conclusão do trabalho espera-se contribuir tanto para as próprias instituições, proporcionando eficácia no diagnóstico de evasão e no planejamento da gestão dos estudantes, quanto para o incentivo a outros alunos a explorarem o tema, acrescentando valor à qualidade da educação, seja das instituições federais presentes no estado da Paraíba ou de outras regiões.

O fenômeno da evasão está ligado ao fracasso de uma instituição, quando esta não cumpre sua principal finalidade, garantir a permanência e a formação de seus estudantes (ARAÚJO; LIMA, 2020). Posto isto, e tudo o que foi ressaltado, neste trabalho foi realizada uma AED para descobrir padrões em comum entre os alunos evadidos de cursos superiores no IFPB, bem como foram utilizados algoritmos de *Machine Learning* visando obter um modelo capaz de prever o abandono precoce de um estudante ao fornecer informações relevantes a serem utilizadas pelos gestores dos *Campi* já mencionados. O modelo foi treinado a partir de uma base de dados construída com dados estudantis disponíveis na Plataforma Nilo Peçanha.

1.2 OBJETIVO GERAL

O presente trabalho tem como objetivo geral realizar uma análise preditiva com ML para obter um modelo capaz de auxiliar no ato de prever a evasão estudantil de cursos superiores no IFPB, bem como extrair *insights* por meio de uma AED para descobrir características do fenômeno da evasão.

1.3 OBJETIVOS ESPECÍFICOS

Os seguintes objetivos específicos foram determinados para se atingir o objetivo principal:

- Obter os microdados das matrículas dos estudantes presentes na PNP, preparálos e modelá-los;
- Compor uma base de dados (*Data Warehouse*) que servirá de fonte para os algoritmos de ML;
- Realizar uma Análise Exploratória dos dados para obter informações pertinentes sobre as características de um aluno evadido;
- Treinar os algoritmos, gerar os modelos, e escolher o melhor modelo para prever a evasão de um estudante.

1.4 METODOLOGIA

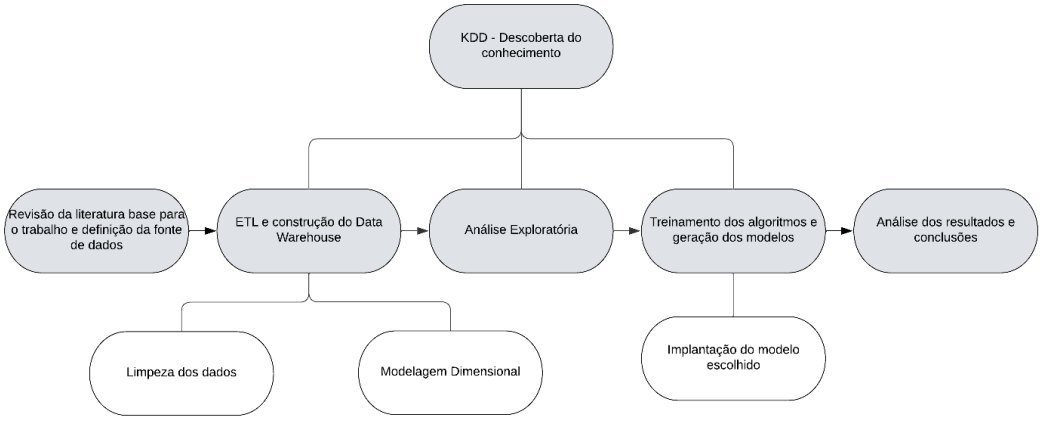
O corrente trabalho realiza uma Mineração de Dados sobre a Plataforma Nilo Peçanha em uma tarefa de descoberta de conhecimento. Como tarefa inicial, foi realizada uma revisão em artigos, livros e sites sobre *Data Mining*, *Machine Learning*, AED e demais conceitos que sustentam este TCC e que nele foram citados, bem como a definição da fonte de dados a ser utilizada para o processo de mineração.

Na próxima etapa foi feita a aplicação do processo de *Knowledge Discovery in Databases* (KDD) em conjunto com o *Extract Transform and Load* (ETL) para construção de um *Data Warehouse* (DW) e geração de conhecimento com base nos dados obtidos da PNP. Essa etapa é uma das mais essenciais para o trabalho, contando com tarefas como Modelagem Dimensional e pré-processamento de dados, deixando-os devidamente preparados para serem utilizados pelos algoritmos de ML, além da Análise Exploratória de Dados.

Em seguida, veio a etapa geração dos modelos com a aplicação de algoritmos de classificação e a seleção do melhor modelo, para ter-se informações consideráveis a serem utilizadas pelo IFPB, que poderá direcionar estratégias de combate à evasão,assim finalizando o processo KDD. Nesta etapa, também temos o *deploy* (implantação) do modelo final para ser utilizado pelos usuários através de um navegador web.

Por fim, foi feita a análise dos resultados obtidos no intuito de avaliar sua eficácia e a elaboração das conclusões sobre o trabalho realizado. A Figura 2 exibe o fluxo de tais atividades.

Figura 2 – Representação gráfica do ciclo das atividades



Fonte: Elaborado pelo autor

1.5 ORGANIZAÇÃO DO TRABALHO

O presente trabalho está organizado em seis capítulos. O Capítulo 2 se dá pelo referencial teórico, onde está uma discussão sobre a literatura existente que forma a base desse Trabalho de Conclusão de Curso (TCC), tais como as tarefas de classificação e seus algoritmos. No Capítulo 3 encontra-se a construção de um *Data Warehouse*, sendo este a fonte de dados que foi utilizada pelos algoritmos de ML. Em seguida, Capítulo 4, encontra-se a Análise Exploratória de Dados para extração de informações sobre a evasão, seguido da Análise preditiva, Capítulo 5, com a aplicação dos algoritmos de ML. Por último, no Capítulo 6, são apresentados os resultados obtidos e as considerações finais sobre o trabalho, bem como a sugestão de trabalhos futuros.

2 REFERENCIAL TEÓRICO

Esta seção apresenta os conceitos em que se fundamenta o trabalho. Inicialmente foi comentado sobre os microdados da PNP, seguido de concepções como KDD, *Data Warehouse*, Aprendizado Supervisionado,algoritmos de classificação e métricas de avaliação.

2.1 DADOS ABERTOS DA PLATAFORMA NILO PEÇANHA

A Plataforma Nilo Peçanha foi instituída com o objetivo de disseminar dados estatísticos de toda a Rede Federal de Educação. Ela surgiu em 2018 pela Secretaria de Educação Profissional e Tecnológica do Ministério da Educação (SETEC/MEC), tendo em vista a necessidade de se melhorar a gestão pública com base em indicadores de desempenho (MORAES et al., 2020). A plataforma é *online* e de livre acesso, possui um sistema de *Business Intelligence* (BI) que serve para prover suas informações, contando com painéis interativos que fornecem diversas maneiras de visualizar os dados.

> No tocante às preocupações mais estritamente pedagógicas, as produções estatísticas podem auxiliar as instituições que compõem a Rede na tarefa de analisar seus processos escolares, construindo conhecimento, por exemplo, a respeito da qualidade educacional dos cursos e de seus graus de inclusão social. Pode, ainda, de maneira objetiva, mensurar as taxas de evasão escolar, variável historicamente crítica na Rede Federal. Além disso, os levantamentos estatísticos nos permitem avaliar se os objetivos e as finalidades, legalmente previstos para a Rede Federal, estão sendo cumpridos (MORAES et al., 2020, p.7).

É possível acessar um conjunto de microdados da PNP através do Portal de Dados Abertos do Ministério da Educação[1]. Esses microdados possuem indicadores sobre Eficiência Acadêmica, Matrículas, Servidores e Financeiro, sendo separados por ano, onde podem ser baixados em arquivos no formato *Comma-Separated Values* (CSV), que é um formato em que os dados são salvos em forma de tabela.

Para este trabalho, foram utilizados os microdados de matrículas dos anos de 2018 a 2021, que contém referências como unidade de ensino, renda, faixa etária, raça e ciclo de matrícula dos alunos, que nesse caso, foram utilizados unicamente microdados relativos aos alunos de cursos superiores do IFPB.

[1] <http://dadosabertos.mec.gov.br/pnp>

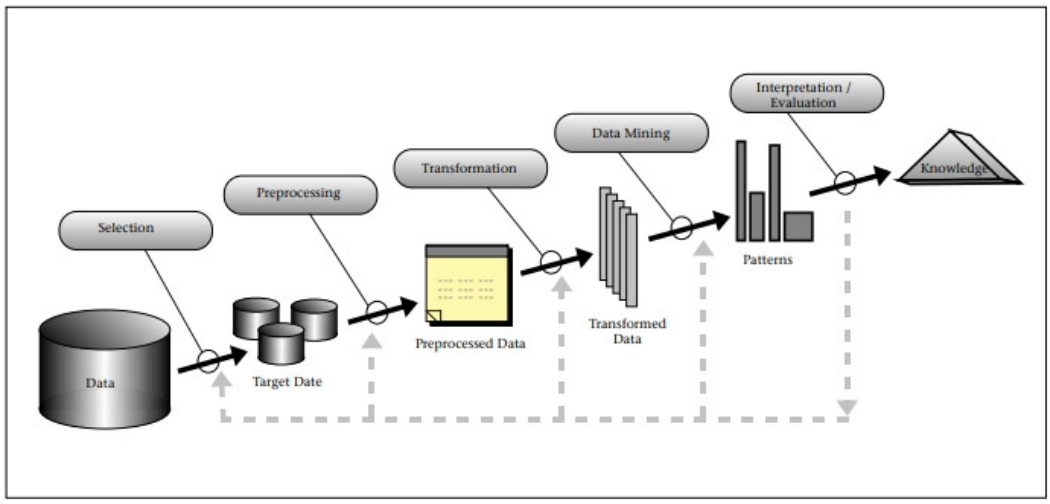
2.2 *KNOWLEDGE DISCOVERY IN DATABASES* E *DATA MINING*

Knowledge Discovery in Databases (KDD), ou Descoberta de Conhecimento em Banco de dados, é um processo que permite extrair conhecimentos úteis a partir de Bancos de dados, onde uma de suas etapas consiste na mineração dos dados (*Data Mining*), sendo ela o ponto central do KDD (FAYYAD et al., 1996).

Como já abordado, *Data Mining* proporciona a obtenção de um conhecimento aplicável através de uma "análise de grandes conjuntos de dados, a fim de descobrir padrões e regras significativos"(PUJARI, 2001, p.1). Sendo assim, ela nada mais é que parte de um processo maior, que é o KDD, proporcionando a extração de conhecimento prezada por ele.

Na Figura 3 são exibidas todas as etapas do KDD. Inicialmente é realizado um entendimento dos dados e do que deverá ser feito, formulado um objetivo para então ser feita a parte de Seleção (*Selection*), que seria criar um conjunto de dados a partir daqueles que estão disponíveis, isto é, definir que parte dos dados servirá de base para aplicação do KDD. Feito isso, adentramos à etapa de Pré-processamento (*Pre-processing*), composta por tarefas de substituição ou remoção de valores faltantes ou repetidos e demais valores que apresentem problemas ao serem lidos pelos algoritmos de ML.

Figura 3 – Processo KDD



Fonte: (FAYYAD et al., 1996)

Em sequência, vem a etapa de Transformação dos dados (*Transformation*), onde ocorre uma redução de dimensionalidade, que em outros termos, significa reduzir o número de variáveis a serem utilizadas, podendo contar também com a conversão de seus valores. Após isso, é escolhido o tipo de tarefa a ser utilizada, a exemplo da tarefa de classificação escolhida para este trabalho.

Logo após, são definidos quais algoritmos serão utilizados, baseando-se na tarefa previamente escolhida. Até que então chega-se na etapa de *Data Mining*, e é neste estágio que se encontra a aplicação dos algoritmos de ML para extração de padrões, ou seja, tem-se a mineração dos dados em busca de um conhecimento essencial.

Por fim, é feita uma interpretação das informações obtidas, uma avaliação dos resultados dos modelos para então ter-se um conhecimento que poderá ser aplicado em ações futuras, assim finalizando o processo de KDD. Importante destacar que no caso de obtenção de maus resultados pelos modelos treinados tem-se o retorno para etapas anteriores a fim de obter melhores informações. Todas essas etapas que consistem o KDD foram definidas em (FAYYAD et al., 1996).

2.3 *DATA WAREHOUSE*

Um *Data Warehouse* nada mais é que um Banco de Dados (BD) que armazena informações integralizadas advindas de diferentes fontes com o intuito de auxiliar na análise de dados e tomada de decisões (HÜSEMANN et al., 2000).

A partir de diversos BD é feito um processo para unificar os seus dados em uma única fonte a fim de se extrair informações relevantes. Vale ressaltar que não necessariamente um DW é constituído de múltiplas fontes, considerando que seu objetivo principal pode ser realizado com uma única, facilitar processos de análise de dados, estando comumente relacionado com *Data Mining* e *Machine Learning*.

Para que seja possível a construção de um DW antes utiliza-se uma técnica conhecida como Modelagem Dimensional (CHAUDHURI; DAYAL, 1997). Nesse tipo de Modelagem, o objeto a ser analisado, isto é, aquilo que se quer conhecer passa por um processo onde ele é dimensionado para ser armazenado dentro do DW. Tal objeto é conhecido como fato, podendo este possuir diversas dimensões.

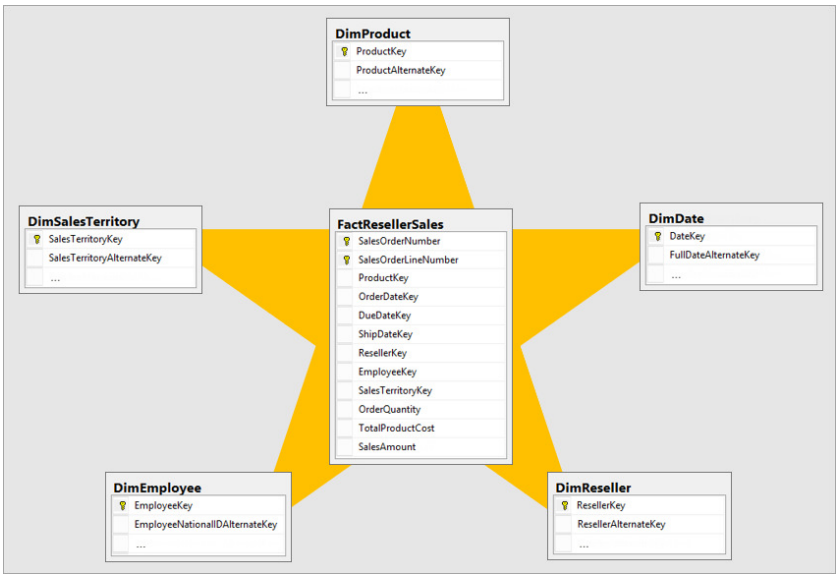
> O fato é a informação central, o tema ao qual se quer analisar. Um fato possui medidas que são valores a serem analisados e pré-calculados. Um fato também possui dimensões que são os diversos pontos de vista sobre o qual se quer analisar o fato (AMARAL, 2016, p.42).

Então, para se conhecer uma determinada situação é preciso esclarecer detalhes sobre ela, onde cada detalhe se torna uma dimensão do fato ou situação. Quando é aplicada a Modelagem Dimensional tem-se um esquema conhecido como *Star Schema*, ou Modelo Estrela (KIMBALL; ROSS, 2011).

O *Star Schema* tem esse nome porque as tabelas são estruturadas de modo que lembrem uma estrela. Esse tipo de Modelagem se desvia dos problemas de uma grande quantidade de tabelas com relacionamentos entre si do Modelo Relacional comum.

No exemplo da Figura 4 a tabela fato está no centro da estrela, sendo ela referente a vendas. As demais tabelas ao redor são as dimensões, ou seja, os detalhes sobre uma venda. É possível observar dimensões como *DimEmployee* (que seria um empregado que realizou uma venda), *DimDate* (a data em que foi realizada) e *DimProduct* (o produto que foi vendido).

Figura 4 – Modelagem Dimensional - *Star Schema*

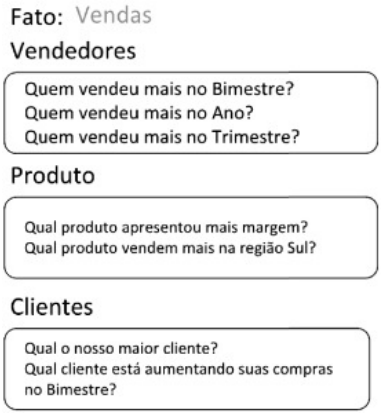


Fonte: (KFOLLIS, 2021)

Observe também como as chaves primárias (chave identificadora) das tabelas de dimensão são passadas para tabela fato, tendo assim uma ligação entre o fato e suas dimensões.

Todas as dimensões podem ser facilmente encontradas ao realizar questionamentos sobre o fato, como pode ser encontrado em (AMARAL, 2016). Perceba como na Figura 5 algumas perguntas sobre uma fato Vendas concebeu dimensões como:Vendedor, Produto e Cliente.

Figura 5 – Modelagem Dimensional - Fatos e Dimensões



Fonte: (AMARAL, 2016)

2.3.1 *Extract, Transform and Load*

O *Extract, Transform and Load* (ETL), ou Extrair, Transformar e Carregar, em tradução livre, é um processo em que é realizado a extração de dados de suas respectivas fontes, transformação dos mesmos ao executar um processo de limpeza de dados para então carregá-los no *Data Warehouse* (SIMITSIS, 2004).

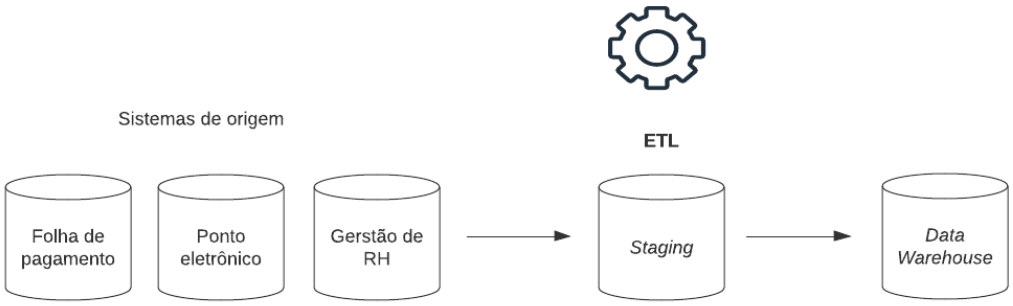
O processo de ETL é uma das etapas mais importantes na construção de um DW, pois os dados são devidamente preparados para serem utilizados, evitando que dados que não fazem sentido ou que apresentem algum problema sejam carregados ao DW final. Além disso, como um DW pode ser alimentado por diferentes fontes, naturalmente irá se encontrar dados em formatos diferentes que irão precisar passar por uma conversão (que se inclui na etapa de transformação) para enfim serem reunidos no DW sem demais problemas.

A primeira etapa (*Extract*), como o próprio nome sugere, é extrair os dados de sua fonte original. Um vez extraídos eles irão passar por "várias transformações em potencial, como limpeza dos dados (corrigir erros ortográficos, resolver conflitos de domínio, lidar com elementos ausentes ou analisar em formatos padrão)[...]"(KIMBALL; ROSS, 2011, p.19), todo esse processo seria a segunda etapa (*Transform*). É nessa etapa que também realiza-se o processo de Modelagem Dimensional.

A etapa final (*Load*) seria carregar esses dados para o DW, uma vez que eles já foram devidamente tratados na etapa anterior. Todo o processo ETL é normalmente realizado em uma área chamada de *Staging* (AMARAL, 2016), que tem por objetivo fornecer um ambiente que irá intermediar o armazenamento e o processamento dos dados (MACHADO, 2004).

A Figura 6 exemplifica todo o fluxo da construção do DW. Os dados são extraídos de suas fontes de origem para a área de *Staging*, onde são transformados e depois carregados para o DW finalizando sua produção. Dessa forma, tem-se um ambiente onde as informações estão centralizadas, com dados estruturados de uma forma compreensível, assim facilitando a análise e consulta sobre os dados, bem como a identificação de informações relevantes para darem suporte à tomada de decisões.

Figura 6 – Processo de ETL



Fonte: Adaptado pelo autor de (AMARAL, 2016)

2.4 APRENDIZADO SUPERVISIONADO

No *Machine Learning* os algoritmos podem ter o aprendizado classificado de maneiras diferentes. Uma delas é conhecida como Aprendizado Supervisionado, sendo esta uma maneira de treinar um algoritmo através de um conjunto de dados de entrada classificados para uma saída desejada, onde o algoritmo irá encontrar padrões existentes entre os dados de entrada e a saída esperada que, posteriormente, o tornará capaz de classificar com exatidão um valor ainda não conhecido pelo mesmo (PETERSSON, 2021).

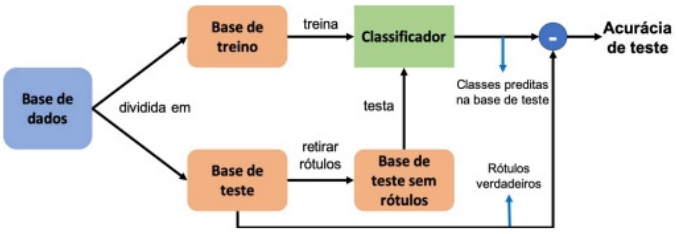
Nesse tipo de aprendizado, a pessoa que está fornecendo os dados ao algoritmo seria o supervisor, aquele que está dizendo como ele irá aprender. No Aprendizado Supervisionado (AS) já se conhece previamente o que se quer descobrir, apenas é preciso treinar o algoritmo com dados corretos para que ele, por si só, realize essa descoberta.

2.4.1 Tarefa de classificação

Uma tarefa naturalmente realizada no AS é a de classificação. Essa tarefa carrega um conceito central chamado *classe*, que pode ser entendida como um grupo ou conjunto da qual alguma entidade irá ou não pertencer a depender de suas características (*features*).

Partindo de um conjunto de dados subdivido em dois grupos, um grupo de treino contendo atributos que servem de noção para que o algoritmo saiba como se configura cada *classe* (que representa 70% do conjunto e contribuirá para a geração de um modelo) e um grupo de teste contendo valores de exemplo das *classes* que queremos prever (que representa 30% do conjunto), o algoritmo irá então tentar definir a que *classe* pertence os dados existentes no conjunto de teste com base no que identificou nos dados utilizados no grupo de treino, onde a partir daí pode-se medir a capacidade preditiva deste modelo (acurácia e demais métricas) com base no quanto ele classificou corretamente esses dados (ESCOVEDO; KOSHIYAMA, 2020). Porém, vale salientar, que essa divisão entre 70% para dados de treino e 30% para dados de teste é arbitrária, ficando a critério do supervisor o percentual de divisão a ser utilizado. A Figura 7 apresenta um exemplo do fluxo de uma tarefa de classificação.

Figura 7 – Fluxo de uma tarefa de Classificação



Fonte: (ESCOVEDO; KOSHIYAMA, 2020)

Uma vez que o modelo é gerado, pode-se submeter para ele novos dados além dos dados de teste, ou seja, dados que não estavam presentes em todo o conjunto de dados original. Com isso, o modelo classificador irá tentar descobrir a que *classe* pertence essa nova amostra. Se as suas métricas de classificação são ruins, significa que este modelo precisa de mais treinamento, como, por exemplo, adicionar uma base de dados maior, encontrar parâmetros diferentes, balancear as *classes* nos dados, reduzir a dimensionalidade dos dados e tentar diversos outros métodos visando aumentar sua precisão.

A capacidade de fazer previsões bem sucedidas com dados não observados a partir dos dados observados (dados de treino) é chamada de generalização, ou seja, reunir numa classe geral, termo ou proposição, um conjunto de seres ou fenômenos similares.

2.5 ALGORITMOS DE CLASSIFICAÇÃO

Como já evidenciado, o Aprendizado Supervisionado traz com si diversos algoritmos capazes de realizar tarefas de classificação em conjuntos de dados. Existem

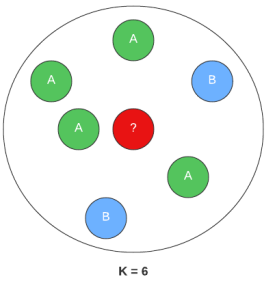
diversos algoritmos para este propósito, e nesta seção serão abordados os algoritmos utilizados no trabalho.

2.5.1 *K-Nearest Neighbors*

Um deles é o *K-Nearest Neighbors* (KNN), que significa K-Vizinhos mais Próximos. O objetivo desse algoritmo é encontrar qual *classe*, dado um conjunto de teste, se aproxima mais (é mais parecida ou mais próxima) de uma *classe* já conhecida por ele pelo conjunto de treino. Para saber qual amostra de dados se parece mais com outra o KNN se utiliza de fórmulas matemáticas para encontrar distâncias, tais como a Distância Euclidiana, Distância Manhattan e a Distância Minkowski (ESCOVEDO; KOSHIYAMA, 2020).

A Figura 8 mostra um exemplo de como funciona o KNN. O ponto em vermelho seria a amostra de dados a ser classificada, seja como uma *classe* A ou B. O valor de K é um número informado pelo supervisor que será utilizado pelo algoritmo para considerar apenas as K-classes mais próximas da amostra alvo (círculo vermelho). O algoritmo irá então realizar um cálculo de distância entre a amostra alvo e todas as outras *classes* até que se escolha apenas as seis mais próximas do alvo, já que esse foi o valor de K informado e, como existe um maior número da *classe* A ali presente, o algoritmo classifica essa amostra, até então desconhecida, como sendo pertencente à A (JOSÉ, 2018).

Figura 8 – Exemplo do KNN



Fonte: Adaptado pelo autor de (JOSÉ, 2018)

Para encontrar o grupo, presente na Figura 8, com as seis *classes* mais próximas da *classe* desconhecida, o KNN calcula a distância para todas as amostras até encontrar K-amostras que mais se aproximam da *classe* alvo fornecida, e o tipo de *classe* mais presente nas K-amostras encontradas será atribuído aos dados que inserimos no modelo. A Distância Euclidiana é a medida mais comum utilizada pelo KNN, fórmula observada na Figura 9. Soma-se o quadrado das diferenças entre os pares de elementos *ij* partindo de um até o número de dimensões da amostra (valor m)

e tira-se a raiz quadrada desse somatório, assim obtendo-se a distância entre estes pares *ij*.

Figura 9 – Fórmula da Distância Euclidiana

$$Dist_{ij} = \sqrt{\sum_{k=1}^{m}(x_{ik} - x_{jk})^2}$$

Fonte: Adaptado pelo autor de (MULAK; TALHAR, 2015)

É um algoritmo de fácil compreensão e muito utilizado para tarefas que exigem a classificação apenas entre duas *classes* distintas, ou seja, quando as amostras de dados pertencem ou não a uma determinada *classe*, e também apresenta praticidade na criação de sistemas de recomendação devido a sua capacidade de encontrar amostras parecidas.

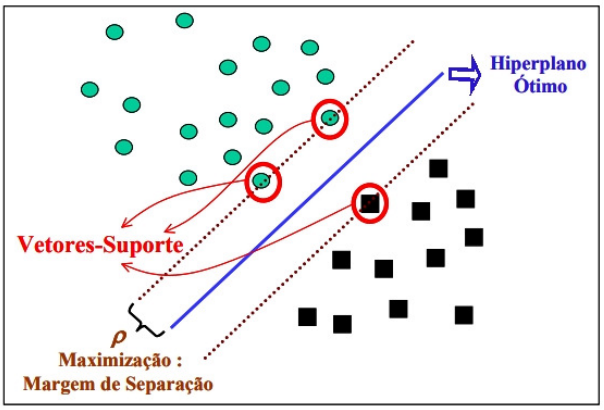
2.5.2 *Support Vector Machine*

O *Support Vector Machine* (SVM), que em português significa Máquina de Vetores de Suporte, é um algoritmo capaz de "[...] trabalhar com cenários lineares e não lineares, permitindo alto desempenho em diversos contextos."(BONACCORSO, 2017, p.133). Ou seja, é possível construir um modelo a partir de dados aglomerados (não lineares) onde, inicialmente, não é possível definir uma separabilidade para os mesmos. No SVM os Vetores de Suporte são como pontos de referência para a construção de um hiperplano, sendo a base de funcionamento desse algoritmo. A depender da linearidade dos dados o SVM irá criar dimensões diferentes.

> Essencialmente, o SVM realiza um mapeamento não linear para transformar os dados de treino originais em uma dimensão maior. Nesta nova dimensão, o algoritmo busca pelo hiperplano que separa os dados linearmente de forma ótima. Com um mapeamento apropriado para uma dimensão suficientemente alta, dados de duas classes podem ser sempre separados por um hiperplano. (ESCOVEDO; KOSHIYAMA, 2020, p.127).

Se você tem duas *classes*, o hiperplano ótimo seria a linha que melhor as divide, tomando como referência os Vetores de Suporte (que são os pontos com a menor distância do hiperplano) que formam a margem do classificador. A Figura 10 mostra como o hiperplano (linha azul) é traçado com base nos pontos limite de cada grupo próximos a ele.

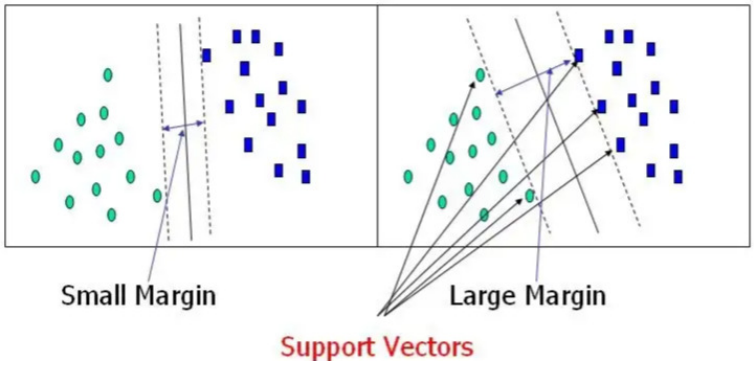
Figura 10 – Exemplo SVM



Fonte: (SEMOLINI et al., 2002)

Na Figura 11 observa-se um exemplo de maximização da margem, onde no lado esquerdo (*Small Margin*) o hiperplano é traçado de uma maneira imprecisa, mas ao maximizar a margem (*Large Margin*) o melhor hiperplano possível é encontrado, garantindo um maior grau de separabilidade.

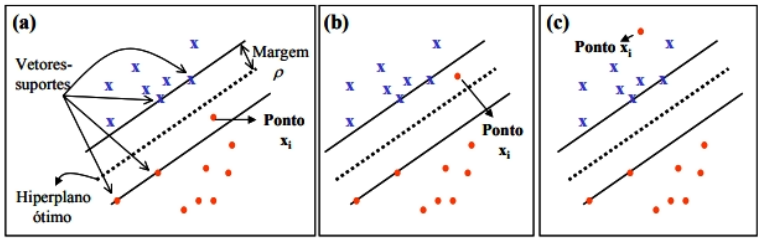
Figura 11 – Exemplo SVM - Margem maximizada



Fonte: (GANDHI, 2018)

Quando se tem dados não lineares, ou seja, em que é impossível se traçar um hiperplano ótimo, surge um conceito que se chama margem flexível (*soft margin*) que permite que pontos ultrapassem a margem, já que não é concebível encontrar um hiperplano sem erros (SEMOLINI et al., 2002). A Figura 12 mostra que pontos dentro da margem foram permitidos para garantir a construção do hiperplano.

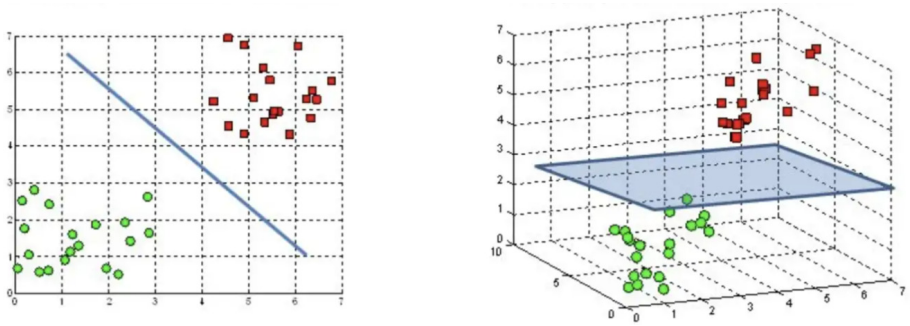
Figura 12 – Exemplo SVM - Margem *soft*



Fonte: (SEMOLINI et al., 2002)

"Ao trabalhar com problemas não lineares, é útil transformar os vetores originais [dados originais] projetando-os em um espaço dimensional superior onde possam ser separados linearmente."(BONACCORSO, 2017, p.141). Em outras palavras, se torna mais fácil de separar dados não lineares quando estes estão em uma dimensão maior. Se tem-se dados lineares o hiperplano é meramente uma linha, mas se eles não são lineares e não permitem a construção de um limiar correto o hiperplano tem sua dimensão aumentada juntamento com os dados. A Figura 13 demonstra esse conceito da diferença dimensional nos dados e no hiperplano.

Figura 13 – Exemplo SVM - Dimensão do hiperplano



Fonte: (GANDHI, 2018)

O SVM é um poderoso algoritmo de ML para tarefas de classificação, sendo bem flexível e otimizado, traz consigo diversas maneiras de lidar com os dados de treinamento e oferece diferentes formas de adaptação a depender da dimensionalidade dos dados.

2.5.3 Árvore de Decisão

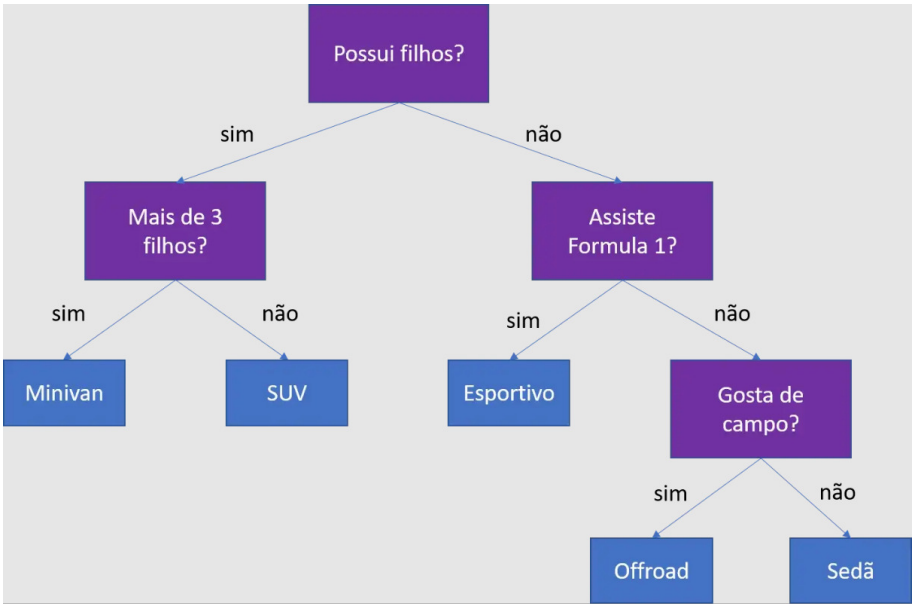
Uma Árvore de Decisão (*Decision Tree*) é um algoritmo que representa uma estrutura de dados com hierarquia formada a partir de uma base de dados rotulada com regras que definem tal estrutura (ALPAYDIN, 2020).

A Árvore possui nós raiz, ramo e folha, onde o raiz se encontra no topo da estrutura. Onde cada ramo da Árvore se inicia é onde está o nó de decisão que irá influenciar na ação do algoritmo dizendo-o para qual nó deve seguir. Os nós folha são os últimos da estrutura e é por eles que as *classes* são atribuídas.

> Dada uma entrada, em cada nó, um teste é aplicado e um dos ramos é tomado dependendo do resultado. Este processo começa na raiz e é repetido recursivamente até que um nó folha seja atingido, ponto em que o valor escrito na folha constitui a saída (ALPAYDIN, 2020, p.213).

Em um exemplo retirado de (ARAUJO, 2020), presente na Figura 14, observa-se como funciona a estrutura de uma Árvore de Decisão. Neste exemplo, a Árvore auxilia na decisão de que tipo de carro comprar. Todos os quadros em roxo abaixo do nó raiz (topo da árvore) são nós de decisão, que com base em uma condição irá desviar o fluxo do algoritmo. Os quadros em azul são os nós folha que contém o resultado final, isto é, que carro deve ser comprado.

Figura 14 – Exemplo de uma Árvore de Decisão



Fonte: (ARAUJO, 2020)

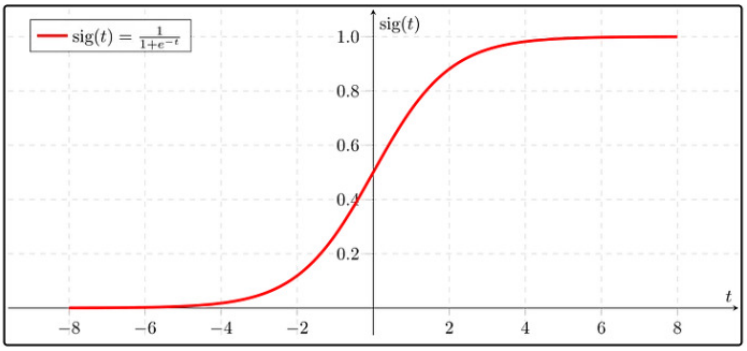
É um algoritmo de fácil entendimento em que o fluxo do processo pode ser facilmente visualizado. Como nas palavras de (BONACCORSO, 2017, p.115) "Uma árvore de decisão é como ir a um médico que faz uma série de perguntas a fim de determinar a causa de seus sintomas.". E a partir desses, pode ser feita uma escolha, uma medida que melhor se adequa à situação.

2.5.4 Regressão Logística

Um modelo de Regressão Logística (*Logistic Regression*) é muito utilizado para tarefas de classificação binárias."Quando a resposta é binária, normalmente assume a forma de 1/0, com 1 geralmente indicando um sucesso e 0 uma falha"(HILBE, 2011, p.1). Por exemplo, para este trabalho tem-se duas classes: evadido ou concluinte. Logo, temos uma classificação binária onde podemos dizer que concluinte seria zero (0) e evadido seria um (1). Por convenção, definimos a *classe* alvo como positiva (número 1).

A Regressão Logística (RL) se parece com um outro modelo conhecido como Regressão Linear que é utilizado para prever um valor numérico aleatório (pode ser qualquer valor), porém na RL os resultados são convertidos em probabilidades por meio de uma função Sigmoide que pode ser observada na Figura 15. Essa função tem como característica o formato de "S" e sua saída é sempre um valor entre 0 e 1.

Figura 15 – Regressão Logística - Função Sigmoide



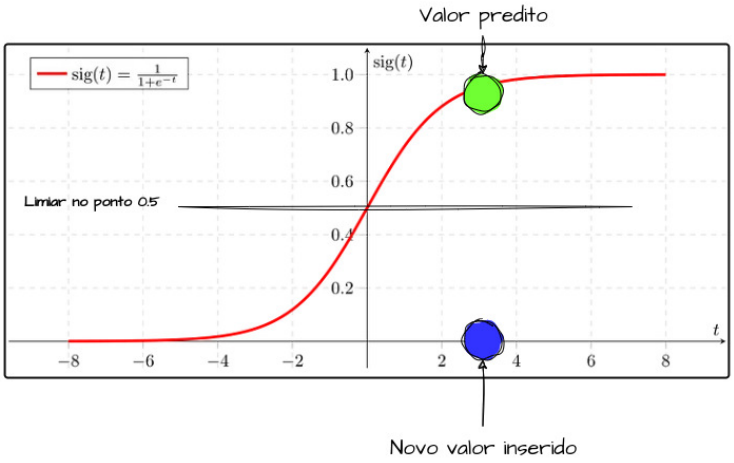
Fonte: (SWAMINATHAN, 2018)

Como o eixo y da função é convertido em uma probabilidade o modelo de RL consegue prever a probabilidade de novos valores pertencerem à classe 0 ou 1 por meio de um limiar que por padrão é 0.5 (que seria 50%).

Então digamos que treinamos um modelo de Regressão Logística com a base de dados da PNP e agora queremos passar um novo dado sobre um novo estudante. Como resultado nós iríamos obter uma classe 0 ou 1, ou seja, se o novo dado estiver acima de 0.5 de chances de pertencer à classe 1 (evadido), então a saída será 1, do contrário (se estiver abaixo de 0.5), a saída será 0 (concluinte).

No exemplo da Figura 16 tem-se uma demonstração de um saída para a classe 1, presumindo uma probabilidade acima de 0.5 encontrada para o novo valor.

Figura 16 – Regressão Logística - Exemplo de saída



Fonte: Adaptado pelo autor de (SWAMINATHAN, 2018)

2.6 MÉTRICAS DE AVALIAÇÃO

A Acurácia (*Accuracy*) é uma métrica calculada com base nas previsões corretas do modelo sobre o total de previsões que ele realizou (HOSSIN; SULAIMAN, 2015), ou seja, ela nos fornece uma visão geral de desempenho. Se de 100 previsões o modelo estava correto em 91 delas, então tem-se uma Acurácia de 91%. Já a métrica de Precisão (*Precision*) fornece um valor que mensura o quanto das classificações positivas que um modelo fez estavam corretas. Se tem-se uma Precisão de 50%, significa que o modelo estava correto em 50% das previsões positivas que efetuou.

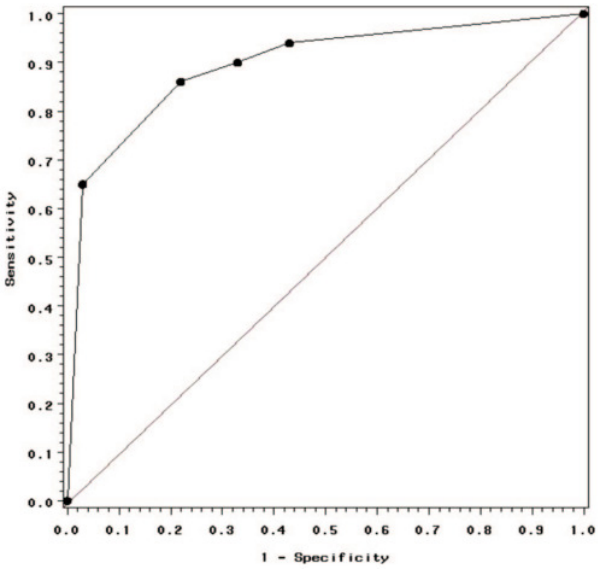
A curva *Receiver Operating Characteristic* (Características operacionais do receptor, ROC) define o quanto um modelo é capaz de distinguir duas *classes*, positiva ou negativa (0 ou 1) (RODRIGUES, 2018). Essa curva é construída com base nas Taxa de Verdadeiro Positivo (TVP), que seriam as amostras positivas classificadas corretamente, e na Taxa de Falso Positivo (TFP), que seriam as amostras negativas classificadas incorretamente como positivas, ambas obtidas para diferentes limiares.

Um limiar é o ponto limite que define a saída do modelo, esse conceito foi demonstrado no Capítulo 2 no algoritmo de Regressão Logística. Dado um valor de entrada, o modelo irá encontrar uma probabilidade desse novo valor pertencer a uma determinada *classe*, se esse valor for acima do limiar definido, a saída é 1 (positiva), se for abaixo, a saída é 0 (negativa), e a partir desses resultados a TVP e a TFP são calculadas.

A curva ROC é construída com base em diferentes limiares para o modelo. Para cada limiar testado a TVP e a TFP são medidas e um ponto é inserido no gráfico.

Na Figura 17 observamos um exemplo de uma curva ROC.

Figura 17 – Curva ROC

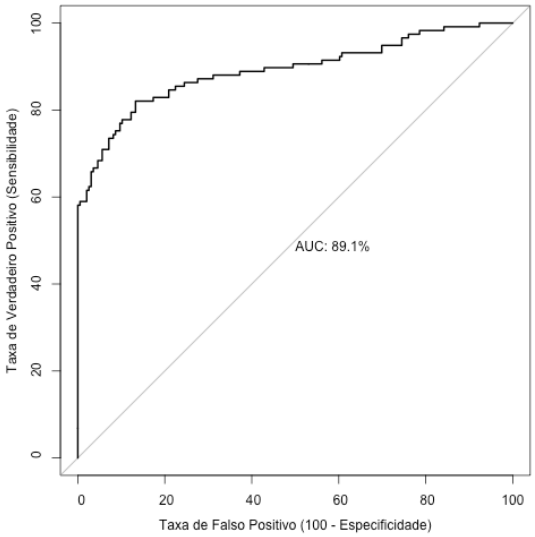


Fonte: (MANDREKAR, 2010)

As coordenadas (*x,y*) de cada ponto são definidas de acordo com os resultados obtidos para cada limiar. No eixo *x* tem-se a especificidade (*Specificity*) do modelo que está relacionada com a TFP, então, presume-se, que quanto mais para a direita um ponto está no eixo *x*, mais amostras negativas foram classificadas como positivas. Já no eixo *y* tem-se a sensibilidade do modelo (*Sensibility*) que está relacionada com a TVP, logo, quanto mais para cima um ponto está nesse eixo, mais amostras positivas foram classificadas como positivas. Na literatura, a sensibilidade também é conhecida como *Recall* (rechamada ou cobertura), sendo ela um valor de probabilidade de saída positiva dada uma entrada também positiva.

Como a curva ROC representa uma distribuição de diversos resultados obtidos, tornando mais difícil sua interpretação, surgiu-se a necessidade de resumir estes resultados em uma única métrica para melhor compreensão do grau de separabilidade do modelo. A *Area under the ROC curve* (Área sob a curva ROC, AUC) é simplesmente uma maneira de representar todos os dados da curva ROC em um único valor obtido da área sob a própria curva (AVELAR, 2019).

Figura 18 – AUC - Área sob a curva ROC



Fonte: (PRATES, 2020)

Na Figura 18 constatamos um exemplo do valor AUC, cujo varia entre 0 e 1, ou poderíamos dizer, que varia entre 0% e 100%. Quanto maior o AUC, maior a capacidade do modelo de distinguir duas *classes*. A partir dessa métrica é possível comparar diferentes modelos diretamente. Se um modelo A tem um AUC maior do que um modelo B, então certamente o primeiro é capaz de distinguir de uma maneira mais precisa as classes positivas das negativas, sendo mais habilidoso.

Portanto, visando esse grau de separabilidade e o aumento na TVP, as métricas primárias de avaliação durante o treinamento dos algoritmos foram a área sob a curva ROC, dado que estaríamos aumentado as chances do modelo de separar aluno evadido de aluno concluinte corretamente, e a cobertura, para medir a capacidade de identificação de amostras positivas.

3 PRODUÇÃO DO *DATA WAREHOUSE*

Seguindo-se as orientações do KDD percebe-se que suas etapas iniciais, tais como *Selection, Preprocessing e Transformation* equivalem a todo o processo realizado no ETL. Sendo assim, ao aplicarmos o ETL na construção do *Data Warehouse* nós já estamos completando parte das etapas presentes no KDD.

3.1 TECNOLOGIAS UTILIZADAS

Abaixo foram descritas as tecnologias utilizadas na construção do DW:

PostgreSQL[1]**:** é um banco de dados relacional que faz parte de um projeto de código aberto, apresenta robustez e grande desempenho, sendo muito adotado por grandes empresas.

Pandas[2]***:*** trata-se de uma biblioteca presente na linguagem *Python* muito utilizada para manipular tabelas, a exemplo dos arquivos CSV.

Docker[3]***:*** é um software utilizado para virtualizar o uso de aplicações permitindo executá-las em diferentes sistemas.

SQLAlchemy[4]***:*** é um *framework Python* usado para aumentar a produtividade em tarefas de acesso a banco de dados relacionais.

Jupyter Notebook[5]***:*** é uma plataforma que permite o uso de algoritmos, textos e gráficos através de um navegador *web*, sendo muito comum em análise de dados com *Python*.

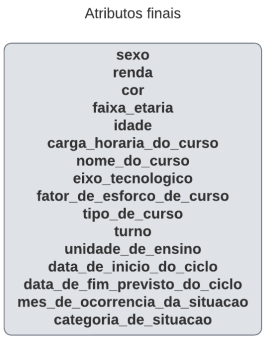
3.2 EXTRAÇÃO E TRANSFORMAÇÃO DOS DADOS

Inicialmente foi realizado o *download* dos arquivos que representam os micro-dados de matrículas no formato CSV existentes na PNP, onde foram posteriormente manipulados pela biblioteca *pandas* executada no *Jupyter Notebook*. Através dessa biblioteca foi possível renomear colunas das tabelas para melhor compreensão e compatibilidade com o BD *PostgreSQL*, como também foram removidas colunas que explicitamente não tinham valores relevantes.

[1] <https://www.postgresql.org/>
[2] <https://pandas.pydata.org/>
[3] <https://www.docker.com/>
[4] <https://www.sqlalchemy.org/>
[5] <https://jupyter.org/>

Os arquivos CSV baixados foram referentes aos anos base e 2018 a 2021. Foram selecionados apenas alunos evadidos e concluintes em seus referidos cursos através de uma filtragem pelo campo **categoria _de_situacao** exibido na Figura 19 que armazenava valores que indicavam o estado da matrícula de um aluno, sendo selecionado apenas os valores "Concluinte" e "Evadido". Alunos que tinham uma situação de "Em curso" foram descartados devido a impossibilidade de inferência de evasão ou conclusão dos mesmos.

Figura 19 – Conjunto de dados finalizado



Fonte: Elaborado pelo autor

Foi um largo processo até o conjunto final observado na Figura 19. Também foram selecionados apenas alunos de cursos superiores pelo campo **tipo_de_curso** que guardava os seguintes valores: "Licenciatura", "Bacharelado" e "Tecnologia". Por meio da função *to_sql* presente no *SQLAlchemy* os dados foram submetidos a uma tabela *staging* presente na base de dados *PostgreSQL*, que estava em execução pelo *Docker*. Então, foi realizada a etapa de Transformação que contou com a remoção de valores duplicados, ausentes ou incorretos, bem como foi feita a eliminação de variáveis que se apresentavam irrelevantes para o trabalho, como as diversas variáveis de vagas ofertadas, que seriam correspondentes às vagas disponibilizadas no início de um curso, onde a maioria de seus valores constava o valor zero.

Diversos campos foram removidos, variáveis referentes à região foram eliminadas, uma vez que os dados são de uma única região (Paraíba). Além disso, foi realizado um treinamento inicial para verificar a influência de algumas variáveis nos modelos, onde tal processo também contribuiu para a remoção de mais campos, tais como os campos de: Vagas ofertadas, Número de inscritos e Modalidade de ensino. No fim, 16 atributos restaram. O significado de cada um deles pode ser observado no Quadro 1.

Quadro 1 – Significado dos atributos

Atributo	Significado
sexo	Sexo do aluno.
renda	Renda Familiar Per capita (RFP) de um aluno.
cor	Cor ou raça de um aluno.
faixa_etaria	Faixa etária de idade de um aluno.
idade	Idade de um aluno.
codigo_da_matricula	Código da matrícula que identifica um aluno.
cod_do_ciclo_matricula	Código do ciclo (início e fim) da matrícula de um aluno.
cod_curso	Código que identifica um curso.
categoria_de_situacao	Situação em que um aluno se encontra:evadido ou concluinte.
data_de_inicio_do_ciclo	Data de quando um aluno iniciou um curso.
data_de_fim_previsto_do_ciclo	Data prevista para um aluno terminar seu curso.
mes_de_ocorrencia_da_situacao	Data de mudança de estado da matrícula de um aluno.
carga_horaria_do_curso	Carga horária de um curso.
turno	Turno de um curso.
eixo_tecnologico	Área em que um curso se encontra.
tipo_de_curso	Tipo de curso:Licenciatura, Bacharelado ou Tecnologia.
nome_do_curso	Nome do curso.
fator_de_esforco_de_curso	Carga horária do curso em função da quantidade de aulas práticas que, tecnicamente, demandem menor Relação Matrícula por Professor.
unidade_de_ensino	Nome do *Campus.*
cod_sistec	Código que identifica um *Campus.*

Fonte: Adaptado de (PNP, 2022)

3.2.1 Modelagem Dimensional

Para a Modelagem Dimensional foi definido o que seria o fato e suas dimensões. Uma vez que o objeto de estudo central é a evasão dos alunos, tal fenômeno foi determinado como um fato.

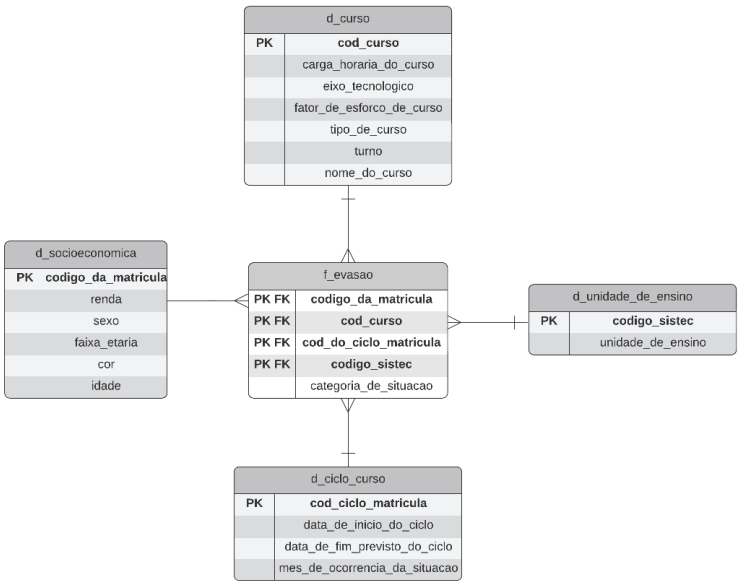
No intuito de identificar as dimensões buscou-se compreender como estava a

situação da evasão de um aluno com relação às suas características socioeconômicas, ao seu curso, à unidade de ensino e também ao seu ciclo de matrícula, a exemplo das variáveis **data_de_inicio_do_ciclo** e **data_de_fim_previsto_do_ciclo**.

Feitas todas essas conclusões, foram estabelecidas as tabelas **d_socioeconomica, d_curso, d_ciclo_curso** e **d_unidade_de_ensino** para representar as dimensões, assim como a tabela **f_evasao** para representar o fato, como pode ser observado na Figura 20.

Cada *primary key* (chave identificadora PK) das tabelas de dimensões são exportadas para a tabela de fato como *foreign key* (chave estrangeira FK) para que seja possível interligar essas tabelas por meio de um atributo em comum. Com isso, torna-se mais prática as consultas na base de dados e também o processo de análise. Todo esse esquema foi montado com o objetivo de facilitar as consultas na base de dados, bem como estruturá-los e organizá-los de maneira a facilitar a busca por respostas que evidenciem, por exemplo, como está a situação da matrícula do aluno (na tabela **f_evasao**) com relação a sua renda, faixa etária ou cor (na tabela **d_socioeconomica**).

Figura 20 – Modelo Dimensional (*Star Schema*)



Fonte: Elaborado pelo autor

3.3 CARREGAMENTO DOS DADOS

Uma vez definida a modelagem dos dados foi possível extraí-los da tabela *staging* e carregá-los às suas respectivas tabelas definidas na Modelagem Dimensional, assim finalizando a produção do *Data Warehouse*. Mais abaixo encontra-se um exemplo

de inserção dos dados para a dimensão curso no Algoritmo 1.

Nota-se a utilização de funções como *UNACCENT* para remover acentos das palavras e *LOWER* para torná-las minúsculas, uma vez que havia palavras iguais, mas escritas com acentos diferentes ou escritas em maiúsculo. Também foi utilizado uma seleção distinta com o *DISTINCT ON* para retornar apenas os cursos que diferiam em seu código, já que na tabela de dimensão não pode haver informações repetidas, o que é diferente na tabela de fatos, onde, por exemplo, um curso com o mesmo código pode se mostrar várias vezes.

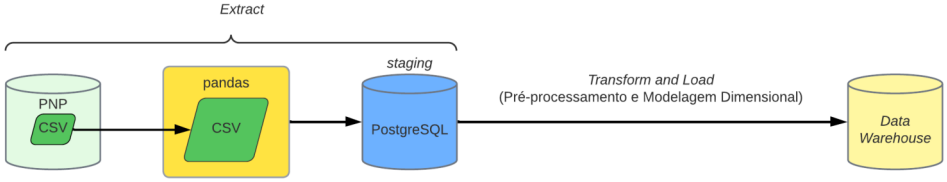
Algoritmo 1 – Exemplo de script de inserção

```
INSERT INTO public.d_curso(cod_curso,nome_do_curso,
 carga_horaria_do_curso,
 eixo_tecnologico,fator_de_esforco_de_curso,
 tipo_de_curso,turno)(
  SELECT DISTINCT ON (cod_curso,nome_do_curso) cod_curso,
    UNACCENT(LOWER(nome_do_curso)) AS nome_do_curso,
    carga_horaria_do_curso,
    UNACCENT(LOWER(eixo_tecnologico)) AS eixo_tecnologico,
    fator_de_esforco_de_curso,
    UNACCENT(LOWER(tipo_de_curso)) AS tipo_de_curso,
    UNACCENT(LOWER(turno)) AS turno
    FROM staging
);
```

Fonte: Elaborado pelo autor

O processo de inserção foi repetido até se preencher todas as tabelas corretamente. Todo o fluxo pode ser verificado na Figura 21 onde cada etapa do ETL foi devidamente concluída até a finalização do DW. O processo de extração se deu desde o *download* dos arquivos CSV até a sua inserção na tabela *staging* intermediada pela linguagem *Python*. Um vez na tabela *staging* os dados foram pré-processados, modelados e carregados no *Data Warehouse*.

Figura 21 – Fluxograma do produção do *Data Warehouse*



Fonte: Elaborado pelo autor

4 ANÁLISE EXPLORATÓRIA DE DADOS

4.1 TECNOLOGIAS UTILIZADAS

Foram utilizadas as seguintes tecnologias na AED:

Matplotlib[1] ***:*** como descrito na própria biblioteca:"Matplotlib é uma biblioteca abrangente para criar visualizações estáticas, animadas e interativas em Python. O Matplotlib torna as coisas fáceis fáceis e as difíceis possíveis".

Seaborn[2] ***:*** é uma biblioteca *Python* para visualização de dados com gráficos estatísticos baseada no *Matplotlib*.

Plotly[3] ***:*** é uma biblioteca *Python* também utilizada para visualização de dados, porém com o *Plotly* é possível criar gráficos mais robustos e dinâmicos.

Streamlit[4] ***:*** é um *framework Python* utilizado para construir aplicações web com modelos de *Machine Learning*, mas neste trabalho também foi utilizado para visualização de dados.

Além disso, também foram utilizadas a biblioteca ***Pandas*** e a plataforma ***Jupyter Notebook*** mencionadas no Capítulo 3.

4.2 APRESENTAÇÃO DE *INSIGHTS*

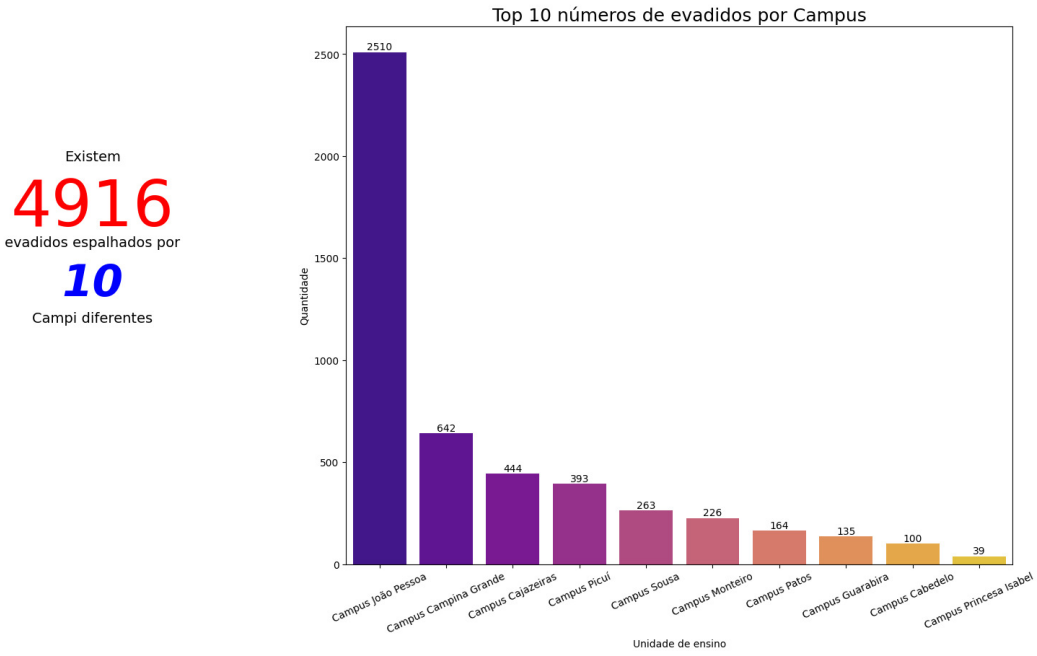
Uma vez com a base de dados pronta o caminho para realizar a AED ficou livre. Foi possível retirar *insights* importantes a serem utilizados para tomadas de decisão. Na Figura 22 podemos ver como estava distribuída a evasão pelos diferentes *Campi*. Os *Campi* João Pessoa, Campina Grande e Cajazeiras apresentaram os três maiores números de evasões.

Porém, uma análise por contagem, nem sempre irá representar a realidade, pois nesse caso o maior número de evasões no *Campus* João Pessoa, por exemplo, pode estar ligado apenas ao fato de que nessa instituição a quantidade de alunos é maior que nas outras.

---

1 <https://matplotlib.org/>
2 <https://seaborn.pydata.org/>
3 <https://plotly.com/python/>
4 <https://streamlit.io/>

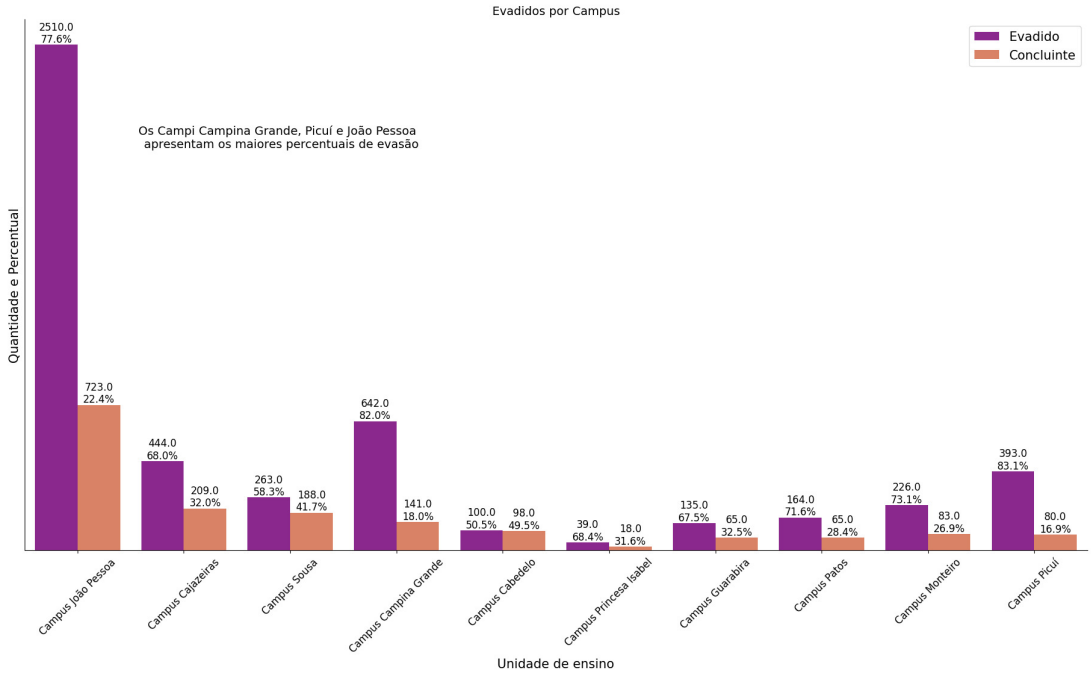
Figura 22 – Evadidos por *Campus*



Fonte: Elaborado pelo autor

Portanto, é necessária uma análise percentual para averiguar a proporção de evadidos em cada *Campus* ao comparar-se com a proporção de concluintes. Na Figura 23 são exibidos os percentuais de evasão agrupados por concluinte e evadido de acordo com cada *Campus*, onde é possível notar que os *Campi* Picuí e Campina Grande apresentaram os maiores percentuais de evasão, com 83.1% e 82%, respectivamente.

Figura 23 – Percentual de evadidos por *Campus*

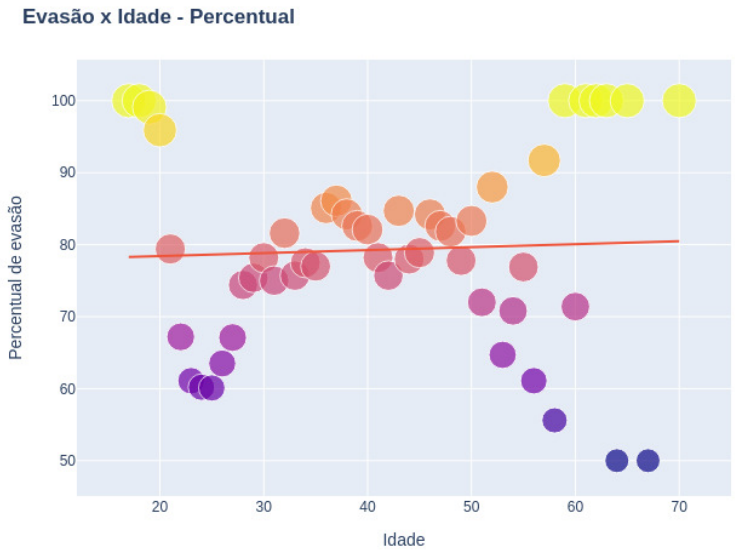


Fonte: Elaborado pelo autor

Outro *insight* expressivo foi obtido ao observar a relação entre a idade e a evasão. Inicialmente, foi constatado que o número de evasões aumentava à medida que a idade diminuía. Porém, ao analisar se o percentual de evasão, e não sua frequência absoluta, aumentava ou diminuía junto com a idade, observou-se que não havia relação entre essas duas variáveis, evidenciando que o número de evasões era maior entre os mais jovens devido ao fato dos alunos mais jovens serem maioria nos dados.

A análise pode ser observada na Figura 24, onde também podemos afirmar que a probabilidade de evasão é relativamente maior na faixa de 30 a 50 anos. No eixo x deste gráfico tem-se a idade do aluno, no eixo y tem-se o percentual de evasão para a idade em questão, e no seu centro, está uma linha de tendência, que se estivesse muito inclinada para cima ou para baixo, haveria uma correlação entre essas duas variáveis.

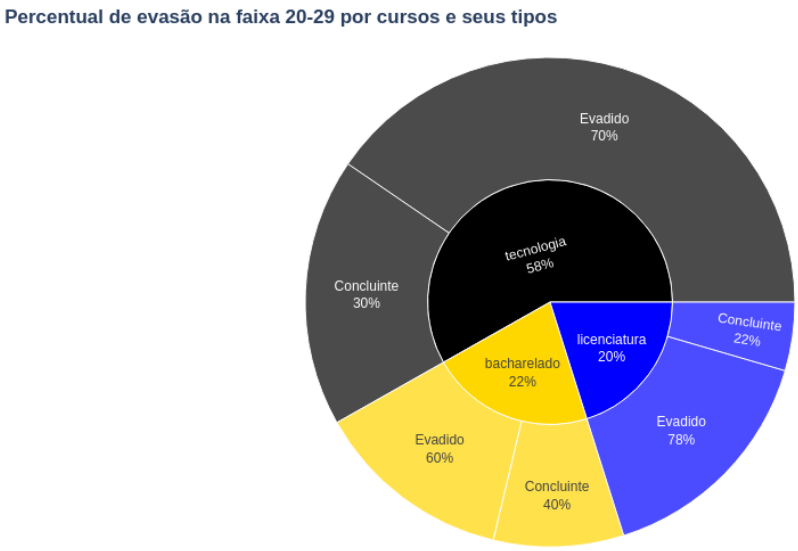
Figura 24 – Análise de correlação da idade com a evasão



Fonte: Elaborado pelo autor

Durante as análises, para entender a evasão nos cursos, optou-se por dividir os dados em grupos de faixa etária. O grupo principal a ser analisado foi o grupo com alunos na faixa de 20 a 29 anos, dado que a maior parte das evasões e dos dados eram de estudantes nessa faixa de idade, sendo um grupo capaz de fornecer uma visão do comportamento geral dos dados, devido a sua superioridade numérica. A partir disso, foi possível observar que os cursos de Licenciatura representavam a maior parte das evasões, seguido por Tecnologia e Bacharelado, como demonstrado na Figura 25, que exibe um gráfico que segmenta os percentuais de evasão por tipo de curso (círculo do centro) e os percentuais de evasão de acordo com situação de evadido ou concluinte (círculo externo).

Figura 25 – Concentração da evasão por cursos e tipos - Faixa 20-29



Fonte: Elaborado pelo autor

A evasão nos cursos para o grupo na faixa de 30 a 49 anos também foi analisada, porém não apresentou nenhuma diferença significativa para o grupo anterior. Demais grupos acima dessas faixas não foram analisados unicamente devido a baixa quantidade de registros para estudantes acima de 49 anos que poderia resultar em análises não muito precisas. Muitos outros *insights* foram obtidos, tais como os percentuais de evasão com relação à caraterísticas socioeconômicas como faixa etária e renda, onde as cores preta e parda apresentaram os maiores números, assim como as rendas mais baixas de no máximo 1,5 salários mínimo.

Ainda, os cursos que apresentavam os maiores percentuais de evasão foram identificados. Para os cursos de Tecnologia, por exemplo, os maiores percentuais de evasão se apresentaram para os cursos de Redes de Computadores, Análise e Desenvolvimento de Sistemas e Sistemas para Internet com 83%, 79% e 78% de evasão, na devida ordem. Os demais *insights* podem ser vistos na aplicação web[5] ou no arquivo *Notebook*[6] anexados no rodapé desta página.

Em posse dessas informações os *Campi* do IFPB podem direcionar medidas preventivas e adaptar estratégias de retenção de acordo com a necessidade, uma vez

[5] <https://data-visualization-and-forecasting-student-dropout.streamlit.app/>
[6] <https://nbviewer.org/github/math3usvalenca/machine-learning-no-combate-a-evasao-estudantil/blob/main/analise-de-dados-estudantis/AED.ipynb>

que se conhece o perfil de um aluno evadido e quais *Campi* e cursos com as maiores estatísticas de evasão.

5 ANÁLISE PREDITIVA

5.1 TREINAMENTO DOS ALGORITMOS

Para realizar a análise preditiva foram utilizados os algoritmos: *Logistic Regression* (Regressão Logística), *K-Nearest Neighbors*, *Decision Tree* e *Support Vector Machine*. Buscou-se encontrar os melhores parâmetros para treinar os modelos com a função *RandomizedSearchCV* da biblioteca *scikit-learn*. As estatísticas dos modelos treinados encontram-se na Tabela 2. Todo o procedimento utilizado no treinamento dos modelos pode ser conferido no link[1] em anexo no final desta página.

Tabela 2 – Métricas dos modelos treinados

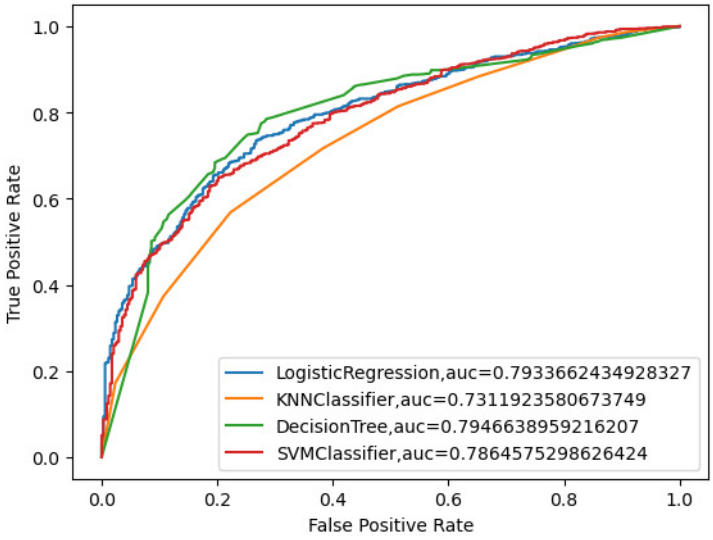
Algoritmo	Precisão	Acurácia	Recall
KNN	80%	75%	88%
Decision Tree	**83%**	**78%**	88%
SVM	81%	77%	91%
Logistic Regression	80%	77%	**92%**

Fonte: Elaborado pelo autor

Os resultados foram satisfatórios, com uma Precisão na faixa de 80% em todos os algoritmos utilizados. Além disso, a cobertura (*Recall*) chegou a valores de 92%, caso do *Logistic Regression*. Isso indica um resultado animador, pois como mencionado anteriormente, essa métrica mede a capacidade do modelo em detectar corretamente classes positivas (que seriam alunos evadidos). Por fim, os modelos foram comparados pela área sob a curva ROC, onde o *Logistic Regression* apresentou um AUC muito próximo do *Decision Tree* e acabou sendo escolhido pelo fato de apresentar um valor de cobertura mais alto (92%). Os dados obtidos são apresentados na Figura 26.

[1] <https://nbviewer.org/github/math3usvalenca/machine-learning-no-combate-a-evasao-estudantil/blob/main/aplicacao-de-algoritmos/machine_learning.ipynb>

Figura 26 – Comparação dos modelos - Área sob a curva ROC



Fonte: Elaborado pelo autor

Em posse do modelo final, foi construída uma aplicação *web* com código fonte disponível em um repositório no site *Github* e com uma versão *online* na plataforma *Streamlit* [2]. Os usuários têm a opção de utilizar a aplicação de maneira local através do código-fonte ou *online* e podem informar os mesmos atributos utilizados nos dados de treinamento dos modelos para obter como resultado uma probabilidade de um aluno informado pertencer ao grupo de evadidos, como visto na Figura 27.

Para utilizar a ferramenta de maneira local é obrigatório que a linguagem *Python* esteja instalada no computador do usuário. Com isso, deve-se apenas abrir o terminal da máquina e executar os passos do Algoritmo 2, onde o primeiro comando (linha 1) é para realizar o *download* do código-fonte, em seguida (linha 3) tem-se o comando para instalar as dependências funcionais da aplicação e por último (linha 5) toma-se o comando para executar a aplicação localmente que poderá ser acessada diretamente de um navegador *web*.

[2] <https://data-visualization-and-forecasting-student-dropout.streamlit.app/>

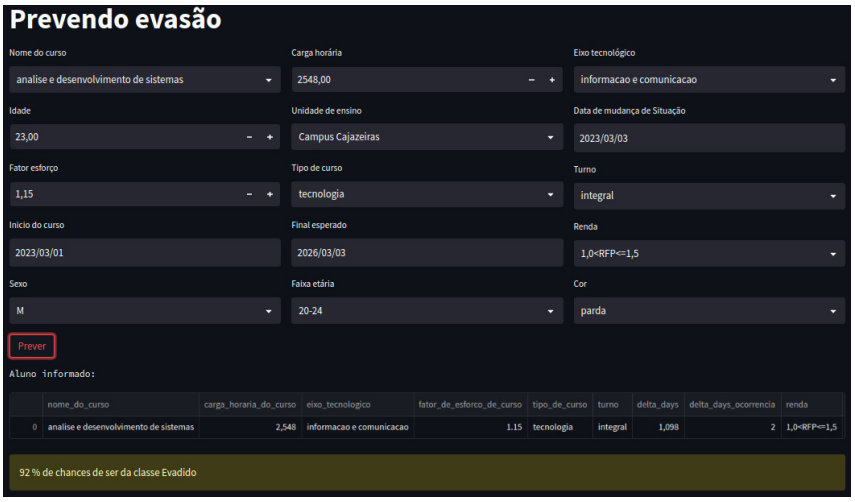
Algoritmo 2 – Exemplo de como usar a aplicação localmente

```
git clone https://github.com/math3usvalenca/data-visualization-and-prediction-of-student-dropout.git

pip install -r requirements.txt

streamlit run app.py
```

Fonte: Elaborado pelo autor

A ferramenta pode ser utilizada, por exemplo, no início do curso para identificar alunos mais propensos à evasão e na definição de estratégias para evitá-la. Além disso, os *insights* obtidos na AED também são de suma importância na elaboração de tais estratégias.

Figura 27 – Modelo preditivo - Aplicação *web*



Fonte: Elaborado pelo autor

6 CONSIDERAÇÕES FINAIS

A finalidade deste trabalho foi auxiliar no combate à evasão estudantil de cursos superiores no IFPB por meio de uma análise preditiva e exploratória de dados. Dados da Plataforma Nilo Peçanha foram preparados de maneira adequada para serem utilizados no treinamento e teste de modelos em uma tarefa de classificação binária, visando identificar alunos com risco de evasão.

Foram utilizados dois critérios para analisar o modelo final a ser consumido: *Recall* e a área sob a curva ROC. O *Recall* foi uma métrica importante para o objetivo de minimizar os falsos negativos e aumentar os verdadeiros positivos, ou seja, ampliar a capacidade de identificação de alunos que pertencem à *classe* evadido. A área sob a curva ROC foi utilizada para avaliar a capacidade geral do modelo em distinguir *classes* positivas (evadidos) de negativas (concluintes).

O modelo final escolhido foi a Regressão Logística, pois demonstrou-se ser mais adequado devido a um bom resultado tanto no *Recall*, quanto da área sob a curva ROC. A maior limitação para o trabalho deu-se pelo fato da impossibilidade de acesso a dados de histórico escolar, tais como frequência das aulas, notas disciplinares e disciplinas cursadas que certamente contribuiriam para resultados mais rigorosos.

Em todo o caso, durante a AED, foram obtidos *insights* importantes sobre as características que contribuem para evasão de alunos de cursos superiores no IFPB. Tais *insights* podem ser aproveitados para a construção de políticas de intervenções visando a redução na taxa de evasão e melhoria no desempenho dos alunos.

Uma aplicação *web* foi construída para tornar o uso do modelo uma tarefa mais prática ao permitir que usuários informem dados sobre um novo aluno e obtenham uma probabilidade do mesmo pertencer ao grupo de evadidos.

Em suma, os objetivos do trabalho foram devidamente atingidos com a aplicação de métodos de *Machine Learning* e uma análise exploratória sobre dados abertos da rede federal de ensino. Os resultados podem contribuir para o IFPB adotar medidas proativas na retenção de alunos e na melhoria da eficiência acadêmica.

Além do mais, com a aplicação *web* o uso do modelo se torna mais acessível por gestores e professores, demonstrando-se uma ferramenta prática de se utilizar para os usuários finais. Contudo, é preciso ressaltar que os resultados apresentados por esta ferramenta não devem ser tomados como uma decisão final, ela é um auxílio para

tomada de decisões e requer cuidados quanto ao seu uso, pois devido a natureza dos dados, mesmo com uma alta probabilidade de evasão obtida, não é possível realmente garantir que tal aluno irá evadir em um ponto futuro, mas pode-se entender que ele tem um potencial para tal. Esta ferramenta serve para apoiar as decisões e previsões, mas sempre levando-se em consideração o conhecimento humano dada as limitações e incertezas de todo modelo de ML, seja por conta dos dados utilizados, seja por conta da própria aleatoriedade do futuro ou do problema em questão.

6.1 TRABALHOS FUTUROS

Para trabalhos futuros, sugere-se a utilização de dados de histórico escolar para resultados mais completos que enriqueceriam os modelos e os *insights* obtidos. Outra possibilidade, seria a implementação de algoritmos e métodos diferentes, utilizando, por exemplo, algoritmos de redes neurais artificiais e métodos *Ensemble* que combinam os resultados de diversos modelos para realizar a previsão, onde os principais algoritmos são o *AdaBoost*, *Gradient Boosting* e *XGBoost*.

Por fim, considerando a adoção de políticas de intervenção dado os resultados da análise preditiva a da AED, também sugere-se para estudos futuros a tarefa de mensurar a efetividade dessas políticas e analisar se houve diminuição significativa nas estatísticas de evasão no IFPB.

REFERÊNCIAS

ALPAYDIN, E. **Introduction to machine learning**. [S.l.]: MIT press, 2020.

AMARAL, F. **Introdução à ciência de dados: mineração de dados e big data**. [S.l.]: Alta Books Editora, 2016.

ARAUJO, R. **Aprendizado com Árvores de Decisão**. 2020. Acessado em: 28-dez-2023]. Disponível em: <https://ricardomatsumura.medium.com/aprendizado-com-%C3%A1rvores-de-decis%C3%A3o-73d874664d1>.

ARAÚJO, E. Barbosa de; LIMA, A. M. de. Evasão nos institutos federais: a produção científica da pós-graduação brasileira no período 2014-2018. **Congresso Nacional de Educação-CONEDU**, 2020.

AVELAR, A. **O que é AUC e ROC nos modelos de Machine Learning — eam.avelar**. 2019. Acessado em: 16-Jul-2023. Disponível em: <https://medium.com/@eam.avelar/o-que-%C3%A9-auc-e-roc-nos-modelos-de-machine-learning-2e2c4112033d#:~:text=ROC%20%C3%A9%20uma%20curva%20de,o%20classificador%20errou%20a%20predi%C3%A7%C3%A3o.>

BIANCHI, J. Previsão de evasão em cursos de ensino superior através de aprendizado de máquina associado à análise de disciplinas aprovadas. Universidade Federal de Santa Maria, 2017.

BONACCORSO, G. **Machine learning algorithms**. [S.l.]: Packt Publishing Ltd, 2017.

CHAUDHURI, S.; DAYAL, U. An overview of data warehousing and olap technology. **ACM Sigmod record**, ACM New York, NY, USA, v. 26, n. 1, p. 65–74, 1997.

DUARTE, M. A construção da vida ou um novo modelo para a intervenção na carreira. **Desenvolvimento Vocacional: Avaliação e intervenção**, p. 21–30, 2010.

ECKERSON, W. W. Predictive analytics. **Extending the Value of Your Data Warehousing Investment. TDWI Best Practices Report**, v. 1, p. 1–36, 2007.

ESCOVEDO, T.; KOSHIYAMA, A. **Introdução a Data Science: Algoritmos de Machine Learning e métodos de análise**. [S.l.]: Casa do Código, 2020.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. **AI magazine**, v. 17, n. 3, p. 37–37, 1996.

FILHO, R. L. L. S.; MOTEJUNAS, P. R.; HIPÓLITO, O.; LOBO, M. B. d. C. M. A evasão no ensino superior brasileiro. **Cadernos de pesquisa**, SciELO Brasil, v. 37, p. 641–659, 2007.

GANDHI, R. **Support Vector Machine — Introduction to Machine Learning Algorithms**. 2018. Acessado em: 29-dez-2022. Disponível em: <https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47>.

GOMEZ-URIBE, C. A.; HUNT, N. The netflix recommender system: Algorithms, business value, and innovation. **ACM Transactions on Management Information Systems (TMIS)**, ACM New York, NY, USA, v. 6, n. 4, p. 1–19, 2015.

HILBE, J. M. Logistic regression. **International encyclopedia of statistical science**, v. 1, p. 15–32, 2011.

HOSSIN, M.; SULAIMAN, M. N. A review on evaluation metrics for data classification evaluations. **International journal of data mining & knowledge management process**, Academy & Industry Research Collaboration Center (AIRCC), v. 5, n. 2, p. 1, 2015.

HÜSEMANN, B.; LECHTENBÖRGER, J.; VOSSEN, G. **Conceptual data warehouse design**. [S.l.]: Universität Münster. Angewandte Mathematik und Informatik, 2000. v. 168.

JOSÉ, I. **KNN (K-Nearest Neighbors)**. 2018. Acessado em: 21-dez-2022. Disponível em: <https://medium.com/brasil-ai/knn-k-nearest-neighbors-1-e140c82e9c4e>.

KFOLLIS. **Understand star schema and the importance for Power BI - Power BI — learn.microsoft.com**. 2021. Acessado em: 02-Jan-2023. Disponível em: <https://learn.microsoft.com/en-us/power-bi/guidance/star-schema>.

KIMBALL, R.; ROSS, M. **The data warehouse toolkit: the complete guide to dimensional modeling**. [S.l.]: John Wiley & Sons, 2011.

LOPES, G. R.; ALMEIDA, A. W. S.; DELBEM, A.; TOLEDO, C. F. M. Introdução à análise exploratória de dados com python. **Minicursos ERCAS ENUCMPI**, v. 2019, p. 160–176, 2019.

MACHADO, F. N. R. **Tecnologia e projeto de Data Warehouse**. [S.l.]: Saraiva Educação SA, 2004.

MANDREKAR, J. N. Receiver operating characteristic curve in diagnostic test assessment. **Journal of Thoracic Oncology**, Elsevier, v. 5, n. 9, p. 1315–1316, 2010.

MORAES, G. H.; JÚNIOR, W. Tavares da S.; KENCHIAN, G. Guia de referência metodológica PNP - 2020. Brasília/DF, Editora Evobiz, p. 1–181, 2020.

MULAK, P.; TALHAR, N. Analysis of distance measures using k-nearest neighbor algorithm on kdd dataset. **Int. J. Sci. Res**, v. 4, n. 7, p. 2319–7064, 2015.

PARSONS, C. **What's a Machine Learning Model? — blogs.nvidia.com**. 2021. <https://blogs.nvidia.com/blog/2021/08/16/what-is-a-machine-learning-model/>. Acessado em: 07-ago-2023.

PETERSSON, D. **What is Supervised Learning? — techtarget.com**. 2021. <https://www.techtarget.com/searchenterpriseai/definition/supervised-learning>. Acessado em: 29-dez-2022.

PNP. **PNP - Plataforma Nilo Peçanha**. 2022. Acessado em: 14-Nov-2022. Disponível em: <https://www.gov.br/mec/pt-br/pnp>.

PRATES, W. R. **Curva ROC e AUC em Machine Learning - Ciência e Negócios — cienciaenegocios.com**. 2020. Acessado em: 16-Jul-2023. Disponível em: <https://cienciaenegocios.com/curva-roc-e-auc-em-machine-learning/>.

PRIMÃO, A. P. et al. Uso de algoritmos de machine learning para prever a evasão escolar no ensino superior: um estudo no instituto federal de santa catarina. 2022.

PUJARI, A. K. **Data mining techniques**. [S.l.]: Universities press, 2001.

RODRIGUES, V. **Entenda o que é AUC e ROC nos modelos de Machine Learning — medium.com**. 2018. Acessado em: 16-Jul-2023. Disponível em: <https://medium.com/bio-data-blog/entenda-o-que-%C3%A9-auc-e-roc-nos-modelos-de-machine-learning-8191fb4df772>.

SEMESP. **Evasão – Dados Brasil – 11º Mapa do Ensino Superior**. 2021. Acessado em: 13-Nov-2022. Disponível em: <https://www.semesp.org.br/mapa/edicao-11/brasil/evasao/>.

SEMOLINI, R. et al. Support vector machines, inferência transdutiva e o problema de classificação. **Campinas, SP**, 2002.

SIMITSIS, A. Modeling and optimization of extraction-transformation-loading (etl) processes in data warehouse environments. **National Technical University of Athens: PhD Thesis, Athens, Greece**, 2004.

SWAMINATHAN, S. **Logistic Regression — Detailed Overview — towardsdatascience.com**. 2018. <https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc>. Acessando em: 07-Jun-2023.

TUKEY, J. W. et al. **Exploratory data analysis**. [S.l.]: Reading, MA, 1977. v. 2.

UNICEF. **Dois milhões de crianças e adolescentes de 11 a 19 anos não estão frequentando a escola no Brasil, alerta UNICEF**. 2022. Acessado em: 13-Nov-2022. Disponível em: <https://www.unicef.org/brazil/comunicados-de-imprensa/dois-milhoes-de-criancas-e-adolescentes-de-11-a-19-anos-nao-estao-frequentando-a\-escola-no-brasil>.

VIANA, F. S.; SANTANA, A. M.; RABÊLO, R. d. A. L. Avaliaçao de classificadores para prediçao de evasao no ensino superior utilizando janela semestral. In: SBC. **Anais do XXXIII Simpósio Brasileiro de Informática na Educação**. [S.l.], 2022. p. 908–919.

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
Campus Cajazeiras
Rua José Antônio da Silva, 300, Jardim Oásis, CEP 58.900-000, Cajazeiras (PB)
CNPJ: 10.783.898/0005-07 - Telefone: (83) 3532-4100

Documento Digitalizado Ostensivo (Público)

Entrega de versão final de TCC

Assunto: Entrega de versão final de TCC
Assinado por: Francisco Trajano
Tipo do Documento: Anexo
Situação: Finalizado
Nível de Acesso: Ostensivo (Público)
Tipo do Conferência: Cópia Simples

Documento assinado eletronicamente por:

- **Francisco Matheus Valença Trajano**, **ALUNO (201912010018) DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - CAJAZEIRAS**, em 20/09/2023 16:03:49.

Este documento foi armazenado no SUAP em 20/09/2023. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse https://suap.ifpb.edu.br/verificar-documento-externo/ e forneça os dados abaixo:

Código Verificador: 948880
Código de Autenticação: 454a92ff6d