

# Gerador de Grades Horárias Utilizando Inteligência Artificial

1<sup>st</sup> Matheus Rodrigues Borges Antonio

*Graduando Bacharelado Interdisciplinar em Ciência e Tecnologia*

*Universidade Federal de São Paulo*

São José dos Campos, Brasil

mantonio@unifesp.br

**Abstract**—Esse documento é um relatório de formalização do projeto de Geração de Grades Horárias com IA. Trata-se de uma abordagem que utiliza regras de associação juntamente com estratégias de resolução de timetabling problems para gerar uma grade para o semestre que não haja conflito de aulas de acordo com a matriz horária e que maximize os ganhos de escolha das disciplinas, promovendo um maior bem-estar do corpo discente e assim auxiliando na redução da evasão no ensino superior.

**Index Terms**—grades, IA, regras, associação, shceduling, timetabling, universidade

## I. MOTIVAÇÃO

A motivação primária desse projeto é realizar uma implementação usando conceitos vistos durante a disciplina de forma a auxiliar na vida dos estudantes da universidade de forma direta e visível. Busca-se tornar esse trabalho acessível para todo e qualquer aluno da faculdade de forma que haja o pleno entendimento de seu funcionamento e de suas características, para que os conceitos também sejam aprendidos, não somente pelo corpo discente, mas também por todo e qualquer indivíduo que acesse esse artigo e os seguintes, democratizando o acesso à ciência e à pesquisa.

## II. INTRODUÇÃO

### A. Sobre a evasão no ensino superior

Um fenômeno amplamente discutido no âmbito acadêmico é a evasão de alunos durante a graduação. De acordo com o professor Roberto Leal Lobo e Silva Filho [1], ex-reitor da USP, a evasão só leva a perdas: sociais, de recurso e de tempo de todos os envolvidos - discentes e docentes. De acordo com dados do INEP de 2016 [2], 27% das vagas nas universidades estão inativas, e mais de 15% dos alunos abandonaram seus cursos. E esse abandono, somente no ano de 2016, gerou um prejuízo de mais de 17 bilhões de reais no Brasil. Que a evasão é uma problemática persistente em nosso sistema educacional já sabemos, mas por que esse evento ocorre? Quais são possíveis meios de reduzir esse efeito nas universidades, sobretudo as públicas?

### B. As causas

Quando paramos para refletir sobre os motivos da evasão escolar, diversas razões aparecem em mente: financeiro, localização, tempo, conciliação com trabalho, dificuldades nas disciplinas, etc. Mas o que as pesquisas nos revelam? De

acordo com um artigo produzido por um grupo de doutores da PUCRS [3], dentre os motivos mais frequentes para a evasão no ensino superior, estão: problemas financeiros para manter os estudos, indecisão profissional e indecisão do futuro da carreira, seguido pelo baixo desempenho e reprovação nas matérias. Tirando os fatores externos, observamos que outros fatores como indecisão profissional e baixo desempenho podem ser intervindos com programas e ação direta das universidades. Nesse contexto, onde a Unifesp se encaixa e o que pode ser realizado?

### C. O âmbito que estamos

A Unifesp de São José dos Campos apresenta uma matriz de matérias muito ampla e isso permite aos alunos uma grande flexibilidade ao se montar a grade horária do semestre. O fato de haver uma formação interdisciplinar abre uma gama infinita de possibilidades de grades contendo disciplinas de diferentes áreas, como por exemplo, um discente seguindo para a área da computação cursando matérias de economia ou de biologia, ou um aluno de engenharia biomédica realizando matérias de computação. Todo esse cenário torna nossa universidade muito atrativa no estado inteiro e também em outras regiões, entretanto, num universo tão amplo com diversos caminhos para se seguir, qual é a combinação mais adequada que eu devo adotar para o meu semestre? E para minha formação como um todo? Relacionando a pluralidade do Instituto de Ciência e Tecnologia e as causas da evasão no ensino superior, há uma ação que pode ser tomada: auxiliar os discentes a escolher o melhor conjunto possível de matérias para o semestre subsequente. Sabemos muito bem que há uma matriz recomendada para se seguir de acordo com a formação específica a se adotar, mas nem sempre a disponibilidade dos horários e a realidade dos alunos permitem com que essa matriz seja seguida estritamente.

## III. CONCEITOS FUNDAMENTAIS

### A. O que é Inteligência Artificial?

Levando em conta todo esse contexto citado acima, podemos levantar estratégias para buscar a melhor forma de se obter uma grade ideal para cada aluno. Uma possível solução é a implementação de um projeto que utilize inteligência artificial para montar uma grade. Mas antes, faz-se importante explicar o que é IA e como podemos implementar esse conceito

em termos gerais e em específico no projeto de grades. Basicamente, a inteligência artificial [4] é um conjunto de tecnologias e técnicas que permitem com que um computador realize funções mais avançadas, antes só possíveis com a intervenção humana. Como exemplo, temos IA que consegue classificar fotos, analisar dados e fazer recomendações, como é o exemplo do gerador de grade. É um artifício poderoso e por ter esse potencial gigantesco de exploração, é normal se indagar como que a inteligência artificial funciona. Ela não é um monólito: a IA abrange diversos campos, como ciência da computação, dados estatísticos, neurociência e muitas outras áreas. Para que essa tecnologia consiga reproduzir de forma coesa ações humanas, é necessário que ela aprenda, geralmente por aprendizado de máquina e aprendizado profundo, que, de forma geral, consiste no treinamento do modelo partindo de um conjunto de dados.

#### B. Desafio: conjunto de dados

Agora que já sabemos todos esses conceitos, precisamos trazer essas ideias para uma possível solução para a problemática da geração de grades horárias. Como podemos implementar um modelo que use de inteligência artificial e que aprenda, partindo de um conjunto de dados, como montar uma grade horária de acordo com os dados especificados por um aluno qualquer da Unifesp. Pode parecer uma tarefa simples, mas o grande desafio não consiste em treinar o modelo para realizar tal tarefa, mas sim como podemos mexer com os dados relacionados às grades horárias de semestres passados da faculdade? Para nos auxiliar nessa tarefa, pode ser interessante buscar na literatura e no mercado projetos que já implementaram ideias semelhantes, para nortearmos nossa metodologia e como iremos agir.

#### C. Conhecendo as Regras de Associação

Uma possível maneira de trabalharmos com o conjunto de dados e obtermos as informações que irão guiar as recomendações de grades do nosso modelo é utilizando regras de associação. Basicamente, é um modelo não supervisionado associativo, que busca encontrar relações de ocorrências frequentes em um conjunto de dados. Uma regra de associação apresenta o formato:

- $X \rightarrow Y$ , onde  $X$  e  $Y$  são conjuntos distintos;
- Um exemplo cabível para a abordagem usada por esse projeto seria: Álgebra Linear - Inteligência Artificial. As regras de associação possuem métricas de avaliação, sendo as mais utilizadas e que serão empregadas nesse projeto: suporte, confiança e lift. O suporte é o percentual que mede a quantidade de transações que possuem os elementos de  $X$  e  $Y$ . A confiança, por sua vez, é o percentual que mede a quantidade de vezes que os elementos de  $Y$  aparecem juntos com os elementos de  $X$  nas transações. O lift é uma medida mais específica dada pela razão entre a taxa de confiança da regra pelo suporte do consequente, que no caso é o  $Y$ , é o lift que ajuda a revelar regras menos visíveis e que podem agregar muito valor na descoberta de conhecimento.

#### D. Algoritmo Apriori [?]

O algoritmo Apriori é um dos mais conhecidos quando o assunto é a extração de regras de associação e é ele que será utilizado nessa abordagem. O algoritmo se baseia na ideia de que se um conjunto de itens é frequente, então os subconjuntos gerados a partir desse item também são frequentes. Partindo dessa premissa, é possível gerar *itemsets* candidatos e ir eliminando os que não forem frequentes. Dessa forma, pode-se ir elevando a ordem dos conjuntos de itens, até que se encontre um conjunto vazio, determinando o tamanho máximo do conjunto de itens frequentes que compõem as regras para os parâmetros mínimos determinados.

### IV. TRABALHOS RELACIONADOS

A ideia de criar um gerador de grades horárias que utilize de IA provavelmente já foi explorada antes por diversos pesquisadores, estudantes e até empresas ao redor do mundo. Nesse momento, faz-se interessante buscar artigos na literatura que descrevam os passos realizados, de forma a auxiliar no norteamo e desenvolvimento desse projeto. Além disso, também é interessante perceber que existem soluções para problemas semelhantes ao abordado no mercado. Como exemplo, temos o sistema Cronos [5], software brasileiro capaz de organizar uma matriz horária de acordo com a disponibilidade e restrição de cada professor com suas respectivas aulas, utilizando de algoritmos que resolvam problemas conhecidos na literatura como *School Timetabling Problem* (STP). Apesar de não apresentar o mesmo contexto do problema abordado nesse projeto, devemos nos atentar que uma grade de aluno nada mais é que uma matriz horária reduzida, em que as restrições são os horários e os pré-requisitos das disciplinas. Sendo assim, realizar uma busca na literatura sobre os problemas que envolvem timetabling pode ser um caminho interessante a se seguir.

#### A. School Timetabling Problem usando meta-heurísticas [6]

Nessa dissertação de mestrado de um aluno da UFRGS, há a apresentação do problema STP como sendo uma problemática difícil de ser solucionada pela quantidade de características a serem levadas em consideração, tal como aulas, turmas, professores, salas de aula, etc. Para encontrar uma solução para essa problemática que é enfrentada todo começo de semestre, o autor nos apresenta a possibilidade do uso de meta-heurísticas, inclusive através de algoritmos vistos durante a disciplina de Inteligência Artificial, como algoritmos gulosos que seguem um critério, ou algoritmos genéticos. Entretanto, foco dessa dissertação é buscar uma solução utilizando da meta-heurística conhecida como *simulated annealing*, ou têmpera simulada (também visto em aula), de forma a aceitar formulação de grades piores do que no momento atual. Quanto pior for a solução vigente em relação à anterior, menor a probabilidade de se escolher um caminho naquela vizinhança, de forma a controlar as tomadas de decisão, possibilitando uma solução viável. Levando em conta as restrições do problema de timetabling, foi usado um critério guloso para escolha inicial, enquanto a têmpera simulada realizava melhorias nas escolhas.

#### *B. University Timetabling Problem (UTP) usando programação linear [7]*

Aqui também temos uma dissertação de mestrado de um aluno da UFRGS, mas desta vez o tema abordado é o problema de timetabling aplicado as universidades, em que grande parte dos alunos escolhem suas disciplinas. Essa abordagem é mais próxima da problemática em que estamos trabalhando e utiliza de pesquisa operacional, em específico programação linear inteira, para alcançar uma solução ótima. Por não usar de heurísticas, a solução se torna exata, entretanto há um custo computacional muito maior e os modelos matemáticos criados para a resolução são mais complexos. O artigo também realiza uma busca exaustiva na literatura e recolhe dados de resoluções já feitas para o problema UTP, nos revelando que as 4 formas mais comuns de resolução tratam-se de: programação linear inteira, algoritmo genético, têmpera simulada e busca tabu.

#### *C. Gerador de grades horárias na universidade utilizando algoritmos genéticos [8]*

Nessa dissertação de mestrado de uma aluna da UFU, há o método de algoritmos genéticos para a resolução do timetabling problem. Nessa abordagem, cada indivíduo da população possui uma grade de horários, composta por n-uplas de disciplina, professor, turno e valor de aptidão. A ideia básica deste algoritmo genético é semelhante aos demais: o algoritmo é iniciado com uma população inicial, em que cada indivíduo representa uma solução. Então, ocorre a reprodução para a obtenção de novas soluções, juntamente com os fatores genéticos inerentes, tal como evolução, mutação, cruzamento, elitismo, etc. Algumas soluções são escolhidas dentro das existentes para formar novas soluções melhores que atendam uma determinada função de adaptação e esse processo é repetido até que a condição inicial imposta seja satisfeita e uma solução ótima para o problema que englobe todas as restrições seja criada.

#### *D. Timetabling Problem usando busca Tabu [9]*

Esse artigo escrito por pesquisadores turcos mostram a abordagem da busca tabu para a resolução da problemática. Basicamente, uma busca tabu apresenta um comportamento semelhante com a têmpera simulada, em que a cada iteração, uma solução é escolhida como a atual e uma vizinhança é criada a partir dela. Na próxima iteração, o algoritmo irá executar um movimento para uma próxima solução. Os movimentos anteriores são armazenados na chamada lista tabu, para prevenir com que haja um looping infinito. Entretanto, há uma chance de que um movimento feito anteriormente seja uma solução global, portanto há um critério denominado critério de aspiração que permite com que um movimento saia da lista tabu para ser visitado novamente. O critério de aspiração implementado nessa abordagem do artigo é a seguinte: caso um movimento já explorado tenha um valor de função de objetivo menor do que a melhor solução atual, então tire-o imediatamente da lista tabu para que ele possa ser explorado novamente no futuro. Aliando isso juntamente com uma combinação de movimentos simples (alocar aula a

um espaço livre) e movimentos de troca (trocar duas aulas de lugar), juntamente com o critério de aspiração, há a obtenção de um resultado ótimo ou muito aproximado da solução ótima.

#### *E. Abordagem utilizando critérios gulosos para busca de solução [10]*

Mais um artigo de pesquisadores turcos, desta vez adotando critérios gulosos na busca pela solução ótima ou aproximadamente ótima na geração de grades. Foram desenvolvidas diversas heurísticas para a implementação do algoritmo, sendo elas divididas em duas partes: heurísticas de ordenação de disciplinas e heurísticas de ordenação de salas. Para nosso projeto de gerador de grade para os alunos, as heurísticas de ordenação de salas não são relevantes, visto que a própria instituição de ensino provê isso ao criar a matriz horária. Entretanto, algumas heurísticas da seleção de aulas merecem atenção especial, tal como: disciplinas com menos dias de aula (todas no mesmo dia), disciplinas que possuem menos alunos (menos concorrência), etc.

#### *F. Recomendação de horários e planejador de semestre usando algoritmos gulosos [11]*

Pesquisa feita por cientistas da Malásia. Esse projeto possui um foco maior num planejador de atividades a fim de auxiliar na produtividade dos alunos. Mesmo sendo um abordagem diferente do uso de buscas para resolução de problemas de scheduling, a metodologia empregada para definição de prioridades e as heurísticas usadas na busca podem ser estudadas e aplicadas neste trabalho. O algoritmo baseado em critérios gulosos adotado pelos pesquisadores em grande parte dos casos encontrava a solução ótima ou subótima e engloba até atividades extra-curriculares, tal como hobbies e prática de esportes, de fato uma abordagem mais complexa mas muito interessante ao mesmo tempo.

#### *G. Fusão de abordagens: algoritmos gulosos e genéticos [12]*

Projeto proposto por pesquisadores chineses, com uma abordagem interessante de juntar a solução local presente nos algoritmos gulosos com os métodos evolutivos presentes nos algoritmos genéticos. Assim como visto anteriormente, cada indivíduo é uma solução no formato de uma tupla, contendo as informações a serem usadas. A grande diferença nessa abordagem é que se utiliza de critérios gulosos para se criar a população inicial, dessa forma garantindo que o modelo obtenha a solução global ótima e evitando uma parada precoce antes dos critérios e restrições serem atendidos.

#### *H. Regras de associação para escolhas de disciplinas [13]*

Pesquisa realizada por um aluno da universidade de Cambridge, mostra a metodologia empregada para o uso de regras de associação em pesquisas educacionais. No projeto, as RA são usadas para tentar compreender melhor a escolha de disciplina por parte dos alunos. Durante a pesquisa, há a alavancagem das regras com maior suporte, confiança, lift e outros parâmetros. Chega-se a conclusão que há poucas regras que revelam informações inovadoras, mas as regras

de associação mostram-se como uma abordagem interessante quando não se sabe nada sobre os dados.

#### *I. Mineração de Dados e Regras de Associação na avaliação dos dados de estudantes [14]*

Essa é uma abordagem um pouco diferente feita por pesquisadores da Índia. Há uma análise sobre as notas de alunos durante a graduação e após a graduação, e são criadas regras de associação que tentam entender como essas notas se relacionam. Esse artigo é interessante para esse projeto pois há explicitamente a metodologia usada no pré-processamento e formatação dos dados, além de que o uso das notas pode ser aplicado nesta pesquisa.

#### *J. Sistema inteligente de sugestões de disciplinas usando regras de associação [15]*

Projeto desenvolvido por pesquisadores da Jordania. Essa pesquisa é muito próxima do viés abordado por este trabalho, com metodologias e motivações semelhantes. Trata-se de um modelo baseado em regras de associação que sugere quais matérias os alunos devem pegar baseando-se em decisões feitas por alunos em condições semelhantes. Os dados são pré-processados em transações, sendo que cada transação contém O ID do estudante, o semestre indicado, qual é o curso e a nota. Após essa transformação nos dados, o algoritmo de regras de associação *apriori* é utilizado para gerar as regras. Após isso, as regras são analisadas de acordo com a confiança e o suporte e regras não válidas são descartadas ou corrigidas. Caso as regras tragam inovação e utilidade, serão implementadas no modelo.

### V. OBJETIVO

Levando em conta as buscas feitas na literatura e ideias prévias, o objetivo desse trabalho é criar um gerador de grades horárias que utilize Inteligência Artificial em seu funcionamento em duas etapas distintas: primeiramente, utiliza-se regras de associação para determinar possíveis combinações boas de disciplinas em um semestre e após isso seria realizada uma busca gulosa, utilizando as informações descobertas com as regras de associação como heurística.

### VI. METODOLOGIA PRIMÁRIA

Primeiramente, nesta seção de metodologia, faz-se necessário explicar os motivos que levaram as escolhas de estratégias a se usar nesse projeto. Nesse caso, estamos lidando com um problema de scheduling consideravelmente menor do que os STP E UTP, com menos parâmetros e menos restrições. Nesse caso, a estratégia de regras de associação como heurística para determinar quais matérias se escolher em casos de restrição de matriz horária ou pré-requisitos mostra-se como uma ferramenta interessante para a abordagem. É importante, nesse momento, relembrar as etapas da mineração de dados vista em aula: conhecimento do domínio, pré-processamento, extração de padrões, pós-processamento e utilização do conhecimento. Para cada etapa, será empregada uma metodologia e uma forma de abordagem.

#### *A. Conhecimento do Domínio*

Essa etapa é, basicamente, tudo o que está sendo abordado nesse artigo de formalização de tema. É aqui em que há a pesquisa e a identificação do problema, tal como a definição do objetivo que queremos atingir e de que forma iremos alcançar essa meta. Nesse caso, o conhecimento do domínio se deu por meio de pesquisas e buscas na literatura sobre o problema abordado, tal como formas de solucioná-lo.

#### *B. Pré-Processamento*

Talvez essa seja a etapa mais desafiadora e longa desse projeto. O grande desafio aqui é a obtenção dos dados, já que é complexo ter o acesso à grades de semestre passadas de alunos. Para isso, fora empregado o seguinte método: um formulário foi disponibilizado aos alunos, para que eles informassem um semestre do ano passado, quais matérias foram selecionadas para compor a grade dese semestre e a relação de aprovação/reprovação/exclusão das unidades curriculares. Dessa forma, qualquer aluno que queira se voluntariar pode ajudar na criação desse projeto, afinal esse trabalho tem a principal motivação de auxiliar o corpo discente da faculdade. Além da obtenção dos dados brutos, é necessário realizar um polimento para que padrões sejam extraídos e que um algoritmo de criação de regras de associação seja aplicado, logo é necessário que haja um tratamento dos dados, formatando-os para o mesmo padrão (o formulário poderá haver respostas em formatos diferentes) e transformá-los para outro tipo de dado, preferencialmente em uma lista de listas, já que as regras de associação funcionam com base em transações.

#### *C. Extração de Padrões*

Nessa etapa, iremos escolher a tarefa como sendo regras de associação e o algoritmo *apriori*. O objetivo aqui é associar as escolhas de disciplinas em cada semestre entre si e também associar a taxa de aprovação da grade. Ou seja, para ilustrar com um exemplo: metade dos alunos que escolheram computação gráfica e compiladores no mesmo semestre reprovaram nas duas disciplinas. Logo, essa combinação talvez não seja a mais adequada. Agora, 75% dos que escolheram Geometria Analítica e Algoritmos e Estrutura de Dados foram aprovados nas duas matérias. Logo, essa combinação pode ser interessante na hora de montar a grade.

#### *D. Pós-processamento*

Agora que o algoritmo de regras de associação foi executado, é necessário analisar as associações descobertas e avaliar se as descobertas realmente trazem um conhecimento novo e útil para a implementação da solução. Como estamos trabalhando com regras de associação, faz-se importante atentar-se para o suporte, confiança e LIFT das regras, de forma a ter um equilíbrio entre o suporte e a inovação advinda da regra. Exemplo: 95% dos alunos que escolheram química geral, também escolheram linguagem de programação. Essa constatação não traz nenhuma novidade, visto que essas duas disciplinas são obrigatórias para todos os calouros no 1º semestre de curso. Entretanto, se eu descubro que 20% dos

alunos combinaram Análise de Investimento e Risco e AED2 e tiveram uma aprovação de 85%, com certeza essa descoberta traz um valor grande para a busca da solução.

### E. Utilização do Conhecimento

Após ter analisado profundamente as regras de associação descobertas e extraído as regras mais gerais e mais específicas baseando-se nos parâmetros, agora é necessário criar heurísticas com base nessas regras para que o algoritmo possa encontrar uma solução que, além de possível, seja ótima. Partindo da matriz horária do semestre, a grade recomendada por semestre e as heurísticas encontradas, o algoritmo de montagem de grade pode usar os seguintes critérios gulosos:

- Dado as restrições informados pelo aluno (número de aulas, turno, termo, etc), priorize as matérias recomendadas para o semestre de acordo com o termo do aluno;
- Caso haja algum conflito de horário ou de pré-requisito, utilize das heurísticas descobertas anteriormente para encontrar uma matéria que se encaixe adequadamente com a subgrade montada no item anterior;

A qualidade da grade montada pelo modelo poderá ser quantificada por meio de pesquisa entre os alunos dos diversos eixos da faculdade, de forma a ter uma espécie de metrica que determina o "quão próxima da realidade esta grade está". Uma forma de se realizar isso é por meio de um formulário onde são apresentadas grades geradas pelo modelo e os alunos votam com uma nota de acordo com a qualidade da grade para aquele semestre.

## VII. METODOLOGIA EXPERIMENTAL

### A. Base de Dados

Para que todos os elementos desse projeto sejam concretizados, é necessário uma base de dados consistente. Essa subseção está diretamente relacionada com a etapa de Pré-Processamento da mineração citada anteriormente e irá definir nossa base de dados a ser trabalhada. Como mencionado anteriormente, houve a criação de um formulário, disponibilizado entre os alunos da universidade alvo, no caso a Universidade Federal de São Paulo, campus São José dos Campos. A disponibilização foi realizada através de redes sociais e no formulário os discentes deveriam escolher entre um dos semestres disponíveis: 1º semestre de 2022 ou 2º semestre de 2022 e após isso, informar a grade montada no semestre e a relação de aprovação/reprovação/exclusão das disciplinas que compuseram essa grade. O formulário permite a exibição das respostas como uma planilha, o que permite uma manipulação mais simples e adequada dos dados. Entretanto, como esperado, os dados foram disponibilizados em diversos formatos e tipos. Sendo assim, foi necessário formatar as disciplinas da grade das respostas para que um padrão fosse formado e a formatação escolhida foi a seguinte:

- Forma genérica: Disciplina1, Disciplina1(relação de aprovação), ..., Disciplina\_n, Disciplina\_n(relação de aprovação)

- Exemplo prático: Inteligência Artificial, Inteligência Artificial(aprovado), Algoritmos e Estrutura de Dados II, Algoritmos e Estrutura de Dados II(aprovado)

Sendo assim, temos uma tupla composta pelo semestre indicado, as disciplinas, separadas por vírgulas, e cada matéria sucedida por ela mesma acrescida da relação de aprovação naquela matéria entre parenteses. Essa formatação foi escolhida já pensando na execução do algoritmo de regras de associação, em que cada linha da planilha será usada como uma transação e será comparada com as demais transações. Contudo, para que isso seja executado de forma adequada, é preciso que os dados sejam transformados, de forma que cada item da planilha seja transformada numa lista para que no final a saída seja uma lista de listas. Portanto, um código em python foi gerado para fazer as transformações necessárias, incluindo a remoção de colunas não utilizadas pelo algoritmo e a transformação em lista.

```
data_list = []
size = len(data)
for i in range(size):
    data_list.append(data.values[i, :])
```

Nesse trecho de código, uma lista vazia foi iniciada e através de um laço de repetição os dados presentes na planilha foram adicionados nessa lista. O resultado obtido possui este formato:

- array(['1º Semestre 2022', 'Cálculo em Várias Variáveis, Cálculo em Várias Variáveis(reprovado), Circuitos Digitais, Circuitos Digitais(reprovado), Microeconomia, Microeconomia(reprovado), Probabilidade e Estatística, Probabilidade e Estatística(aprovado), Algoritmos Estruturas de Dados I (excluido)'], dtype=object), ...

Aqui podemos constatar uma transformação a ser realizada, tendo em vista que todas as disciplinas compõem apenas um item da lista, quando o ideal para as regras de associação é a separação desses itens da transação:

- saída: [semestre, matérias + aprovação]
- ideal: [semestre, matéria1, matéria1\_aprovação, ..., matéria\_n, matéria\_n\_aprovação]

Levando isso em consideração, é necessário realizar outra transformação, para dividir as matérias em elementos distintos de cada lista dentro da lista maior. Para isso, utilizaremos a função split do python para, a cada vírgula, realizar uma separação interna nas listas:

```
data = []
for i in range(len(data_list)):
    data.append(data_list[i][1].split(','))
```

Dessa forma, a lista de listas estará pronta para a execução do algoritmo de regras de associação, com uma saída no formato:

- saída: [['Cálculo em Várias Variáveis', 'Cálculo em Várias Variáveis(reprovado)', 'Circuitos Digitais', 'Circuitos Digitais(reprovado)' ... ]

### B. Protocolo Experimental I: Modelo Geracional de Grades

Para criar o gerador que leva o tema deste projeto, teremos que nos atentar para duas coisas: as regras descobertas

utilizando o conjunto de dados criado e também como as disciplinas serão relacionadas entre si na hora da montagem da grade. Para isso, constata-se a seguinte estratégia:

- **Arranjo das Matérias:** para que o modelo possa funcionar de maneira ideal, é necessário que cada disciplina que componha a grade dos diversos bacharelados contidos no campus São José dos Campos seja organizada em tuplas contendo as informações requeridas para a escolha de prioridades. No bacharelado Interdisciplinar em Ciência e Tecnologia, são 6 grades diferentes para cada curso específico e 2 turnos, sendo assim faz-se interessante a criação de 6 conjuntos de tuplas, em que cada tupla é formada por: 'Nome da Disciplina', 'Pré-Requisitos', 'Quantas Matérias são requeridas por ela', 'Termo', 'Horário'. Dessa forma, poderá ser possível escalonar as matérias de acordo com os requisitos e as regras descobertas/banco de regras.

Para a realização do gerador de grades, foi levado em consideração os seis eixos específicos disponibilizados no ICT-UNIFESP: Bacharelado em Ciência da Computação, Bacharelado em Matemática Computacional, Biomedicina, Engenharia Biomédica, Engenharia da Computação e Engenharia de Materiais. Todas as disciplinas recomendadas desses eixos para a formação no Bacharelado em Ciência e Tecnologia foram mapeados em planilhas, como exemplo temos Algoritmo e Estrutura de Dados I:

- Algoritmo e Estrutura de Dados I, Lógica de Programação, 12, 2, -, 13:30, -, 13:30, -

Com essa formatação, podemos entender que a disciplina Algoritmo e Estrutura de Dados I possui como pré-requisito Lógica de Programação, 12 disciplinas subsequentes dependem dela, ela é do termo 2, e suas aulas ocorrem terça e quinta às 13:30. Os horários das disciplinas foram obtidos através da matriz horária disponibilizada no site da UNIFESP. Após esse mapeamento, as planilhas foram carregadas em um notebook do *Google Colab* através da biblioteca *pandas* e transformados em *dataframes* para a utilização. Então, o modelo recebe como entrada o curso específico desejado pelo aluno, o termo em que ele está, o tamanho máximo da grade e se ele tem alguma reprovação ou impedimento de semestres anteriores. Partindo desses parâmetros, o modelo acessa o *dataframe* de acordo com a primeira entrada, em seguida, todas as disciplinas do termo referido são buscadas e adicionadas em uma lista. Após isso, a lista é ordenada de acordo com a quantidade de dependentes de cada disciplina, de forma a priorizar unidades curriculares que apresentam quantidades maiores de outras matérias que dependem dela. Após isso, os horários dessas disciplinas são compactados em tuplas e há a comparação entre os horários de cada disciplina. Caso não haja nenhum conflito de horários, há a verificação se existe algum impedimento de pré-requisito para cursar aquela matéria. Caso não haja, a matéria é adicionada a lista final de escolhas, respeitando o tamanho máximo imposto pelo próprio aluno.

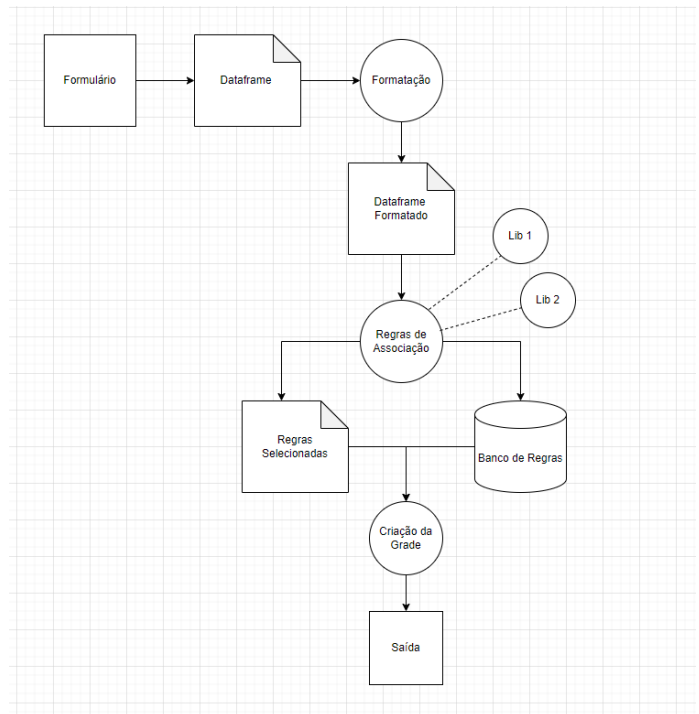


Fig. 1. Pipeline Experimental do modelo

### C. Protocolo Experimental II: Regras Associativas

Agora que já temos o *dataset* completamente formatado, podemos começar o nosso protocolo experimental e a execução do algoritmo que irá criar nossas regras. Para isso, iremos utilizar o algoritmo *apriori*, munindo-se da abordagem presente em duas bibliotecas diferentes: *apyori* e *mlxtend* (*machine learning extensions*). O funcionamento dos algoritmos presentes nessas bibliotecas é semelhante, já que ambos são implementações da ideia fundamental do *apriori*. Entretanto, suas formas de uso e formatações apresentam diferenças significativas:

- *Apyori*: o algoritmo recebe uma lista com as transações, no caso as disciplinas e uma forma de se visualizar os resultados é enlistando as regras. Dessa forma, inserimos uma lista de listas com as transações e recebemos uma lista de listas com os *itemsets* frequentes.
- *MLxtend*: o algoritmo recebe um *dataframe*, então a lista de listas será convertida em uma tabela em que todas as disciplinas e as relações de aprovação serão uma coluna e as transações que tiverem esses elementos serão marcadas com um 1, enquanto as que não tiverem, serão marcadas com um 0. Abaixo segue o trecho e código que realiza essa transformação:

```

encoder = TransactionEncoder()
e= encoder.fit(data).transform(data)

df = pd.DataFrame(e)
df = df.replace(False,0)
df = df.replace(True,1)

```

O resultado escolhido foi o da biblioteca *mlxtend*. Apesar de ambos trabalharem na base do mesmo algoritmo, o formato de tabelas permite uma visualização muito melhor dos dados e já nos informa detalhes relevantes para a análise das regras, tal como suporte, confiança, lift, etc. Para a execução do algoritmo, foi utilizado um suporte mínimo de 0.045, esse valor foi obtido escolhendo um valor inferior próximo ao resultado da divisão de 2 pela quantidade de transações no *dataset*. Dessa forma, evitaremos com que cada transação apresente uma quantidade elevada de relações, se restringindo a itemsets que sejam frequentes a no mínimo 2 transações. Essa abordagem foi escolhida tendo em vista que com esse suporte mínimo de 2 transações, é possível gerar todas as relações não-únicas, que aliado à ferramenta de filtragem presente nas tabelas interativas do *Google Colab*, ferramenta utilizada para execução dos códigos, é possível encontrar qualquer regra regrada de acordo com as especificações fornecidas. Nesse momento, já se faz possível analisar as regras descobertas e selecionar as candidatas para serem utilizadas no gerador de grades. Para nortear as decisões tomadas, iremos tomar como base as medidas de avaliações de suporte, de confiança e de lift das regras descobertas para auxiliar na escolha de regras chave para o modelo a ser desenvolvido.

#### D. Unindo o Gerador de Grades com as Regras Associativas

Agora que temos um gerador funcional que retorna as materias juntamente com o horário de cada disciplina e um conjunto de regras de associação relacionados com essas disciplinas, a tarefa é: checar se o conjunto de materias gerados por essa grade contém matérias em comum com as regras escolhidas previamente. Caso haja uma regra que contemple uma combinação presente dentro da grade, um estatamento relacionado com a regra associativa deve ser retornado como saída juntamente com a grade gerada, de forma a trazer a informação descoberta para o público alvo.

### VIII. RESULTADOS

Partindo do algoritmo apriori disponibilizado na biblioteca *mlxtend* e de um conjunto de dados com 43 transações envolvendo mais de 150 de disciplinas, foi possível gerar um conjunto de 8554 regras descobertas. Nessa seção, serão apresentadas algumas análises a respeito das regras descobertas tal como a escolha das regras chaves a serem utilizadas no modelo de geração de grades. Como evidenciado no gráfico abaixo, grande parte das regras possui o suporte mínimo estabelecido na execução do algoritmo, ou seja, cerca de mais de 7800 regras aparecem somente em duas transações. Esse valor tende a diminuir, de forma a apresentar os seguintes números:

- 3 transações: 427 regras;
- 4 transações: 204 regras;
- 5 transações: 82 regras;
- 6 transações: 14 regras;
- 8 transações: 4 regras;

É uma distribuição que faz sentido, quanto maior o tamanho do conjunto de itens, mais raro a regra se torna. Agora analisaremos a confiança. A grande maioria das regras apresentou uma

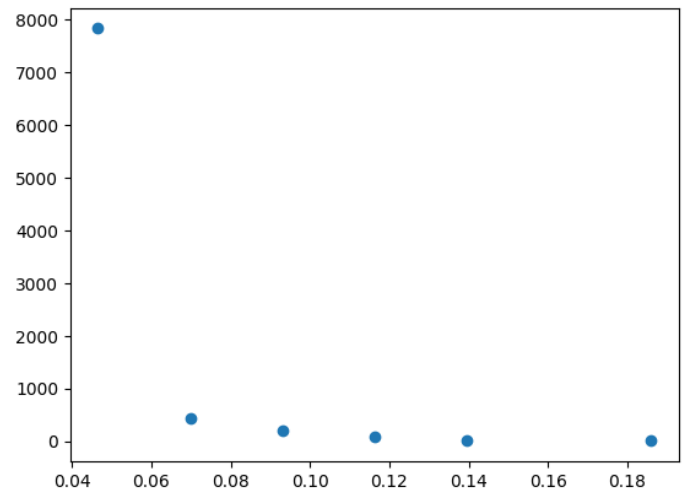


Fig. 2. Distribuição de frequência do suporte das regras

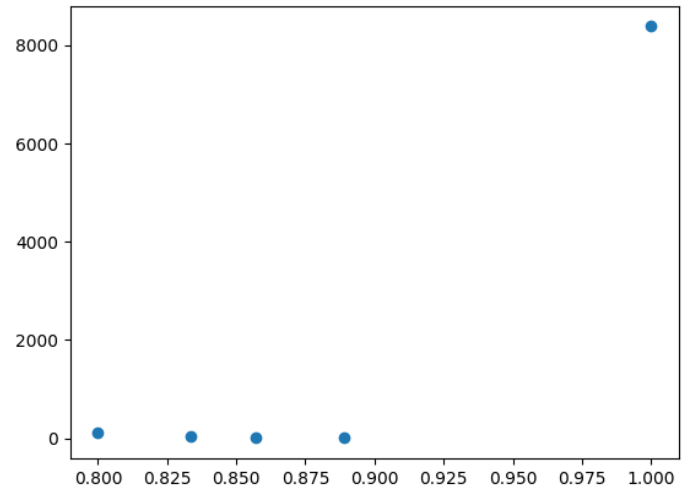


Fig. 3. Distribuição de frequência da confiança das regras

confiança de 1, ou seja, em todas as aparições do antecedente, o consequente também apareceu. Levando em consideração que grande parte das regras é composta somente por duas transações, podemos deduzir que grande parte das regras pode apresentar redundância ou pouca relevância. Salvas exceções, em que há um suporte alto e uma confiança alta, podendo haver a revelação de conhecimento valioso para a montagem do modelo. As distribuições foram:

- Confiança 100%: 8379 regras;
- Confiança 80%: 119 regras;
- Confiança 83.3%: 50 regras;
- Confiança 85.71%: 5 regras;
- Confiança 88.89%: 1 regra;

Todas as regras mostram uma alta taxa de confiança, isso se dá provavelmente pelo baixo número de transações que envolvem as regras, fazendo mais fácil com que padrões sejam formados. Sendo assim, faz-se importante analisar o *lift* das regras. O



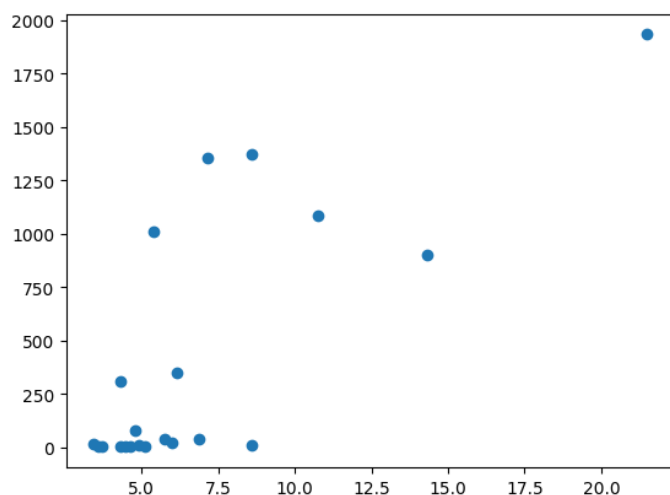


Fig. 4. Distribuição de frequência do lift das regras

lift das regras apresenta uma variação muito maior do que as demais medidas de avaliação, por depender tanto do suporte quanto da confiança, variando entre 3.68 e 21.5, sendo que o lift maior mais comum do que os demais.

#### A. Regras Descobertas e Criação de Estatamentos

Partindo das regras de associação geradas com suas medidas de avaliação (suporte, confiança e *lift*), o intuito nesse momento é combinar regras que possam se mostrar interessante para gerar estatamentos, que serão informados aos alunos juntamente com as disciplinas para o semestre. Algumas das regras e estatamentos selecionados e criados:

- Algoritmos e Estruturas de Dados I - (aprovado) - suporte: 0.186 - confiança: 0.88 - lift: 4.77
- Matemática Discreta - (reprovado) - suporte: 0.1395 - confiança: 0.857 - lift: 3.6857
- Algoritmos e Estruturas de Dados I, Matemática Discreta(reprovado) - Algoritmos e Estruturas de Dados I(aprovado) - suporte: 0.093 - confiança: 1.0

Partindo dessas regras e baseando no conhecimento das disciplinas é possível compor os estatamentos:

- Algoritmos e Estruturas de Dados I: 'Uma das matérias mais escolhidas e uma das mais importantes para os eixos de computação, possui dificuldade moderada, atente-se nela.'
- Matemática Discreta: Essa disciplina é importante em diversos eixos, porém possui uma dificuldade elevada e um alto nível de reprovação, sobretudo se aliado à outras disciplinas trabalhosas, como Algoritmos e Estruturas de Dados. Reserve bastante tempo do seu semestre para os estudos.

Outro conjunto de regras descobertas pode ser levado em consideração:

- Geometria Analítica(aprovado) - Geometria Analítica - suporte: 0.116 - confiança: 1.0 - lift: 6.142

- Algoritmos e Estruturas de Dados I, Fenômenos Mecânicos - Geometria Analítica - suporte: 0.0697 - confiança: 1.0 - lift: 6.142

Com esse conjunto de regras vista e a relação entre as matérias e os termos, podemos confeccionar as seguintes informações:

- Geometria Analítica: Matéria essencial para engenharias e eixos computacionais, se relaciona bem com Fenômenos Mecânicos.
- Fenômenos Mecânicos: É uma das matérias obrigatórias de todos os eixos, se relaciona bem com Geometria Analítica.

Uma disciplina isolada a ser analisada é Ciência, Tecnologia, Sociedade e Ambiente:

- Ciência, Tecnologia, Sociedade e Ambiente - (aprovado) - suporte: 0.1162 - confiança: 0.833 - lift: 7.1666

Com isso aliado a conhecimentos advindos das disciplinas da universidade, constatamos que:

- Ciência, Tecnologia, Sociedade e Ambiente: É uma matéria obrigatória e com alto nível de aprovação, contendo apenas uma aula por semana é uma boa alternativa para a grade.

Com disciplinas de outros eixos além da computação também foi possível analisar as regras e extrair informações:

- Biologia Geral - (aprovado)- suporte: 0.0465 - confiança: 1.00 - lift: 21.5 (valor máximo encontrado)
- Circuitos Elétricos I - Fenômenos Eletromagnéticos - suporte: 0.0465 - confiança: 1.00 - lift: 8.6
- Fenômenos Eletromagnéticos - (aprovado) - suporte: 0.0465 - confiança: 1.00 - lift: 8.6
- Fenômenos Eletromagnéticos - (reprovado) - suporte: 0.0465 - confiança: 1.00 - lift: 8.6

Com essas regras e conhecimentos prévios acerca dos cursos, podemos constatar que:

- Biologia Geral: mostra-se como uma opção consistente para a grade dos outros eixos;
- Circuitos Elétricos I: essa disciplina é fundamental nos eixos específicos de engenharia e é comumente associada com Fenômenos Eletromagnéticos.
- : Fenômenos Eletromagnéticos: Apresenta taxa de aprovação de 50%, deve-se tomar precauções acerca da sua dificuldade, ainda mais se relacionados com matérias como Circuitos Elétricos.

Um outro grupo pode ser analisado:

- Circuitos Digitais(aprovado) - Algoritmos e Estruturas de Dados II - suporte: 0.0465 - confiança: 1.0 - lift: 21.5 (máximo encontrado)
- Circuitos Digitais(aprovado) - Circuitos Digitais - suporte: 0.0465 - confiança: 1.0 - lift: 14.333
- Circuitos Digitais - Cálculo em Várias Variáveis - suporte: 0.069 - confiança: 1.00 - lift: 14.333

Com base nessa regra podemos relacionar que:

- Circuitos Digitais: Além de ser uma disciplina de vários eixos, tem um taxa moderada de aprovação e se relaciona com diversas outras matérias, como Algoritmos e Estruturas de Dados II e Cálculo em Várias Variáveis.



- Algoritmos e Estruturas de Dados II: Matéria importante para o eixo computacional, com dificuldade moderada e relacionando-se adequadamente com Circuitos Digitais e outras disciplinas do termo.
- Cálculo em Várias Variáveis: Disciplina com dificuldade moderada-alta, é necessário uma dedicação de tempo maior nessa matéria.

Essas e outras regras semelhantes foram adicionadas ao modelo por meio de uma lista contendo o nome da disciplina e o estatamento relacionado a ela.

#### B. Grades Geradas com as Informações Relacionadas

A entrada de informações para o modelo de geração de grades é da seguinte forma: curso específico / termo / tamanho máximo da grade / impedimentos. Todas as informações são separadas por barras, tendo a seguir um exemplo: bcc/2/6/nenhuma disciplina

A saída obtida será uma lista com as disciplinas recomendadas, já escalonadas por ordem de prioridade e checagem de conflito de horários, juntamente com os estatamentos, da seguinte forma: ['Algoritmos e Estruturas de Dados I', 'Geometria Analítica', 'Matemática Discreta', 'Fenômenos Mecânicos', 'Ciência Tecnologia Sociedade e Ambiente', 'Séries e Equações Diferenciais Ordinárias']

- Fenômenos Mecânicos : É uma das matérias obrigatórias de todos os eixos, se relaciona bem com Geometria Analítica.
- Geometria Analítica : Matéria essencial para engenharias e eixos computacionais, se relaciona bem com Fenômenos Mecânicos.
- Ciência Tecnologia Sociedade e Ambiente : É uma matéria obrigatória e com alto nível de aprovação, contendo apenas uma aula por semana é uma boa alternativa para a grade.
- Algoritmos e Estruturas de Dados I : Uma das matérias mais escolhidas e uma das mais importantes para os eixos de computação, possui dificuldade moderada, atente-se nela!!
- Matemática Discreta : Essa disciplina é importante em diversos eixos, porém possui uma dificuldade elevada e um alto nível de reprovação, sobretudo se aliado à outras disciplinas trabalhosas, como Algoritmos e Estruturas de Dados. Reserve bastante tempo do seu semestre para os estudos.

Caso eu reduza o tamanho máximo da minha grade, os elementos de maior índice, considerado menos prioritários, irão deixar de compor a saída, juntamente com os estatamentos relacionados a essas matérias. Caso um dos impedimentos seja uma matéria pré-requisito de alguma disciplina que componha as candidatas para as disciplinas recomendadas, ela não será adicionada à lista.

#### C. Discussão - Metodologia

Creio que um dos pontos a serem levantados na discussão dos resultados é o *dataset* utilizado para a execução do algoritmo de regras de associação. Trata-se de uma base

de dados obtida através de alunos voluntariados do ICT-UNIFESP, portanto não é um conjunto de informações muito extenso, o que seria o ideal para a execução do algoritmo *a priori*. Dessa forma, as regras descobertas não abrangem em sua totalidade todas as disciplinas de todos os eixos específicos e suas combinações. Outro ponto a ressaltar é que grande parte das regras geradas são redundantes ou não revelam informações interessantes para o modelo trabalhado. O modelo de geração não apresenta conflito de horários entre as disciplinas, porém poderia haver a possibilidade da adição de matérias de outros termos, mas para encontrar a melhor combinação, seria necessário um *dataset* de maiores dimensões para a descoberta de mais regras associativas.

#### D. Validação [16]

Fora buscado na literatura projetos que utilizem de regras de associação e fora encontrado um trabalho feito por um mestrando da UTFPR em que há a mineração de dados utilizando regras de associação em dados de saúde, em especial de pronto atendimento, e as regras descobertas serão validadas por um especialista da área em que o trabalho é aplicado. Analisando os projetos em que as regras de associação são utilizadas, pode-se perceber um padrão semelhante para a constatação da utilidade e inovação das regras descobertas. Sendo assim, trazendo essa abordagem para o escopo desse projeto, faz-se interessante realizar a validação por meio do próprio corpo discente da universidade, por meio de avaliação. Dentro desse viés, pode-se configurar um caso de avaliação das regras descobertas:

- Gerar declarações acerca das regras descobertas e levar aos alunos que já cursaram as matérias relacionadas para que a avaliação seja realizada. Exemplo: Regra descoberta - > 80% dos alunos que escolheram computação gráfica e matemática discreta no mesmo semestre, reprovaram. Declaração referente à regra: computação gráfica e matemática discreta são matérias difíceis, ainda mais se feitas juntas. Partindo desse estatamento, os alunos que de ciência da computação deverão analisar o nível de relevância e proximidade com a realidade da afirmação.

Para as grades horárias geradas através destas regras, pode-se criar duas medidas de avaliação e validação do modelo:

- Gerar grades horárias variando os parâmetros possíveis e disponibilizá-las para os alunos dos diversos cursos específicos presentes na universidade, para que eles possam avaliar se eles adotariam aquela grade para o semestre vigente ou não;

#### E. Discussão - Validação

Um formulário foi distribuído para alunos veteranos no ICT-UNIFESP contendo alguns conjuntos de matérias e estatamentos sobre elas, geradas pelo modelo. Dentre as opções disponíveis sobre o conjunto de matérias:

- a) Usaria essas disciplinas - 86%;
- b) Faria uma alteração; - 10%
- c) Faria mais de uma alteração - 4%;

Dentre os estatamentos, poderia ser escolhido:

- a) Concordo plenamente - 78%;
- b) Concordo parcialmente - 18%
- c) Discordo parcialmente - 4%
- d) Discordo plenamente - 0%

Esses resultados revelam que o modelo traz algum benefício para o corpo discente da universidade e pode ser utilizado, sobretudo pelos alunos calouros.

## IX. ENTREGA E PLANOS FUTUROS

Levando em consideração todas as informações e pesquisas citadas, espera-se que até o término do projeto, possa ser entregue um modelo que consiga gerar as grades para um determinado semestre, partindo da matriz horária determinada pela instituição, das restrições (parâmetros) impostas por cada aluno ao usar o modelo e das informações implementadas no modelo através das regras de associação descobertas durante a fase de mineração de dados. No momento, é esse a meta a ser entregue para a disciplina de Inteligência Artificial, entretanto é importante ressaltar que o trabalho não irá se acabar com o fim do semestre, ele será desenvolvido posteriormente para que haja uma interface de uso para todos os alunos, seja por meio de site ou aplicativo.

## REFERENCES

- [1] R. L. Lobo, P. R. Montejunas, O. Hipólito, M. B. de Carvalho. Evasão no Ensino Superior Brasileiro. Cadernos de Pesquisa, São Paulo, Brasil, v. 37, n.132, 2007.
- [2] INEP, Censo da Educação Superior, Brasil, 2016.[Online]. Disponível em: <https://www.gov.br/inep/pt-br>
- [3] R. P. Teixeira, M. J. Mentges, A. J. Kampff. Evasão no Ensino Superior: Um Estudo Sistemático. Brasil, 2019
- [4] Google Cloud. What Is Artificial Intelligence? [Online]. Disponível em: <https://cloud.google.com/learn/what-is-artificial-intelligence?hl=pt-br>
- [5] Sistema Cronos. [Online]. Disponível em: <https://www.cronostimetable.com/>
- [6] C. J. B. Poulsen. Desenvolvimento de um modelo para o School Timetabling Problem baseado na Meta-Heurística Simulated Annealing. Porto Alegre, Brasil, 2012.
- [7] G. B. Bucco. Construção de um modelo de programação linear para o University Timetabling Problem. Porto Alegre, Brasil, 2014.
- [8] C. D. L. Hamawaki. Geração Automática de Grade Horária Usando Algoritmos Genéticos: O Caso da Faculdade de Engenharia Elétrica da UFU. Uberlândia, Brasil, 2005.
- [9] Ç. H. Aladağ, G. Hocaoglu. A Tabu Search Algorithm To Solve A Course Timetabling Problem. Turquia, 2007
- [10] B. Coşar, B. Say, T. Dökeroğlu. A New Greedy Algorithm for the Curriculum-based Course Timetabling Problem. Turquia, 2022.
- [11] K. A. F. A. Samah et al. A Greedy-based Algorithm in Optimizing Student's Recommended Timetable Generator with Semester Planner. Malásia, 2022.
- [12] K. Wang, W. Shang, M. Liu, W. Lin. A Greedy and Genetic Fusion Algorithm for Solving Course Timetabling Problem. Beijing, China, 2018.
- [13] Sutch, T. Using association rules to understand subject choice at AS/A level. Cambridge Assessment Research Report. Cambridge, UK: Cambridge Assessment, 2015
- [14] V. Kumar, A. Chadha. Mining Association Rules in Student's Assessment Data. Índia, 2012.
- [15] R. Shatnawi, Q. Althebyan, B. Ghalib, M. Al-Maolegi. Building A Smart Academic Advising System Using Association Rule Mining. Jordania, 2014.
- [16] R. J. Feuser. Mineração de Dados com Regras de Associação Aplicada em Dados de Unidade de Saúde de Pronto Atendimento. Pato Branco, Paraná, Brasil. 2017
- [17] W. Romão et al. Extração de Regras de Associação em C&T: O algoritmo Apriori. Florianópolis, Santa Catarina, Brasil, 1999.