UNIVERSIDADE FEDERAL DE PELOTAS
Centro de Desenvolvimento Tecnológico
Programa de Pós-Graduação em Computação

Dissertação

Aplicação de Técnicas de Mineração de Dados e Learning Analytics para Predição de Evasão de Alunos nos Cursos de Ciência da Computação e Engenharias da UFPel

Alexandre Gomes da Costa

Pelotas, 2021

Alexandre Gomes da Costa

Aplicação de Técnicas de Mineração de Dados e Learning Analytics para Predição de Evasão de Alunos nos Cursos de Ciência da Computação e Engenharias da UFPel

Dissertação apresentada ao Programa de Pós-Graduação em Computação do Centro de Desenvolvimento Tecnológico da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Julio Carlos Balzano de Mattos
Coorientador: Prof. Dr. Tiago Thompsen Primo

Pelotas, 2021

Universidade Federal de Pelotas / Sistema de Bibliotecas
Catalogação na Publicação

C837a Costa, Alexandre Gomes da

Aplicação de técnicas de mineração de dados e learning analytics para predição de evasão de alunos nos cursos de Ciência da Computação e Engenharias da UFPel / Alexandre Gomes da Costa ; Julio Carlos Balzano de Mattos, orientador ; Tiago Thompsen Primo, coorientador. — Pelotas, 2021.

91 f. : il.

Dissertação (Mestrado) — Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, 2021.

1. Mineração de dados educacionais. 2. Learning analytics. 3. Técnicas de predição. 4. Kdd. 5. Descoberta de conhecimento em base de dados. I. Mattos, Julio Carlos Balzano de, orient. II. Primo, Tiago Thompsen, coorient. III. Título.

CDD : 005

Alexandre Gomes da Costa

Aplicação de Técnicas de Mineração de Dados e Learning Analytics para Predição de Evasão de Alunos nos Cursos de Ciência da Computação e Engenharias da UFPel

Dissertação aprovada, como requisito parcial, para obtenção do grau de Mestre em Ciência da Computação, Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas.

Data da Defesa: 08 de março de 2021

Banca Examinadora:

Prof. Dr. Júlio Carlos Balzano de Mattos (orientador)
Doutor em Computação pela Universidade Federal do Rio Grande do Sul, Brasil.

Prof. Dr. Tiago Thompsen Primo (co-orientador)
Doutor em Computação pela Universidade Federal do Rio Grande do Sul, Brasil.

Prof. Dr. Cristian Cechinel
Doutor em Ingeniería de la Información y del Conocimiento pela Universidad de Alcalá, Espanha.

Prof. Dr. Marilton Sanchotene de Aguiar
Doutor em Computação pela Universidade Federal do Rio Grande do Sul, Brasil.

Prof. Dr. Rafael Dias Araújo
Doutor em Ciência da Computação pela Universidade Federal de Uberlândia, Brasil.

Aos meus pais e irmão, Clóvis, Lúcia e Andrigo, pelo apoio e incentivo em todos os momentos da minha vida. Por acreditarem em mim, e não medirem esforços para a concretização dos meus sonhos. Sem vocês, nada seria possível.

Aos meus familiares, amigos e colegas. Mesmo com a distância, sempre se fizeram presentes na minha vida e estarão sempre em meu coração. Obrigada pelo companheirismo, apoio e amizade incondicional.

AGRADECIMENTOS

Aos meus pais e irmão, Clovis, Lucia e Andrigo, que nunca mediram esforços para me ensinar o caminho do bem, e sempre me apoiaram em todas as etapas da minha vida. Sem vocês, eu não chegaria até aqui. Muito obrigada por tudo! O amor que sinto por vocês é incondicional.

À minha família e amigos. Obrigada por acreditar no meu sonho e sempre me motivar a seguir em frente. É muito bom saber que posso contar com vocês em todos os momentos. Amo vocês!

Ao meu orientador, Professor Júlio Carlos Balzano de Mattos, pela oportunidade de realizar este trabalho. Obrigada pela confiança. Agradeço por todos os ensinamentos compartilhados e por me guiar nos primeiros passos da pós-graduação.

Ao meu co-orientador, Professor Tiago Thompsen Primo, por toda a ajuda e ensinamentos durante a realização deste trabalho. Sua contribuição foi essencial para a concretização de todas as etapas deste trabalho desenvolvido neste Programa de pós-graduação.

Gostaria de agradecer aos meus colegas de trabalho em especial o José Hiram Salengue Noguez, por ter me apresentado o Professor Júlio. Obrigado pela convivência agradável no dia-a-dia.

Tudo o que não puder contar como fez; Não o faça! Se há razões para não contar; há para não o fazer.
— KANT

RESUMO

COSTA, Alexandre Gomes da. **Aplicação de Técnicas de Mineração de Dados e Learning Analytics para Predição de Evasão de Alunos nos Cursos de Ciência da Computação e Engenharias da UFPel**. Orientador: Julio Carlos Balzano de Mattos. 2021. 91 f. Dissertação (Mestrado em Ciência da Computação) – Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2021.

Os sistemas de gestão para educação armazenam uma grande quantidade de dados oriundos de diversas modalidades de interação entre alunos e professores, mas também entre os alunos e o ambiente educacional. Analisar e encontrar padrões nesta quantidade de dados manualmente é inviável, por isso a utilização de Mineração de Dados Educacionais (MDE) é largamente utilizada. Este trabalho apresenta modelos de predição de alunos em risco de evasão usando apenas os dados dos três primeiros semestres cursados pelos alunos (N=1514) no curso de Ciência da Computação e alunos (N=6351) de doze cursos de Engenharia da Universidade Federal de Pelotas (UFPel). Ambos conjuntos de dados utilizaram os mesmos atributos, que foi no total de 22 atributos entre socio-econômicos e acadêmicos. Neste trabalho é utilizada a metodologia CRISP-DM e os dados extraídos no sistema acadêmico da UFPel (Cobalto). Foram selecionados Para as duas bases de dados (Ciência da Computação e Engenharias) são apresentados resultados para cinco algoritmos de predição. Para o curso de Ciência da Computação, o melhor resultado foi com o modelo de Regressão Logística que obteve um precisão de 90,16% e uma revocação de 90,34%. Já para os doze cursos de Engenharia, o resultado obtido para o modelo de Floresta Aleatória foi de uma precisão de 83,40% e uma revocação de 79,48%. Em ambas as bases de dados os resultados indicam que é possível criar um modelo de predição utilizando apenas os dados dos três primeiros semestres.

Palavras-chave: mineração de dados educacionais. learning analytics. técnicas de predição. kdd. descoberta de conhecimento em base de dados.

ABSTRACT

COSTA, Alexandre Gomes da. **Application of Data Mining Techniques and Learning Analytics for Computer Science and Engineering Students Dropout Prediction at UFPel**. Advisor: Julio Carlos Balzano de Mattos. 2021. 91 f. Dissertation (Masters in Computer Science) – Technology Development Center, Federal University of Pelotas, Pelotas, 2021.

Educational Management Systems store a large amount of data from interaction of not only students and professors but also of students and the educational environment. Analyze and find patterns manually from a huge amount of data is hard, so Educational Data Mining (EDM) is widely used. This work presents a model that can predict the student's risk of dropout using data from the first three semesters attended by Computer Science (N=1516) and Engineering (N=6351) Undergraduate students from UFPel. Both data sets used the same attributes, which was a total of 22 attributes between socio-economic and academic. This work uses the CRISP-DM methodology e data from UFPel Management System (called Cobalto). The results are shown for five algorithms. For Computer Science, the Logistic Regression algorithm a precision of 91.24% and a Recall of 92.17% is presented. For Engineering, the Random Forest algorithm a precision of 83.40% and a Recall of 79.48% is presented. For both data bases (Computer Science and Engineering), the results indicate that it is possible to use a prediction model using only the data from the first three semesters of the course.

Keywords: educational data mining. learning analytics. prediction techniques. kdd. knowledge-discovery in databases.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
AC	Ampla concorrência
AED	Análise Exploratória de Dados
AM	Aprendizagem de Máquina
ANDIFES	Associação Nacional dos Dirigentes das Instituições Federais de Ensino Superior
AUC	Area Under the ROC Curve
COCEPE	Conselho Coordenador do Ensino, da Pesquisa e da Extensão
Colab	Colaboratory
CRISP-DM	Cross Industry Standard Process for Data Mining
ENEM	Exame Nacional do Ensino Médio
FN	False Negative
FP	False Positive
FPR	False Positive Rate
GOL	Gestão On-line
IBk	Instance Based Learner
IES	Instituições de Ensino Superior
KDD	*Knowledge Discovery in Database*
L1	Candidatos com renda familiar bruta per capita igual ou inferior a 1,5 salário mínimo que tenham cursado integralmente o ensino médio em escolas públicas (Lei nº 12.711/2012).
L2	Candidatos autodeclarados pretos, pardos ou indígenas, com renda familiar bruta per capita igual ou inferior a 1,5 salário mínimo e que tenham cursado integralmente o ensino médio em escolas públicas (Lei nº 12.711/2012).
L5	Candidatos que, independentemente da renda (art. 14, II, Portaria Normativa nº 18/2012), tenham cursado integralmente o ensino médio em escolas públicas (Lei nº 12.711/2012).

L6	Candidatos autodeclarados pretos, pardos ou indígenas que, independentemente da renda (art. 14, II, Portaria Normativa nº 18/2012), tenham cursado integralmente o ensino médio em escolas públicas (Lei nº 12.711/2012).
L9	Candidatos com deficiência que tenham renda familiar bruta per capita igual ou inferior a 1,5 salário mínimo e que tenham cursado integralmente o ensino médio em escolas públicas (Lei nº 12.711/2012).
L10	Candidatos com deficiência autodeclarados pretos, pardos ou indígenas, que tenham renda familiar bruta per capita igual ou inferior a 1,5 salário mínimo e que tenham cursado integralmente o ensino médio em escolas públicas (Lei nº 12.711/2012).
L13	Candidatos com deficiência que, independentemente da renda (art. 14, II, Portaria Normativa nº 18/2012), tenham cursado integralmente o ensino médio em escolas públicas (Lei nº 12.711/2012).
L14	Candidatos com deficiência autodeclarados pretos, pardos ou indígenas que, independentemente da renda (art. 14, II, Portaria Normativa nº 18/2012), tenham cursado integralmente o ensino médio em escolas públicas (Lei nº 12.711/2012).
LA	Learning Analytics
MD	Mineração de Dados
MDE	Mineração de Dados Educacionais
MLP	Multilayer Perceptron
PEC-G	Programa Estudante Convênio de Graduação
ReLU	Rectified Linear Unit
REUNI	Programa de Apoio a Planos de Reestruturação e Expansão das Universi-dades Federais
RNA	Redes Neurais Artificiais
ROC	Receiver Operating Characteristic
SISU	Sistema de Seleção Unificada
SMO	Sequential Minimal Optimization
SQL	Standard Query Language
SVM	Support Vector Machine
TIC	Tecnologias da Informação e Comunicação
TN	True Negative
TP	True Positive
TPR	True Positive Rate
UFPB	Universidade Federal da Paraíba
UFPel	Universidade Federal de Pelotas

UFRJ	Universidade Federal do Rio de Janeiro
UFSJ	Universidade Federal de São João del-Rei
UnB	Universidade de Brasília

SUMÁRIO

1 INTRODUÇÃO

O uso constante de Tecnologia da Informação e Comunicação em diversas áreas vem gerando um grande volume de dados (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Tecnologias como a internet, redes sociais, ambientes virtuais de aprendizagem, dispositivos móveis, aplicativos embarcados, leitores de código de barras, sensores, leitores biométricos e sistemas de informação em geral são alguns exemplos de recursos que vêm aumentando o número de dados das mais diversas naturezas.

Atualmente, áreas como a educação produzem uma grande quantidade de dados relacionados a alunos e professores todos os dias. Para isto, esta área utiliza sistemas para fazer o controle de interações acadêmicas, gestão de projetos de pesquisa, ensino ou extensão, ou até mesmo sistemas para fazer o controle de gestão de pessoas são alguns dos exemplos de sistemas que podem gerar um volume considerável de dados.

A partir desse volume de dados é possível analisar problemas recorrentes relacionados a educação. Um desses problemas é a evasão escolar que ainda é um desafio a ser superado. A evasão é um problema que atinge não só as Instituições de Ensino Superior (IES) privadas, mas também as públicas. Segundo dados do Inep (2018), em 2017, o índice de matrículas desvinculadas em todo o Brasil foi de 16,41%. Já para as IES públicas esse índice no mesmo período foi de 11,56%. Outro valor a ser considerado é o número de matrículas trancadas, foi de 11,17% em todo o Brasil e 8,10% IES públicas.

Comparando os índices da Universidade Federal de Pelotas (UFPel) com os do Inep (2018) estes índices não mudam muito. Em 2017 o índice de matrículas desvinculadas na UFPel foi de 11,53% que está abaixo do índice de 11,56% apresentado pelo Inep (2018). Mas olhando para a evasão curso a curso, no ano de 2017, observou-se uma situação preocupante. Por exemplo, temos cursos como o de Geoprocessamento onde a taxa de evasão semestral foi de 29,07%. Esse fenômeno é conhecido na estatística como paradoxo de Simpson, uma tendência aparece em um determinado grupo de dados e desaparece quando estes dados são combinados. Ou seja, olhando para os cursos individualmente vários deles apresentam uma taxa de

evasão bem elevada, mas quando essas taxas são combinadas a taxa de evasão não é tão relevante (WAGNER, 1982). Olhando para esses dados algumas perguntas surgem tais como “Qual é o perfil do aluno que tende a evadir?” ou “Qual é a quantidade de alunos que trancaram e acabaram evadindo?”, por exemplo.

Analisar essa crescente quantidade de dados é inviável sem o auxílio de um software (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Para ajudar nesta questão a área de Descoberta de Conhecimento em Base de Dados (*Knowledge Discovery in Database* - KDD) é focada em extrair conhecimento em cima de granes volumes de dados. O termo mais conhecido relacionado a essa área é a Mineração de Dados (MD) que é uma das etapas do processo de KDD.

Segundo Koedinger et al. (2008) Mineração de Dados (MD) aplicada à educação é um campo interdisciplinar mais conhecido como Mineração de Dados Educacionais (MDE). Baker et al. (2010) define MDE como a área de investigação científica centrada no desenvolvimento de métodos para fazer descobertas dentro dos tipos de dados que vêm de ambientes educacionais e usando esses métodos para entender melhor os alunos e a aprendizagem deles. A Figura 1 mostra que a área de MDE pode ser a combinação de 3 grandes áreas (ciência da computação, estatística e educação) e serve de exemplo para justificar a interdisciplinaridade da área.

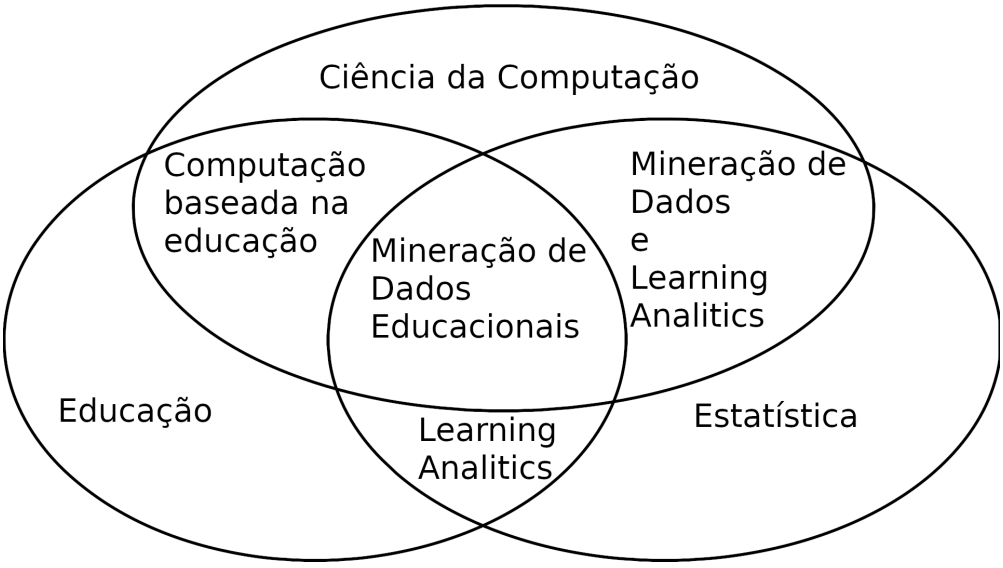

Figura 1 – Principais áreas relacionadas com mineração de dados educacionais. Adaptado de Koedinger et al. (2008)

Este trabalho explora a utilização de MDE visando classificar e identificar perfis de alunos com tendência a evadir. Uma tarefa de classificação possui dois grupos. Um grupo contêm normalmente um atributo apenas que vai servir para fazer a predição de um valor (atributo-alvo). Outro grupo corresponde aos atributos que vão servir para fazer a predição do valor (atributos de predição). Tarefas de classificação são largamente utilizadas para fazer a predição de alunos em risco de evasão escolar

Goldschmidt; Bezerra; Passos (2015); como será apresentado neste trabalho.

Baker et al. (2010) agrupa o problema de evasão em uma categoria ou tarefa de detectar o comportamento do aluno, onde o objetivo é detectar os alunos que têm algum tipo de problema ou comportamento incomum, por exemplo: pouca motivação, plágio, evasão escolar, etc. Destaca também que as principais técnicas usadas para resolver esses tipos de problemas são de classificação e agrupamentos.

Este trabalho possui como motivação o grande problema de evasão encontrado nos cursos superiores, em especial em alguns cursos sendo principalmente cursos das engenharias e exatas, e a dificuldade de traçar o perfil destes alunos. Existe uma enorme quantidade de dados históricos de cursos de graduação presencias da UFPel armazenados no sistema acadêmico da instituição que podem aprimorar/auxiliar o trabalho de combate a evasão. Por outro lado, a grande maioria dos trabalhos relacionados com predição de evasão de alunos vem de cursos EAD. Enquanto alguns trabalhos fazem questionários, provas, exercícios, etc., no presente trabalho serão coletados dados que foram gerados a partir de resultados que os alunos obtiveram em diversas disciplinas, cursos entre outros. Utilizando a base de dados do Sistema Integrado de Gestão da UFPel (Cobalto), que tem dados de alunos de 1980 até os dias atuais. A base do Cobalto conta também com os dados históricos do sistema Gestão On-line (GOL) que foi o sistema acadêmico da UFPel de 2006 a 2013.

Este trabalho apresenta a investigação dos motivos que levam os alunos evadir com ajuda de técnicas de MDE através dos dados acadêmicos dos alunos do curso de Ciência da Computação (dados de 2000 a 2018) e de doze cursos de Engenharia (dados de 2010 a 2018) da UFPel. Como estudo inicial foi escolhido o curso de Ciência da Computação por possuir elevados índices de evasão e a partir dele o modelo foi sendo refinado. Após o modelo utilizado no Curso de Ciência da Computação foi também aplicado em carácter exploratório para doze cursos de Engenharia por possuírem características similares de alta evasão. Para este propósito, foram analisado e coletado dados dos 3 primeiros semestres de alunos dos cursos, para responder as seguintes questões de pesquisa: (Q1) Quais são os atributos que mais influenciam no processo de evasão dos estudantes em cursos de computação?; (Q2) Quais os classificadores e técnicas podem ser utilizados nessa tarefa? (Q3) É possível aplicar os modelos utilizados para classificar os alunos em risco de evasão da Ciência da Computação para classificar alunos das engenharias?

O trabalho está organizado como segue. O Capítulo 2 apresenta o referencial teórico utilizado no trabalho como os conceitos utilizados e também os trabalhos relacionados encontrados na literatura. A metodologia que norteou o desenvolvimento do trabalho, baseado na Cross Industry Standard Process for Data Mining (CRISP-DM), é desenvolvida no Capítulo 3. Já o Capítulo 4 mostra os experimentos e os resultados obtidos para o Curso de Ciência da Computação e o conjunto das Engenharias.

Por fim, o Capítulo 5 apresenta as considerações finais do trabalho e as perspectivas futuras.

2 REFERENCIAL TEÓRICO

Neste Capítulo serão abordados os temas relacionados a predição de alunos que evadiram com o foco nos cursos de Ciência da Computação e Engenharias usando técnicas de mineração de dados. Para isso, serão apresentados conceitos de evasão escolar e mineração de dados.

2.1 Caracterização

Para desenvolver essa pesquisa foram utilizados dados, registrados no Cobalto, do curso de Ciência da Computação e dos cursos de Engenharias da UFPel conforme apresentado na Tabela 1. O nome atual do curso de Ciência da Computação foi definido em 2001, pois inicialmente tinha o nome de Bacharelado em Informática. O curso foi aprovado, pelo Conselho Universitário da UFPel em 1992 e iniciou as atividades em 07/03/1994. Outro curso mais antigo analisado é o curso de Engenharia Agrícola que teve suas atividades iniciadas em 07/01/1973. Todos os demais cursos, com exceção da Engenharia Industrial Madeireira, de Engenharia da UFPel foram criados no contexto do Programa de Apoio a Planos de Reestruturação e Expansão das Universidades Federais (REUNI) instituído pelo Governo Federal do Brasil através do Decreto 6.096, de 24 de abril de 2007 (DECRETO REUNI, 2021).

Os currículos dos cursos tanto de Ciência da Computação como das Engenharias da UFPel compreendem um conjunto de disciplinas obrigatórias, disciplinas optativas e atividades complementares. O curso de Ciência da Computação é integral (com aula em dois turnos) estruturado em 3200 horas em 8 semestres. Nas versões atuais dos currículos os alunos da Engenharias precisam cumprir o número de horas mínima apresentado na Tabela 1 em 10 semestres de forma integral também. A única exceção é o curso de Engenharia de Produção que é um curso noturno estruturado em 11 semestres. Além disso, todos os cursos envolvidos neste trabalho são cursos semestrais.

O processo de avaliação do ensino e de aprendizagem segue as normas e os procedimentos estabelecidos pelo Conselho Universitário e Conselho Coordenador

Tabela 1 – Cursos de Engenharia da UFPel utilizados no trabalho.

Cód. do curso	Nome	Data de Início	CH (Horas)
6400	Engenharia Hídrica	01/03/2009	3680
6500	Engenharia de Petróleo	01/03/2009	3784
6700	Engenharia de Produção	22/03/2010	3609
6100	Engenharia de Materiais	09/03/2009	3750
6300	Engenharia Civil	01/03/2009	3924
6200	Engenharia Ambiental e Sanitária	01/03/2009	4103
5200	Engenharia Industrial Madeireira	18/10/2005	3890
7000	Engenharia Eletrônica	01/08/2010	3627
5600	Engenharia Geológica	10/07/2008	4369
3910	Engenharia de Computação	22/03/2010	3427
6900	Engenharia de Controle e Automação	01/08/2010	3750
700	Engenharia Agrícola	07/01/1973	4321

do Ensino, Pesquisa e da Extensão da UFPel. Onde a aprovação em cada disciplina é feita a cada semestre e depende da frequência do aluno, que precisa de pelo menos 75% em aulas teóricas e 75% em aulas práticas. A média das verificações constitui a nota semestral, considerando-se aprovado o aluno que obtiver nota semestral igual ou superior a 7.

O aluno que ficar com a média semestral inferior a 7 e igual ou superior a 3 tem a possibilidade de fazer um exame final, que trata de toda a matéria apresentada no período. A média final do aluno no caso do exame é resultante da divisão por 2 (dois) da soma da nota semestral com a do exame. O aluno que obtiver média semestral inferior a 3 será reprovado, sem a possibilidade de fazer o exame.

As condições de aprovação para alunos que fizerem exame são diferentes. É considerado aprovado na disciplina o aluno que fizer exame e obtiver média igual ou superior a 5.

O curso ainda prevê a possibilidade de parte da carga horária de cada disciplina ser cumprida de forma semipresencial até o limite de 20% da carga horária total. E também pode ser ofertada disciplina integralmente à distância, a critério do Colegiado de Curso. O total de dessa atividade não pode ultrapassar 20% da carga horária total do Curso, além de fazer uso das TICs (Tecnologias da Informação e Comunicação) conforme legislação em vigor.

Os artigos 154, 155 e 156, do Regulamento do Ensino de Graduação na UFPel (UFPEL, 2018), tratam da perda do vínculo institucional do aluno.

O artigo 154 lista os casos em que o aluno poderá perder o vínculo com a instituição, que são: não confirmar a matrícula junto ao colegiado do curso; sofrer sanção em decorrência de processo administrativo disciplinar; e não realizar matrícula no mínimo de créditos.

Já o artigo 155 trata dos casos em que o aluno perderá o vínculo institucional, que são conforme os abaixo:

I – ingressar nas vagas reservadas, conforme legislação vigente, e não cumprir as etapas previstas no edital;

II – solicitar o cancelamento de sua matrícula junto a CRA, podendo ser feito a qualquer tempo;

III – descumprir protocolos de convênios;

IV – decisão judicial;

V – falecimento;

VI – transferência para outra instituição de ensino superior;

VII – realizar troca de curso através de reopção;

VIII – abandonar o curso;

IX – não integralizar o curso dentro do tempo máximo estabelecido pelo COCEPE[1].

A evasão de cursos é um problema geral, independendo da área, e na Ciência da Computação não é diferente.

2.2 Evasão Escolar

A evasão é um problema complexo onde governos e instituições demonstram uma preocupação em reduzir as taxas de evasão das instituições publicas (MANHÃES et al., 2011). Para Silva filho et al. (2007) a evasão é fonte de ociosidade de professores, funcionários, equipamentos e espaço físico, isso tanto para o setor privado quanto para o setor público.

Programas como o REUNI e SISU trouxeram mudanças tanto para as universidades como também os perfis de alunos. Estas mudanças vêm sendo estudadas em trabalhos como o de Lima brito (2013), que faz uma análise dos limites na ampliação de vagas e redução da evasão da implementação do REUNI na UnB de 2008 a 2011.

Geralmente a evasão pode ser classificada em evasão de curso, da instituição e do sistema de ensino. Segundo um relatório apresentado pela Andifes et al. (1996) a evasão do curso é a saída definitiva do aluno de seu curso de origem, sem ter concluído, por motivos como: abandono, desistência, jubilamento, mudança de cursos, transferência e outros. Já a evasão da instituição é o desligamento da IES na qual o

[1] Conselho Coordenador do Ensino, da Pesquisa e da Extensão (Cocepe)

aluno estava matriculado. E a evasão do sistema é quando o estudante abandona de forma definitiva a ensino superior.

No trabalho de Silva filho et al. (2007) a evasão é analisada sob duas abordagens similares, evasão anual/semestral média e evasão total. A primeira mede o percentual de alunos matriculados em um sistema de ensino, IES ou curso que não se matriculou nem se formou no ano/semestre seguinte. Já o segundo mede o número de alunos que entraram em um sistema de ensino, IES ou curso que não colou grau no período de integralização curricular máximo. **Nesta pesquisa será estudado a evasão total de curso, que considera o período máximo de integralização curricular**.

Os trabalhos de predição de evasão que utilizam técnicas de MDE geralmente são divididos nos que utilizam dados invariáveis no tempo (dados socioeconômicos, demográficos, dentre outros) e nos que utilizam dados variantes no tempo (notas em disciplinas, presença, etc.). Lykourentzou et al. (2009) mostra que modelos que utilizam dados invariantes no tempo trazem precisões inferiores quando comparados com os modelos que utilizam dados variantes no tempo. Neste trabalho estes dois tipos de dados serão combinados.

2.3 Cross Industry Standard Process for Data Mining

Nesta seção será apresentada a metodologia que norteou o desenvolvimento desta pesquisa. A metodologia *Cross Industry Standard Process for Data Mining* CRISP-DM que tem como principal objetivo fornecer uma direção para conduzir o processo de KDD (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). A Figura 2 apresenta o ciclo de vida dividido em 6 fases da metodologia CRISP-DM. A seguir será apresentado uma breve descrição das seis fases da metodologia:

(a) **Compreensão do negócio:** Esta é a fase onde se deve identificar o problema a ser resolvido. Esta fase compreende também uma descrição do *background*, dos objetivos e também dos critérios de sucesso (VIGLIONI, 2007). Esta fase não será descrita, pode ser observada em maiores detalhes no Capítulo 2.

(b) **Compreensão dos dados:** É a fase responsável por fazer a coleta dos dados e a análise exploratória de dados (AED). Esta fase tem que dizer como os dados foram adquiridos, qual o seu formato, qual foi a quantidade de dados, descrever cada atributo selecionado, fazer visualizações dos dados e além disso qualquer informação pertinente aos dados.

(c) **Preparação dos dados:** Compreende as atividades de pré-processamento dos dados para a próxima fase. Normalmente se faz a seleção, limpeza, formatação dos dados, e ainda são gerados novos atributos derivados dos atributos existentes.

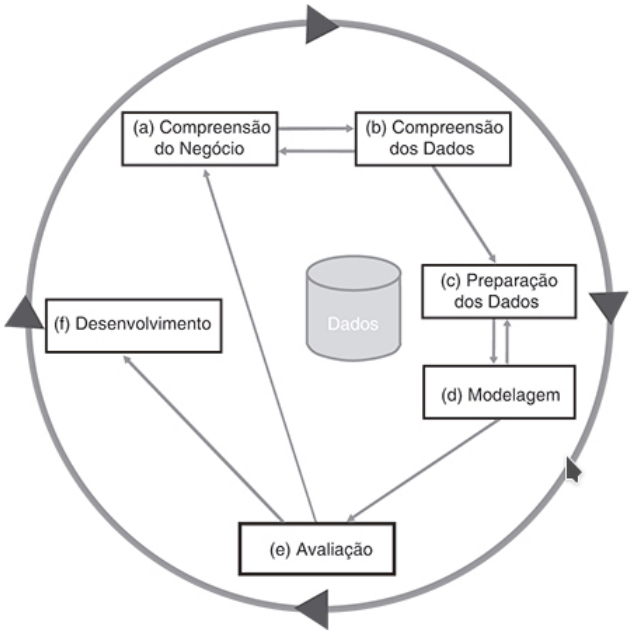

Figura 2 – Fases do modelo de referência CRISP-DM. Adapitado de Shearer (2000). Fonte: (GOLDSCHMIDT; BEZERRA; PASSOS, 2015).

(d) **Modelagem:** Corresponde a fase de aplicação dos algoritmos de mineração de dados selecionados sobre os dados preparados. É a etapa de mineração de dados do processo de KDD (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Nessa fase é criado um modelo para testar a sua qualidade e validade. É comum usar a taxa de erro como medida de qualidade do modelo em aprendizado supervisionado (VIGLIONI, 2007).

(e) **Avaliação:** Consiste em avaliar o modelo gerado, examinando os passos seguidos e validando se realmente foram alcançados os objetivos elencados na fase de compreensão do negócio (VIGLIONI, 2007). A partir da avaliação é possível propor revisões das fases anteriores e redefinir os próximos passos (GOLDSCHMIDT; BEZERRA; PASSOS, 2015).

(f) **Desenvolvimento:** É a fase onde se faz o planejamento e acompanhamento a serem realizadas com o modelo gerado pelas fases anteriores (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Esta fase não faz parte do escopo deste trabalho.

2.4 Mineração de Dados Educacionais

A definição de MDE dada por Costa et al. (2012) é de que é uma área que busca desenvolver ou adaptar métodos de KDD para resolver problemas do contexto educacional. Com esses métodos busca-se entender melhor o estudante em todo o seu processo de aprendizagem. Ainda segundo os autores em MDE há a necessidade

de adaptar os algoritmos e métodos de MD para tratar características inerentes aos dados no contexto educacional. Uma dessas características está relacionada a falta de padronização dos dados que acaba demandando um tempo maior na etapa de pré-processamento (BAKER; ISOTANI; CARVALHO, 2011).

O termo Mineração de Dados Educacionais vem do inglês *Educational Dada Mining* (EDM) e foi citado pela primeira vez no Workshop sobre Mineração de Dados Educacionais. Este workshop foi parte da XX Conferência Nacional de Inteligência Artificial, que aconteceu em Pittsburgh nos Estados Unidos em 2005.

Depois deste workshop houve outros em 2006 e 2007, até que Montreal, Canadá, hospedou a primeira conferência de MDE (*First International Conference on Educational Data Mining*). O evento se consolidou e ganhou regularidade anual e hoje a área de MDE é consolidada globalmente.

Baker; Isotani; Carvalho (2011) e Romero; Ventura (2013) categorizam os métodos de MDE em predição, agrupamento e mineração de relações.

2.4.1 Predição

O objetivo da previsão é descobrir um valor desconhecido de atributos que descrevem um aluno (ROMERO; VENTURA, 2013). Baker et al. (2010) divide a predição em 3 tipos, classificação, regressão e estimação de densidade.

Na classificação o conjunto de dados é dividido em dois grupos um é o atributo-alvo, ou seja, o atributo que deverá ser feito a previsão do valor e o outro são atributos que descrevem o aluno, geralmente é chamado de atributos previsores. Na classificação o atributo-alvo pode ser categórico ou discreto (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Atributos categóricos são aqueles que podem ser associados a categorias, por exemplo, estado civil do aluno, e atributos numéricos discretos são aqueles que podem representar um número finito de valores como por exemplo a idade dos alunos. A análise de regressão encontra a relação entre uma variável dependente e uma ou mais variáveis independentes (ROMERO; VENTURA, 2013). No caso deste trabalho a variável dependente é o atributo evasão que vai ser determinada usando os algoritmos com as variáveis independentes que são os atributos do conjunto de dados. A estimação de densidade é pouco usada em MDE em razão da falta de independência estatística em dados educacionais (BAKER; ISOTANI; CARVALHO, 2011).

2.4.2 Agrupamento

O objetivo do agrupamento é buscar dados que se agrupem naturalmente, e com isso classificar os dados em diferentes grupos ou categorias (BAKER; ISOTANI; CARVALHO, 2011). Essa técnica tenta fazer o agrupamento automático dos dados através dos graus de semelhança entre os grupos de atributos. Um exemplo é achar grupos de alunos para investigar as diferenças e similaridades entre alunos.

2.4.3 Mineração de relações

A tarefa de mineração de relações consiste na tentativa de aprender quais das variáveis está mais fortemente ligada a uma determinada variável, essa variável pode ser conhecida e importante, ou pode ter relação com outra variável presente no conjunto de dados (BAKER et al., 2010). O objetivo é encontrar alguma possível relação entre as variáveis do banco de dados. Por exemplo, ao analisar um conjunto de dados seria possível identificar uma regra que faz a associação entre a variável “objetivo do aluno”, uma variável binária que pode ter os valores alcançado ou não alcançado, e uma outra variável binária “pedir ajudar ao professor” que pode ter os valores sim ou não. Neste contexto, se o aluno tem como objetivo aprender geometria, mas está com dificuldade (i.e. a variável objetivo do aluno tem valor não alcançado), então é provável que ele peça ajuda do professor (i.e. a variável pedir ajuda ao professor tem valor positivo).

2.5 Algoritmos

As técnicas de Mineração de Dados utilizada neste trabalho são conhecidas como classificação. E os classificadores utilizados neste trabalho são: árvore de decisão, redes bayesianas, redes neurais e regressão logística. Que serão descritos no decorrer da Seção.

2.5.1 Árvore de decisão

Árvore de decisão é um modelo representado por nós e ramos que é parecido com uma árvore (HAN; PEI; KAMBER, 2011). A árvore começa pelo nó raiz que fica no topo da árvore. Cada nó da árvore é um nó de decisão, ou seja, cada nó contém um teste sobre uma variável independente e o resultado desse teste forma o ramo da árvore. Os nós folhas representam os valores preditos para a variável dependente. Ou seja, um exemplo é classificado de modo que ele parte do nó raiz até atingir uma folha da árvore, que vai corresponder a um rótulo da classe (ROMERO; VENTURA, 2013). Uma das principais vantagens de utilizar árvores de decisão é que elas são simples e tendem a facilitar para uma estrutura de decisão. Em geral o algoritmo de podagem percorre a arvore em profundidade e para cada nó ele calcula o erro do nó e a soma dos erros dos nós descontentes do nó corrente. Se essa soma o erro do nó for igual ou menor que o erro dos nós descendentes o algoritmo transforma o nó atual em um nó folha.

É possível que o modelo de árvore de decisão sofra com o sobre-ajuste (*overfit*). Usar métodos de poda de uma árvore pode resolver esse problema (HAN; PEI; KAMBER, 2011). A potagem da arvore pode ser pré ou pós, a primeira é feita antes da arvore ser gerada e a segunda é quando a arvore já foi gerada.

2.5.2 Naive bayes

O algoritmo Naive Bayes é um classificador probabilístico baseado no Teorema de Bayes. O algoritmo ganhou notoriedade na área de Aprendizado de Máquina para classificar textos baseado na frequência das palavras usadas. Um exemplo da utilidade desse modelo é na classificação de e-mails como SPAM ou não SPAM.

A ideia do Naive Bayes é utilizar o teorema de Bayes, para determinar a qual classe pertence uma observação, tupla ou registro. O teorema de Bayes é representado pela a equação abaixo que mostra como calcular a probabilidade da classe A dado uma observação B.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

como $P(x_1, ..., x_m|C_k) = P(x_1|C_k) * ... * P(x_n|C_k)$, pode-se generalizar que $P(C_k|x_1, ..., x_m) \cong P(x_1|C_k) * ... * P(x_n|C_k) * P(C_k)$, onde x_i é um atributo do conjunto de dados e C_k representa uma classe sendo avaliada. No final a saída é dada pela *Maximum a Posteriori* representada pela equação abaixo.

$$y = argmax_k \; P(C_k) * \prod P(x_i|C_k)$$

Um exemplo prático seria o de determinar se um funcionário irá ou não sair da empresa. No exemplo a observação “reclamou” vai servir para determinar se o funcionário irá sair ou não.

No treinamento do algoritmo Naive Bayes será gerado a probabilidade para cada possibilidade da observação e da classe. Para o exemplo seria como segue abaixo.

$$P(reclamou \mid saiu) = \frac{Quantidade(reclamou \; E \; saiu)}{Quantidade(saiu)}$$

$$P(nãoreclamou \mid saiu) = \frac{Quantidade(não \, reclamou \; E \; saiu)}{Quantidade(saiu)}$$

$$P(reclamou \mid nãosaiu) = \frac{Quantidade(reclamou \; E \; não \; saiu)}{Quantidade(nãosaiu)}$$

$$P(nãoreclamou \mid nãosaiu) = \frac{Quantidade(não \, reclamou \; E \; não \; saiu)}{Quantidade(não \; saiu)}$$

Depois que o algoritmo calculou todas as probabilidades ele já está treinado. Usando essas probabilidades dentro do Teorema de Bayes é possível fazer uma classificação para uma nova entrada. Se a nova entrada fosse “Reclamou” o algoritmo calcularia as duas probabilidades abaixo.

$$P(reclamou \mid saiu) = \frac{Quantidade(reclamou\ E\ saiu)}{Quantidade(saiu)}$$

$$P(reclamou \mid nãosaiu) = \frac{Quantidade(reclamou\ E\ não\ saiu)}{Quantidade(nãosaiu)}$$

E a classe que retornasse maior probabilidade entre as duas seria escolhida. Este exemplo possui apenas uma entrada, mas em problemas reais pode existir diversas entradas.

Neste trabalho será utilizado a biblioteca Scikit Learn, que na versão atual implementa 3 tipos de Naive Bayes Gaussian, Multinomial e Bernoulli. Nesta pesquisa será utilizada a Gaussian, pois todos os atributos do trabalho serão convertidos para número.

2.5.3 Regressão logística

A área de aprendizagem de máquina vem resgatando alguns modelos estatísticos e a regressão logística é um deles. Este modelo é análogo a um modelo de regressão linear, mas para resolver problemas de classificação. Este tipo de problema aparece quando é preciso categorizar alguma variável por classes (HOSMER JR; LEMESHOW; STURDIVANT, 2013).

Para implementar um modelo de regressão logística é preciso adicionar uma função de achatamento após transformação linear. Essa função de achatamento geralmente é uma função logística ou sigmoide. Isso vai fazer com que o modelo converta a transformação linear em uma probabilidade, de forma que quanto maior o valor maior é a probabilidade prevista. Abaixo segue a função de achatamento.

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

2.5.4 Redes neurais

As Redes neurais artificiais foram definidas para possuírem a característica básica de um neurônio biológico. Basicamente um neurônio de uma rede neural é um componente que faz a soma ponderada de várias entradas, aplica a uma função de ativação que passa o resultado a frente. Cada entrada é multiplicada por um peso e todo produto de entradas serão somados para calcular o nível de ativação do neurônio. O resultado do neurônio é processado por uma função de ativação. A Figura 3 é um exemplo de um neurônio artificial descrito acima.

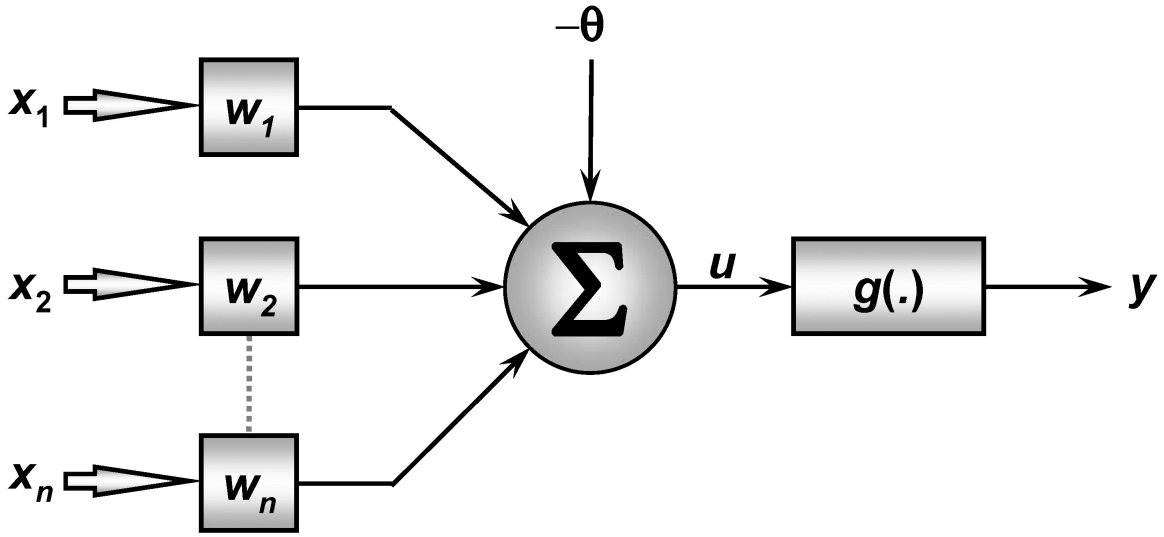

Figura 3 – Exemplo de um neurônio artificial.

Tanto a regressão logística quanto a regressão linear são tipos de *perceptrons*[2], que se diferem apenas quanto a função de ativação. Enquanto a regressão logística usa uma sigmoide como *step function*[3], a regressão linear usa a função identidade $f(x) = x$. Em uma rede neural uma das funções mais utilizadas é a unidade linear retificada (ReLU), que é uma função não linear diferencial.

Quando se usa vários neurônios em paralelo é que se tem uma rede neural. Esses neurônios em paralelo formam a camada oculta, onde cada saída é uma entrada para o neurônio da próxima camada. Quando se tem mais de uma camada de neurônios essa rede passa a ser chamada de rede neural profunda.

2.6 Métricas de avaliação

Esta Seção mostrará as métricas utilizadas para avaliar esta pesquisa. Foram utilizadas 4 métricas: acurácia, precisão, revocação (*Recall*) e AUC, do inglês *Area Under the ROC Curve*.

A Figura 4 apresenta a orientação utilizada para gerar a matriz de confusão dos algoritmos e mostra a posição do *True Negative* (TN), *False Positive* (FP), *False Negative* (FN) e *True Positive* (TP) na matriz.

A acurácia é uma das métricas de modelos de ciência de dados mais utilizadas. Em geral ela mostra o quanto o modelo está acertando. A fórmula abaixo mostra o cálculo da acurácia utilizando os termos da matriz de confusão.

[2] *Perceptron* é uma especie de rede neural artificial que foi inventada em 1958 por Frank Rosenblatt no Cornell Aeronautical Laboratory.

[3] *Step function* também chamada de função de Heaviside ou função degrau, é uma função singular e descontínua com valor zero quando o seu argumento é negativo e um valor unitário quando o argumento é positivo.

		Valor predito	
		Não Evadiu	Evadiu
Valor real	Não evadiu	TN	FP
	Evadiu	FN	TP

Figura 4 – Categorias da matriz de confusão.

$$acurácia = \frac{TN + TP}{(TN + TP + FN + FP)}$$

No caso deste trabalho, utilizar a apenas a acurácia não é interessante, pois os dados estão desbalanceados. Nos dados obtidos do Cobalto se um modelo simples classificasse todos os alunos como evadidos esse modelo obteria 66,5% de acurácia, porque dos 744 alunos 495 estão em situação de evasão.

A precisão é boa para determinar quanto o custo do FP é alto. Ela normalmente é considerado em situações em que os FPs são mais importantes do que os FNs. Abaixo segue a fórmula da precisão.

$$precisão = \frac{TP}{(TP + FP)}$$

O Recall mostra a proporção de todos os valores positivos foi identificada corretamente. O Recall é a probabilidade do modelo predizer que o aluno evadiu dado que o aluno realmente evadiu. Ele geralmente é utilizado em situações contrárias a da precisão, ou seja, onde FNs são considerados mais problemáticos do que os FPs. O Recall é calculado da seguinte forma.

$$recall = \frac{TP}{(TP + FN)}$$

AUC vem do inglês *Area under the ROC Curve*, para poder explicar esta métrica é preciso falar da curva ROC antes.

A curva ROC (*Receiver Operating Characteristic*) é um gráfico que mostra o desempenho dos modelos de classificação. Ela representa True Positive Rate (TPR) versus False Positive Rate (FPR) em diferentes limiares de classificação. O TPR é equivalente ao recall e o FPR é definido como segue abaixo.

$$FPR = \frac{FP}{(FP + TN)}$$

Para calcular cada ponto da curva ROC, é possível utilizar um modelo de regressão logística modificando os limiares de classificação até gerar todos os pontos. Porém

existe um algoritmo que calcula a área sob a curva ROC chamado de AUC.

AUC calcula a área abaixo de todos os limiares de classificação possíveis da curva ROC. Quanto mais alto é o valor do AUC melhor é a capacidade do modelo de classificação em distinguir entre as classes positivas e negativas, no caso deste trabalho se o aluno evade ou não.

2.7 Trabalhos Relacionados

Nesta Seção serão apresentados os trabalhos relacionados a esta pesquisa. Será feita uma contextualização desses trabalhos para depois ser traçado um paralelo entre o que tem sido pesquisado e o que está sendo proposto neste trabalho. Para fazer essa contextualização foram selecionados três dos principais *surveys* que fizeram um grande levantamento do que estava sendo pesquisado na área entre 2010 e 2014, e alguns trabalhos que têm uma relação direta com o tema de pesquisa deste trabalho. Como critérios de seleção para os trabalhos foram os seguintes:

- Os trabalhos tinha que utilizar Mineração de dados educacionais para resolver o problema;
- O conjunto de dados tinha que envolver alunos de graduação presencial.

Manhães et al. (2011) provaram que é possível identificar alunos em risco de evasão através das primeiras notas semestrais dos alunos ingressantes. A base de dados utilizada no trabalho foi do sistema acadêmico da instituição e contou com alunos que cursaram Engenharia Civil na UFRJ de 1994 a 2005. O trabalho consistiu em três experimentos, onde foi modificado apenas a forma de treinamento dos algoritmos 10 fold *Cross-validation*, *train/test percentage split (data randomized)* e *supplied test set*, em que 2/3 para treinamento e o restante para teste. Cada experimento foi submetido a 10 algoritmos utilizados em trabalhos relacionados. Os experimentos alcançaram acurácia média entre 75% e 80%, além disso a predição incorreta de risco de evasão foi considerada como erro grave do classificador.

Júnior; Noronha; Kaestner (2014) apresentaram uma abordagem de extração do conhecimento combinado com séries temporais e mineração de dados. Os autores utilizaram as séries temporais para fazer a seleção dos atributos e isso facilitou o trabalho de levantamento e agregação dos dados. Estes dados foram submetidos a 5 algoritmos de classificação (J48, Multilayer perceptron, SMO, IBk, Naive Bayes) e o melhor desempenho foi do J48 com uma acurácia de aproximadamente 80%. Os autores também chegaram a conclusões importantes, tais como, que 80,6% dos alunos que retiram livros na biblioteca e tem mais de 18 anos estão ativos no curso, os autores afirmam que isso é um indício da correlação entre estes dois atributos. Eles destacam que isso é uma correlação entre os dados considerados, e não uma

relação de causa e efeito. Os resultados indicam que existe uma correlação entre a desistência de um aluno e o não empréstimo de livros.

No trabalho de Rigo et al. (2014) é apresentado um estudo de fatores envolvidos no fenômeno de evasão escolar e descrevem a utilização de um sistema para MDE e LA durante 18 meses em cursos de graduação na modalidade de Educação a Distância. Ao todo foram executados 4 experimentos onde foram acompanhados 603, 250, 925 e 713 alunos. Para cada estudo de caso que trata o trabalho foi utilizada a técnica *RNA Multilayer Perceptron*. O melhor resultado com relação a predição da evasão foi no experimento 4 onde a melhor média de acertos foi de 83,7%.

Kantorski et al. (2016) propõem prever a evasão de cursos de graduação presenciais em universidades públicas. Foram extraídos dados pessoais, acadêmicos, sociais e econômicos de alunos e construídos modelos de predição através de algoritmos de aprendizagem de máquina. Os autores destacam que a vantagem da proposta foi a otimização dos resultados pela combinação de vários modelos de mineração de dados para gerar uma única predição e isso permite um resultado mais abrangente. Nos testes alcançaram uma acurácia de 98% e mais de 70% de sucesso na predição de alunos que evadiram do curso.

Pascoal et al. (2016) apresentaram uma abordagem para fazer o diagnóstico da evasão de alunos, a partir de dados socioeconômicos e acadêmicos. Os autores coletaram dados de 241 alunos do curso de Ciência da Computação da UFPB e submeteram eles a um modelo de Naive Bayes treinado pela validação cruzada de 10 folds. Os autores concluíram que as informações acadêmicas dos alunos têm um maior impacto na predição do que os dados socioeconômicos. O modelo apresentou uma acurácia geral de 85,48% que demonstrou a viabilidade do método.

No trabalho de Barbosa; Santos; Pordeus (2017) foi apresentado uma abordagem para o problema da evasão, onde os autores classificam os alunos em 3 grupos: os que vão se formar, os que vão abandonar e o que os autores classificaram como incertos sobre o seu futuro. Os experimentos foram feitos com 5 classificadores, que classificaram os alunos em dois grupos e rejeitaram aqueles que não tinham clareza a que grupo deveriam pertencer. Para fazer a classificação os autores escolheram a seguinte estratégia, eles treinavam dois classificadores e os dados de entrada passavam pelos dois. Quando ambos os classificadores concordavam na resposta a classe correspondente era reproduzida, caso contrário, o teste era rejeitado. A abordagem dos autores foi validada pela curva de rejeição, que é uma matriz que compara a acurácia versus a taxa de rejeição. Os autores obtiveram resultados significativos em seu trabalho com acurácia média de até 88,32%.

Em Paz; Cazella (2017) os autores buscaram identificar os perfis de alunos com tendência a evadir de uma universidade comunitária utilizando mineração de dados educacionais. A pesquisa deles partiu de 3 hipóteses uma de que alunos de semes-

tres iniciais possuem maior tendência a evasão, outra de que alunos que residem em municípios fora do campus onde estudam tem tendência a evasão e a última de que os incentivos fornecidos podem relação com a evasão. Os autores coletaram dados de 4697 alunos dos cursos de graduação da universidade do segundo semestre de 2016 de todos os campi dessa. No total foram coletados 322 atributos de 12 Tabelas, onde foram selecionados 6 atributos fora o atributo classe, que indica se o aluno evadiu ou não. Esses dados foram submetidos ao algoritmo J48 que foi treinado utilizando validação cruzada com K igual a 10. Os autores relatam que obtiveram resultados acima de 90% na predição da evasão. Também confirmam a percepção de que o incentivo e o currículo dos alunos estão diretamente ligados a tendência de evasão. Por outro lado, não se comprovou a hipótese de que alunos que não moram no município do campus tendem a evadir mais.

Lanes; Alcântara (2018) apresentaram um estudo que visa identificar estudantes que apresentam risco de evasão a partir do seu primeiro ano no curso de graduação. Os experimentos foram realizados com informações extraídas do sistema acadêmico da FURG. O conjunto de dados contou com 916 registros de 12 cursos de graduação de áreas distintas. Os dados foram discretizados e categorizados para gerar o *dataset* final. Foi utilizada a ferramenta Weka e aplicado o algoritmo J48 para processar o *dataset* e obter a árvore de decisão. Os resultados mostram que os potenciais alunos em risco de evadir podem ser identificados com acurácia de 90,7% usando o algoritmo J48.

Beltran et al. (2019) desenvolveram uma plataforma de aprendizado de máquina, que utiliza os dados socio-econômicos e acadêmicos de alunos do ensino superior do Peru, para classificar os alunos que apresentam risco de evadir. Os autores usaram a correlação de Pearson para gerar uma segunda versão da base de dados com 19 atributos, a primeira versão possuía 36 atributos. Em seguida usaram 3 algoritmos de agrupamento (Expectation Maximization, Hierárquico Aglomerativo e k-Means) nas duas bases, e foram escolhidas as 7 melhores partições de cada base que gerou um total de 14 combinações das bases. No final os autores usaram as 16 bases para treinar os algoritmos AdaBoost, Baggging, IBK, J48, MLP e Naive Bayes. Os métodos foram avaliados usando o F-measure. Os modelos foram treinados com 2.696 instâncias e treinados em 3.393 instâncias de alunos que ainda estavam cursando. Os autores relataram que obtiveram resultados acima de 90%.

Carrano et al. (2019) apresentam uma metodologia que combina diferentes técnicas de mineração de dados, para gerar modelos de classificação que auxiliem aos gestores os alunos com tendência a evadir e identificam quais são os atributos mais importantes relacionados a evasão de alunos. Os autores avaliaram a metodologia utilizando os dados de todos os alunos da graduação presencial da UFSJ. Em sua análise identificaram que o desempenho do aluno, a satisfação e a assiduidade são

informações fundamentais para determinar se o aluno vai evadir ou não. Além disso, essa conclusão elenca o gestor institucional como ator fundamental no processo de combate a evasão. Os autores obtiveram resultados da área da curva ROC por volta de 0,80, que pode variar de 0,0 a 1,0.

A Tabela 2 mostra a lista de todos os trabalhos citados acima. Todos os trabalhos têm o domínio da aplicação e técnica utilizadas em comum, que são dados retirados de sistemas acadêmicos e utilizam mineração de dados educacionais respectivamente.

2.8 Considerações do Capítulo

Foram selecionados trabalhos diretamente relacionados a predição da evasão escolar em sistemas acadêmicos, mas também foram analisados parcialmente trabalho não relacionados diretamente com o objetivo principal da pesquisa. Outros trabalhos que usaram dados retirados de Ambientes Virtuais de Aprendizagem (AVA) como Braz et al. (2019), Burlamaqui et al. (2017), Detoni; Cechinel; Araújo (2015), Fernandes et al. (2017), Queiroga; Cechinel; Araújo (2015), Ramos et al. (2018), Santos; Siebra; Oliveira (2014) e Silva et al. (2015). Além de outros trabalhos que não foram com alunos do ensino superior tais como Bezerra et al. (2016), Calixto; Segundo; Gusmão (2017) e Sales et al. (2019), dentre outros.

A principal diferença deste trabalho é a aplicação de técnicas de visualização e MDE em cima de dados reais de alunos do curso presencial de Ciência da Computação e das Engenharias da UFPel. Outro ponto a ser levado em consideração é que foi utilizado preliminarmente apenas um curso para tentar não generalizar e evitar problemas como o paradoxo de Simpson. Após o modelo foi aplicado, em carácter exploratório, em um conjunto maior de cursos. Também foi feito uma caracterização da evasão em cima de dados de dezenove anos do curso para Ciência da Computação e nove anos para as Engenharias. Por fim, diferentemente dos outros trabalhos foram utilizados dados apenas dos três primeiros semestres do curso.

Tabela 2 – Lista de trabalhos relacionados a pesquisa.

Trabalho	Volume de dados	Dimensões analisadas	Técnicas
Manhães et al, 2011	Periodo: 1994 a 2005. Registros: 887 alunos.	Foram selecionados 12 atributos	Mineração de Dados Educacionais
Junior et al, 2014	Periodo: 2012/1 a 2013/2. Registros: 3605 alunos	Foram utilizados 11 atributos	Mineração de Dados Educacionais: J48, Naive Bayes, SVM, Multilayer Perceptron e IBk.
Rigo et al, 2014	Periodo: 2012 e 2013. Registros: 2491 alunos.	Não fala sobre os atributos selecionados para o estudo.	Mineração de Dados Educacionais e LA: MLP com backpropagation.
Kantorski et al, 2016	Periodo: 2000 a 2015.	Foram selecionados 33 atributos	Mineração de Dados Educacionais
Pascoal et al, 2016	Periodo: 2001 a 2013.	Foram selecionados 22 atributos entre dados acadêmicos e dados socioeconômicos	Mineração de Dados Educacionais: Naive Bayes
Barbosa; Santo; Pordeus, 2017	Periodo: 2005 a 2016.	Foram selecionados 10 atributos	Mineração de Dados Educacionais: FNNRW, MLP, SVM, Naive Bayes, kNN
Paz et al, 2017	Periodo: 2016/2.	Foram selecionados 322 atributos.	Mineração de Dados Educacionais: J48
Lanes et al, 2018	Periodo: não informado. Registros: 916 alunos.	Foram selecionados 10 atributos	Mineração de Dados Educacionais: J48
Beltran et al, 2019	Periodo: 2010 e 2017	A base foi separada em V1 e V2. V1: 36 atributos e 6086 instâncias; V2: 19 atributos e 6086 instâncias.	Mineração de Dados Educacionais: AdaBoost, Bagging, IBK, J48, MLP e Naive Bayes.
Carrano et al, 2019	Periodo: 2010 e 2017.	Foram selecionados 77 atributos, 7 são informações pessoais, 31 informações acadêmicas e 39 informações socioeconômicas.	Mineração de Dados Educacionais: árvores de decisão

3 METODOLOGIA DA ABORDAGEM PROPOSTA

No decorrer desta Dissertação foram realizadas pesquisas exploratórias que serviram para definir a metodologia utilizada neste trabalho, e também serviram para publicações como Costa et al. (2020) e Costa; Primo; Mattos (2020).

Este Capítulo será dividido conforme as etapas da metodologia CRISP-DM.

3.1 Compreensão dos dados

A etapa de compreensão dos dados envolve, coletar, descrever, explorar e verificar a qualidade do seu dado. Nesta Seção será apresentada como foi realizada a coleta, descrição e a análise exploratória dos dados. Por serem tarefas manuais foi uma das etapas que demandaram mais tempo deste trabalho.

As ferramentas desta fase foram utilizadas em um sistema Debian Buster. Nesta fase foram utilizadas as seguintes ferramentas:

- Pgadmin3: foi utilizado para as consultas SQL;
- Mozilla Firefox: navegador utilizado para acessar a ferramenta de modelagem;
- Google Colab: ferramenta que cria um ambiente virtual que disponbiliza diversos recurso em Python;
- Python: linguagem de programação;
- Pandas: biblioteca do Python utilizada para manipulação de dados;
- seabor: biblioteca do Python utilizada para gerar os gráficos;

3.1.1 Coleta dos dados

Neste trabalho foram utilizados os dados de alunos do curso de Ciência da Computação e dos cursos de Engenharias registrados no Cobalto. O Cobalto é um sistema integrado de gestão que faz a gestão acadêmica da universidade. Este sistema possui, além dos seus próprios dados, os dados históricos de alunos importados do sistema anterior da IES, denominado GOL.

Os dados extraídos do sistema correspondem aos alunos do curso de Ciência da Computação que ingressaram entre os anos 2000 e 2018. Já os dados das engenharias corresponde aos alunos que ingressaram entre os anos de 2010 e 2018. Os dados sociais e acadêmicos dos alunos foram extraídos através de consultas diretas a base de dados do sistema acadêmico. Vale destacar que esses dados foram anonimizados, ou seja, foi utilizado um código de identificação fictício para cada aluno.

A Tabela 3 apresenta o número total de alunos para a situação final ou atual. A situação “Cursando” representa todos os alunos que ainda estão dentro do período de integralização curricular e não saíram do curso. Já a situação “Retido” são todos os alunos que já passaram do período de integralização curricular e não tem saída. As situações “Formado” e “Evadido” são alunos que já tem uma saída registrada.

Nesta Seção o termo matrícula será usado para identificar a matrícula em uma disciplina, ou seja, quando quiser falar que um aluno se matriculou em 6 disciplinas será falado que o aluno tem 6 matrículas. O total de matrículas de todos alunos do Curso de Ciência da Computação foi de 21.105 e já para os alunos dos Cursos de Engenharias foi de 122.668.

Tabela 3 – Resumo dos totais de dados coletados do curso de Ciência da Computação e dos cursos Engenharias

Cursos	Alunos				
	Total	Cursando	Evadido	Formado	Retido
Ciência da Computação	1514	286	786	330	112
Engenharias	6351	1466	3250	990	645

Nesta pesquisa foram utilizados os dados dos três primeiros semestres de alunos do curso de Ciência da Computação e Engenharias. A ideia de utilizar os dados dos três primeiros semestres é que quanto mais cedo identificar o aluno propenso a evadir maior é a chance de fazer um planejamento de ações para poder modificar essa situação. Trabalhos como o de Lanes; Alcântara (2018) tiveram exito usando esse mesmo tipo de abordagem.

3.1.2 Descrição dos dados

Nesta Seção serão apresentados os atributos retirados da base de dados. Para chegar no conjunto final de atributos foram feitas diversas tentativas. Abaixo segue a lista de todos os atributos utilizados neste trabalho e sua respectiva descrição:

A seguir serão a apresentados estudos e dados estatísticos de cada atributo. O atributo cod_aluno será desconsiderado, pois é apenas um identificador.

Tabela 4 – Relação de atributos retirados da base de dados.

Atributo	Descrição
cod_aluno	Identificador do aluno.
ano_ingresso e semestre_ingresso	Ano e semestre que o aluno ingressou no curso.
ano_saida e semestre_saida	Ano e semestre que o aluno saiu do curso.
aluno_situacao	Situação final do aluno no curso.
nota_final	Nota final do aluno em uma disciplina.
ano_disciplina e semestre_disciplina	Ano e semestre que o aluno cursou a disciplina.
disciplina_situacao	Situação do aluno na disciplina.
genero	Gênero do aluno.
dt_nascimento	Data de nascimento do aluno.
dt_ingresso	Data de ingresso no curso.
etnia	Etnia do aluno.
estado_civil	Estado civil do aluno.
tipo_ingresso	Forma de ingresso do aluno.
cota	Cota de ingresso do aluno.
flg_escola_publica	Aluno veio de escola pública.
flg_curso_superior	Aluno concluiu ensino superior anterior.
flg_benefício	Aluno recebeu auxílio.
naturalidade	Cidade natal do aluno.
flg_pri_sem	Indica que o aluno cursou a disciplina no primeiro semestre.
flg_seg_sem	Indica que o aluno cursou a disciplina no segundo semestre.
flg_ter_sem	Indica que o aluno cursou a disciplina no terceiro semestre.
nr_dis_pri_sem_cur	Número de disciplinas do primeiro semestre do currículo do aluno.
nr_dis_seg_sem_cur	Número de disciplinas do segundo semestre do currículo do aluno.
nr_dis_ter_sem_cur	Número de disciplinas do terceiro semestre do currículo do aluno.
tempo_total_curso	Número de semestres que o aluno cursou ou está cursando no curso.

3.1.2.1 Ano/Semestre de Ingresso

Os atributos ano_ingresso e semestre_ingresso correspondem, respectivamente, ao ano e semestre de ingresso do aluno no curso. A Figura 5 apresenta o número de alunos do curso de Ciência da Computação que ingressaram pelo ano e semestre de ingresso. O período que teve o maior número de ingressantes foi de 2009/1, onde ingressaram 60 alunos. O período com o menor número de ingressantes foi 2002/1. Os dados refletem o que está armazenado na base de dados. Contudo, mesmo assim, algumas inconsistências podem existir. Um exemplo é o caso de alunos que cancelam antes de realmente ingressar no curso. Até o ano 2014 o cadastro era realizado com o código de "cancelamento"(o mesmo do aluno que cancelou durante o curso) e após foi criado um código específico ("cancelamento de ingressante") para capturar este tipo de comportamento.

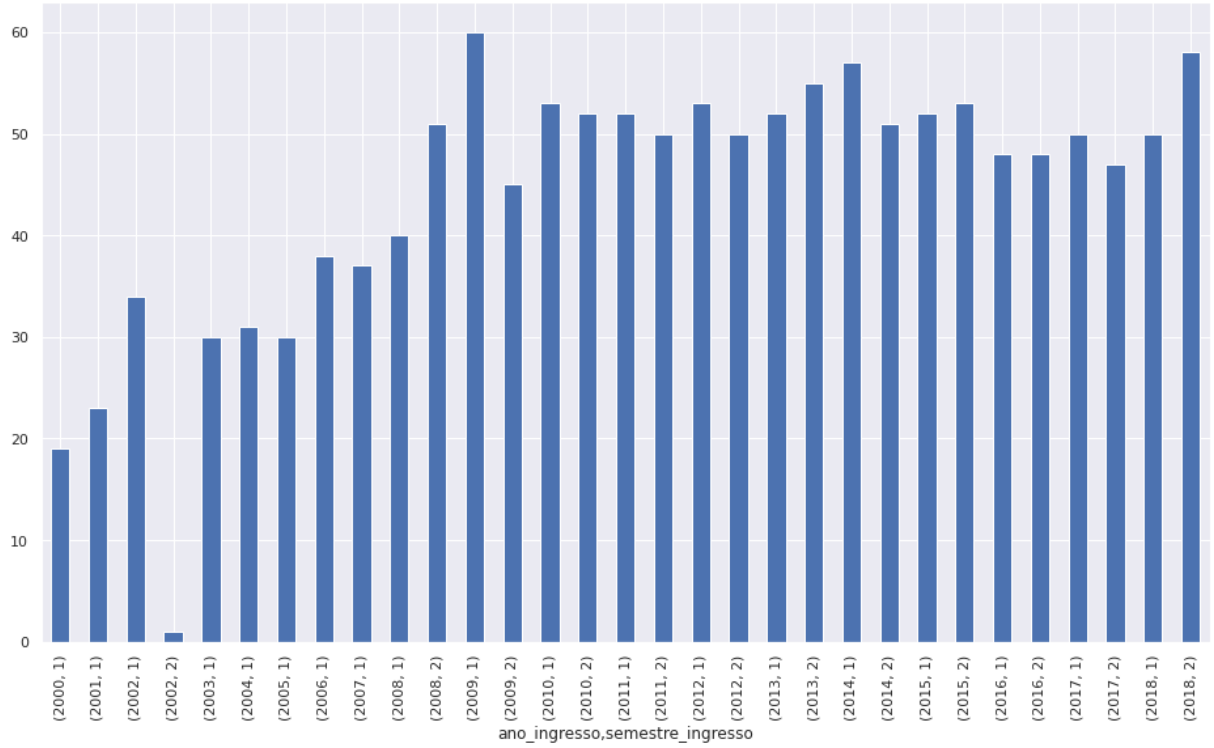

Figura 5 – Número de alunos pelo ano e semestre de ingresso no curso de Ciência da Computação.

A Figura 6 mostra o número de alunos dos cursos das engenharias que ingressaram pelo ano e semestre de ingresso no período objeto de estudo. O período que teve o maior número de ingressantes foi de 2016/1, onde ingressaram quase 600 alunos. Observa-se um ingresso atualmente em torno de 500 alunos anuais.

O atributo ano_ingresso pode assumir o valor entre 2000 e 2018 para o conjunto de dados da Ciência da Computação e entre 2010 e 2018 para o conjunto de dados das engenharias. Já o semestre_ingresso pode assumir os valores 1 ou 2, que representam respectivamente o primeiro e segundo semestre do ano, tanto para a Ciência

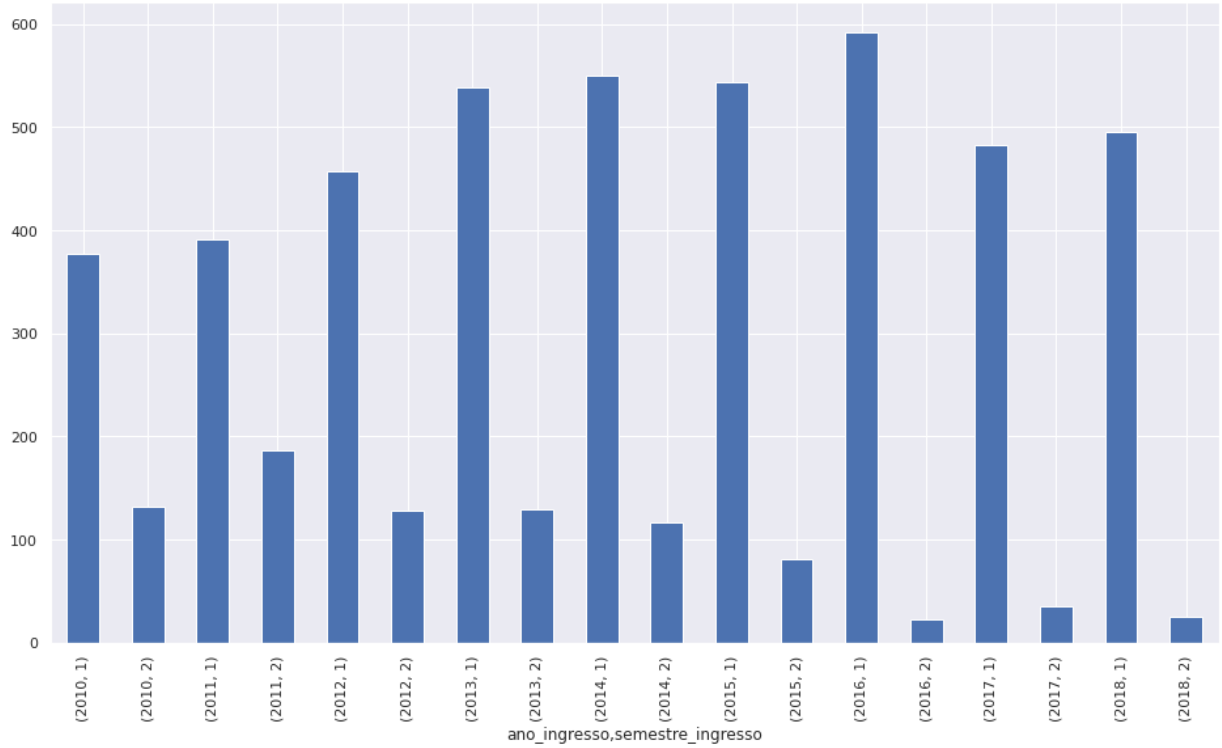

Figura 6 – Número de alunos pelo ano e semestre de ingresso nos cursos de engenharias.

da Computação como também para engenharias.

3.1.2.2 Ano/Semestre de Saída

Estes atributos são o ano e semestre que o aluno perdeu o vínculo com a UFPel. Estas saídas podem envolver Abandono, Cancelamento, Desligado, Desligado Lei nº 12.711 de 29/08/2012, Desligado Res.03/05, Falecido, Fim Mobilidade, Fim período, Formado, Jubilado, Matrícula não confirmada, Reopção, Reopção Compulsória, Trancamento e Transferido, e serão apresentadas na Seção 3.1.2.3.

A Figura 7 apresenta o ano e semestre que o aluno perdeu o vínculo com a instituição no Curso de Ciência da Computação. No casso desta figura são mostrada as saídas de alunos até 2019/2 pois é quando o aluno que ingressou em 2018/2 completa 3 semestres. O período com o maior número de saídas foi 2013/1 com mais de 70 alunos. Além disso eles podem assumir os mesmos valores que o ano_ingresso e semestre_ingresso apresentados na Seção 3.1.2.1. Estes atributos possuem 408 registros com valores nulos, estes são alunos que ainda estão no curso e por isso não tem um período de saída.

Já a Figura 8 mostra o ano e semestre que os alunos dos cursos de Engenharia perderam o vínculo com a instituição. No casso desta figura são mostrada as saídas de alunos até 2019/2 pois é quando o aluno que ingressou em 2018/2 completa 3 semestres. O período com o maior número de saídas foi 2018/2 com quase 350 alunos. A partir de 2014 a tendência de evasão foi sempre maior no segundo semestre

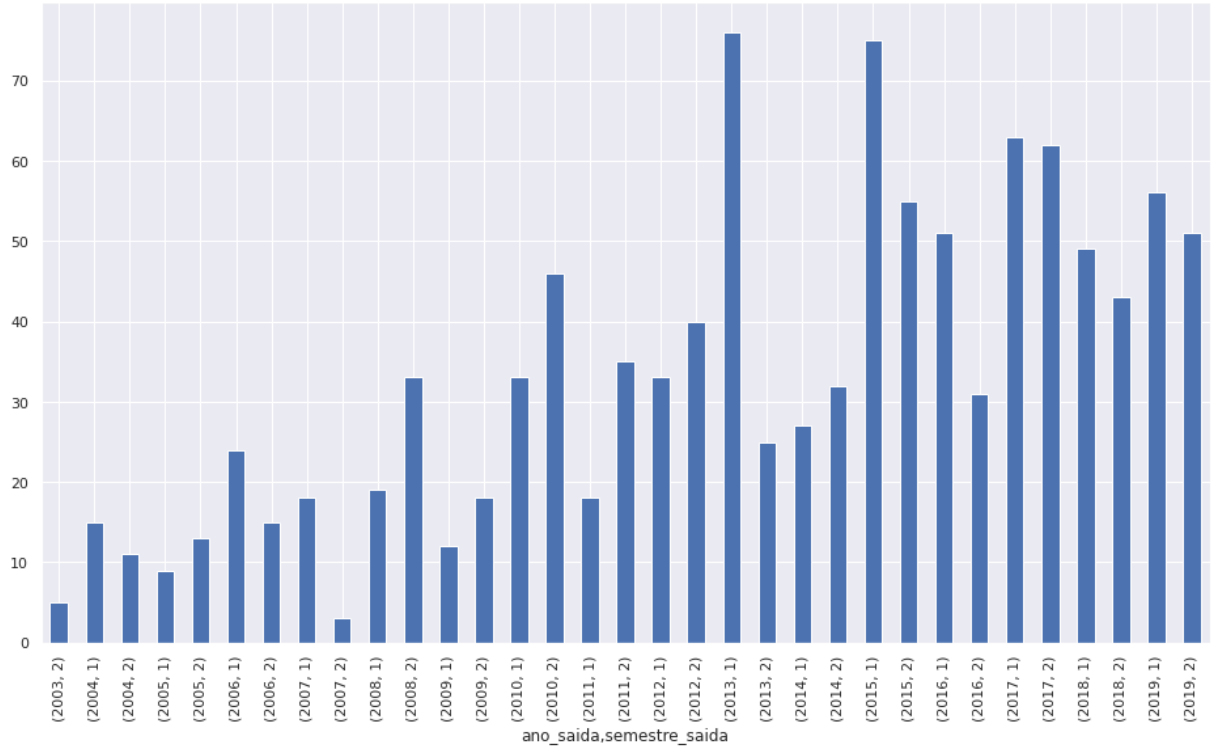

Figura 7 – Número de alunos pelo ano e semestre de saída no curso de Ciência da Computação.

do respectivo ano. Estes atributos possuem 1192 registros com valores nulos.

3.1.2.3 Tipo de Situação do Aluno

Este atributo é a situação do aluno no curso e serve para definir a situação final ou atual do aluno no Curso. A partir deste atributo será criado um novo atributo “evadiu”. Neste trabalho o atributo “evadiu” será a variável preditora (*predicted variable*) que identifica se o aluno evadiu ou não do curso. Para esta variável os modelos têm que prever se o aluno evadiu ou não.

A Tabela 5 mostra os totais de alunos para cada situação encontrada nos conjuntos de dados. Para a Ciência da Computação a maioria das ocorrências são de alunos em situação de “Abandono” que representa mais de 30% de alunos de toda a base de dados, enquanto alunos na situação “Formado” 21,80% do total. Para o curso de Ciência da Computação foram encontradas 15 situações diferentes para o conjunto de dados. Já nas engenharias esse comportamento muda com a maior parte dos alunos na situação de vínculo. O percentual de reopção nas engenharias é de 6,42%, enquanto na Ciência da Computação é de 3,30%.

Na Seção 3.1.2.2 foram encontrados 408 alunos da Ciência da Computação sem o ano e semestre de saída, mas o total de “Aluno com vínculo” é de 398 alunos. Isso acontece pois existem situações que são consideradas como saída do aluno e outras não. Alunos na situação de “Aluno com vínculo” e “Trancamento” são situações que

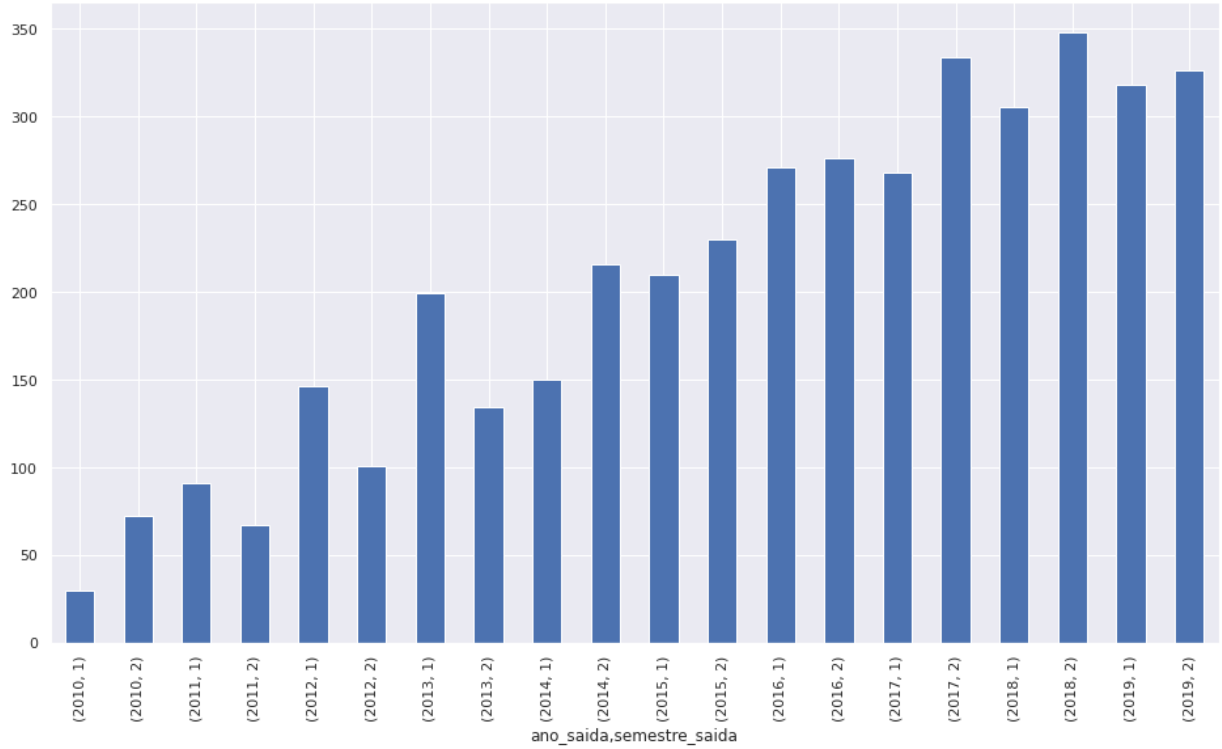

Figura 8 – Número de alunos pelo ano e semestre de saída nos cursos de engenharias.

não são consideradas como situações de saída do aluno. Todas as outras situações são consideradas como saída do aluno.

3.1.2.4 Nota Final

Este atributo é a nota que o aluno obteve em uma disciplina que se matriculou durante o curso. A nota final é um atributo numérico e pode assumir valores de 0 a 10.

A Tabela 6 apresenta os dados estatísticos deste atributo, onde *count* é o número total de registros sem valor nulo, *mean* é a média das notas de todos os alunos em cada disciplina, *std* é o desvio padrão dessas notas, *min* é a menor nota, 25% é o valor máximo do primeiro quartil, 50% é o valor máximo do segundo quartil, 75% é o valor máximo do terceiro quartil e *max* é a maior nota. No curso de Ciência da Computação foram coletadas 21.105 matrículas de alunos em disciplinas, dessas 18.500 (87,66%) tem valores registrado e 2.605 (12,34%) tem valores nulo. Já as engenharias têm 56.669 matrículas em disciplinas registradas no sistema. A média geral das notas no curso de Ciência da Computação foi de 5,44 com o desvio padrão de 3,36, enquanto a média nas engenharias foi de 7,66 com desvio de 1,29.

3.1.2.5 Número de disciplinas cursadas por Ano/Semetre

Estes atributos representam o ano e semestre que o aluno cursou a disciplina e serão usados para definir em qual semestre do curso o aluno fez a disciplina. O atributo ano pode assumir qualquer valor entre 2000 e 2018, e o atributo semestre

Tabela 5 – Número de alunos em cada situação do aluno encontrada no curso de Ciência da Computação e engenharias.

Situação	Ciência da Computação	Engenharias
Abandono	477	1483
Aluno com vínculo	398	2079
Cancelamento	202	1067
Desligado	1	2
Desligado Lei nº 12.711 de 29/08/2012	2	18
Desligado Res.03/05	4	15
Falecido	1	3
Fim Mobilidade	1	35
Fim período	2	51
Formado	330	990
Jubilado	3	1
Matrícula não confirmada	11	57
Reopção	50	408
Reopção Compulsória	0	1
Trancamento	10	32
Transferido	22	109
Totais	1514	6351

Tabela 6 – Dados estatísticos da nota final do aluno

Dado estatístico	Ciência da Computação	Engenharias
count	18500	78079
mean	5,44	5,56
std	3,36	3,37
min	0,00	0,00
25%	2,10	2,20
50%	7,00	7,00
75%	8,20	8,30
max	10,00	10,00

pode ser 1 para primeiro semestre e 2 para segundo semestre. Por exemplo, o aluno que cursou a disciplina de Cálculo A no ano de 2020 e no primeiro semestre terá ano_disciplina e semestre_disciplina com os valores 2020 e 1 respectivamente. A Figura 9 apresenta o total acumulado de disciplinas cursadas por semestre para o curso de Ciência da Computação. No casso desta figura são mostrada as disciplinas cursadas até 2019/2 pois é quando o aluno que ingressou em 2018/2 completa 3 semestres.

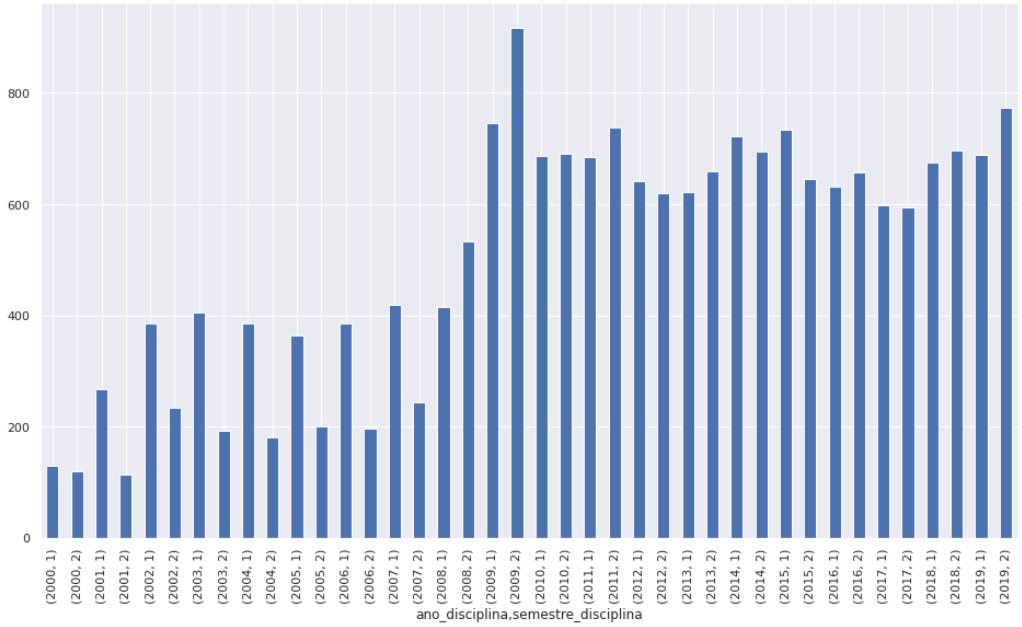

Figura 9 – Número de disciplinas cursadas pelo ano e semestre no curso de Ciência da Computação.

3.1.2.6 Situação da Disciplina

Atributo (disciplina_situacao) usado para registrar a situação da disciplina em um determinado semestre. Este atributo junto com o ano e semestre da disciplina vai ser usado para fazer a contagem de quantas disciplinas o aluno aprovou em um determinado semestre.

Algumas situações serão removidas na etapa de limpeza dos dados. Uma delas é a situação “DSP”, pois é uma disciplina que o aluno não cursou no vínculo atual dele. Outras situações carecem de melhor verificação futura. Este é o caso de situação como “MAT”, “FRC”, “UPD” e “N/I”.

A Tabela 7 apresenta a distribuição da situação das disciplinas encontrada para a Ciência da Computação. Ao todo são 10 situações diferentes para as matrículas encontradas no curso. A maior frequência é da situação Aprovados com 11.803 matrículas nesta situação.

Tabela 7 – Total de matrículas por situação para a Ciência da Computação

Situação da matrícula	Nr. de matrículas	Descrição
APR	11803	Aprovados
REPR	3532	Reprovados
INFR	3080	Infrequentes
MAT	1045	Matriculados
DSP	856	Dispensados
TRC	634	Trancados
CANC	108	Cancelados
FRQ	39	Frequentes
UPD	5	Aprovados usados para dispensa
N/I	3	Não informado
Total	21105	

3.1.2.7 Gênero

Este atributo representa o gênero do aluno e pode assumir dois valores “M” para masculino e “F” para feminino. A Tabela 8 mostra os totais de alunos para cada gênero para Ciência da Computação e Engenharias. Tanto na Ciência da computação quanto nas Engenharias a maior frequência é de homens, com 86,13% e 65,75% respectivamente.

Tabela 8 – Total de alunos por gênero

Atributo	Ciência da Computação	Engenharias
M	1304	4176
F	210	2175
Totais	1514	6351

3.1.2.8 Datas de Ingresso e Nascimento

Os atributos dt_ingresso e dt_nascimento são a data de ingresso do aluno no curso e data de nascimento do aluno respectivamente. Estes atributos vão gerar um novo atributo que vai representar a idade do aluno quando ingressou no curso.

A Figura 10 mostra a contagem do número total e ano de nascimento dos alunos do curso de Ciência da Computação. Alunos que nasceram entre os anos de 1993 e 1995 foram encontrados com maior frequência no curso. Dos 1514 alunos do curso de Ciência da Computação 36 não possuem data de nascimento registrada no sistema, ou seja, 2,38% do conjunto de dados total.

A Figura 11 mostra o número total e ano de nascimento dos alunos dos cursos de engenharia. Nas engenharias tem uma frequência maior de alunos que nasceram no ano de 1994. Foi encontrado um dado discrepante onde o ano de nascimento foi 2010 e esse registro foi removido do conjunto de dados final na fase de limpeza dos dados.

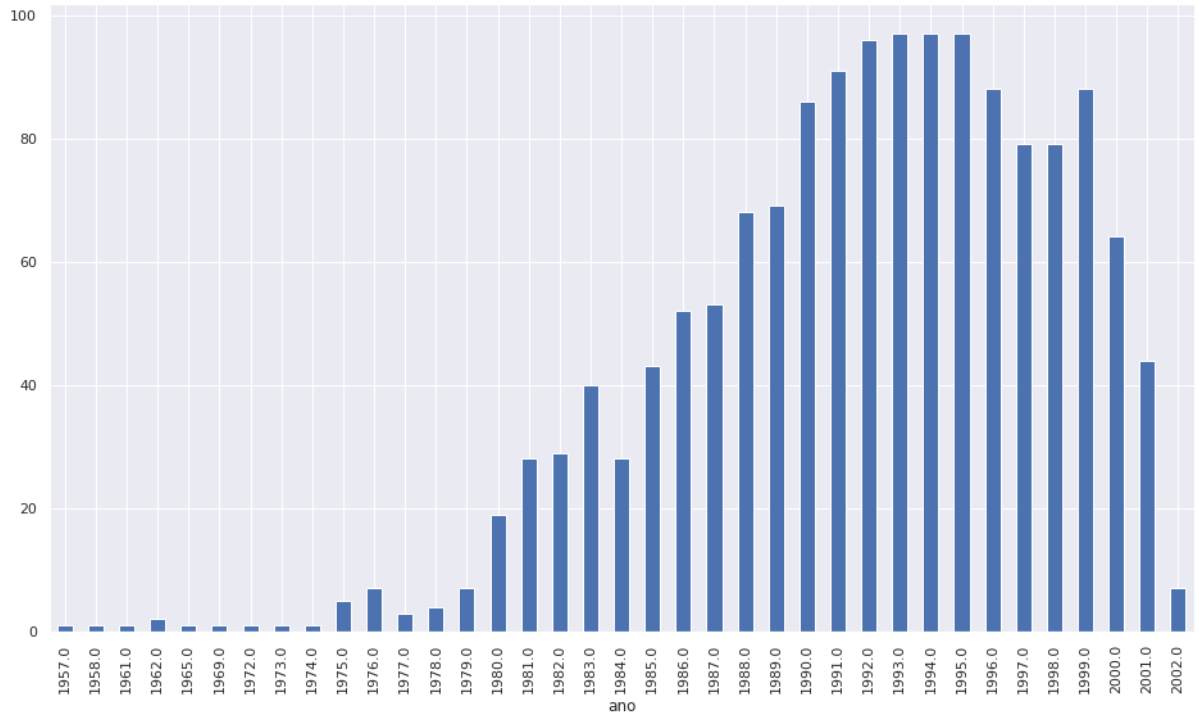

Figura 10 – Número de alunos do curso de Ciência da Computação por ano de nascimento até o ano de ingresso no curso.

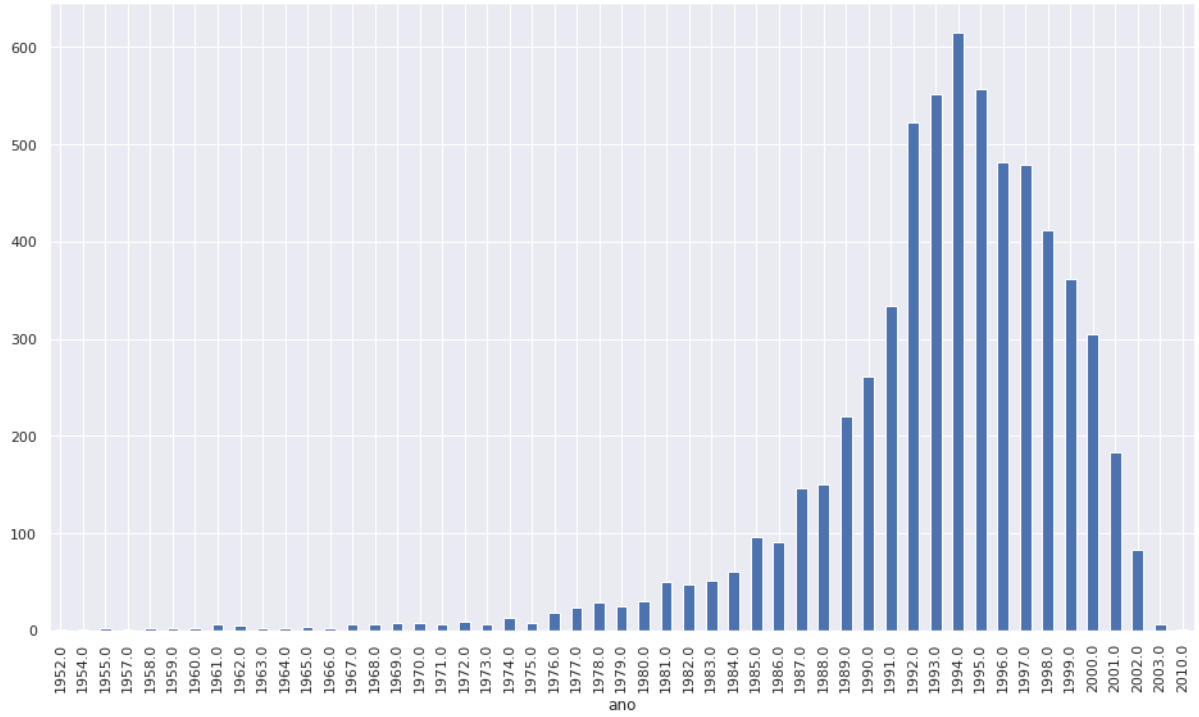

Figura 11 – Número de alunos dos cursos de Engenharias por ano de nascimento até o ano de ingresso nos cursos.

3.1.2.9 Etnia

Este atributo é a etnia declarada pelo aluno no curso de Ciência da Computação e Engenharias. A Tabela 9 apresenta o número total de alunos pela sua respectiva etnia. No curso de Ciência da Computação faltam informações de 457 alunos dos 1514, isso representa 30,18% do total. Já nas Engenharias faltam informações de 922 (14,5%) alunos de 6351. Um valor que chama a atenção é de alunos que não querem declarar tanto na Ciência da Computação quanto nas engenharias.

Tabela 9 – Total de alunos por etnia

Etnia	Ciência da Computação	Engenharias
BRANCA	749	3579
NÃO QUERO DECLARAR	167	966
PARDA	73	489
PRETA	64	356
AMARELA	4	34
INDÍGENA	0	5
Totais	1057	5429

3.1.2.10 Estado Civil

Este atributo é o estado civil do aluno quando ingressou no curso. A Tabela 10 mostra o número total de alunos e seu respectivo estado civil. A maior ocorrência é de alunos no estado civil “SOLTEIRO” com 980 e nas Engenharias de 4913. Na Ciência da Computação faltam informações de 495 alunos e isso representa 32,69% da base de dados. E nas engenharias faltaram informações de 1178 alunos.

Tabela 10 – Total de alunos pelo estado civil

Estado civil	Ciência da Computação	Engenharias
SOLTEIRO	980	4913
CASADO	28	173
OUTROS	8	60
DIVORCIADO	2	21
VIÚVO	1	3
SEPARADO JUDICIALMENTE	0	3
Totais	1019	5173

3.1.2.11 Tipo de Ingresso

O atributo tipo_ingresso representa a forma de ingresso do aluno no curso. Este atributo sofreu uma mudança bem significativa depois de 2007, devido a UFPel ter adotado como forma de ingresso o “SiSU/ENEM”. Isso fez com que alunos que tem

a forma de ingresso “Vestibular” fossem diminuindo naturalmente para dar lugar ao principal processo seletivo da instituição.

A Tabela 11 mostra o número total de alunos e sua respectiva forma de ingresso no curso de Ciência da computação e nos cursos de Engenharias. Na Ciência da Computação a maior ocorrência é de “SiSU/ENEM” com 903 alunos, que representa 59,64% do total de alunos. Nas Engenharias também destaca a forma de ingresso “SiSU/ENEM” como a principal forma de ingresso dos cursos. Ingressos como PEC-G[1] e outros são regimentados por decreto ou lei específica.

Tabela 11 – Número total de alunos pela forma de ingresso

Forma de ingresso	Ciência da Computação	Engenharias
Atividades Isoladas	1	40
Convênio PEC-G	0	3
Decisão Judicial	3	0
Mobilidade Acadêmica	1	40
PAVE	118	458
Portador de diploma - Processo Simplificado	0	12
Portador de Diploma de Curso Superior	8	104
Regime Especial	1	15
Reingresso	8	49
Reopção	36	543
SiSU (Vagas ociosas PAVE)	0	6
SiSU/ENEM	903	4792
Transferência	50	238
Transferência Compulsória	2	5
Transferência PEC-G	0	1
Vestibular	383	44
Vestibular - Quilombolas/Indígenas	0	1
Totais	1514	6351

3.1.2.12 Cota

Este atributo identifica o ingresso por ampla concorrência ou por cota e determinando em qual cota o aluno ingressou. A Tabela 12 mostra os totais de alunos pelo respectivo tipo de ingresso. Na Ciência da Computação 318 alunos ingressaram pela cota de ampla concorrência que representa aproximadamente 50% do total das formas de ingresso. Nas Engenharias não foi diferente, onde 1984 alunos ingressaram pela ampla concorrência. No caso da Ciência da Computação pela forma de ingresso do curso nem sempre ter sido feita por cotas este atributo tem muitos dados faltando

[1] Decreto Federal N° 7.948/2013

e não foi utilizado no trabalho.

Tabela 12 – Número total de alunos pela cota de ingresso. Leis n° 12.711/2012 e 13.409/2016 regulamentam o ingresso nas universidades públicas.

Cota de ingresso	Ciência da Computação	Engenharias
AC	318	1984
L05	125	587
L01	95	508
L06	45	261
L02	44	227
L13	2	5
L09	0	3
L14	0	1
L10	0	1
Totais	629	3577

3.1.2.13 Escola Pública

Este atributo indica se o aluno veio de escola pública ou não. Devido a migrações de bases existem eventuais ruídos que precisarão ser identificados, corrigidas e/ou eliminadas.

A Tabela 13 apresenta os totais de alunos provenientes de escola pública no ensino médio para o curso de Ciência da Computação e Engenharias. No exemplo deveria possuir o valor “S” ou “N”, mas foram encontrados os valores “P” e “V” e por isso terá que ser feito uma verificação a qual classe os valores “P” e “V” correspondem.

Tabela 13 – Número de alunos provenientes de escola pública ou não

Escola pública	Ciência da Computação	Engenharias
S	982	3828
N	391	1970
P	23	177
V	22	145
Totais	1418	6120

3.1.2.14 Curso Superior

Este atributo indica se o aluno cursou algum curso superior antes do curso atual. Caso o aluno tenha curso superior anterior o valor desse atributo é “S” caso contrário é “N”.

A Tabela 14 apresenta o número de alunos que possuem ou não curso superior anterior. Na Ciência da Computação aproximadamente 98% dos alunos não possuem curso superior anterior e nas Engenharias este cenário não muda ficando com 97%.

Tabela 14 – Total de aluno que possui ou não curso superior anterior

Curso superior	Ciência da Computação	Engenharias
N	1491	6174
S	23	177
Totais	1514	6351

3.1.2.15 Benefício

Este atributo informa se o aluno possui ou não benefício de assistência estudantil (de qualquer tipo). Pode assumir o valor “S” se possui o benefício ou “N” caso contrário.

A Tabela 15 mostra os totais de alunos que conseguiram benefício em algum momento do curso. No curso de Ciência da computação mais de 77% dos alunos não possui benefício. Enquanto nas Engenharias esse mesmo conjunto de alunos foi de 76,54%. No caso da Ciência da Computação, o problema deste atributo é que o número de registro de benefícios ficou mais relevante a partir de 2004 com 206 registros de benefícios. Acredita-se que anteriormente esses registros não eram sistematizados em bases de dados.

Tabela 15 – Total de alunos que possui ou não benefício de assistência estudantil

Benefício	Ciência da Computação	Engenharias
N	1167	4861
S	347	1490
Totais	1514	6351

3.1.2.16 Naturalidade

Este atributo apresenta a cidade de nascimento do estudante. E para melhor representação, o atributo ’naturalidade’ foi adaptado para 6 categorias conforme interesse da pesquisa e definição de meso e microregiões do IBGE (IBGE, 2021). As categorias definidas foram:

- PELOTAS - Alunos que nasceram na cidade de Pelotas;
- MICROPELOTAS - Alunos que nasceram na Microregião de Pelotas;
- MESOPELOTAS - Alunos que nasceram na Mesoregião de Pelotas;
- ESTADO - Alunos que nasceram no estado do Rio Grande do Sul;
- PAIS - Alunos que nasceram fora do estado do Rio Grande do Sul;
- ESTRANGEIRO - Alunos que nasceram em qualquer cidade fora do Brasil.

Neste atributo faltam informações de 335 alunos para o curso de Ciência da Computação, que representa 22,13% do total de alunos. Já para as Engenharias faltam informação de 973 alunos, que representa 15,32% do total.

Tabela 16 – Total de alunos pela naturalidade.

Naturalidade	Ciência da Computação	Engenharias
PELOTAS	562	2301
MICROPELOTAS	105	582
MESOPELOTAS	56	221
ESTADO	252	906
PAIS	204	1332
ESTRANGEIRO	0	36
Totais	1179	5378

3.1.2.17 Atributos de Informações do Primeiro, Segundo e Terceiro Semestre

Os atributos flg_pri_sem, flg_seg_sem, flg_ter_sem representam se o aluno cursou o primeiro, segundo ou terceiro semestre respectivamente. Já os atributos nr_dis_pri_sem_cur, nr_dis_seg_sem_cur e nr_dis_ter_sem_cur representam o número de disciplinas cursadas no primeiro, segundo ou terceiro semestre respectivamente.

Estes atributos vão servir para criar o coeficiente de dificuldade, que vai ser o número de disciplinas aprovadas no semestre dividido pelo número de disciplinas no currículo do curso no respectivo semestre. Uma disciplina só vai entrar para a contagem do número de disciplinas aprovadas no semestre se a disciplina for do mesmo semestre que o aluno se encontra. Por exemplo, um aluno que está no 3º semestre do curso e cursou uma disciplina que pertencia ao primeiro semestre do curso, essa disciplina não entra para a contagem, mas no caso de o aluno estar no 3 semestre do curso e aprovar uma disciplina do terceiro semestre, essa disciplina entrará para a contagem. Isso é uma forma de tentar penalizar por não ter cursado no tempo correto.

$$coeficiente = \frac{número\ disciplinas\ aprovadas\ no\ semestre}{número\ disciplinas\ total\ no\ semestre\ do\ currículo}$$

3.1.3 Análise exploratória dos dados

Neste trabalho foi feito uma AED para entender a natureza dos dados sem fazer suposições e tentar tirar alguns *insights*. A AED é uma técnica realizada após a etapa de pré-processamento de dados e coleta de dados.

Para fazer algumas análises foi criado um campo chamado “Situação categoria”, que agrupa as situações dos alunos apresentada na Seção 3.1.2.3 em 4 classes “Cursando”, “Retido”, “Formado” e “Evadido”. A Tabela 17 mostra a relação das situações

dos alunos com as novas classes. As classes “Cursando” e “Retido” foram determinadas pela seguinte regra: alunos que estavam nas situações “Aluno com vínculo” ou “Trancado”, e além disso quem estava além do período normal de integralização (no caso da Ciência da Computação são 8 semestres para versão atual do currículo) no curso foram considerados como “Retido” caso contrário foram considerados como “Cursando”.

Tabela 17 – Relação de situações e novas classes.

Situação	Categorias
Abandono	Evadido
Aluno com vínculo	Cursando ou Retido
Cancelamento	Evadido
Desligado	Evadido
Desligado Lei nº 12.711 de 29/08/2012[2]	Evadido
Desligado Res.03/05	Evadido
Falecido	Evadido
Fim Mobilidade	Evadido
Fim período	Evadido
Formado	Formado
Jubilado	Evadido
Matrícula não confirmada	Evadido
Reopção	Evadido
Trancamento	Cursando ou Retido
Transferido	Evadido

3.1.3.1 Idade

Foi considerada a idade que o aluno tinha quando ingressou no curso. Por existirem alunos com data de nascimento nulo foram considerados um número inferior de alunos. Na Ciência da Computação, por exemplo, apenas 1.436 alunos dos 1.514 coletados da base de dados.

A Figura 12 é um histograma do curso de Ciência da Computação, que mostra a densidade de alunos por idade. A partir do gráfico é possível verificar que a idade dos alunos está concentrada no início, ou seja, 77% dos alunos são jovens com idade entre 16 e 21 anos.

Outro ponto importante é que o atributo idade não segue uma distribuição normal. A prova disso foi alcançada utilizando o teste Shapiro-Wilk (SHAPIRO; WILK, 1965), onde a hipótese nula é que os dados seguem uma distribuição normal (H_0: distribuição dos dados = normal) e hipótese alternativa é que os dados não seguem uma distribuição normal (H_1 : distribuição dos dados $\neq$ normal). Então, com um nível de significância (α) de 5%, rejeita-se H_0, com um p-valor de $2,2 \times 10^{-16}$. Portanto a distribuição da idade de ingresso dos alunos não segue uma distribuição normal com um

grau de significância de 5%.

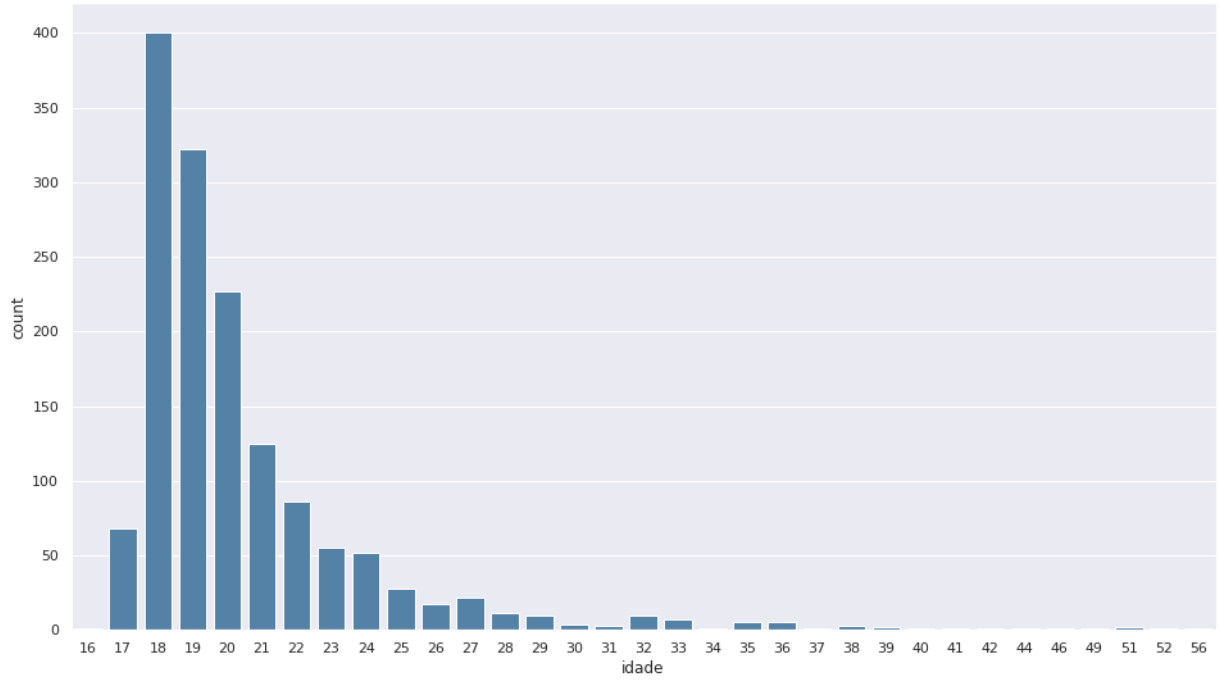

Figura 12 – Número de alunos por idade de ingresso no curso de Ciência da Computação.

A Figura 13 é o histograma dos cursos de Engenharias. Assim como na Ciência da Computação o gráfico mostra que a concentração de alunos está mais a esquerda, mostrando que existe uma concentração maior de alunos jovens no curso e em maior número entrando com 18 anos.

3.1.3.2 Relação da Idade e Situação Final do Aluno

Nesta Seção será verificada a relação entre a idade dos alunos com a situação final deles, para tentar identificar alguma tendência, principalmente entre alunos nas situações formado e evadidos. A Figura 14 mostra a distribuição de idade de ingresso pela situação final ou atual dos alunos da Ciência da Computação. No casso do atributo idade é preferível olhar para a mediana e intervalo interquartis (IQR) do que para media e desvio padrão, pois a distribuição dos dados neste atributo não seguem uma distribuição normal, como já foi provado no início deste sessão.

Para verificar se existe uma diferença entre a distribuição da idade de alunos formados e evadidos, foi utilizado o o teste de Mann-Whitney (HART, 2001) - que tem como hipótese nula que a mediana dos alunos formados é igual a mediana dos alunos evadidos e hipótese alternativa que a mediana dos alunos formados é diferente da mediana dos alunos evadidos. O teste de Mann-Whitney mostrou que a mediana das idades dos alunos formados não é diferente da mediana das idades dos alunos evadidos (W = 24893; p = 0,08199). A mediana dos alunos formados (19 e 3, mediana e IQR) foi igual a dos alunos evadidos (19 e 4).

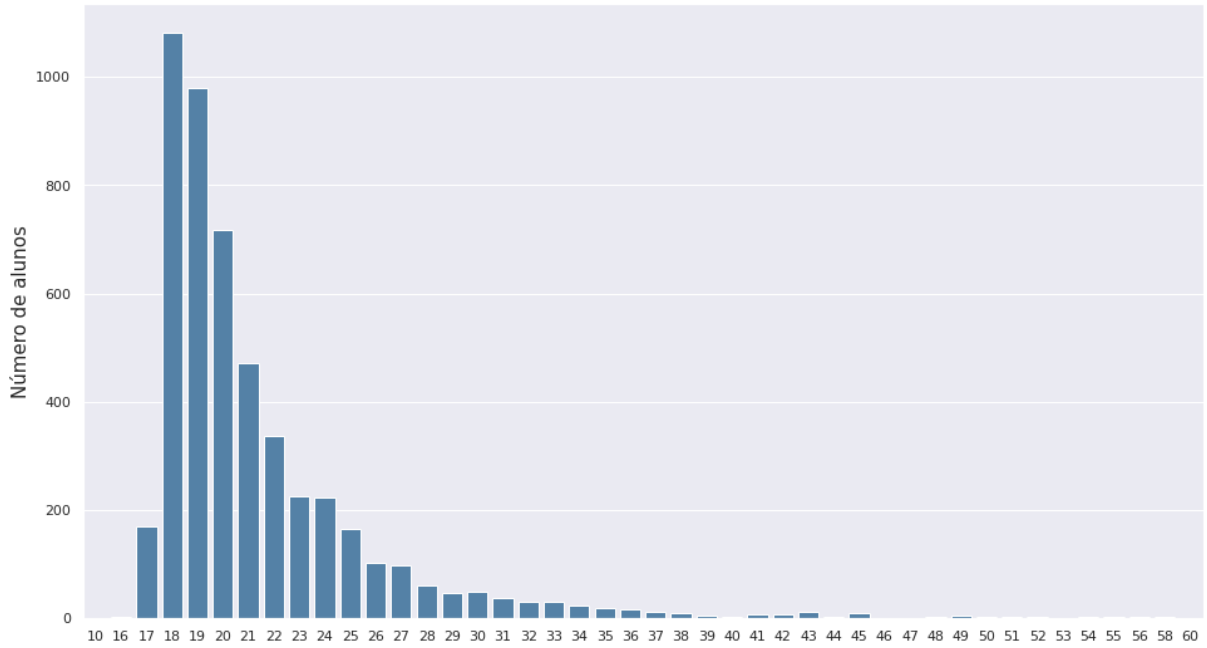

Figura 13 – Número de alunos por idade de ingresso nos cursos de Engenharias.

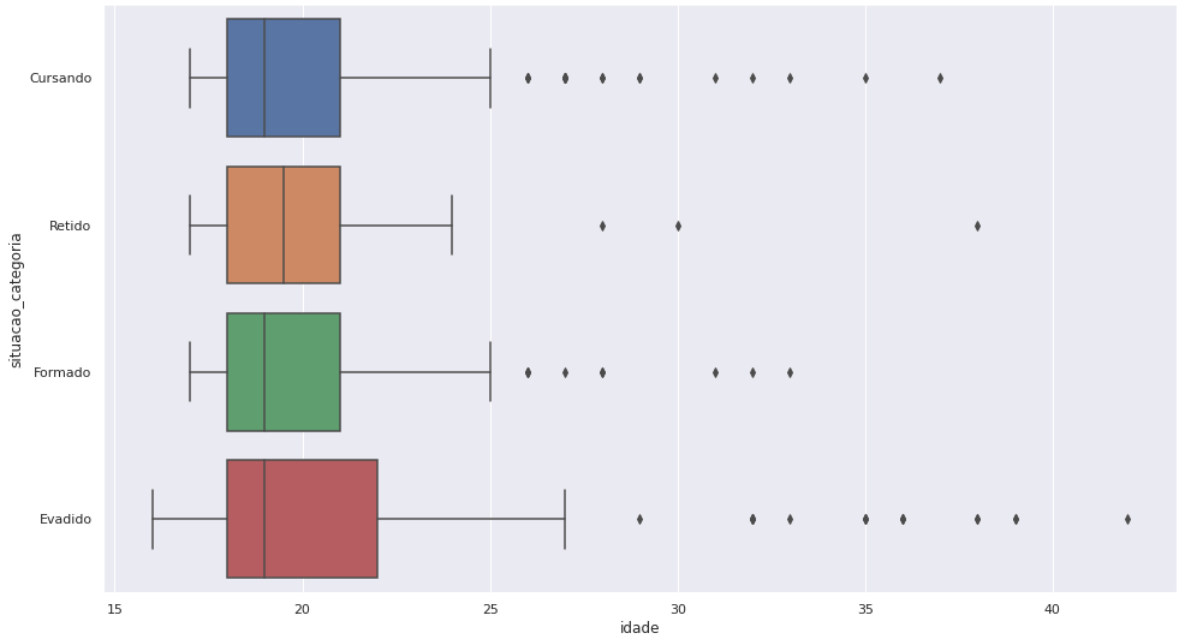

Figura 14 – Distribuição das idades dos alunos da Ciência da Computação pela situação final ou atual.

A Figura 15 mostra a distribuição de idade pela situação final ou atual dos alunos das Engenharias. Assim como na Ciência da Computação foi realizado o teste de Mann-Whitney usando as mesmas hipóteses. O teste de Mann-Whitney mostrou que a mediana das idades dos alunos formados é diferente da mediana das idades dos alunos evadidos (W = 817525; p = $3,65 \times 10^{-09}$). A mediana dos alunos formados (19 e 4, mediana e IQR) foi igual a dos alunos evadidos (20 e 4).

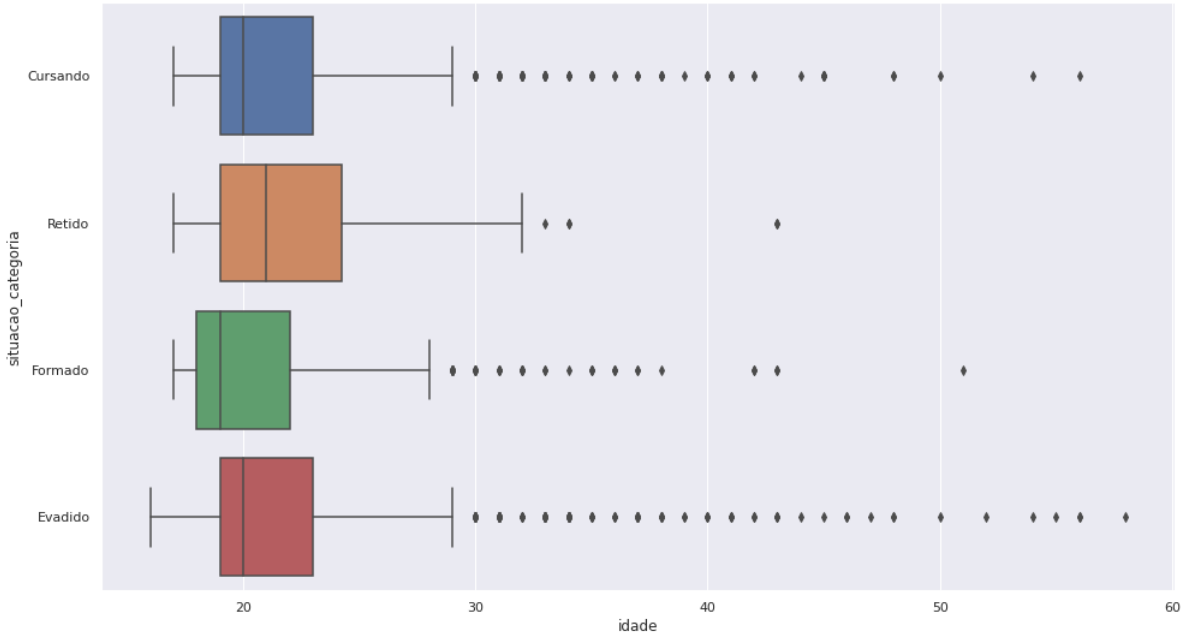

Figura 15 – Distribuição de idade dos alunos das Engenharias pela situação final ou atual.

3.1.3.3 Relação da Idade com Média geral do Aluno

Nesta Seção será apresentada uma relação entre a idade com a média geral do aluno. Estes dados são apresentados para o curso de Ciência da Computação. A Figura 16 mostra a relação entre a média geral do aluno com a idade e traça a regressão linear entre as duas.

A Figura 17 apresenta um gráfico que expõe informações como máximo, mínimo, quartil e *outlier* da média geral do aluno pela idade. A partir deste gráfico é possível verificar que o conjunto de dados possui alguns *outliers* que devem ser removidos na etapa de limpeza dos dados. Nas idades onde não aparecem os mínimos, máximos e quartis significa que existem penas um ou nenhum valor para aquela idade e é apresentado apenas um traço.

3.1.3.4 Números de alunos por período de ingresso e situação

A Figura 18 apresenta uma série histórica do curso de Ciência da Computação. O eixo x representa o ano e semestre de ingresso do aluno e o eixo y o número de alunos no período agrupados pela situação final do aluno. De 2000 até 2007 o curso

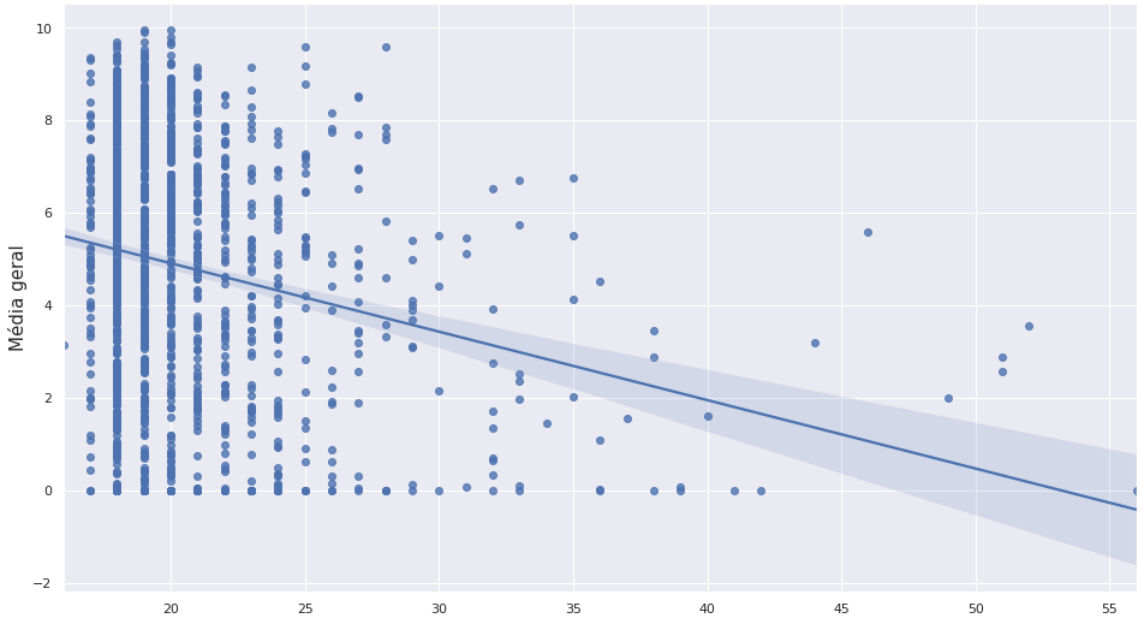

Figura 16 – Média geral pela idade de alunos com a regressão linear.

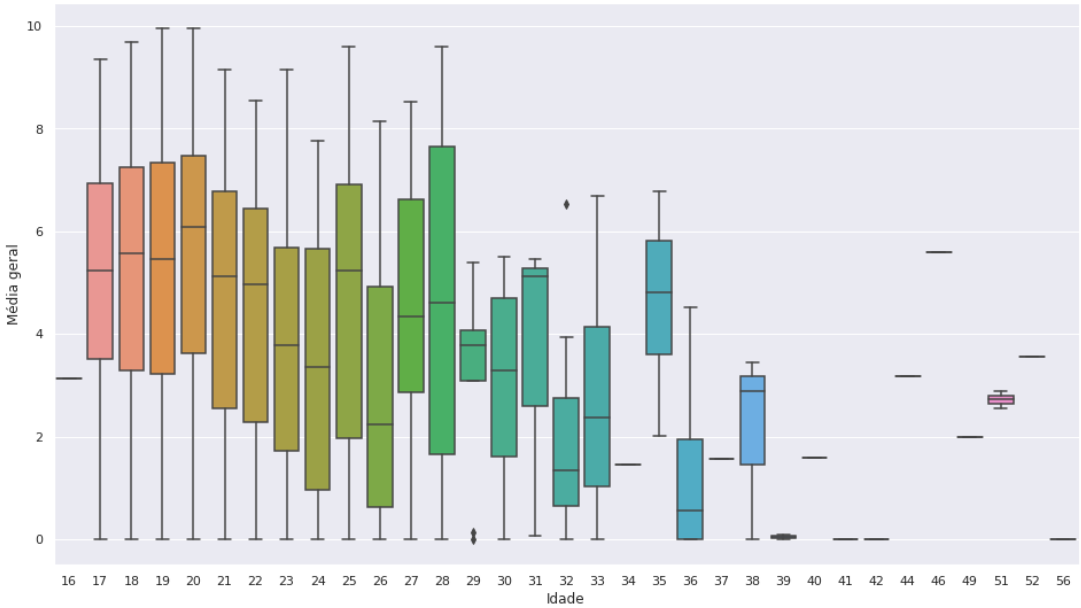

Figura 17 – Média geral pela idade dos aluno mostrando maximo, minimo, quartis e *outliers*.

tinha uma população de alunos formados maior do que a de alunos que evadiram. Depois de 2007 o número de alunos formados começa a ser inferior ao número de evadidos, além disso o número de alunos retidos fica cada vez mais significante com o passar do tempo.

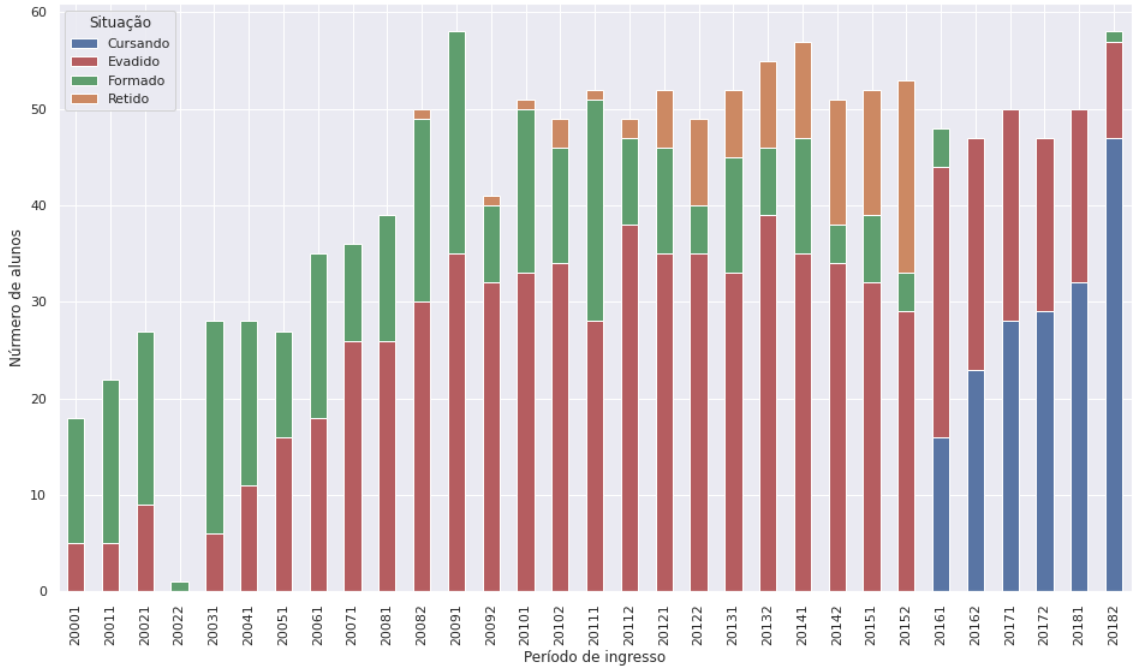

Figura 18 – Número de alunos e respectiva situação pelo período de ingresso da Ciência da Computação.

A Figura 19 apresenta a série histórica das Engenharias. Nas Engenharias em todos os períodos o número de alunos formados foi sempre menor que o número de evadidos. Vale destacar ainda que existem alunos que ingressaram em 2010/1 e ainda continuam no curso, estes alunos são considerados como retidos.

3.1.3.5 Número de alunos por semestre de saída e situação

A Figura 20 apresenta o número total de alunos da Ciência da Computação no semestre de saída do aluno, agrupados em conjunto a respectiva situação. Um ponto que vale destacar é que o número de alunos formados cresce significativamente a partir do oitavo semestre e chega ao ponto mais alto no décimo semestre. Isso se deve ao período de integralização curricular que em alguns currículos foi de 8 semestres e outros de 9 semestres. Ainda é possível destacar que 19,33% evade até o terceiro semestre e 32,45% depois do terceiro semestre.

A Figura 21 apresenta o número total de alunos das Engenharias no semestre de saída do aluno, agrupados em conjunto a respectiva situação. Assim como na Ciência da Computação, o segundo semestre é o que apresenta o maior índice de evasão. Outra similaridade é que o número de alunos formados cresce a partir do oitavo semestre e chega ao ponto mais alto no décimo semestre, que é é tempo adequado de

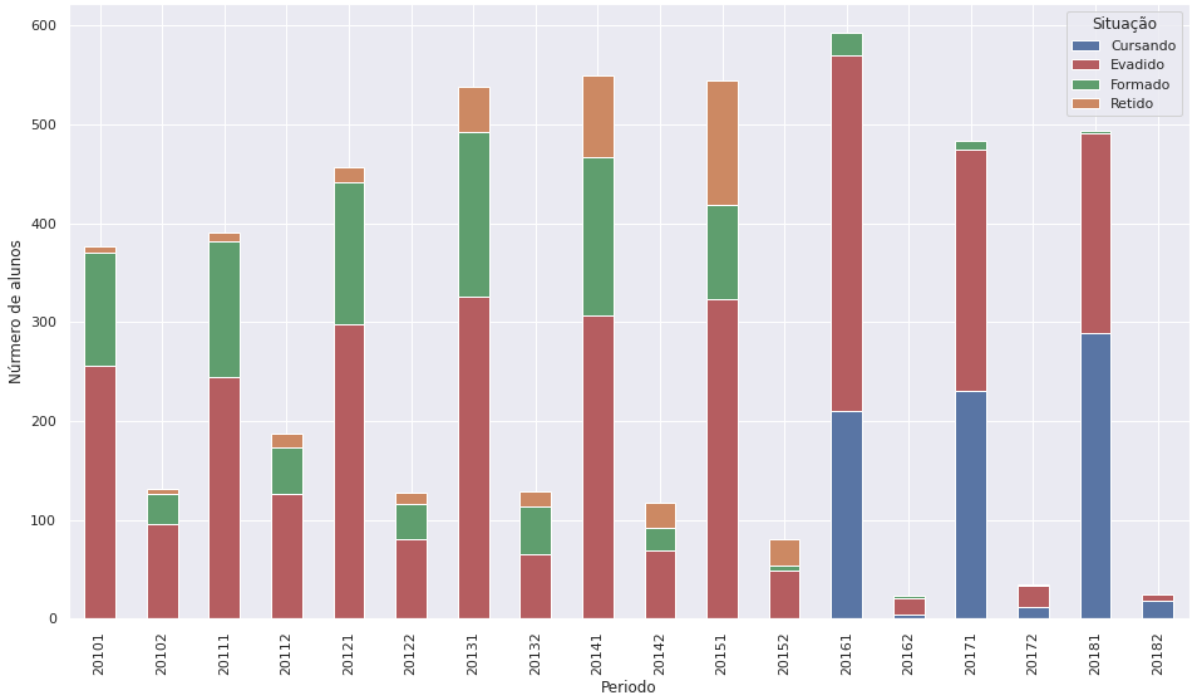

Figura 19 – Número de alunos e respectiva situação pelo período de ingresso nas Engenharias.

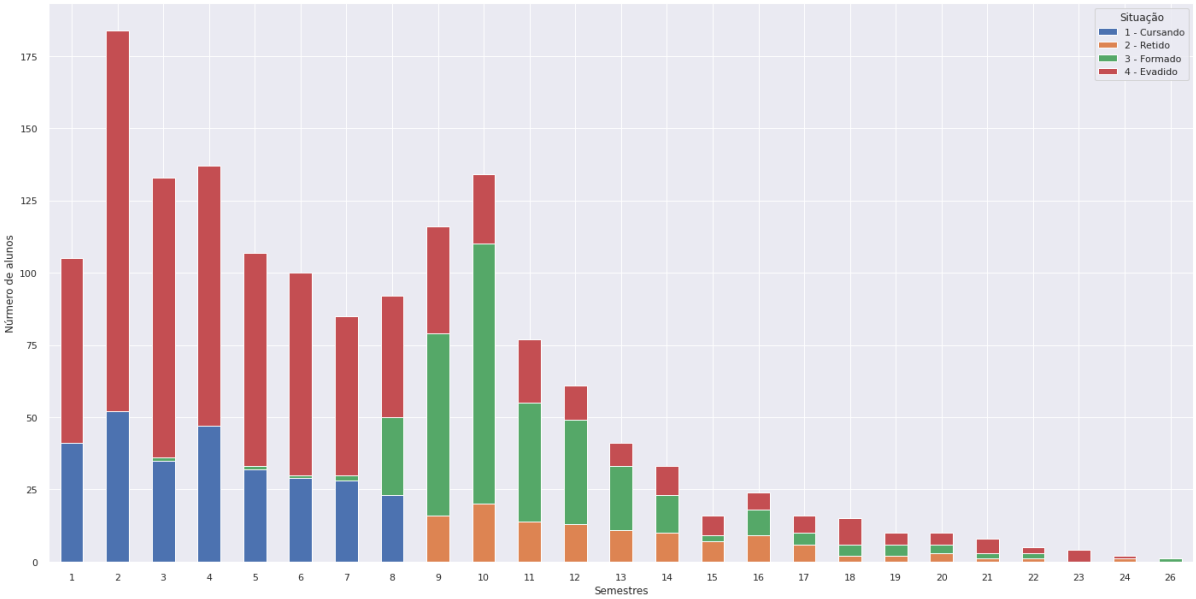

Figura 20 – Número de alunos, da Ciência da Computação, pelo semestre de saída e agrupados pela situação.

conclusão.

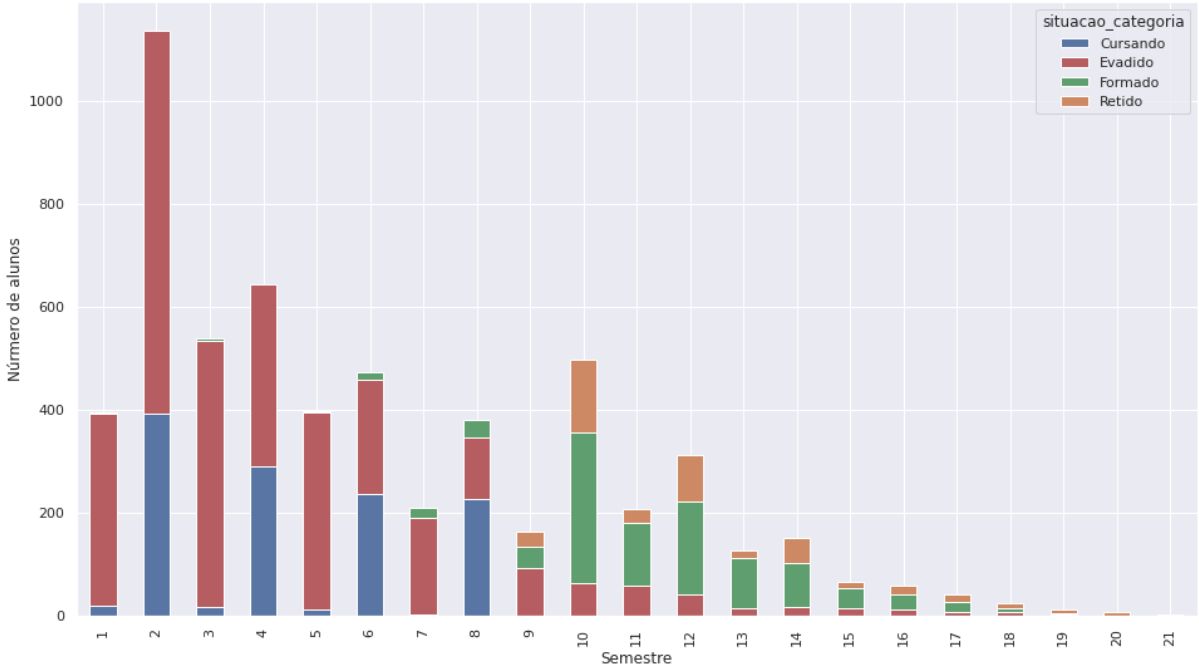

Figura 21 – Número de alunos, das Engenharias, pelo semestre de saída e agrupados pela situação.

3.1.3.6 Média geral dos alunos nos primeiros semestres

A Figura 22 apresenta a média geral dos alunos por semestre(três semestres iniciais). No terceiro semestre alunos que conseguiram concluir o curso tem uma média geral mais alta que alunos nas outras situações. Por outro lado, alunos que evadiram tiveram a média mais baixa nos 3 períodos iniciais do curso de Ciência da Computação.

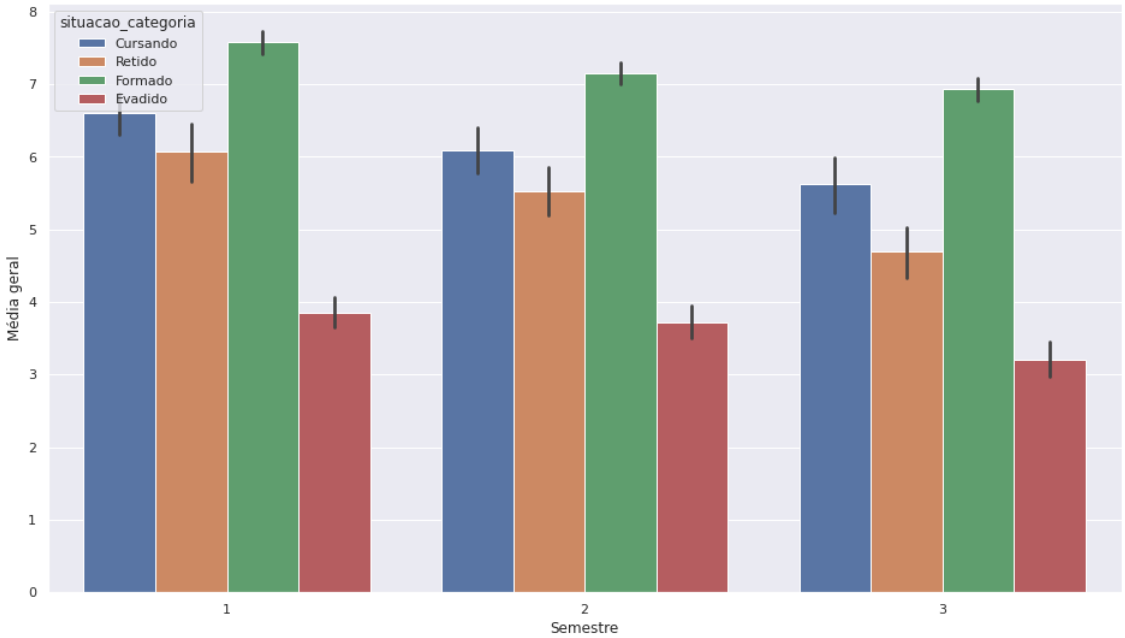

Figura 22 – Média geral dos alunos do Curso de Ciência da Computação nos três primeiros semestres.

3.2 Preparação dos dados

Esta Seção está organizada em Seleção, Limpeza, Formatação, Criação e Descrição do conjunto final de dados, onde será descrito o que foi feito em cada uma dessas etapas.

3.2.1 Seleção

Após a fase de compreensão dos dados iniciou-se a de preparação dos dados, nesta fase primeiro foi feita a seleção dos dados. Geralmente a seleção de dados tem dois enfoques distintos redução de dados horizontal e vertical (GOLDSCHMIDT; BEZERRA; PASSOS, 2015).

A redução de dados horizontal normalmente se escolhe qual caso vai ser dado atenção. Neste enfoque foi escolhido o curso de Ciência da Computação, por ser o curso de maior afinidade das pessoas que desenvolvera esta pesquisa. Já a redução de dados vertical visa escolher o conjunto de atributos mais relevantes para alcançar o objetivo. Neste enfoque foram escolhidos os atributos apresentados na Seção 3.1.2. A escolha dos atributos foi baseada nos atributos de trabalhos encontrados em trabalhos relacionados.

3.2.2 Limpeza

A fase de limpeza de dados envolve verificar as informações inconsistentes, correções de erros e preencher ou eliminar valores desconhecidos e redundantes, e também retirar valores que não fazer parte do domínio em estudo (GOLDSCHMIDT; BE-

ZERRA; PASSOS, 2015). Em geral se faz a limpeza de informações ausentes, limpeza de inconsistências e limpeza de valores não pertencentes ao domínio.

3.2.2.1 Limpeza de informações ausentes

Neste passo os valores ausentes são eliminados do conjunto de dados. Foram eliminados todos os registros de alunos com valores ausentes dos atributos flg_escola_publica, nota_final, dt_nascimento, dt_ingresso, naturalidade.

O atributo cota é um caso a parte, pois estão sendo avaliados dados de alunos que entraram entre 2000 e 2018 para Ciência da Computação e de 2010 a 2018 para as Engenharias, e o primeiro registro de cota aconteceu em 2014/1. Para manter este atributo todos os registros anteriores a este período foram considerados de alunos que entraram por ampla concorrência.

Como exemplo, para a Ciência da Computação, antes de eliminar os valores ausentes o conjunto de dados contava com 21.105 registros de 1.514 alunos. Após essa remover os valores ausentes ficaram 13.492 registros de 1.085 alunos.

Existem alguns métodos de preenchimento de valores tais como preenchimento manual, preenchimento com medidas estatísticas, dentre outros. Mas esses métodos têm um custo para o desempenho final dos modelos e não foram utilizados.

3.2.2.2 Limpeza de inconsistências

Esta função tem como objetivo identificar e eliminar valores inconsistentes do conjunto de dados (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). A inconsistência pode envolver um registro apenas ou um conjunto de registros e eles podem ser removidos ou corrigidos.

Um exemplo que aconteceu no conjunto de dados deste trabalho foi a disciplina dispensada, onde todas as disciplinas na situação dispensado não deveriam conter a nota final do aluno na disciplina, mas neste caso havia uma disciplina com nota final e na situação dispensada.

Outro exemplo foi o atributo que determina se o aluno veio de escola pública ou não. Este atributo deveria conter o valor “S” para alunos que vieram de escola pública e “N” para alunos que não vieram de escolas públicas, mas foram encontrados os valores “P” e “V”. Os dois últimos valores pertenciam ao sistema anterior ao Cobalto e descobriu-se que o valor “P” equivale a alunos que vieram de escolas públicas e o valor “V” equivale a alunos que não vieram de escolas públicas.

3.2.2.3 Limpeza de valores que não pertencem ao domínio

Esta função compreende a busca e eliminação de valores que não pertençam ao domínio dos atributos do problema (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Ela pode ser considerada um caso particular da função de limpeza de inconsistências

e é preciso ter um conhecimento prévio do problema. Um exemplo deste trabalho foram os “Alunos com vínculo” que fogem do objetivo deste trabalho, pois não dá para determinar se eles evadiram ou não.

Como exemplo, para a Ciência da Computação, após finalizar esta função sobraram 9.725 registros de 744 alunos do conjunto de dados total. Este também foi o conjunto final de dados da fase de limpeza dos dados.

3.2.3 Construção de atributos

A operação de construção de atributos consiste em criar novos atributos a partir de atributos existentes. Estes novos atributos costumam ser chamados de atributos derivados (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Exemplos deste trabalho foram idade, semestre que o aluno cursou uma determinada disciplina, médias do primeiro, segundo e terceiro semestre, média dos três primeiros semestres, coeficiente de dificuldade do primeiro, segundo e terceiro semestre, e a média do número de disciplinas cursadas nos três primeiros semestres. Esses atributos foram gerados para tentar contextualizar melhor os dados retirados do sistema e obter um melhor desempenho dos algoritmos de classificação.

A idade foi gerada a partir da data de ingresso do aluno e a data de nascimento. Foram extraídos o ano das datas e subtraído o ano da data de ingresso pelo ano da data de nascimento, e os meses e dias foram desconsiderados para fazer este cálculo.

O atributo que representa o semestre que o aluno cursou uma determinada disciplina foi gerado através de uma fórmula usando o ano/semestre de ingresso e ano/semestre que o aluno cursou a disciplina.

$$semestre = 2(ano\_dis - ano\_ing) + (sem\_dis - sem\_ing) + 1$$

Onde ano_dis e sem_dis correspondem ao ano e semestre que o aluno cursou a disciplina respectivamente, e ano_ing e sem_ing correspondem ao ano e semestre que o aluno ingressou no curso também respectivamente.

As médias do primeiro, segundo e terceiro semestre foram geradas utilizando a técnica de *pivot table*, que reorganiza os valores da Tabela criando novas estatísticas. Estes atributos foram gerados utilizando o método *pivot_table* do Pandas, que é uma biblioteca do Python que faz diversas manipulações de dados. Como exemplo, para Ciência da Computação, após a execução dessa função o conjunto de dados ficou com 744 registros de 744 alunos, pois o método *pivot_table* calculou a média do primeiro, segundo e terceiro semestres em colunas do conjunto de dados.

No mesmo processo que criou as médias do primeiro, segundo e terceiro semestres foram criados o número total de disciplinas aprovadas no primeiro, segundo e terceiro semestres, através dos atributos flg_pri_sem, flg_seg_sem e flg_ter_sem que

indicam se a disciplina pertence e foi cursada no primeiro, segundo e terceiro semestres respectivamente.

O coeficiente de dificuldade é uma estratégia para penalizar o aluno que não seguiu o currículo de forma integralizada. Esta estratégia compara o currículo do curso com as disciplinas que o aluno cursou. O coeficiente é a divisão das disciplinas que o aluno aprovou no semestre atualmente cursado pelo número total de disciplinas que o aluno deveria cursar no semestre. Ou seja, se o aluno se matriculou e passou em todas as disciplinas do primeiro semestre do currículo no seu primeiro semestre de curso, então esse aluno conseguiu integralizar todo o primeiro semestre. Caso contrário ele fica com uma integralização parcial do semestre. Por exemplo, se o aluno entrou no curso, se matriculou em 5 das 5 disciplinas do primeiro semestre e aprovou em 4, esse aluno vai ficar com o *score* de 4/5 no coeficiente de dificuldade. Disciplinas que o aluno solicitou dispensa foram consideradas como integralizada. Disciplinas que entraram no histórico do aluno por transito de notas não foram consideradas integralizadas. No final foram criados 3 coeficientes de dificuldade, um para cada semestre.

O atributo evadiu que será o atributo-alvo dos modelos de classificação foi gerado a partir da situação final ou atual do aluno. Alunos com situação final de “Formado” recebeu o valor falso e o aluno com a situação diferente dessa recebeu o valor verdadeiro.

3.2.4 Codificação dos dados

A codificação dos dados é uma operação que modifica o domínio de valores de um determinado atributo. O importante é que os dados sejam codificados para melhor suprir as limitações de determinados algoritmos de MD (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Um exemplo seria uma rede neural que necessita que os dados estejam representados em formato numérico.

O tipo de conhecimento que se deseja buscar é fortemente influenciado pela maneira que a informação é codificada (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Uma codificação pode ser de numérica para categórica ou vice-versa. Neste trabalho foi feito tanto uma codificação numérica para categórica, quanto uma codificação categórica para numérica.

3.2.4.1 Numérica para categórica

A codificação numérica para categórica foi feita no atributo idade, onde foi utilizado uma abordagem de mapeamento de intervalos, que também é conhecida como discretização. As idades foram agrupadas em 5 categorias até 17, de 18 a 20, de 21 a 23, de 24 a 26 e de 27 a 56 anos. Os intervalos foram definidos de forma arbitrária.

3.2.4.2 Categórica para numérica

A codificação categórica para numérica faz a representação dos valores categóricos em valores numéricos. Um exemplo desse tipo de representação é a representação binária padrão onde cada categoria é transformada em um valor binário.

Neste trabalho foi usado uma variação desse tipo de codificação que é chamado em inglês de *dummy*. Dummy transforma um atributo categórico em tantos atributos quanto categorias que existem no atributo e cada atributo gerado vai ter o valor 0 ou 1. Por exemplo, o atributo gênero que no caso do sistema pode ser masculino e feminino, seriam criados dois atributos masculino e feminino onde quando um deles for 1 o outro é 0 e vice-versa. Portanto uma pessoa do gênero masculino teria o atributo masculino igual a 1 e atributo feminino igual a 0.

3.2.5 Conjunto final de dados

Para a Ciência da Computação, após todo o processo de preparação dos dados o conjunto de alunos do estudo se limitou a 744 alunos e 22 atributos que foram selecionados para o estudo. Destes alunos, 66,5% (495) estavam na situação de evasão do curso e um total de 33,5% (249) estavam na situação de conclusão do curso de 2000 a 2018.

3.3 Modelagem

Esta Seção corresponde a fase de Modelagem da metodologia proposta no Capítulo 3. Nesta fase os algoritmos de aprendizagem de máquina (AM) são aplicados ao conjunto de dados extraído do sistema acadêmico da universidade.

Este trabalho será dividido em 4 etapas que vão descrever as ferramentas e algoritmos que foram utilizados.

3.3.1 Etapa 1

Na fase inicial deste trabalho foram buscadas ferramentas de fácil utilização em vista de otimizar o tempo. Nesta fase foi utilizada a ferramenta RapidMiner Studio na versão 9.7 (RAPIDMINER, 2021) que é um designer de fluxo de trabalho visual para análise preditiva que traz ciência de dados e aprendizado de máquina.

O RapidMiner Studio na versão 9.7 é possível acelerar o processo de aprendizagem e construção de modelos utilizando as abordagens guiadas *Turbo Rep*, *Auto Model* e *Deployments*. Neste trabalho optou-se pela abordagem *Auto Model*, por passar por todo o processo de mineração de dados. O *Auto Model* é um processo dividido em 6 passos *Load Data*, *Select Task*, *Prepare Target*, *Select Inputs*, *Model Types* e *Results*.

O primeiro passo é o *Load Data*, que é onde o conjunto de dados é carregado.

Existe diversas formas de carregar os dados nesta versão do RapidMiner e para este trabalho foi escolhida a de importar do computador local. Nesta etapa foram feitos testes com os dados do curso de Geoprocessamento e de Ciência da Computação, pois eram cursos de conhecimento das pessoas envolvidas na pesquisa.

Em seguida vem o passo *Select Task*, que é onde tem que selecionar o tipo de tarefa de mineração de dados será realizada. A versão utilizada neste trabalho oferece 3 tipos de tarefas *Predict*, *Clusters* e *Outliers*. Foi utilizada a tarefa *Predict*, que faz a predição de uma coluna do conjunto de dados. A coluna utilizada para fazer a predição foi a de "evasao" que pode ser "S" quando o aluno evadiu e "N" caso tenho se formado.

No passo *Prepare Target* serão dadas algumas opções com relação ao atributo alvo. Uma delas é o *Class of Highest Interest*, que permite selecionar a classe que os algoritmos devem focar os resultados. Isso é importante pois valores de desempenho como *Precision* e *Recall* precisam saber qual classe devem interpretar como resultado positivo. No caso deste trabalho é a classe "S", pois ela indica que o aluno evadiu do curso.

O próximo passo é *Select inputs*, que ajuda selecionar os atributos que vão melhorar o desempenho dos modelos. Um ponto importante é que se procura padrões nos dados e se não tiver variações neles eles não serão uteis. Esse passo ajuda a identificar atributos que tenham uma correlação muito alta com o atributo alvo, que tenham todos ou quase todos os valores diferentes ou idênticos, ou que tenham valores faltando. Esses atributos problemáticos serão identificados e marcados, para serem agregados ou não ao conjunto de dados final.

Em *Model Types* é onde selecionam os modelos que serão usados. A versão 9.7 do *RapidMiner* disponibiliza 8 modelos *Naive Bayes*, *Generalized Linear Model*, *Logistic Regression*, *Deep Learning*, *Decision Tree*, *Random Forest*, *Gradient Boosted Trees (XGBoost)*, *Support Vector Machine (SVM)*. Neste trabalho os dados, tanto da Ciência da Computação quanto do Geoprocessamento, foram submetidos aos 8 modelos de aprendizagem de máquina.

Após executar os modelos são mostrados os resultados que é a última fase do *Auto Model*. Nesta fase são mostrados os resultados dos modelos, como por exemplo: *Accuracy, AUC, Precision, Recall, F-Measure, Sensitivity, Specificity*, dentre outras formas de avaliar os resultados dos modelos.

3.3.2 Etapa 2

A implementação dessa etapa foi apresentada em Costa et al. (2020) e Costa; Primo; Mattos (2020) e usando Python e Scikit-learn. Scikit-learn é uma biblioteca desenvolvida em Python, que possui diversas ferramentas de aprendizagem de máquina. Esta implementação foi separada em dois módulos, onde um fez a preparação dos dados e outro que submeteu os dados aos algoritmos de aprendizagem de má-

quina.

A preparação dos dados seguiu um roteiro que é normalmente utilizado em Ciência de Dados que é: seleção, limpeza, formatação e transformação. Foram selecionados 18 atributos e dados de 744 alunos do curso de Ciência da Computação. Destes alunos 66,5% (495) estavam na situação de evasão do curso e um total de 33,5% (249) estavam na situação de conclusão do curso de 2000 a 2018.

Para submeter os dados aos modelos os dados foram separados em um conjunto de treinamento e outro de testes, na proporção de 3/4 para treinamento e o restante para testes utilizando *train_test_split*. A função *train_test_split* foi utilizada com a opção stratify ativada, esta opção mantém a proporção de evadidos e não evadidos igual no treinamento e teste. Após essa separação os dados de treinamento foram balanceados para tentar reduzir a quantidade de falsos negativos, que é quando os algoritmos predizem que o aluno não vai evadir, mas na verdade ele evadiu. Para fazer o balanceamento foi utilizada a função *SMOTE* do Scikit-learn, para fazer o *Oversampling* no conjunto de dados. Por fim, os dados foram submetidos a três algoritmos de AM, sendo eles *Logistic Regression*, *Decision Tree* e *RandomForest*.

O trabalho apresentado em Costa et al. (2020) e Costa; Primo; Mattos (2020) tiveram resultados tais como a previsão alunos em risco de evasão com uma acurácia de 91,05% para o melhor resultado e que o atributo mais influenciou a predição foi a média do terceiro semestre que se destacou no *Feature Importance* de dois dos 3 algoritmos. A Tabela 18 apresenta os resultados alcançados nos trabalhos apresentados em Costa et al. (2020) e Costa; Primo; Mattos (2020).

Tabela 18 – Resultados da execução dos algoritmos de Costa et al. (2020) e Costa; Primo; Mattos (2020).

Algoritmo	Accuracy	Precision	Recall	F1-score	AUC
Logistic Regression	90,00%	95,80%	89,06%	92,31%	90,50%
Decision Tree	87,37%	91,94%	89,06%	90,48%	86,47%
Random Forest	91,05%	95,12%	91,41%	93,23%	90,86%

3.3.3 Etapa 3

A etapa 3 é a principal etapa da metodologia é nessa etapa que acontece a busca efetiva por conhecimento. A execução dessa etapa é a aplicação efetiva de algoritmos sobre os dados na tentativa de extrair conhecimento (GOLDSCHMIDT; BEZERRA; PASSOS, 2015). Neste trabalho foi feita uma abordagem de aprendizado supervisionado, onde a entrada é um conjunto de valores de uma ou mais variáveis e a saída é o valor do atributo alvo.

Nesta etapa foram utilizados 5 algoritmos, os 3 da Seção 3.3.2, mais Naive Bayes e Redes Neurais, também foi utilizado o método de validação cruzada estratificada. As ferramentas utilizadas foram Python e a biblioteca Sklearn do Python.

Para o algoritmo de *Decision Tree* foi utilizado o método “DecisionTreeClassifier” da biblioteca Sklearn. O método foi executado com os valores padrões do método. Foi configurado com a estratégia Gini Impurity para medir a qualidade da divisão e *Best* para a estratégia de escolher a divisão em cada nó. Outra configuração foi a da profundidade máxima da árvore, que foi a de expandir os nós até que todas as folhas contenham menos do que 2 filhos.

O método “RandomForestClassifier” do módulo “ensemble” do Sklearn foi utilizado para implementar o classificador *RandomForest*. O método também foi mantido em sua configuração padrão que utiliza o *Gini Impurity* e profundidade máxima da árvore de menos do que 2 filhos.

O modelo de *Logistic Regression* foi implementado pelo método “LogisticRegression”, que se encontra no módulo *“linear_model”* do Sklearn. O método foi utilizado com as configurações padrões da versão 0.23.2 da biblioteca Sklearn.

Para implementar o modelo de Rede Neural foi utilizada o método “MLPClassifier” do módulo “neural_network” do Sklearn. Foram utilizadas todas as configurações padrões do método “MLPClassifier”.

A implementação do modelo Naive Bayes foi através do método “GaussianNB” do módulo “naive_bayes” da bibliotéca Sklearn. Assim como os outros foram mantidas as configurações padrões.

As configurações completas de cada modelo foi coloca no Apêndice A deste trabalho.

3.3.4 Etapa 4

Esta etapa é similar à etapa 3, utilizou-se os mesmo algoritmos e configurações, porem ao invés de utilizar os dados da Ciência da Computação foram utilizados os dados de alunos dos cursos de engenharia. Estes dados foram coletados da base do Cobalto limitando o período de 2010 a 2018, pois em geral, os cursos de Engenharias iniciaram nesse período.

3.4 Considerações do Capítulo

Este Capítulo apresentou a metodologia utilizada para o desenvolvimento deste trabalho. Esta metodologia baseou-se na metodologia CRISP-DM. Foi apresentado todo o processo de compreensão dos dados e apresentando cada atributo utilizado, e ainda uma análise exploratória dos dados. Em seguida, foi abordado o processo de preparação dos dados que iniciou na seleção, até obter o conjunto final de dados. Por fim, as três fases de modelagem que este trabalho passou, para chegar na configuração final.

4 EXPERIMENTO E RESULTADOS

Este Capítulo corresponde à fase de avaliação da metodologia proposta no Capítulo 3. Nesta fase os resultados dos algoritmos de AM serão avaliados. Essa avaliação será dada no sentido de tentar responder as questões de pesquisa já citadas.

Este Capítulo está divido em 3 seções, para apresentar os resultados alcançados neste trabalho. A Seção 4.1 apresenta os experimentos para o curso de Ciência da Computação denominados experimentos 1 e 2. Esta Seção dos experimentos da Ciência da Computação é dividida em duas Seções que apresentam a *Performance* que avalia o desempenho alcançado em cada experimento e a outra Seção apresenta o *Feature Importance* que mostra os atributos com maior peso para determinar se o aluno evadiu ou não.

A Seção 4.2 apresenta o experimento 3 realizado com os doze cursos de Engenharias. São apresentados os resultados de *Performance* e *Feature Importance* da aplicação dos modelos aos dados para estes cursos.

4.1 Experimentos com o Curso de Ciência da Computação

Para alcançar os resultados deste trabalho foram realizados dois experimentos apenas com dados do curso de ciência da computação. Os dois experimentos utilizaram dados socioeconômicos e acadêmicos dos alunos, porém a diferença ficou em qual tipo de dado acadêmico foi utilizado.

O primeiro experimento (experimento 1) utilizou o mesmo conjunto de dados utilizado no trabalho apresentado em Costa et al. (2020). Neste experimento foram utilizados, como dado acadêmico, as médias dos 3 primeiros semestres do aluno.

O segundo experimento (experimento 2) também teve o mesmo conjunto de dados utilizado do trabalho apresentado em Costa et al. (2020), mas foram retiradas as médias dos 3 primeiros semestres e adicionado o fator de dificuldade apresentado na Seção 3.2.3.

Para os dois experimentos foram mantidas as mesmas configurações dos algoritmos e utilizando a validação cruzada com 10 conjuntos estratificada. Assim como no

trabalho em Costa et al. (2020), houve uma preocupação para manter a identidade dos alunos anônimas.

4.1.1 Performance

Antes de buscar os atributos que mais influência na predição dos algoritmos é preciso obter uma boa acurácia. Além da acurácia, os modelos serão avaliados pela precisão, *recall* e AUC. É possível obter cada uma dessa métricas a partir da matriz de confusão resultante da execução dos algoritmos.

Para comparar os experimentos será avaliado apenas o erro, ou seja, o FP que é quando o algoritmo prevê que o aluno vai evadir, mas não evade e o FN que é quando prevê que o aluno não vai evadir, mas ele evade. O mais problemático é o FN, pois se o modelo predizer que o aluno não vai evadir, mas na verdade esse aluno evade ele será um aluno a menos para o gestor poder intervir, mas também, é preciso ter atenção no FP, pois estaria gastando recursos da instituição com um aluno que não precisaria de ajuda.

A Figura 23 mostra a matriz de confusão do experimento 1 e 2 do algoritmo de **Árvore de Decisão**. O algoritmo obteve melhores resultados no experimento 1 alcançando um menor erro, com 63 FN e FP. Enquanto o experimento 2 alcançou o resultado de 72 FN e 78 FP, isso é 9 e 15 alunos a mais em relação ao experimento 1 respectivamente. No caso do experimento 1 63 alunos foram classificados em situação de evasão, mas na verdade não é um aluno em situação de evasão. Também no experimento 1 também 63 alunos não foram classificados como fora de risco de evasão, mas se encontra em situação de risco. O experimento 1 classificou melhor que o experimento 2 9 alunos falso negativos e 15 alunos falso positivos.

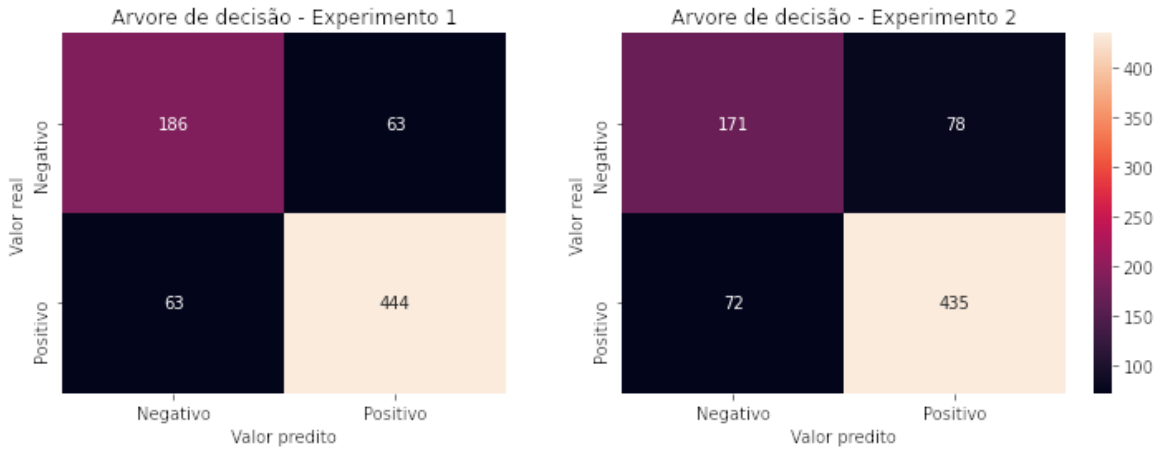

Figura 23 – Matriz de confusão dos experimentos 1 e 2 de Árvore de Decisão.

A Figura 24 mostra a matriz de confusão do experimento 1 e 2 dos algoritmos de **Floresta Aleatória**. Para estes dois algoritmos os resultados foram melhores para os dados do segundo experimento, onde o FN foi de 50 alunos e o FP foi de 59 alunos.

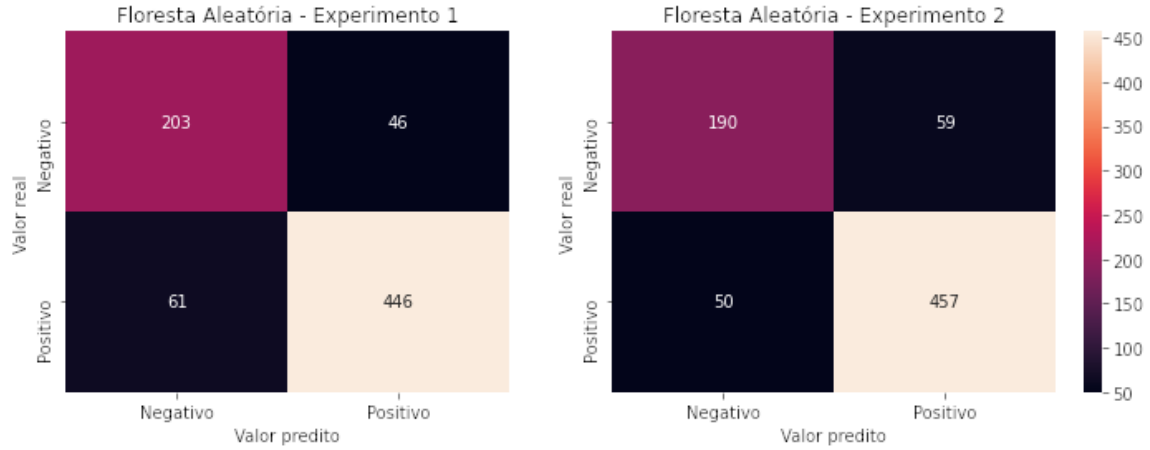

Figura 24 – Matriz de confusão dos experimentos 1 e 2 de Floresta Aleatória.

A matriz de confusão do experimento 1 e 2 do algoritmo de **Regressão Logística** são apresentados na Figura 25. Os resultados do experimento 2 para este algoritmo não foi equilibrado, pois ele obteve um FN baixo, mas o FP ficou muito alto. Já o resultado do experimento 1 ficou mais equilibrado com 49 FN e 50 FP.

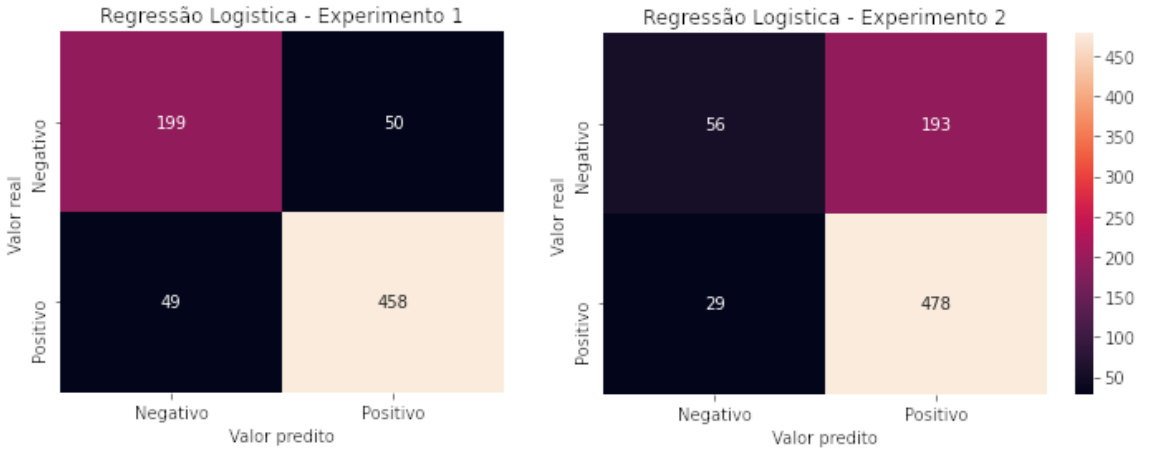

Figura 25 – Matriz de confusão dos experimentos 1 e 2 de Regressão Logística.

A Figura 26 mostra a matriz de confusão do experimento 1 e 2 do algoritmo de **Redes Neurais**. Para este algoritmo o resultado do FN foi melhor no experimento 2, mas o resultado do FP foi melhor no experimento 1.

Por fim, a Figura 27 mostra a matriz de confusão do experimento 1 e 2 do algoritmo **Naive Bayes**. No caso deste algoritmo o que chamou a atenção a diferença dos resultados do FP e FN do experimento 1 para os do experimento 2.

É difícil avaliar os resultados olhando apenas para o FP e FN. Por isso os resultados serão avaliados pela acurácia (accuracy), precisão (precision), sensibilidade (recall) e AUC.

A Tabela 19 apresenta os resultados do experimento 1, onde foi utilizado as médias dos 3 primeiros semestres. O algoritmo que obteve a melhor AUC foi o Naive Bayes com 88,07% e o pior foi Árvore de Decisão com 81,14%. Em relação ao Recall

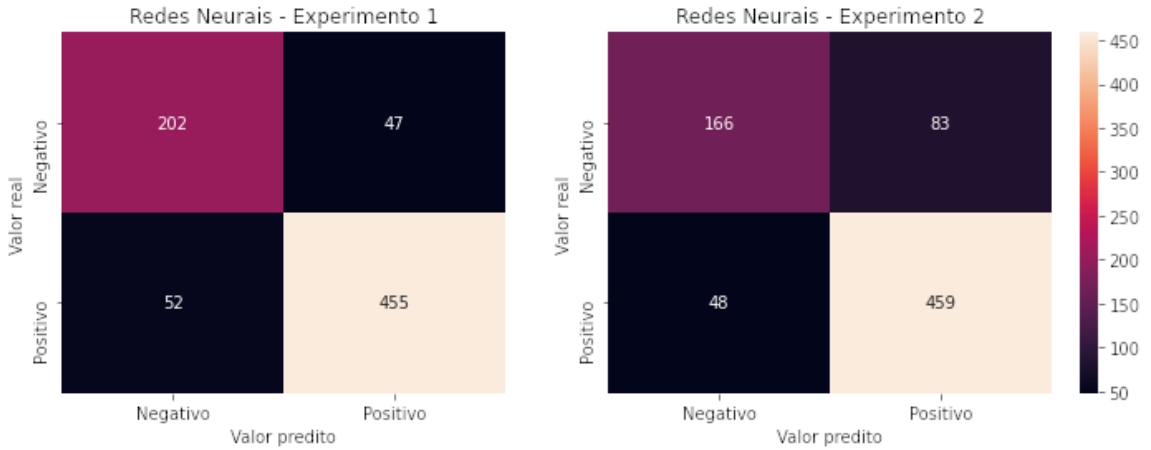

Figura 26 – Matriz de confusão dos experimentos 1 e 2 de Redes Neurais.

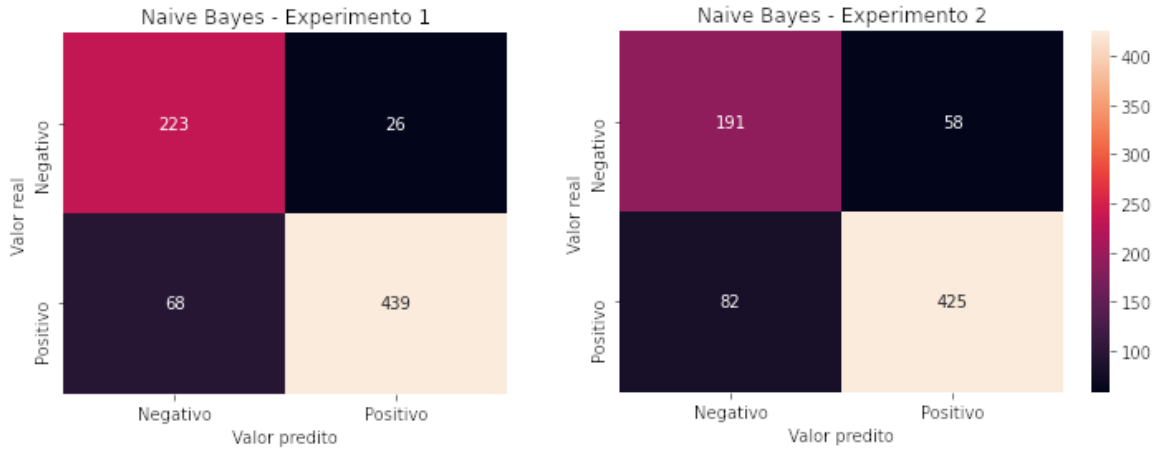

Figura 27 – Matriz de confusão dos experimentos 1 e 2 do algoritmo Naive Bayes.

o modelo que se saiu melhor foi o de Regressão logística com 90,34% e o pior foi Naive bayes com 86,59%. Redes Neurais foi o modelo que obteve o resultado mais equilibrado obteve o segundo melhor desempenho tanto na AUC quanto no Recall.

Tabela 19 – Resultado da execução dos algoritmos com os dados do experimento 1.

Algoritmo	Accuracy	Precision	Recall	AUC
Árvore de decisão	83.33%	87.57%	87.57%	81.14%
Floresta Aleatória	85.85%	90.65%	87.97%	84.75%
Regressão Logística	86.90%	90.16%	90.34%	85.13%
Redes Neurais	86.90%	90.64%	89.74%	85.43%
Naive Bayes	87.57%	94.41%	86.59%	88.07%

A Tabela 20 mostra os resultados do experimento 2, que utilizou o coeficiente de dificuldade dos 3 primeiros semestres. A melhor AUC no segundo experimento foi do modelo de Floresta Aleatória, com 83,22%, e o pior foi o de Regressão logística, com 58,39%. Vale destacar que o modelo de Regressão logística teve o maior Recall para este experimento 94,28%, porém o modelo não aprendeu, pois, a acurácia foi de 70,63%, que fica 3,13% acima de um modelo que se aposta sempre na evasão.

Tabela 20 – Resultado da execução dos algoritmos com os dados do experimento 2.

Algoritmo	Accuracy	Precision	Recall	AUC
Árvore de decisão	80.16%	84.80%	85.80%	77.24%
Floresta Aleatória	85.58%	88.57%	90.14%	83.22%
Regressão Logística	70.63%	71.24%	94.28%	58.39%
Redes Neurais	82.67%	84.69%	90.53%	78.60%
Naive Bayes	81.48%	87.99%	83.83%	80.27%

4.1.2 Feature Importance

Nesta Seção é apresentada a técnica de *Feature Importance* para analisar quais são os atributos mais relevantes para o modelo que obteve melhor desempenho para cada experimento. Esta técnica atribui uma pontuação a cada atributo com base na sua relevância para prever a variável de saída.

Para analisar a *Feature Importance* de um modelo é importante que este modelo tenha obtido um bom desempenho na predição, pois não teria relevância analisar a *Feature Importance* de modelos que não conseguem fazer uma boa predição.

4.1.2.1 Experimento 1

Para determinar qual modelo teve o melhor desempenho fui utilizada a métrica AUC, pois Jin Huang; Ling (2005) mostram que a AUC é uma métrica melhor do que a acurácia para esse tipo de problema. O modelo selecionado com a maior AUC a partir da Tabela 19 foi o Naive Bayes com 88,07%.

A Figura 28 mostra a *Feature Importance* do modelo Naive Bayes. O atributo com maior relevância para prever se o aluno evadiu ou não foi a média do terceiro semestre, seguida pela média do primeiro e segundo semestres. Outros atributos como gênero, se aluno teve curso superior anterior, e outros não tiveram muita relevância para a predição.

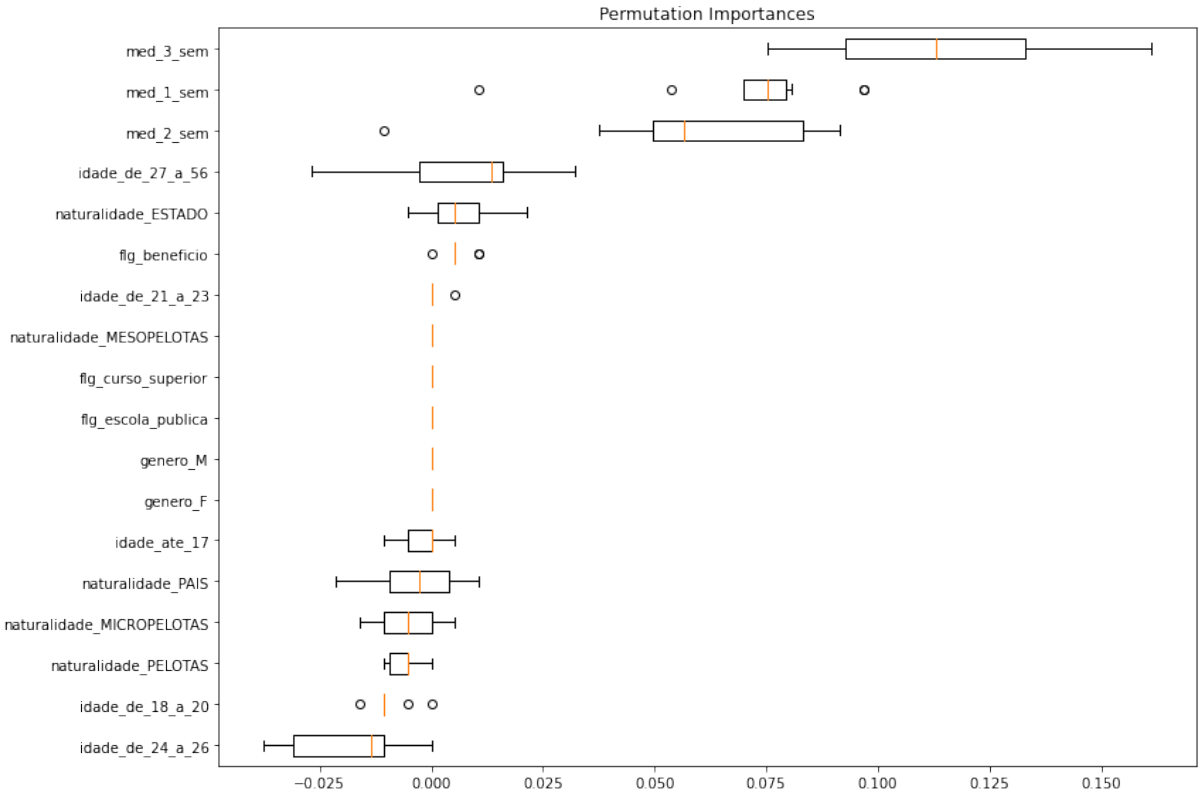

Figura 28 – *Feature Importance* do modelo Naive Bayes.

4.1.2.2 Experimento 2

O caso do experimento 2 a relação dos resultados apresentados na Tabela 20 mostra que o modelo de Floresta Aleatória obteve uma AUC de 83,2% e foi o melhor resultado para este experimento.

A Figura 29 mostra a *Feature Importance* do modelo de Floresta Aleatória. Onde o atributo com maior relevância para prever se o aluno evadiu ou não foi o coeficiente de dificuldade do terceiro semestre, seguido pelos coeficientes de dificuldade do segundo e primeiro semestres. Outros atributos como alunos com idade entre 27 e 56 anos, e alunos com idade entre 21 e 23 anos também foram relevantes para a predição.

4.2 Experimentos com os Cursos de Engenharia

O experimento 3 é uma união dos atributos do experimento 1 e 2, porém utilizando dados dos alunos de doze cursos de Engenharia da UFPel (apresentados na Tabela

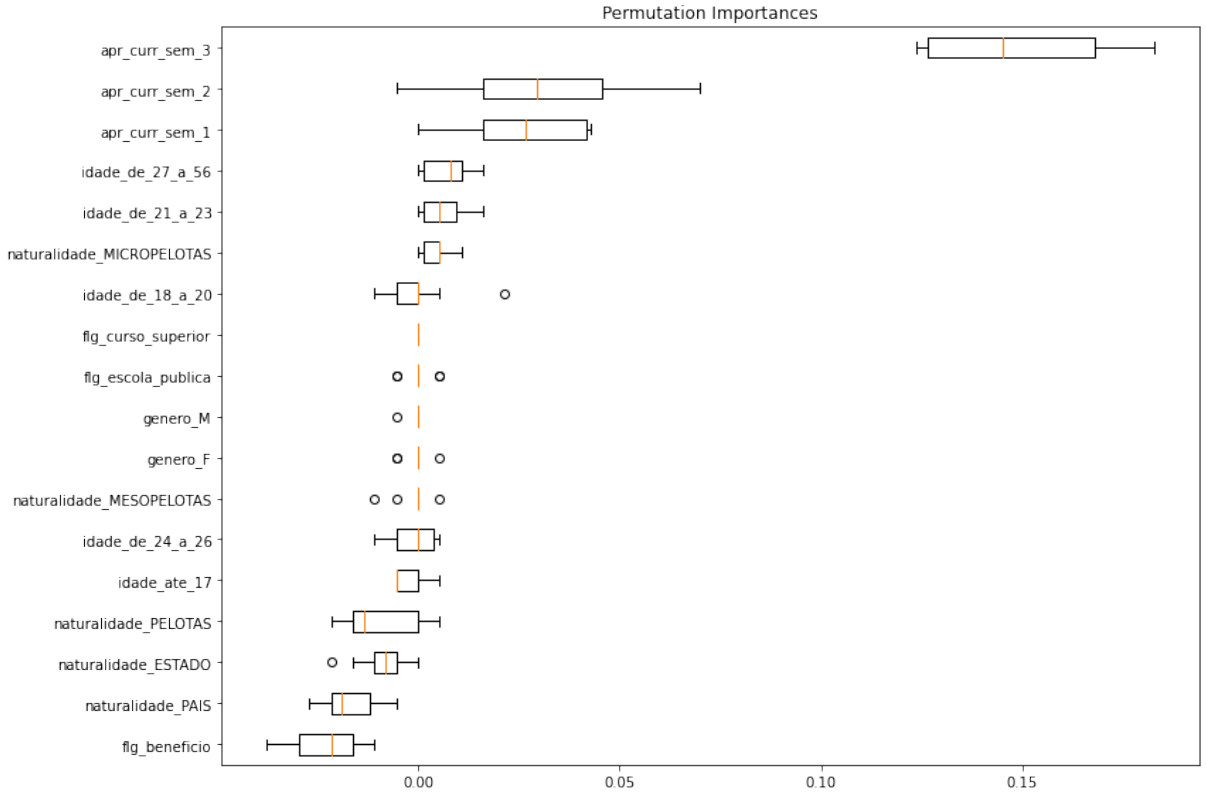

Figura 29 – *Feature Importance* do modelo de Floresta Aleatória.

1) de 2010 a 2018. O período dos dados teve que ser ajustado, pois o curso mais novo nesse conjunto iniciou em 2010. Este experimento teve um carácter exploratório no sentido de avaliar se as técnicas e metodologia utilizadas para o curso de Ciência da Computação possam ser aplicadas aos cursos de Engenharia de forma geral. Esse conjunto de dados passou pelo mesmo processo de preparação dos dados dos outros experimentos, ficando com 2640 instâncias de alunos diferentes e 22 atributos.

A Figura 30 mostra a matriz de confusão do modelo de **Árvore de Decisão** das Engenharias. O modelo teve um erro de 26,74% entre FP e FN, com 13,98% no FP e 12,77% no FN. Este modelo foi o que teve o menor erro de FN entre todos os modelos.

A Figura 31 mostra a matriz de confusão do modelo de **Floresta Aleatória** das Engenharias. O modelo teve o menor erro entres os modelos com 22,88% entre FP e FN, onde 13,86% correspondem a taxa do FN.

A Figura 32 mostra a matriz de confusão do modelo de **Regressão Logística** das Engenharias. O modelo teve um erro de 26,25% entre FP e FN, onde o FN foi de 13,83%.

A Figura 33 mostra a matriz de confusão do modelo de **Redes Neurais** das Engenharias. O modelo teve um erro de 25,30% entre FP e FN, onde 11,55% correspondem a taxa de FP e 13,75% a taxa de FN.

Por fim, a Figura 34 mostra a matriz de confusão do modelo de **Naive Bayes** das Engenharias. O modelo teve um erro de 25,80% entre FP e FN, onde 19,51% é o erro

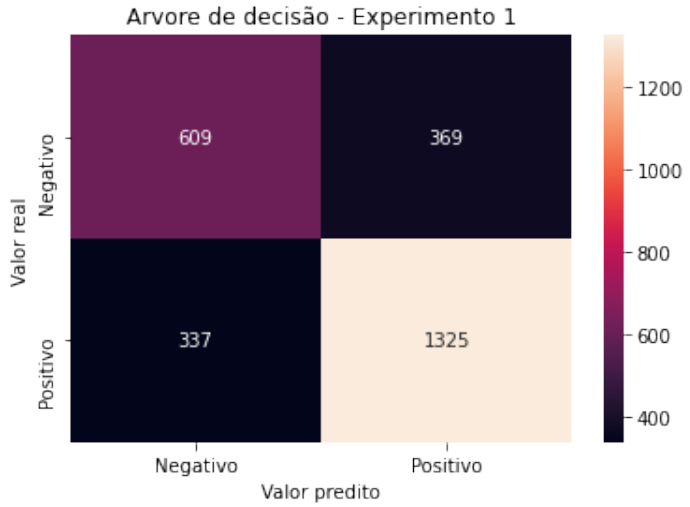

Figura 30 – Matriz de confusão do modelo de Árvore de Decisão das Engenharias

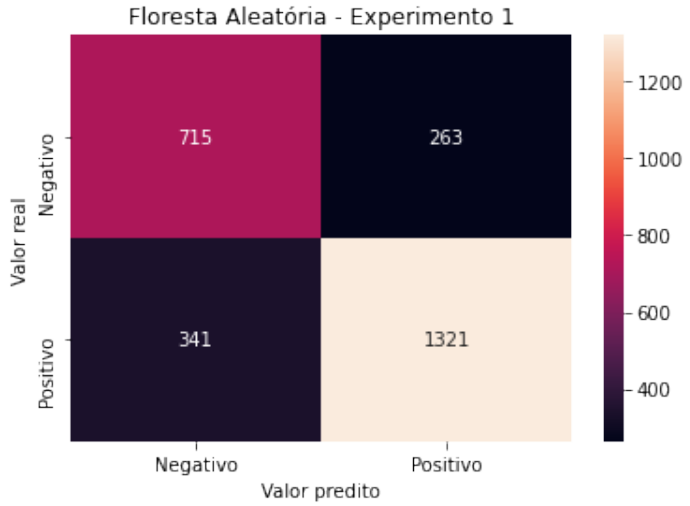

Figura 31 – Matriz de confusão do modelo de Floresta Aleatória das Engenharias

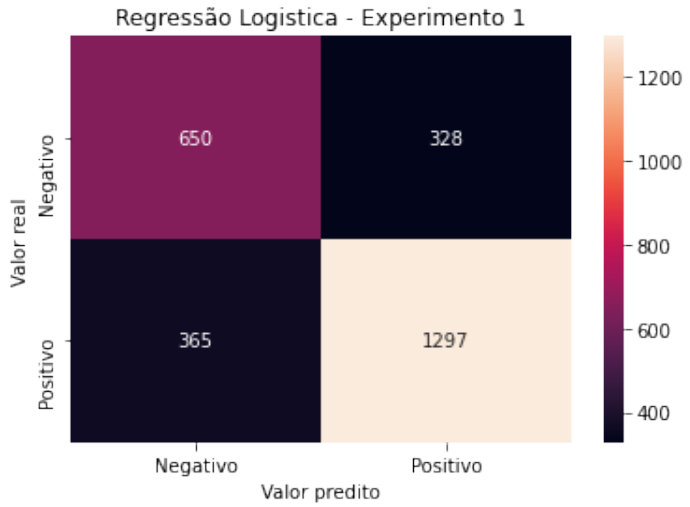

Figura 32 – Matriz de confusão do modelo de Regressão Logística das Engenharias

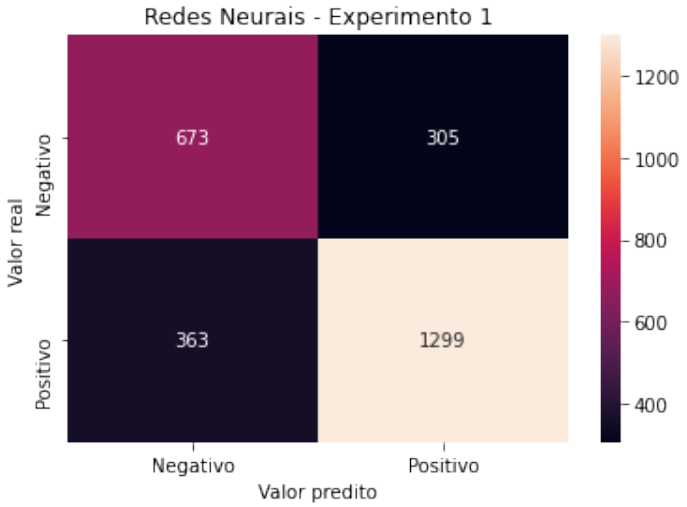

Figura 33 – Matriz de confusão do modelo de Redes Neurais das Engenharias

mais problemático para o problema da evasão de alunos. Este modelo teve o pior erro entre todos os modelos.

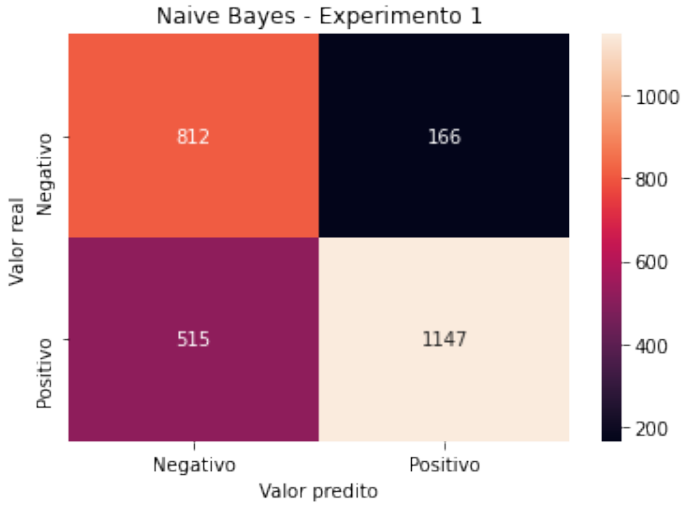

Figura 34 – Matriz de confusão do modelo de Naive Bayes das Engenharias

Assim como na Ciência da Computação serão utilizadas as seguintes métricas: acurácia, precisão, *recall* e AUC, calculadas a partir da matriz de confusão.

A Tabela 21 mostra os resultados do experimento 3, onde foi utilizado as médias dos três primeiros semestres e o coeficiente de dificuldade. O algoritmo que obteve a melhor AUC foi o de Floresta Aleatória com 76,30% e o pior foi o de Árvore de Decisão com 71,00%. Em relação ao Recall o modelo que se saiu melhor foi o de Árvore de Decisão com 79,72% e o pior foi Naive Bayes com 69,01%.

A melhor média entre as quatro métricas foi do modelo de Floresta Aleatória e, por isso, será analisado o *Feature Importance* desse modelo. A Figura 35 mostra a *Feature Importance* do modelo de Floresta Aleatória. O atributo com maior relevância para prever se o aluno evadiu ou não foi o coeficiente de dificuldade do terceiro

Tabela 21 – Resultado da execução dos algoritmos com os dados do experimento 3.

Algoritmo	Accuracy	Precision	Recall	AUC
Árvore de decisão	73.26%	78.22%	79.72%	71.00%
Floresta Aleatória	77.12%	83.40%	79.48%	76.30%
Regressão Logística	73.75%	79.82%	78.04%	72.25%
Redes Neurais	74.70%	80.99%	78.16%	73.49%
Naive Bayes	74.20%	87.36%	69.01%	76.02%

semestre, seguida pela média do terceiro e segundo semestre respectivamente. Outros atributos como gênero, se aluno teve curso superior anterior, e outros não tiveram muita relevância para a predição.

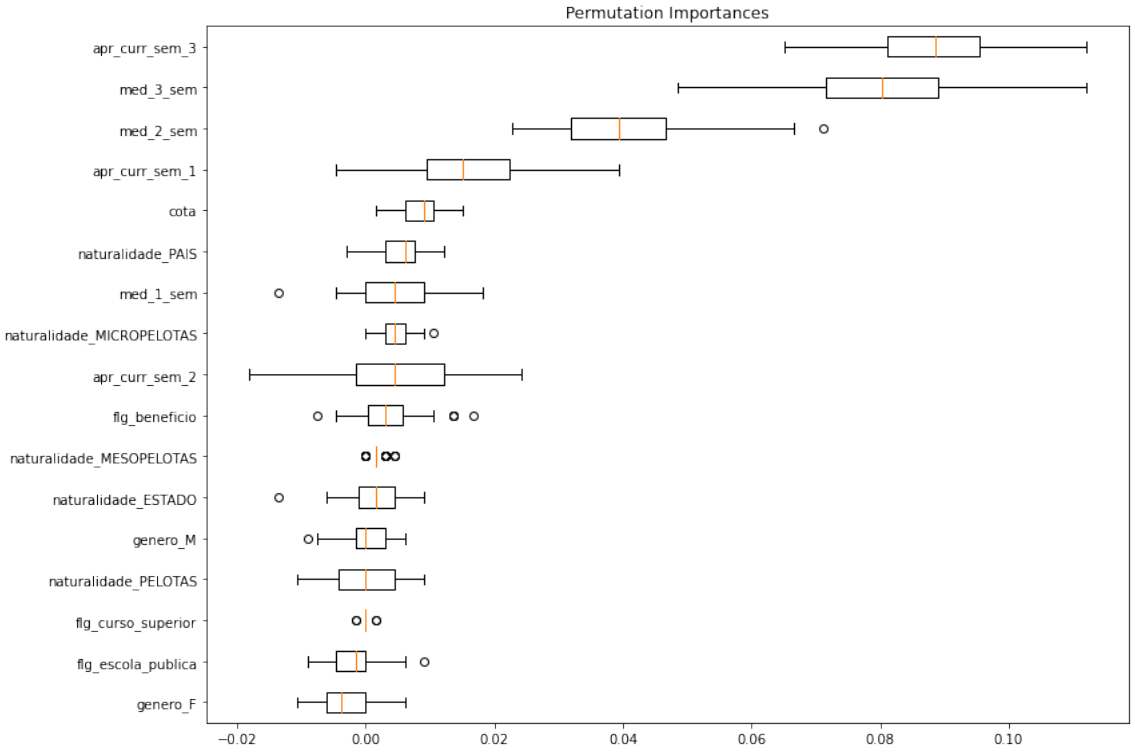

Figura 35 – *Feature Importance* do modelo de Floresta Aleatória.

4.3 Considerações do Capítulo

Neste Capítulo foram analisados os resultados obtidos com a execução dos modelos de aprendizado de máquina. Os resultados foram separados em dois experimentos (experimento 1 e 2) para o curso de Ciência da Computação e analisados individualmente. wOs modelos de aprendizado de máquina foi os mesmo para ambos os experimentos.

Os resultados do experimento 1 mostraram que é possível fazer a predição de alunos em risco de evasão através das médias dos 3 primeiros semestres com uma AUC de até 88%. Neste experimento o modelo com melhor resultado a partir da AUC foi o Naive Bayes com recall de 86,59%, precisão de 94,41% e acurácia de 87,57%.

Já os resultados do experimento 2, que usa o coeficiente de dificuldade, teve como melhor resultado da AUC (83%) foi o modelo de Floresta Aleatória. Diferente do experimento 1, o melhor modelo não teve o melhor Recall, mas teve a melhor precisão com 88,58%.

Os resultados do experimento 3 (doze cursos de Engenharia) mostram que tratar o curso individualmente é melhor do que tentar generalizar o modelo para atender todos os cursos. Isso se confirma pois embora o experimento tenha mais atributos e um volume de dados maior não alcançou resultados como os dois experimentos no curso de Ciência da Computação.

Neste trabalho foi dado uma atenção maior para o Recall, pois no contexto deste trabalho dizer que o aluno não vai evadir e esse aluno evadir é um problema mais do que o contrário. Por exemplo, se a universidade tivesse um programa de apoio que fosse colocar os alunos com probabilidade de evadir em monitorias, isso não seria uma perda para o aluno que não fosse evadir. Por isso o Recall é uma métrica importante neste caso e é preciso considerar ele mais que do que as outras métricas.

5 CONSIDERAÇÕES FINAIS

Este trabalho apresentou os resultados para a predição da evasão de alunos através de dados dos três primeiros semestres do curso de Ciência da Computação da UFPel que ingressaram entre os períodos de 2000 e 2018. Também foi realizado um experimento exploratório com um conjunto de dados de doze cursos de Engenharia que ingressaram de 2010 a 2018. Para fazer essa classificação foi utilizado o processo de KDD e algoritmos de aprendizagem de máquina. Os resultados mostraram que é possível prever a evasão com uma acurácia de até 87,57%.

Neste trabalho foram elaboradas duas questões que nortearam toda a pesquisa. Uma para tentar identificar quais atributos mais influenciam no processo de evasão de alunos no curso. A outra para identificar quais modelos ou técnicas que podem ser utilizados para realizar esta tarefa.

Para responder a primeira questão de pesquisa foi utilizado os *Feature Importance* resultante do treinamento dos algoritmos. O experimento 1 mostrou que dados acadêmicos dos alunos influenciam na predição de alunos em risco de evasão. O experimento 2, que utilizou o coeficiente de dificuldade, mostrou que este atributo também é o que mais influencia na predição. Em ambos os experimentos foi possível observar que atributos como o local de nascimento do aluno e a cota também são atributos que podem influenciar na predição.

Para responder a segunda questão de pesquisa foram apresentados os resultados que os algoritmos alcançaram. No experimento 1 o modelo que se destacou foi o Naive Bayes, que mostrou que é possível prever alunos em risco de evasão com uma acurácia de 87,57%, e um Recall e AUC de 86,59% e 88,07% respectivamente. Já o experimento 2 o modelo que mostrou que é possível predizer alunos em risco de evasão com uma acurácia de 85,58% foi o de Floresta Aleatória, que teve um Recall e AUC de 90,14% e 83,22% respectivamente.

Para responder a terceira questão de pesquisa foi realizado um experimento que reuniu os dados de alunos de todos os cursos de engenharias. Os resultados para prever alunos em risco de evasão nos cursos das engenharias alcançaram uma acurácia, precision, recall e AUC de 77,12%, 83,40%, 79,48% e 76,30%, respectivamente,

com o modelo de Floresta Aleatória. Esses resultados mostram que tratar o curso individualmente é melhor que tentar generalizar o modelo para atender todos os cursos.

Como trabalhos futuros pretende-se expandir a análise deste modelo de predição para outros cursos, primeiramente da área de exatas e engenharias e após demais cursos. Desta forma, será possível analisar a viabilidade ou não deste modelo em outros cursos e caso necessário adaptá-lo. Além disso, pretende-se implantar os modelos desenvolvidos neste trabalho no sistema de gestão da UFPel, o Cobalto, para fornecer os recursos necessários para que professores e gestores possam planejar políticas e ações para combater a evasão na universidade.

REFERÊNCIAS

ANDIFES; ABRUEM; SESU; MEC. **Diplomação, Retenção e Evasão nos Cursos de Graduação em Instituições de Ensino Superior Públicas**. [S.l.]: Comissão Especial de Estudos sobre a Evasão nas Universidades Públicas . . . , 1996.

BAKER, R.; ISOTANI, S.; CARVALHO, A. Mineraçao de dados educacionais: Oportunidades para o brasil. **Brazilian Journal of Computers in Education**, [S.l.], v.19, p.03, 2011.

BAKER, R. et al. Data mining for education. **International encyclopedia of education**, [S.l.], v.7, n.3, p.112–118, 2010.

BARBOSA, A.; SANTOS, E.; PORDEUS, J. P. A machine learning approach to identify and prioritize college students at risk of dropping out. **Anais do XXVIII Simpósio Brasileiro de Informática na Educação (SBIE 2017)**, [S.l.], v.1, p.1497, 2017.

BELTRAN, C. A. R.; XAVIER-JÚNIOR, J. C.; BARRETO, C. A. S.; NETO, C. O. Plataforma de Aprendizado de Maquina para Detecção e Monitoramento de Alunos com Risco de Evasão. **Congresso Brasileiro de Informática na Educação (CBIE)**, [S.l.], p.1591, 2019.

BEZERRA, C. et al. Evasão Escolar: Aplicando Mineração de Dados para Identificar Variáveis Relevantes. **Anais do XXVII Simpósio Brasileiro de Informática na Educação (SBIE 2016)**, [S.l.], v.1, p.1096, 2016.

BRAZ, F.; CAMPOS, F.; STROELE, V.; DANTAS, M. An Early Warning Model for School Dropout: a Case Study in E-learning Class. **Simpósio Brasileiro de Informática na Educação (SBIE)**, [S.l.], p.1441, 2019.

BURLAMAQUI, A. et al. Utilização de técnicas de mineração de dados educacionais para predição de desempenho de alunos de EaD em ambientes virtuais de aprendizagem. **Simpósio Brasileiro de Informática na Educação (SBIE)**, [S.l.], v.1, p.1527, 2017.

CALIXTO, K.; SEGUNDO, C.; GUSMÃO, R. P. de. Mineração de dados aplicada a educação: um estudo comparativo acerca das características que influenciam a evasão escolar. **Anais do XXVIII Simpósio Brasileiro de Informática na Educação (SBIE 2017)**, [S.l.], v.1, p.1447, 2017.

CARRANO, D.; ALBERGARIA, E. T. de; INFANTE, C.; ROCHA, L. Combinando Técnicas de Mineração de Dados para Melhorar a Detecção de Indicadores de Evasão Universitária. **Congresso Brasileiro de Informática na Educação (CBIE)**, [S.l.], p.1321, 2019.

COSTA, A. G. d. et al. Prediction analysis of student dropout in a Computer Science course using Educational Data Mining. In: XV LATIN AMERICAN CONFERENCE ON LEARNING TECHNOLOGIES (LACLO), 2020., 2020. **Anais...** [S.l.: s.n.], 2020. p.1–7.

COSTA, A. G. d.; PRIMO, T. T.; MATTOS, J. C. B. Análise da Predição de Evasão de Alunos da UFPel Utilizando Mineração de Dados Educacionais. **XXII Encontro de Pós-Graduação**, [S.l.], 2020.

COSTA, E.; BAKER, R. S. J.; AMORIM, L.; MAGALHÃES, J. Mineração de Dados Educacionais: Conceitos, Téc- nicas, Ferramentas e Aplicações. **Jornada de Atualização em Informática na Educação (JAIE)**, [S.l.], v.d, p.1–29, 2012.

DECRETO REUNI. Accessado em: 22-02-2021, http://www.planalto.gov.br/ccivil_03/_ato2007-2010/2007/decreto/d6096.htm.

DETONI, D.; CECHINEL, C.; ARAÚJO, R. Modelagem e Predição de Reprovação de Acadêmicos de Cursos de Educação a Distância a partir da Contagem de Interações. **Revista Brasileira de Informática na Educação**, [S.l.], v.23, 2015.

FERNANDES, W. L.; PITANGUI, C.; VIVAS, A.; ASSIS, L. Previsão de Desempenho de Estudantes usando o Algoritmo de Classificação Associativa. **Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação (CBIE 2017)**, [S.l.], v.1, p.734, 2017.

GOLDSCHMIDT, R.; BEZERRA, E.; PASSOS, E. Data mining: conceitos, técnicas, algoritmos, orientações e aplicações. **Rio de Janeiro-RJ: Elsevier**, [S.l.], p.56–60, 2015.

HAN, J.; PEI, J.; KAMBER, M. **Data mining**: concepts and techniques. [S.l.]: Elsevier, 2011.

HART, A. Mann-Whitney test is not just a test of medians: differences in spread can be important. **Bmj**, [S.l.], v.323, n.7309, p.391–393, 2001.

HOSMER JR, D. W.; LEMESHOW, S.; STURDIVANT, R. X. **Applied logistic regression**. [S.l.]: John Wiley & Sons, 2013. v.398.

IBGE. **Instituto Brasileiro de Geografia e Estatística**. Disponível em: <https://biblioteca.ibge.gov.br/index.php/biblioteca-catalogo?view=detalhes&id=22269>. Acesso em: 23.02.2021.

INEP. **INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA. Sinopses Estatísticas da Educação Superior - Graduação**. Disponível em: <http://portal.inep.gov.br/web/guest/sinopses-estatisticas-da-educacao-superior>. Acesso em: 16 08. 2019.

Jin Huang; Ling, C. X. Using AUC and accuracy in evaluating learning algorithms. **IEEE Transactions on Knowledge and Data Engineering**, [S.l.], v.17, n.3, p.299–310, 2005.

JÚNIOR, J. G. d. O.; NORONHA, R. V.; KAESTNER, C. A. A. Análise da Correlação da Evasão de Cursos de Graduação com o Empréstimo de Livros em Biblioteca. **Anais dos Workshops do III Congresso Brasileiro de Informática na Educação (CBIE 2014)**, [S.l.], v.1, p.601, 2014.

KANTORSKI, G. et al. Predição da Evasão em Cursos de Graduação em Instituições Públicas. In: BRAZILIAN SYMPOSIUM ON COMPUTERS IN EDUCATION (SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO-SBIE), 2016. **Anais...** [S.l.: s.n.], 2016. v.27, p.906.

KOEDINGER, K. R.; CUNNINGHAM, K.; SKOGSHOLM, A.; LEBER, B. An open repository and analysis tools for fine-grained , longitudinal learner data. **Proceedings of International Conference on Educational Data Mining**, [S.l.], p.157–166, 2008.

LANES, M.; ALCÂNTARA, C. Predição de Alunos com Risco de Evasão: estudo de caso usando mineração de dados. In: BRAZILIAN SYMPOSIUM ON COMPUTERS IN EDUCATION (SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO-SBIE), 2018. **Anais...** [S.l.: s.n.], 2018. v.29, p.1921.

LIMA BRITO, M. I. de. **Implementação do REUNI na UnB (2008 – 2011)**: limites na ampliação de vagas e redução da evasão. 2013. Dissertação (Mestrado em Ciência da Computação) — Universidade de Brasília (UnB), Universidade de Brasília (UnB).

LYKOURENTZOU, I. et al. Dropout prediction in e-learning courses through the combination of machine learning techniques. **Computers & Education**, [S.l.], v.53, p.950–965, 2009.

MANHÃES, L. M. B. et al. Previsão de Estudantes com Risco de Evasão Utilizando Técnicas de Mineração de Dados. **Anais do XXII SBIE - XVII WIE**, [S.l.], p.150–159, 2011.

PASCOAL, T.; BRITO, D. M. de; ANDRADE, L.; RÊGO, T. G. do. Evasão de estudantes universitários: diagnóstico a partir de dados acadêmicos e socieconômicos. **Anais do XXVII Simpósio Brasileiro de Informática na Educação (SBIE 2016)**, [S.l.], v.1, p.926, 2016.

PAZ, F.; CAZELLA, S. Identificando o perfil de evasão de alunos de graduação através da Mineração de dados Educacionais: um estudo de caso de uma Universidade Comunitária. **Congresso Brasileiro de Informática na Educação (CBIE)**, [S.l.], v.1, p.624, 2017.

QUEIROGA, E.; CECHINEL, C.; ARAÚJO, R. Um Estudo do Uso de Contagem de Interações Semanais para Predição Precoce de Evasão em Educação a Distância. In: WORKSHOPS DO CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2015. **Anais...** [S.l.: s.n.], 2015. v.4, p.1074.

RAMOS, J. L. C. et al. Um estudo comparativo de classificadores na previsão da evasão de alunos em EAD. **Anais do XXIX Simpósio Brasileiro de Informática na Educação (SBIE 2018)**, [S.l.], v.1, p.1463, 2018.

RAPIDMINER. **RapidMiner Best Data Science & Machine Learning Plataform**. [Online; accessed 22-março-2021]. Disponível em: <https://rapidminer.com/>.

RIGO, S. J.; CAMBRUZZI, W.; BARBOSA, J. L.; CAZELLA, S. C. Aplicações de Mineração de Dados Educacionais e Learning Analytics com foco na evasão escolar: oportunidades e desafios. **Revista Brasileira de Informática na Educação**, [S.l.], v.22, p.132–146, 2014.

ROMERO, C.; VENTURA, S. Educational Data Mining: A Review of the State-of-the-Art. **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, [S.l.], v.XX, p.18–20, 2013.

SALES, F. et al. Evasão no Ensino Básico da Rede Pública Municipal de Juiz de Fora: uma Análise com Mineração de Dados. **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)**, [S.l.], p.1371, 2019.

SANTOS, R. N. dos; SIEBRA, C. d. A.; OLIVEIRA, E. S. Uma Abordagem Temporal para Identificação Precoce de Estudantes de Graduação a Distância com Risco de Evasão em um AVA utilizando Árvores de Decisão. **Anais dos Workshops do III**

Congresso Brasileiro de Informática na Educação (CBIE 2014), [S.l.], v.1, p.262, 2014.

SHAPIRO, S. S.; WILK, M. B. An analysis of variance test for normality (complete samples). **Biometrika**, [S.l.], v.52, n.3/4, p.591–611, 1965.

SILVA, F.; SILVA, J. D.; SILVA, R.; FONSECA, L. C. Um modelo preditivo para diagnóstico de evasão baseado nas interações de alunos em fóruns de discussão. **Anais do XXVI Simpósio Brasileiro de Informática na Educação (SBIE 2015)**, [S.l.], v.1, p.1187, 2015.

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

UFPEL, U. F. d. P. **REGULAMENTO DO ENSINO DE GRADUAÇÃO DA UFPEL**. Accessado em: 22-02-2021, https://wp.ufpel.edu.br/scs/files/2018/09/SEI_Resolução-29.2018-Regulamento-Ensino-de-Graduação-I.pdf.

VIGLIONI, G. M. C. **Metodologia Para Previsão De Demanda Ferroviária**. 2007. Dissertação (Mestrado em Ciência da Computação) — Instituto Militar de Engenharia, Praça General Tibúrcio, 80 – Praia Vermelha Rio de Janeiro – RJ; CEP: 22.290-270.

WAGNER, C. H. Simpson's paradox in real life. **The American Statistician**, [S.l.], v.36, p.46–48, 1982.

Apêndices

APÊNDICE A – Configurações dos modelos

Arvore de decisão:

```
DecisionTreeClassifier(
    ccp_alpha=0.0,
    class_weight=None,
    criterion='gini',
    max_depth=None,
    max_features=None,
    max_leaf_nodes=None,
    min_impurity_decrease=0.0,
    min_impurity_split=None,
    min_samples_leaf=1,
    min_samples_split=2,
    min_weight_fraction_leaf=0.0,
    presort='deprecated',
    random_state=None, splitter='best'
)

RandomForestClassifier(
    bootstrap=True,
    ccp_alpha=0.0,
    class_weight=None,
    criterion='gini',
    max_depth=None,
    max_features='auto',
    max_leaf_nodes=None,
    max_samples=None,
    min_impurity_decrease=0.0,
    min_impurity_split=None,
    min_samples_leaf=1,
    min_samples_split=2,
    min_weight_fraction_leaf=0.0,
    n_estimators=100,
```

```
    n_jobs=None,
    oob_score=False,
    random_state=None,
    verbose=0,
    warm_start=False
)
```

Regressão Logistica:

```
LogisticRegression(
    C=1.0,
    class_weight=None,
    dual=False,
    fit_intercept=True,
    intercept_scaling=1,
    l1_ratio=None,
    max_iter=100,
    multi_class='auto',
    n_jobs=None,
    penalty='l2',
    random_state=None,
    solver='lbfgs',
    tol=0.0001,
    verbose=0,
    warm_start=False
)
```

Redes Neurais:

```
MLPClassifier(
    activation='tanh',
    alpha=0.0001,
    batch_size='auto',
    beta_1=0.9,
    beta_2=0.999,
    early_stopping=False,
    epsilon=1e-08,
    hidden_layer_sizes=(100,),
    learning_rate='constant',
    learning_rate_init=0.001,
    max_fun=15000,
```

```
    max_iter=200,
    momentum=0.9,
    n_iter_no_change=10,
    nesterovs_momentum=True,
    power_t=0.5,
    random_state=None,
    shuffle=True,
    solver='adam',
    tol=0.0001,
    validation_fraction=0.1,
    verbose=False,
    warm_start=False
)
```

Naive Bayes:

```
GaussianNB(priors=None, var_smoothing=1e-09)
```