UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS
UNIDADE ACADÊMICA DE EDUCAÇÃO ONLINE
ESPECIALIZAÇÃO EM BIG DATA, DATA SCIENCE E DATA ANALYTICS

Guilherme S. Vieira[1]

PADRÕES SEQUENCIAIS PARA *LEARNING ANALYTICS*: UM ESTUDO SOBRE COMO REDES NEURAIS PODEM PREVER O DESEMPENHO DE ESTUDANTES

São Leopoldo

2021

UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS
UNIDADE ACADÊMICA DE EDUCAÇÃO ONLINE
ESPECIALIZAÇÃO EM BIG DATA, DATA SCIENCE E DATA ANALYTICS

Guilherme S. Vieira[1]

PADRÕES SEQUENCIAIS PARA *LEARNING ANALYTICS*: UM ESTUDO SOBRE COMO REDES NEURAIS PODEM PREVER O DESEMPENHO DE ESTUDANTES

Artigo apresentado como requisito parcial para obtenção do título de Especialista em **Big Data, Data Science e Data Analytics**, pelo Curso de Especialização em **Big Data, Data Science e Data Analytics** da Universidade do Vale do Rio dos Sinos – UNISINOS

Orientadora: Prof[a]. PhD Patrícia A. Jaques Maillard

São Leopoldo

2021

Padrões Sequenciais para *Learning Analytics*: Um estudo sobre como redes neurais podem prever o desempenho de estudantes

Guilherme S. Vieira[1]

[1]Universidade do Vale do Rio dos Sinos (UNISINOS)
Av. Unisinos, 950, Bairro Cristo Rei, São Leopoldo, RS – Brasil

guischa@edu.unisinos.br

Abstract. *A study was conducted for the development of predictive models based on sequential patterns identified through student activity logs in a virtual learning environment to rate student performance before the completion of their course. The objective of this study was to compare the performance of predictive models based on decision tree algorithms and neural networks, as well as to identify whether the use of sequential patterns contributes to the performance of these models rather than to evaluate individual actions taken by students. It was concluded that both sequential patterns and neural networks can improve model performance, but some decision tree-based models were also chosen among the best.*

Resumo. *Foi realizado um estudo para o desenvolvimento de modelos preditivos com base em padrões sequenciais identificados através de logs de atividades de alunos em um ambiente virtual de aprendizado para classificar o desempenho deste aluno antes da conclusão de seu curso. O objetivo deste estudo foi comparar o desempenho de modelos preditivos baseados em algoritmos de árvore de decisão e em redes neurais, bem como identificar se o uso de padrões sequenciais contribui para o desempenho destes modelos ao invés de avaliar as ações individuais realizadas pelos alunos. Concluiu-se que padrões sequenciais e redes neurais podem melhorar o desempenho de um modelo preditivo, mas modelos baseados em árvore de decisão também se destacam entre os melhores.*

1. Introdução

Em 2019, segundo os dados do Censo da Educação Superior, realizado pelo INEP, o índice de evasão dos cursos superiores chegou a 23,90%, sendo este o maior percentual dentre os dados históricos. Ao mesmo tempo em que este índice cresce com o passar dos anos, as instituições de ensino buscam identificar as causas destas evasões e, principalmente, evitá-las, levando estes alunos potencialmente evasivos à conclusão de seus cursos.

Uma técnica para identificar o comportamento dos alunos e categorizar perfis que podem levar à evasão do ensino é a mineração de dados, que ganhou popularidade nos anos recentes em função do aumento de dados gerados por indivíduos e organizações, bem como o aumento exponencial do poder computacional capaz de processar essas grandes massas de dados e identificar correlações que somente com tamanho poder computacional seriam possíveis de se obter. Quando técnicas de mineração de dados é aplicada no contexto do ensino superior para entender os fatores que influenciam o sucesso ou insucesso de estudantes, tem se a "Learning Analystics".

Da mesma forma que o poder computacional aumentou, novos modelos de extração de dados e conhecimento foram surgindo para atender necessidades cada vez mais complexas, como reconhecimento de fala ou classificação de imagens. Ainda no contexto educacional, o histórico de acesso de alunos em um ambiente virtual de aprendizagem pode servir como fonte valiosa para mapear o perfil de estudantes que conseguem ou não finalizar a disciplina cursada. E desta forma, a instituição de ensino pode agir de forma pró-ativa para auxiliar alunos com dificuldades.

Nesse contexto, este estudo visa criar modelos preditivos que, sob diferentes modelos arquiteturais e transformações de dados, sejam capazes de identificar alunos com potencial risco de evasão ou não-conclusão de uma disciplina para que estes sejam devidamente assistidos para atingir seus objetivos.

Embora outros trabalhos tenham buscado desenvolver modelos de predição de sucesso acadêmico ou evasão escolar (ver Seção 3), boa parte destes trabalhos se restringe a um modelo arquitetural ou a uma única disciplina para atingir seus objetivos. No trabalho proposto, são utilizados dados de múltiplas disciplinas, que passam por algoritmos de busca de padrões sequenciais, visando complementar os dados já existentes destes alunos. Além disso, são criados diferentes modelos arquiteturais, tanto de *Machine Learning* quanto de redes neurais, visando encontrar algum modelo que seja coerente para obter um bom desempenho para todas as disciplinas, ou buscando um modelo para cada disciplina, levando em consideração a variação dos dados de cada contexto.

2. Referencial Teórico

Esta seção descreve os temas centrais abordados neste trabalho, sendo eles a evasão no ensino superior, mineração de dados, *Learning Analytics*, *Machine Learning* e *Deep Learning*. Da mesma forma que é importante que um cientista de dados demonstre bons conhecimentos técnicos e de negócio para conseguir produzir bons resultados, conceituar cada um destes temas se faz necessário para que haja uma boa compreensão deste trabalho.

2.1. Evasão no Ensino Superior

A evasão do ensino superior possui impactos tanto para o indivíduo quanto para a sociedade. De acordo com os dados do Instituto Brasileiro de Geografia e Estatística (IBGE), a taxa de desocupação no Brasil é de 16,35% para pessoas que não concluíram o ensino superior, enquanto que, para pessoas com ensino superior concluído, esta taxa cai para 6,44%.

Não existe uma fórmula definitiva para mensurar a evasão no ensino superior pelo fato de que a evasão pode acontecer por diversos motivos. Por conta disto, são sugeridas fórmulas com o objetivo de abranger evasões de forma generalista. Uma destas fórmulas é sugerida pelo estudo realizado por [Silva Filho et al. 2007], onde o percentual de evasão é dado por:

$$E = 1 - \frac{M(n) - I(n)}{M(n-1) - C(n-1)}$$

onde **E** corresponde à evasão, **n** equivale ao ano a ter o índice obtido, **n-1** ao ano

anterior, **M** aos alunos matriculados, **I** aos alunos ingressantes e **C** aos alunos concluintes (egressos).

Utilizando esta equação sobre os dados do Censo da Educação Superior, realizado pelo Instituto Nacional de Estudos e Pesquisas Educacionais (INEP), obtém-se o índice de evasão do ensino superior, que, como mostra o gráfico da Figura 1, evolui consideravelmente desde 2011, atingindo 23,90% de estudantes em 2019, o maior percentual já registrado, representando um aumento de 1,32% em relação ao ano anterior.

Figura 1. Variação da Evasão no Ensino Superior no Brasil
Fonte: Elaborado pelo autor

Os fatores que agravam estes números podem variar de um curso para outro. Eles envolvem desde dificuldades para entrega de atividades propostas dentro dos prazos, ainda que identificadas condições de estudo [Colpani 2018], até motivos familiares e de saúde, em cursos com perfil de maior idade do aluno [de Oliveira et al. 2017]. Estes fatores podem fazer com que a taxa de evasão em determinado curso seja ainda maior do que as médias já descritas anteriormente, podendo chegar a mais da metade dos alunos.

2.2. Mineração de Dados

A mineração de dados consiste em processos de extração, transformação e criação de dados para identificação de padrões que levam a novos conhecimentos, assim como novas hipóteses. O aumento exponencial dos dados gerados pela humanidade contribui para que este tema tenha cada vez mais relevância.

Para [Makhabel 2015] e [SILVA et al. 2016], a mineração de dados consiste na busca de modelos padronizados de uma determinada base de dados. Esses dados podem se mostrar inesperados ao serem analisados individualmente, mas que podem trazer conhecimento de grande valor ao serem submetidos a algoritmos dedicados a tarefas de clusterização e reconhecimento de padrões, por exemplo.

A busca de padronização destes dados, bem como a análise individual deles pode de alguma forma conflitar com a definição de *Business Intelligence* (BI), o que pode causar um desentendimento dos termos, dando assim a impressão de que se tratam da mesma coisa e de que apenas houve uma mudança na terminologia para que o assunto pudesse ganhar popularidade. Apesar de ambos os termos trabalharem com dados, a mineração de dados extrai conhecimento a partir de uma base de dados, enquanto que BI tem como objetivo de prover visualizações de dados de forma que eles possam fazer algum sentido para quem os visualiza. Em BI, a extração de conhecimento depende de

quem está visualizando os dados, enquanto a mineração de dados trabalha para extrair conhecimento a partir dos mesmos.

No entanto, Mineração de dados possui um significado semelhante ao termo *Data Science*, que se tornou popular nos últimos anos. Segundo [Squire 2016], os dois termos são equivalentes considerando sua inserção em metodologias de análise de dados como o *Knowledge Discovery in Databases (KDD)*, ou descoberta de conhecimento em bases de dados. A mineração de dados essencialmente consiste na extração de conhecimento a partir de uma base de dados. Data Science surgiu para que esta extração seguisse uma metodologia científica.

Para [Provost e Fawcett 2016, p.14], "a partir de uma grande massa de dados, a tecnologia da informação pode ser usada para encontrar atributos descritivos informativos de entidades de interesse", o que reflete na importância da mineração de dados. Dentre uma base de clientes que visitam uma loja, qual o perfil de compra deles, e de que forma esse perfil pode ser trabalhado para que eles comprem mais produtos ou para que a loja possa trazer mais produtos que esse perfil de cliente espera? A mineração de dados busca extrair conhecimentos deste tipo.

A mineração de dados e sua extração de conhecimento possuem grande relevância, uma vez que o conhecimento obtido através de uma base de dados auxilia tomadas de decisão que, por exemplo, conseguem melhorar o faturamento de uma loja, tornar um serviço mais atraente para um usuário, e fazer com que instituições de ensino possam agir para reter estudantes que possam, por algum motivo, não concluir seus estudos como esperado.

Conclui-se então, que a mineração de dados consiste na extração de conhecimento a partir de uma base de dados, visando auxiliar na tomada de decisões que podem trazer ganhos para empresas, comércio e serviços. Tanto para quem os administra, quanto para quem os utiliza, pois o conhecimento obtido através destes dados proporciona produtos e serviços mais alinhados às necessidades de seus usuários, que tornam os produtos e serviços utilizados por eles mais rentáveis.

O processo de mineração de dados pode extrair conhecimento de diferentes tipos de dados, assim como gerar diferentes tipos de conhecimento que podem levar a diferentes *insights* para serem usados no processo de tomada de decisão, como os seguintes:

- Mineração de Regras de Associação: A mineração de regras de associação consiste na busca de itens que são encontrados frequentemente em dados como carrinhos de compra, visando obter *insights* que tragam maior lucratividade e aproximação com clientes. Um exemplo conhecido é ilustrado em [Provost e Fawcett 2016], onde uma rede de supermercados encontrou uma grande correlação entre fraldas de bebê e garrafas de cerveja, vistas com grande frequência em grandes carrinhos de compra. Como geralmente estes dois itens são encontrados em locais muito distantes um do outro em mercados, esta rede providenciou a inclusão de freezers próximos ao departamento de fraldas, e esta ação se mostrou efetiva em aumentar o faturamento desta rede de supermercados. Algumas diretivas podem ser utilizadas para garantir a busca das regras mais efetivas, como as seguintes:
  - Suporte: O suporte é um valor usado como limiar de aceitação para

que uma regra de associação ou padrão sequencial seja considerado na mineração de dados. Por exemplo, no processo de busca de padrões sequenciais, foi definido que o suporte mínimo a ser obtido por padrão sequencial é de 0.1% para ser identificado como padrão sequencial da base de dados avaliada. A sequência [1 2] de itens em um carrinho de compras deve constar em pelo menos 0,1% dos carrinhos de compras lidos no processo de mineração para ser contabilizado. Quanto menor o suporte mínimo, maior a quantidade de diferentes padrões sequenciais uma base de dados irá reportar [Makhabel 2015].

– *Gap-constraints*: Este conceito introduzido por [Srikant e Agrawal 1996] fala sobre o espaço de tempo máximo que dois itens devem ter entre si para serem considerados dentro de uma sequência. Por exemplo, a Tabela 1 mostra um log de atividades simplificado com o código de uma ação executada e a hora de realização da mesma. Supondo que para a extração de uma sequência de dados, o tempo máximo entre uma ação e outra deve ser de 60 segundos. Neste exemplo, seriam criadas as sequências [1, 2] e [1, 2, 5].

Tabela 1. Exemplo de log de atividades para demonstração do conceito de max-gap

Ação	Hora
1	23:03:17.000
2	23:03:31.000
1	23:04:52.000
2	23:05:00.000
5	23:05:11.000

2.3. *Learning Analytics*

Learning Analytics consiste no uso de dados obtidos através de sistemas virtuais de aprendizagem para identificar fatores que influem de forma positiva ou negativa no processo de aprendizado. Sua utilização pode identificar alunos que podem estar passando por dificuldades, e assim, agir para que eles consigam retomar o caminho ao sucesso, tornando o seu aprendizado mais eficiente.

Segundo [Mattox 2016], o grande objetivo de se usar *learning analytics* é conseguir medir a efetividade de um curso ou treinamento, podendo classificar essa efetividade através de dados como o levantamento de conhecimentos adquiridos durante um curso ou treinamento.

Neste caso, uma forma simples de medir a efetividade do aprendizado é levantar os dados de notas de provas e trabalhos realizados e conferir o desempenho de cada aluno. Contudo, isto pode não ser o suficiente para conseguir agir de forma pró-ativa para evitar que um aluno tenha um desempenho ruim. Assim como o uso de *learning analytics* pode ser questionado dependendo do tipo de dados que são utilizados para medir a efetividade de um curso. Se apenas dados relacionados a notas forem utilizados para avaliar esta efetividade, o uso de sistemas virtuais de aprendizado pode ser questionado. Afinal, notas de provas e trabalhos são as métricas mais utilizadas no processo de aprendizado.

Tendo estes fatos e números expostos, surge uma discussão sobre como as metodologias de ensino podem mudar de forma a manter os estudantes interessados e como os ambientes de aprendizagem podem se adaptar às necessidades do aluno. O uso de softwares para analisar o progresso individual é visto como uma possibilidade, e neste campo, entra o processo de mineração de dados educacionais (EDM), um campo voltado ao desenvolvimento de modelos para compreender os processos de aprendizagem e torná-los mais eficazes [Baker et al. 2011].

2.4. *Machine Learning* e Redes Neurais

Com o passar dos anos, a quantidade de dados gerados pela humanidade vem apresentando um crescimento exponencial, fazendo com que formas mais complexas de armazenar e processar estes dados sejam disponibilizadas. Com isto, o campo de *Machine Learning* vem ganhando espaço por prover técnicas que permitem que seja feito este processamento das mais variadas formas de dados de forma a obter *insights* e que estes dados mostrem padrões de comportamento [Tripathi 2017].

A capacidade de aprender com os dados e utilizar este aprendizado para realizar previsões também se aplica à definição de *Machine Learning*, criando relações matemáticas entre diferentes variáveis que são criadas através dos dados destas variáveis, e uso das mesmas para prever possíveis saídas para os dados novos [Grus 2016].

Aplicações comuns de *Machine Learning* costumam gerar funções a serem aplicadas sobre os dados de entrada originando uma determinada saída. Este tipo de aplicação possui boa aplicabilidade para casos como regressões lineares e logísticas, que atendem boa parte dos problemas mais comuns em função da estrutura de dados utilizada para atender estas situações ser muito similar.

Porém, existem situações mais complexas que estes modelos não são capazes de atender, como classificação de imagens, reconhecimento de fala e detecção de objetos. Para este tipo de tarefa, não apenas é necessário trabalhar bastante sobre os dados que dão origem a este problema, como é necessário utilizar vários algoritmos em conjunto para serem capazes de agir em cada ponto de um problema mais específico para obter uma saída capaz de agregar valor para um negócio. Neste âmbito, entram as redes neurais, capazes de criar algoritmos poderosos e escaláveis o suficiente para lidar com problemas mais complexos [Géron 2019]. Redes compostas por estes modelos, que vão além de uma entrada e uma saída, são utilizadas em aplicações de aprendizagem profunda, ou *Deep Learning* [Zocca et al. 2017].

Um destes tipos de modelo preditivo é a rede neural artificial, muito utilizada para aprendizagem profunda, que se baseia em como o cérebro humano funciona. Este tipo de modelo preditivo ajuda a resolver problemas como reconhecimento de caligrafia, classificação de objetos baseados em imagens e predições baseadas em séries temporais [Grus 2016].

3. Trabalhos Relacionados

Esta seção irá detalhar pesquisas com objetivos semelhantes. Existem diversos estudos que trabalham com dados educacionais a fim de auxiliar estudantes, com objetivos entre prever o seu desempenho e indicar o papel mais adequado para determinado aluno em um projeto acadêmico. Boa parte destes estudos faz uso de algoritmos baseados em árvore

de decisão e dados tabulares, que também são utilizados neste trabalho. Porém, também são listados estudos que fazem uso de redes neurais e dados sequenciais, que também são características chave deste trabalho.

3.1. Mineração de dados para predição do desempenho de estudantes (2018)

A predição do desempenho de alunos através de um modelo criado através da mineração de dados educacionais também foi trazida por [Nicoletti et al. 2018]. Neste caso, além dos dados específicos da disciplina em estudo, que neste caso diz respeito a Programação e Algoritmos I do curso de Ciência da Computação, são levados em consideração dados pessoais dos alunos, como data de nascimento, sexo e estado civil.

Os dados disponibilizados correspondiam a semestres entre 2009 e 2015, sendo eles compilados e separados por ano, tendo suas variáveis mais relevantes selecionadas utilizando *Feature Selection*, e levadas para compilação de um modelo de árvore de decisão. Para avaliar o desempenho deste modelo, foram realizados três experimentos. Os dois primeiros têm como semelhança o processamento de cada ano de forma separada, tendo como diferença a escolha das variáveis para criação do modelo. Ao utilizar a acurácia como métrica de validação do modelo, viu-se que existem resultados muito diferentes de ano a ano, podendo variar em um dos experimentos, de 49% a 92%.

O terceiro experimento utilizou os dados dos anos de 2009 a 2014 como treino para o modelo preditivo, e os dados de 2015 como teste, visando obter um desempenho mais consistente. O resultado foi de uma acurácia de 88%, sugerindo que uma maior quantidade de dados leva a um modelo mais assertivo. Porém, esta avaliação foi feita somente para o último ano da base de dados, o que não basta para concluir que a adição de dados implica em um melhor desempenho em um modelo preditivo, levando em consideração que em um dos experimentos anteriores, conferiu-se uma acurácia superior à apresentada com todos os semestres anteriores a 2015 utilizados como treino.

3.2. *Framework* para classificação de sequências em dados educacionais (2016)

O modelo proposto por [Jaber et al. 2016] possui um objetivo distinto dos outros já mencionados. Enquanto outros modelos têm como objetivo predizer o desempenho de alunos, ou então classificar se eles serão aprovados ou até mesmo desistentes de seus cursos, este visa identificar em qual papel dentro de um projeto o aluno se enquadra. Esta avaliação se dá por dados de *logs* de comunicação, que geram sequências de iteração temporais, fazendo assim o uso destas sequências como critério para identificar se o aluno se identifica com o papel ao qual foi designado, ou se o melhor a ser feito é trocar de papel visando melhor desempenho no projeto aplicado.

As sequências geradas por estas iterações são coletadas e transformadas de forma a gerar padrões sequenciais com o auxílio de algoritmos dedicados para a identificação de padrões sequenciais dentro de uma base de dados, os quais são dados como classes e então, levados para a criação de modelos preditivos. Neste caso, foram gerados modelos baseados em *K-Nearest Neighbors*, árvore de decisão como a *Random Forest* e máquinas de vetores de suporte foram utilizados. Como critério de avaliação do desempenho, métricas como precisão, *recall*, *F-measure* e a curva de ROC foram utilizadas.

Os resultados deste modelo sugerem que a utilização de algoritmos como *Random Forest* e máquinas de vetores de suporte são muito mais adequadas para obter bons resul-

tados do que *K-Nearest Neighbors*, tendo todas as métricas com resultados entre 88% e 91%.

3.3. Mineração de dados educacionais utilizando classificação e mineração de associações (2019)

O modelo desenvolvido por [Rojanavasu 2019] tem como objetivo identificar se alunos irão seguir uma carreira na área de Tecnologia da Informação (TI) ao finalizarem seus estudos, buscando assim associações dentre as variáveis presentes nas bases de dados utilizadas para a avaliação.

Para a avaliação deste modelo, uma das bases de dados utilizada no seu desenvolvimento contém informações referentes ao processo de admissão de alunos, como data de admissão e região da instituição a ser aplicada, visando encontrar *insights* para melhorias no processo geral de admissão. Outra fonte de dados diz respeito à grade curricular selecionada pelo aluno, com o objetivo de classificar se o seu trabalho será de TI ou de outra área, que neste caso, é classificada como Não-TI.

Com estes dados, foi criado um modelo baseado em árvore de decisão, sendo considerado o mais indicado pelo fato de se tratarem de dados tabulares em um *dataset* pequeno. Sem a necessidade de busca de padrões sequenciais entre as variáveis, este modelo foi capaz de entregar uma acurácia de aproximadamente 73% na classificação da grade curricular dos alunos.

3.4. Predição do desempenho de alunos utilizando padrões sequenciais (2016)

O modelo desenvolvido por este trabalho faz uso de árvore de decisão para classificar o desempenho de estudantes para classificá-los entre alunos que concluirão seus respectivos cursos com sucesso ou não. A base de dados conta com três disciplinas diferentes e cada disciplina teve seu modelo correspondente criado, e os resultados muito semelhantes ao avaliar ações isoladas, com acurácias entre 66% e 67%. Ao fazer uso de padrões sequenciais nos modelos por curso, as acurácias encontradas variaram entre 71% e 100%, mostrando que o uso de padrões sequenciais melhora significativamente o desempenho de modelos desenvolvidos com este tipo de dados.

A proposta deste trabalho difere na forma com que os dados são separados e nos tipos de modelos preditivos a serem criados. Os dados de padrões sequenciais puderam mostrar que sua adição contribui para o desempenho de modelos preditivos, porém, os dados de todos os semestres estão juntos, para serem aleatoriamente separados entre treino e teste.

Sob uma perspectiva prática, os dados de um mesmo semestre podem ser utilizados como treino e teste, o que pode levar a um ganho significativo de desempenho nos modelos preditivos fazendo uso de dados que, em certo momento, não deveriam constar na base de treino, levando a resultados como modelos capazes de prever com precisão máxima todos os alunos de determinado curso que o concluem ou não com sucesso.

Além disto, serão criados modelos de *Deep Learning* para verificar se o uso de redes neurais é capaz de entregar um desempenho superior ao uso de modelos de *Machine Learning* baseados em árvore de decisão.

3.5. Utilização de uma rede *Long-Short Term Memory* para predição com base em dados sequenciais (2016)

[Tang et al. 2016] propõe a criação de uma rede neural baseada no modelo *Long-Short Term Memory* (LSTM), o qual é indicado para classificação de textos e treino de sequências de grande espaço temporal, como *logs* de utilização de sistemas. Estes dois tipos de dados são utilizados para criação de um modelo capaz de produzir novas sequências de interações, visando identificar o futuro comportamento de estudantes com base na utilização de ambientes virtuais de aprendizagem.

Enquanto este trabalho faz uso de dados sequenciais, o modelo proposto por este artigo faz o uso de padrões sequenciais identificados nas sequências de interações coletadas pelos estudantes. É importante diferenciar uma sequência de interações explícitamente encontrada em uma base de dados de um padrão sequencial encontrado através de um algoritmo de busca de padrões sequenciais.

Da mesma forma que estes padrões sequenciais são encontrados, não se pode determinar de forma precisa quando em um espaço de tempo determinado padrão sequencial foi utilizado por um estudante, mas sim quantas vezes aquele padrão foi realizado pelo aluno. Sendo assim, os dados a cerca dos padrões sequenciais são armazenados de forma tabular ao invés de sequencial, fazendo com que o uso de uma rede LSTM não seja a mais indicada para este tipo de situação.

Considerando a forma com que os dados dos padrões sequenciais são armazenados, o mais indicado para a geração de modelos preditivos é o uso de técnicas de *Machine Learning* baseadas em árvores de decisão, como já foram utilizadas em trabalhos relacionados já descritos. Porém, este artigo também fará uso de um modelo de rede neural apropriado para dados tabulares, visando identificar se seu uso pode trazer diferenças significativas de acurácia se comparado com modelos baseados em árvores de decisão.

3.6. Comparação dos trabalhos

A grande maioria dos trabalhos voltados para a mineração de dados educacionais faz uso de dados que de certa forma possuem uma estrutura fixa, os quais são utilizados como base para criação de modelos de *Machine Learning* voltados para classificação, como árvores de decisão, *Naive Bayes* e máquinas de vetores de suporte.

Sendo este trabalho uma continuação direta do modelo proposto por [Bandeira 2016], sua base de dados e etapa de preparação de dados se assemelha com o produzido por este trabalho, tendo como diferença o objetivo direto do mesmo, que é identificar que um modelo preditivo que faz uso de padrões sequenciais consegue um desempenho melhor do que avaliar ações isoladas. Seu grande diferencial em relação aos outros trabalhos relacionados é o uso de padrões sequenciais como forma de avaliar o desempenho dos alunos, o que provou trazer desempenho superior à avaliação de ações isoladas no ambiente de aprendizado utilizado para a coleta destes dados.

Complementando o uso de padrões sequenciais, [Jaber et al. 2016] traz uma rica avaliação no uso de padrões sequenciais para criação de vários modelos baseados em algoritmos de *Machine Learning*, somados a uma boa variedade de métricas capazes de prover uma avaliação consistente do desempenho dos modelos. De forma semelhante, este trabalho fará uso de dados geradores de padrões sequenciais para a criação de modelos

preditivos, assim como fará uso de modelos de *Machine Learning*, tais como *Random Forest*, para avaliação do desempenho destes modelos. O principal diferencial se encontra no teste dos modelos, que se dará de forma incremental ao inserir novos semestres como treinamento, visando obter uma evolução no desempenho dos modelos, e fazendo uso de *Deep Learning* para avaliar se redes neurais são capazes de produzir melhores resultados do que modelos baseados em árvores de decisão.

Outro diferencial do trabalho proposto consiste na validação do modelo utilizando um semestre como treino e o seguinte como base, incrementando a base de treino com dados de mais semestres, de forma semelhante à realizada por [Nicoletti et al. 2018], mas focando não somente no último semestre, de forma a mostrar a evolução do modelo preditivo. Este trabalho utilizará, em um primeiro momento, os dados do primeiro semestre coletado das disciplinas como treinamento para os modelos preditivos, utilizando o semestre seguinte como base de testes. Em um segundo momento, o semestre utilizado como teste no momento anterior se tornará também parte da base de treinamento, tendo o semestre seguinte utilizado como teste, e assim sucessivamente até que todos os semestres sejam testados. Desta forma, teremos condições que concluir se os modelos preditivos criados são capazes de evoluir com o passar do tempo.

Além disso, o trabalho visa avaliar o desempenho obtido não somente com um modelo criado utilizando árvore de decisão, mas fazendo uso também de Redes Neurais Recorrentes, para avaliar se a utilização de *Deep Learning* consegue entregar melhor acurácia na predição de desempenho utilizando padrões sequenciais.

A Tabela 2 exibe de forma sucinta as semelhanças e diferenças dos trabalhos anteriormente mencionados com o proposto neste artigo.

Tabela 2. Trabalhos relacionados

Autor(es)	**Múltiplas Disciplinas**	**Padrões Sequenciais**	**Machine Learning**	**Deep Learning**
Nicoletti et al (2018)	Não	Não	Sim	Não
Jaber et al (2016)	Não	Sim	Sim	Não
Rojanavasu (2019)	Não	Não	Sim	Não
Bandeira (2016)	Sim	Sim	Sim	Não
Tang et al (2016)	Não	Não	Não	Sim

4. Objetivo

O objetivo de uma pesquisa apresenta o assunto a ser desenvolvido, indicando o objetivo geral, que faz uma descrição sucinta do que deve ser atendido na pesquisa, e os objetivos específicos, que servem de instrumento concreto para atender o objetivo geral proposto na pesquisa [Lakatos e de Andrade Marconi 2003].

4.1. Objetivo Geral

- Melhorar a predição do desempenho de alunos de disciplinas à distância a partir do log de atividades de um ambiente virtual de aprendizagem, considerando sequências de iterações realizadas pelos estudantes.

4.2. Objetivos Específicos

- Modelar uma base de dados que possui dados referentes a padrões sequenciais realizados dentro de um ambiente virtual de aprendizagem.
- Avaliar a performance dos modelos criados e definir, através das métricas mais indicadas para o contexto dos dados utilizados, o melhor modelo para cada disciplina.
- Definir visualizações que consigam externar quais modelos possuem o melhor desempenho para cada disciplina e tipo de dados utilizados.

5. Método

A pesquisa tem como objetivo viabilizar a predição do desempenho de alunos tendo como fonte de dados o log de atividades no ambiente virtual de aprendizagem (AVA) Moodle, de uma universidade privada do Rio Grande do Sul. Desta forma, esta pesquisa fará uso do método *Design Science Research* (DSR), uma abordagem de pesquisa que legitima o desenvolvimento de artefatos como um meio para a produção de conhecimentos científicos. [Pimentel et al. 2019].

A abordagem DSR não traz um método consensual sobre como proceder com a pesquisa, mas algumas etapas são recorrentes entre metodologias já conhecidas e aplicadas em *Data Mining*. Uma destas metodologias é o *Cross Industry Standard Process for Data Mining* (CRISP-DM), criado em 1996 com o objetivo de propor um *framework* padrão para o processo de mineração de dados adequado a qualquer projeto, como avaliação de evasão no ensino superior [Castro R. et al. 2018].

A metodologia do CRISP-DM é ilustrada na Figura 2, a qual sugere que o resultado de um dos processos determinará o próximo passo do ciclo de mineração de dados. A modelagem de dados, por exemplo, não necessita necessariamente ser prosseguida pela etapa avaliação do(s) modelo(s) criado(s) para se comportar(em) de acordo com os dados de entrada. Ao perceber-se de que os dados de entrada necessitam de novas modificações, se retorna à etapa de preparação de dados para atender as necessidades do processo de modelagem, até que todos os requisitos do projeto estejam satisfeitos para que se prossiga com o processo avaliativo [Ncr et al. 2004].

As etapas do CRISP-DM existem para que se tenha uma progressão bastante flexível, de acordo com as necessidades de negócio e dos dados. O círculo ao redor das etapas simboliza a natureza cíclica do CRISP-DM, ou seja, o fato de que um modelo preditivo foi de fato implantado no ambiente de produção dos dados não necessariamente significa de que todas as perguntas para as quais o modelo foi desenvolvido serão respondidas. Este modelo pode levar a descobertas que podem levar a novas perguntas, que podem levar a uma modificação do modelo, o que torna a repetir a metodologia com base nos novos dados e descobertas. Em dadas circunstâncias, pode ser que este processo siga um ciclo contínuo de aperfeiçoamento.

O CRISP-DM possui seis etapas, sendo elas:

1. ***Business Understanding* (Entendimento do negócio)**: Esta fase inicial tem como objetivo identificar quais os objetivos devem ser atendidos pela perspectiva do negócio, de forma que estes objetivos sejam convertidos em um processo de mineração de dados capaz de atingi-los.

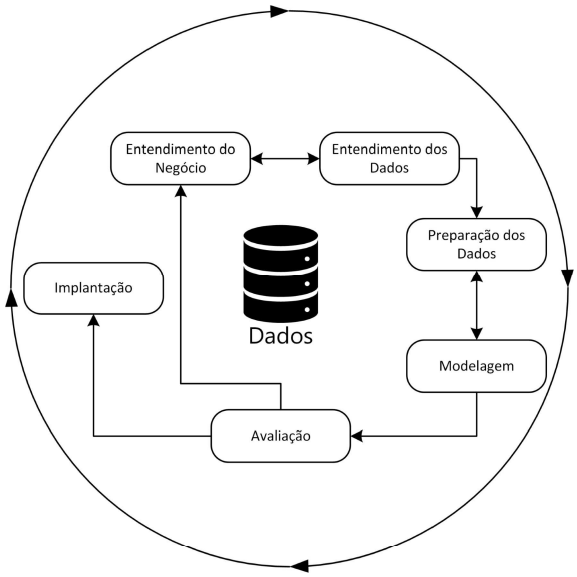


Figura 2. Metodologia do CRISP-DM
Fonte: [Ncr et al. 2004]. Adaptado pelo autor

2. ***Data Understanding*** **(Entendimento dos dados)**: A fase de entendimento dos dados consiste em levantar a origem dos dados a serem utilizados no processo, bem como identificar a forma como eles serão coletados para que o processo de mineração de dados seja conduzido adequadamente.
3. ***Data Preparation*** **(Preparação dos dados)**: Esta etapa cobre todas as modificações que os dados coletados na etapa anterior necessitam para que seja viável a criação de modelos preditivos.
4. ***Modeling*** **(Modelagem)**: Na modelagem, são avaliadas as técnicas mais adequadas para atender os objetivos identificados na etapa de entendimento do negócio, as quais são aplicadas nos dados já preparados.
5. ***Evaluation*** **(Avaliação)**: Na etapa de avaliação, os dados já estão preparados e hábeis para interagir com modelos preditivos já adequados pelos dados de entrada. Aqui, é avaliado se o desempenho deste(s) modelo(s) é suficiente para atender os objetivos de negócio. Caso ainda não seja o suficiente para que se tenha um modelo implantado no ambiente de produção, as etapas anteriores são revisitadas, podendo ser revisto o processo inteiro com o objetivo de se chegar a um modelo satisfatório.
6. ***Deployment*** **(Implantação)**: Aqui, o modelo capaz de satisfazer as necessidades levantadas inicialmente é de fato implementado nos ambientes de produção ao qual foi previsto. Porém, como levantado anteriormente, isto não significa que este modelo não será modificado. As descobertas trazidas por este modelo podem levar a novas perguntas, que podem levar a modificar este modelo de forma a atender novas necessidades.

Falando especificamente a respeito da etapa de modelagem, ainda que seja fácil encontrar frameworks livres e robustos para a realização de experimentos, podem ser desconsiderados parâmetros que são automaticamente configurados para execução do modelo, mas que podem não ser os mais adequados, se levando em conta as características dos dados [Cechinel e Camargo 2019]. Esta pesquisa irá focar na avaliação de parâmetros

que influenciam no desempenho do modelo preditivo que será criado.

A respeito das etapas de Entendimento do Negócio e Entendimento dos Dados, o resultado das partes destes processos já foi especificada antes, como os objetivos geral e específicos do trabalho, que referenciam a definição dos objetivos de mineração de dados, e a verificação da qualidade dos dados, levando em consideração que este trabalho não cobre um processo de extração de dados propriamente dita, já que a coleta dos mesmos já havia sido realizada.

O objetivo deste estudo de caso é desenvolver um modelo preditivo capaz de, através dos logs de atividades de alunos utilizando a plataforma Moodle, identificar alunos que terão ou não êxito na conclusão de seus cursos. Para atingir esse objetivo, este trabalho faz uso de três bases de dados, onde cada uma contém logs de acesso a diferentes funcionalidades do ambiente virtual de aprendizado Moodle. As disciplinas trabalhadas neste artigo são as seguintes:

- Matemática para Administração.
- Fundamentos do Processo Administrativo.
- Oficina de Raciocínio Lógico.

5.1. Preparação dos dados

A preparação dos dados é o processo que responde por aproximar os objetivos levantados no entendimento do negócio das suas devidas realizações. Preparar os dados para que estejam adequados para os modelos a serem desenvolvidos é o que mais consome tempo considerando todas as etapas do CRISP-DM, podendo variar entre 50% a 80%, o que segundo [Mansingh et al. 2017], remanesce como uma arte que depende especificamente dos conhecimentos do analista responsável por trabalhar com os dados.

Durante o processo de carga, transformação e modelagem de dados, é necessário lidar tanto com os dados originalmente providenciados como as tabelas geradas através de diversas transformações. No mundo do Data Science, a forma mais comum de disponibilizar estas bases de dados é em arquivos CSV, que já são conhecidos há muito tempo pela sua ampla utilização em sistemas e em programas como o Microsoft Excel.

Durante todas as etapas do estudo de caso, foi necessário lidar com fontes de dados que começam com os logs de dados dos estudantes, disponibilizados no formato CSV, seguindo com DataFrames dos dados transformados. Porém, levando em conta a necessidade de trabalhar várias vezes com as bases de dados providenciadas, identificou-se que o tempo para carregar os dados em um DataFrame era bastante longo, ultrapassando os 5 minutos apenas para carregar todos os dados que foram trabalhados. Desta forma, seria necessária alguma transformação visando melhorar a performance na carga destes dados.

Para minimizar o tempo de carga dos dados, as bases de dados com os dados de cada disciplina são inicialmente convertidas para o formado Parquet. Este é um formato que armazena os dados orientados a colunas no lugar de linhas, o que dá uma maior performance no sentido de tamanho utilizado no armazenamento e velocidade de carga de toda a base de dados, ao mesmo tempo em que reduz o tamanho do arquivo.

Sendo desenvolvido pela Apache, o Parquet provou ser um formato que, além de comprimir as bases de dados do estudo de caso, é capaz de carregar a mesma quantidade de dados em uma janela de tempo muito menor do que fazendo uso do formato CSV. As

Tabelas 3 e 4 fazem um comparativo com os ganhos de utilização dos arquivos no formato Parquet em comparação com o CSV, tendo como base os arquivos de log de alunos em ambos os formatos.

Tabela 3. Tempo de carga de bases de dados com diferentes sistemas de arquivo

	Tempo de Carga		
Disciplina	**CSV**	**Parquet**	**Redução (%)**
Fundamentos do Processo Administrativo	107,09s	0,53s	99,50%
Matemática para Administração	36,29s	0,24s	99,33%
Oficina de Raciocínio Lógico	66,64s	0,34s	99,48%

Tabela 4. Tamanho das bases de dados com diferentes sistemas de arquivo

	Tamanho		
Disciplina	**CSV**	**Parquet**	**Redução (%)**
Fundamentos do Processo Administrativo	49.999kB	4.422kB	91,15%
Matemática para Administração	24.521kB	1.904kB	92,23%
Oficina de Raciocínio Lógico	44.126kB	2.886kB	93,45%

Como ilustrado nas Tabelas 3 e 4, a utilização de arquivos Parquet reduz o tempo de carga dos dados em mais de 99%, além de reduzir o tamanho das bases de dados em mais de 91%, se mostrando uma etapa muito relevante em termos de performance para todo o processo de preparação de dados. Levando esta eficiência em termos de armazenamento e velocidade, todo o restante do processo passou a utilizar arquivos neste formato, visando a melhor performance possível em termos de carga de bases de dados e utilização eficiente de recursos.

Conforme [Ncr et al. 2004], a finalização do processo de preparação dos dados terá como resultado uma base de dados totalmente adequada para uso pela etapa de modelagem, que utilizará um ou mais algoritmos visando a criação de um modelo preditivo capaz de entregar resultados que atendam os objetivos levantados na etapa de entendimento do negócio. A preparação de dados consiste em cinco etapas que servem de norte para a finalização da base de dados apta a ser usada na modelagem, sendo os seguintes:

1. **Seleção dos dados**: Determina quais atributos são relevantes para a construção dos dados a serem utilizados pelas ferramentas de modelagem.
2. **Limpeza dos dados**: Trata quaisquer ruídos existentes nas colunas que possam, de alguma forma, prejudicar o desempenho dos processos seguintes.
3. **Construção dos dados**: Responsável pela criação de novos dados baseados nos dados já existentes com o objetivo de prover colunas que possam fazer mais sentido ou entregar melhor performance aos modelos criados.
4. **Integração dos dados**: Junta dados de diferentes tabelas para formar uma nova base de dados.
5. **Formatação dos dados**: Realiza qualquer modificação necessária nos dados de forma que seu significado se mantenha intacto, mas que possa ser trabalhado pelas ferramentas de modelagem utilizadas no processo.

Considerando a particularidade dos dados e da natureza cíclica dos processos descritos no CRISP-DM, cada um destes passos não é executado de maneira rígida. A formatação de dados em certo momento é necessária antes que seja feita a construção de novos dados, que são construídos separadamente (agrupados por semestre) e então integrados de forma a se obterem os dados relevantes para a condução do processo de modelagem.

Tendo os dados já otimizados em termos de carga, se torna necessário transformar os dados de forma que cada aluno esteja associado a uma lista de sequências que correspondem pelo histórico de ações realizadas na plataforma Moodle. A Figura 3 ilustra de que forma os dados estão inicialmente disponíveis.

	CodigoCurso	CodigoDisciplina	CodigoLog	ID	CodigoTurma	Data	DiaSemana	CodigoPessoa	Modulo	Acao
0	190	50404	51080868	10461-4428	4428	2012-02-29 15:04:12.120	4	10461	course	view
1	190	50404	51159978	10461-4428	4428	2012-03-01 10:19:13.130	5	10461	course	view
2	190	50404	51161287	10461-4428	4428	2012-03-01 10:38:07.070	5	10461	resource	view
3	190	50404	51161307	10461-4428	4428	2012-03-01 10:38:29.290	5	10461	forum	view forum
4	190	50404	51161316	10461-4428	4428	2012-03-01 10:38:43.430	5	10461	forum	view forum

Figura 3. Amostra da base de dados inicial
Fonte: Elaborado pelo autor

Considerando que a saída esperada deste processo são os dados prontos para serem utilizados na etapa de modelagem, estes devem se resumir em uma coluna que corresponde ao aluno mapeado, e as colunas seguintes correspondentes aos padrões sequenciais identificados, junto com a quantidade de ocorrências realizadas pelos alunos. Para se chegar nesse resultado, a preparação de dados cuida da realização destes itens:

1. Conversão dos eventos em números
2. Obter sequências de iterações
3. Agrupar sequências por aluno
4. Converter sequências no formato de entrada da biblioteca SPMF
5. Gerar arquivos com padrões sequenciais usando o algoritmo SPAM
6. Somar ocorrências de cada padrão por aluno

A base de dados conta com a coluna 'Evento', resultante da concatenação dos valores das colunas 'Modulo' e 'Acao'. Diferentes módulos do ambiente virtual de aprendizagem possuem ações semelhantes. Desta forma, a concatenação de seus conteúdos faz com se tenha uma ideia precisa da ação realizada pelo usuário. A Tabela 5 descreve uma amostra destes dados.

Tabela 5. Amostra de dados das colunas 'Modulo', 'Acao' e 'Evento'

Modulo	Acao	Evento
course	view	course view
resource	view	resource view
forum	view forum	forum view forum

Os modelos preditivos criados na etapa de modelagem trabalham com números para a geração de equações que irão formar a fórmula mais aproximada, de acordo com

os dados de treinamento recebidos, das classificações esperadas por este modelo. Desta forma, é necessário transformar valores categóricos em numéricos para serem trabalhados corretamente, o que é feito com as colunas 'Evento', 'Semestre' e 'Resultado' dando origem às colunas 'Evento_enc', 'Semestre_enc' e 'Resultado_enc', onde cada valor categórico identificado em cada coluna passe a ser representado por um número. A Tabela 6 descreve o resultado desta transformação.

Tabela 6. Amostra de colunas com variáveis categorias codificadas

Evento	Evento_enc
course view	1
resource view	2
forum view forum	3

Com os eventos devidamente categorizados de forma numérica, é necessário identificar todas as sequências de iterações realizadas pelos alunos. Esta transformação específica começa a dividir a base de dados em bases menores, que serão individualmente conduzidas aos mesmos processos de transformação. Este agrupamento será feito por semestre (n semestres por curso), semana do semestre (todas as sequências até a terceira semana do semestre, considerando a metade do curso) e curso inteiro (todas as sequências até a sexta semana, correspondendo a todo o curso) e resultado final dos alunos (sucesso ou insucesso).

Cada uma das partes especificadas é processada para identificar as sequências de iterações da mesma forma. A ideia é separar as sequências realizadas por alunos dependendo do seu desempenho no curso e de cada momento do semestre. A Tabela 7 ilustra cada base de dados que será gerada por semestre.

Tabela 7. Bases de dados geradas por semestre com diferentes semanas limite do semestre e resultados dos alunos

Semestre	Semana	Resultado	Nome da Base de Dados
201901	3	Sucesso	201901_3_Sucesso
201901	3	Insucesso	201901_3_Insucesso
201901	6	Sucesso	201901_6_Sucesso
201901	6	Insucesso	201901_6_Insucesso

A base de dados de origem é utilizada para gerar dados relacionados a ações isoladas e ações sequenciais. Uma ação isolada corresponde a uma linha da base de dados, que possui dados sobre determinada ação realizada por um aluno. Ações sequenciais correspondem a várias ações isoladas, onde a diferença de tempo entre uma ação e outra é curta o suficiente para classificá-las como encadeadas. Em um exemplo simples, a base de dados registra que um aluno acessou a tela de login do AVA, e 15 segundos depois, acessou o fórum de dúvidas da disciplina. Em função do curto tempo entre cada ação, elas são consideradas como uma sequência de ações. Para que ações isoladas possam dar origem a uma ação sequencial, é necessário determinar um intervalo máximo de tempo entre cada iteração. Para este trabalho, o max-gap determinado foi de 60 segundos. Ou seja, se o intervalo de tempo entre duas ações isoladas for inferior a 60 segundos, elas se tornam parte de uma ação sequencial. Caso contrário, cada uma conta como uma sequência.

A identificação dos padrões sequenciais será feita fazendo uso da biblioteca de código-aberto SPMF, desenvolvida por [Fournier-Viger et al. 2016] e especializada em mineração de padrões utilizando diversos algoritmos. Um destes algoritmos é o SPAM, proposto por [Ayres et al. 2002], visando maior eficiência com longas sequências. A Tabela 8 representa os dados de entrada a serem utilizados no algoritmo SPAM, onde cada linha representa um ID, que neste trabalho é representado pelo ID do aluno, e pelas sequências realizadas por este aluno. As sequências de todos os alunos são então processadas pelo algoritmo SPAM, que irá retornar os padrões sequenciais identificados junto com o suporte de cada padrão. A Tabela 9 mostra como esta tabela deve ser transformada para que seja processada pelo algoritmo SPAM, onde o separador '-1' indica o final de uma sequência e o separador '-2' representa o final da linha.

Tabela 8. Exemplo de base de dados com sequências agrupadas por ID

ID	Sequências
1	[1], [1 2 3], [1 3], [4], [3 6]
2	[1 4], [3], [2 3], [1 5]
3	[5 6], [1 2], [4 6], [3], [2]
4	[5], [7], [1 6], [3], [2], [3]

Tabela 9. Base de dados transformada em arquivo de entrada para o SPAM

1 -1 1 2 3 -1 1 3 -1 4 -1 3 6 -1 -2
1 4 -1 3 -1 2 3 -1 1 5 -1 -2
5 6 -1 1 2 -1 4 6 -1 3 -1 2 -1 -2
5 -1 7 -1 1 6 -1 3 -1 2 -1 3 -1 -2

O resultado da identificação de padrões do SPAM será dado em um arquivo descrito como na Tabela 10, onde cada linha representa um padrão, seu suporte em relação aos dados de entrada e as linhas onde o padrão pode ser visto. Por exemplo, o padrão {[2 3], [1]} pode ser encontrado nos IDs 1 e 2. O separador '-1' divide cada conjunto de itens do padrão identificado. A Tabela 11 exibe de uma forma mais legível os dados retornados pelo processamento do SPAM.

Tabela 10. Saída do algoritmo SPAM

2 3 -1 1 -1 #SUP: 2 #SID: 1 2
6 -1 2 -1 #SUP: 2 #SID: 3 4
6 -1 2 -1 3 -1 #SUP: 2 #SID: 3 4

Tabela 11. Padrões identificados pelo algoritmo SPAM

Padrões	Suporte	IDs
{[2 3], [1]}	2	1, 2
{[6], [2]}	2	3, 4
{[6], [2], [3]}	2	3, 4

A Tabela 12 mostra o resultado final da etapa de transformação de dados. Uma tabela onde a primeira coluna representa o ID do aluno, a última o resultado do aluno

(Aprovado ou Reprovado), e as colunas restantes representam os padrões sequenciais encontrados, onde cada linha registra a quantidade de vezes que o aluno executou determinado padrão. As bases de dados geradas são agrupadas por semestre e suporte mínimo, e uma vez finalizadas, estão prontas para alimentarem os modelos preditivos que serão gerados.

Tabela 12. Padrões sequenciais identificados por aluno

ID Aluno	1	2-3	6
1	1	1	0
2	1	1	0
3	0	1	1
4	0	1	1

5.2. Modelagem

Para [Ncr et al. 2004], o resultado final do processo de modelagem será um resumo dos resultados obtidos pelos modelos criados, junto com as devidas métricas que criarão um *ranking* destes modelos. Caso os modelos criados não atinjam o resultado esperado, seus parâmetros de criação devem ser revistos e re-executados para avaliar seu desempenho. Esta revisão deve ser feita até que se acredite ter encontrado os melhores modelos preditivos. Para a criação destes modelos, as seguintes etapas são seguidas:

1. **Seleção da(s) técnica(s) de modelagem**: Definição do(s) modelo(s) ao(s) qual(is) os dados resultantes da transformação de dados serão submetidos.
2. **Geração do Design de Teste**: Definir o roteiro de criação do modelo, indicando a distribuição dos dados e definição das métricas utilizadas para avaliar a performance do(s) modelo(s).
3. **Construção do(s) modelo(s)**: Usando os dados já transformados, é feita a criação do(s) modelo(s) previamente determinado(s).
4. **Avaliação do(s) modelo(s)**: Levantar as métricas de cada modelo criado, a fim de identificar, dentre todos os modelos criados, qual apresentou melhor desempenho.

Para fins de comparação, foram selecionados dois modelos aos quais os dados dos alunos com padrões sequenciais já identificados serão submetidos, um modelo baseado em árvore de decisão e outro baseado em redes neurais. Como já mostrado no capítulo de trabalhos relacionados, existem várias publicações que fazem uso de análise de dados tabulares utilizando árvore de decisão para a criação de modelos preditivos, já que este é o formato de bases de dados mais comum de ser encontrado.

Em complemento a estes dois modelos, um modelo de rede neural recorrente será utilizado, mas com a sequência de todas as ações realizadas pelos alunos sequencialmente sendo utilizada como entrada. Além de utilizar mais um modelo a ser avaliado, este terceiro modelo também poderá ser usado para avaliar se o desempenho de um modelo pode se beneficiar do uso de padrões sequenciais. Para este modelo, os dados de entrada serão as ações isoladas de cada aluno agrupadas sequencialmente, como mostrado na Tabela 13.

Todos os modelos selecionados terão os dados de treino e teste separados da mesma forma. Em um primeiro momento, os dados do primeiro semestre de cada disciplina serão utilizados como treino, e os dados do semestre seguinte utilizados como

Tabela 13. Sequência de ações mapeadas por aluno

ID Aluno	Ações
1	3 1 2 4 0 0 0
2	1 3 3 2 4 2 0
3	1 2 2 3 1 0 0
4	3 1 2 3 2 4 2

teste. Em um segundo momento, os dados do primeiro e segundo semestres serão utilizados como treino, e o terceiro utilizado como teste. Este processo será repetido até que o último semestre de cada disciplina seja utilizado como teste. A Tabela 14 demonstra como será essa distribuição para a disciplina de Oficina de Raciocínio Lógico.

Tabela 14. Plano de execução dos modelos para Oficina de Raciocínio Lógico

Iteração	2012/1	2012/2	2013/1	2013/2
1	Treino	Teste		
2	Treino	Treino	Teste	
3	Treino	Treino	Treino	Teste

Como este trabalho faz uso de padrões sequenciais, serão criados modelos com tipos de dados diferentes para avaliar se o padrões sequenciais conseguem ser relevantes para a obtenção de modelos preditivos de melhor desempenho. Desta forma, considerando os diferentes tipos de algoritmo utilizados junto com os tipos de dados a serem utilizados por modelo, conclui-se que cada iteração irá produzir sete modelos preditivos com as seguintes combinações:

1. Modelo baseado em árvore de decisão utilizando dados com ações isoladas.
2. Modelo baseado em árvore de decisão utilizando dados com padrões sequenciais.
3. Modelo baseado em árvore de decisão utilizando dados com ações isoladas e padrões sequenciais.
4. Modelo baseado em redes neurais utilizando dados com ações isoladas.
5. Modelo baseado em redes neurais utilizando dados com padrões sequenciais.
6. Modelo baseado em redes neurais utilizando dados com ações isoladas padrões e sequenciais.
7. Modelo baseado em redes neurais recorrentes, o qual faz uso das ações isoladas.

Este último modelo segue um modelo de classificação mais utilizado para classificação de texto, como análise de sentimentos de avaliações de filmes, cujo objetivo é identificar, usando um texto avaliativo como entrada, se a avaliação é positiva ou negativa. Em analogia à proposta deste trabalho, cada ação mapeada pelo aluno é identificada como uma palavra. E a sequência de todas estas ações deverá indicar se o aluno foi aprovado ou não na disciplina cursada.

Os outros modelos irão lidar com uma estrutura de dados de entrada correspondente a uma tabela que terá a quantidade de ações isoladas e padrões sequenciais realizadas por aluno. Os dados de ações isoladas e padrões sequenciais serão utilizados tanto em conjunto quanto separados, para avaliar se a adição de dados de padrões sequenciais pode contribuir para o desempenho de um modelo preditivo. As redes neurais criadas para

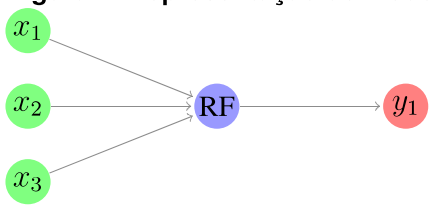
lidar com esse tipo de dado são conhecidas como *Feed-Forward*, onde o resultado de uma camada da rede neural segue somente em direção à camada seguinte. Em contrapartida, o modelo descrito no parágrafo anterior é conhecido como rede neural recorrente, onde uma camada de rede pode reutilizar seus dados de saída como entrada.

O modelo de árvore de decisão será treinado através da biblioteca *Scikit-Learn*, que consolida vários modelos de *Machine Learning*. Dentre estes vários modelos, o escolhido para este estudo é o modelo *Random Forest*, que consiste em gerar várias árvores de decisão, cada uma contendo uma fração das colunas da base de dados. Desta forma, o modelo consegue destacar mais rapidamente os dados mais relevantes para o bom desempenho da classificação dos dados [Géron 2019].

Os modelos seguintes serão criados utilizando a biblioteca *Keras*, que disponibiliza uma forma simples de criar uma rede neural de qualquer complexidade. No caso da biblioteca *Scikit-Learn*, o processo de criação de um modelo é ainda mais fácil, sendo possível através de uma única linha de código. A documentação do *Scikit-Learn* possui uma série de parâmetros que podem ser configurados para melhor se encaixarem com os dados de entrada utilizados. Para os modelos criados neste trabalho, dois parâmetros foram modificados:

- ***n_estimators***: Número de árvores sequenciais a serem criadas.
- ***max_leaf_nodes***: Quantidade de atributos a serem considerados em cada árvore.

Figura 4. Representação do Modelo *Random Forest*



Fonte: Elaborado pelo autor

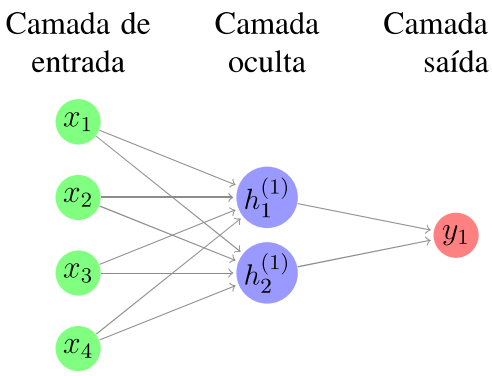
Os modelos de redes neurais serão criados através da biblioteca *Keras*, que permite criar, com poucos comandos, uma rede neural de qualquer complexidade. A biblioteca *TensorFlow*, da Google, passou a integrar esta API do Keras na sua biblioteca a partir da versão 2.0, em função da sua facilidade de criação de redes neurais[1]. Desta forma, a criação da rede pode ser feita utilizando diretamente a biblioteca *Keras* ou também através do *TensorFlow*. Começando pelo modelo de rede neural simples de arquitetura *Feed-Forward*, suas camadas serão criadas de acordo com a Figura 5.

Esta rede neural terá três camadas para lidar com os dados de treinamento, tendo as seguintes especificações:

- **Camada de Entrada**: Através desta camada serão inseridos os dados de treinamento do modelo, sendo que cada coluna do frame (tabela de dados de entradas) corresponde a um dado de entrada. Esta camada irá consolidar todos os dados de entrada em cada neurônio criado, e nesta camada, será criado um neurônio por coluna. Uma rede neural de classificação binária muito mais simples poderia ser formada com apenas um neurônio consolidando todos os dados de entrada.

[1] API do Keras na documentação oficial do TensorFlow [Google 2020].

Figura 5. Modelo de rede neural *Feed-forward*



Fonte: Elaborado pelo autor

Esta camada utiliza a função de ativação Tanh (*Hyperbolic Tangent*, Tangente Hiperbólica), que produz uma saída entre -1 e 1 através desta fórmula:

$$tanh(x) = sinh(x)/cosh(x)$$

- **Camada Oculta**: Os resultados da camada de entrada serão consolidados em neurônios densos, da mesma forma realizada na camada de entrada, sendo que a quantidade de neurônios criados nesta camada será a metade dos criados na camada anterior. Assim como na camada de entrada, os neurônios da camada oculta utilizarão a função de ativação Tanh.
- **Camada de Saída**: A camada de saída também consolida todas as entradas dos neurônios criados, mas como esta camada deve resultar em uma única saída binária, apenas um neurônio é criado. Diferente das outras camadas, os neurônios desta camada utilizarão a função sigmóide de ativação, que produz uma saída entre 0 e 1 com a seguinte fórmula:

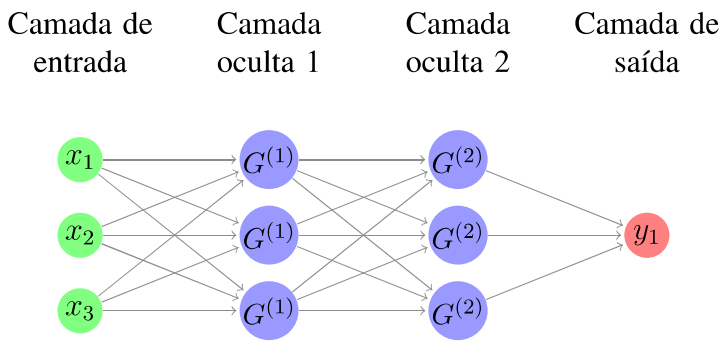
$$Sigmoid(x) = 1/(1 + exp(-x))$$

Entre as camadas da rede, é aplicada uma técnica de regularização chamada *Dropout*, que tem como objetivo desconsiderar alguns dos neurônios da camada anterior como dados de entrada da camada seguinte. A escolha das camadas desconsideradas é aleatória e ocorre a cada etapa de treinamento da rede. O uso desta técnica visa evitar o *overfitting* da rede, que faz com que a rede seja treinada para classificar perfeitamente os dados da base de treinamento, sem ser capaz de classificar com um bom desempenho qualquer outro dado.

Da mesma forma que o modelo neural anterior, o modelo de **rede neural recorrente** será criado utilizando a biblioteca *Keras*. O diferencial deste modelo se dá pelas camadas ocultas utilizadas no mesmo, adequadas para trabalhar com um tipo diferente de dado de entrada do utilizado nos demais modelos. As camadas ocultas desta rede são compostas por células GRU (*Gated Recurrent Unit*), idealizadas por [Cho et al. 2014] como uma variante simplificada do modelo LSTM, ainda que possua um desempenho semelhante.

As camadas da rede ilustrada na figura 6 são as seguintes:

Figura 6. Modelo de rede neural recorrente



Fonte: Elaborado pelo autor

- **Camada de Entrada**: Esta camada é responsável por transformar cada ação realizada pelo aluno (já transformada de texto para número) em um vetor que torne a identificação desta ação única para as camadas ocultas seguintes que irão lidar com a sequência de dados recebidos. A quantidade de colunas que irá formar este vetor de identificação única é parametrizável para se adequar com o tamanho do vocabulário da base de dados.
- **Camadas Ocultas 1 e 2**: As duas camadas ocultas deste modelo utilizam células GRU, que recebem as matrizes geradas pela camada de entrada e conseguem reter parte da informação que cada célula julga como relevante para atualizar os pesos das ações que indiquem o resultado da classe positiva ou negativa do modelo.
- **Camada de Saída**: A camada de saída da rede neural recorrente tem configuração idêntica à camada de saída da rede neural simples, já que ambas têm o mesmo objetivo de resultar em uma saída binária. Desta forma, é criado um neurônio consolidando as saídas da Camada Oculta 2 e utilizando a função sigmóide de ativação.

Depois de definida a estrutura dos modelos de redes neurais, eles devem ser compilados antes da realização do treinamento. No momento da compilação, duas variáveis cruciais para o bom desempenho do modelo devem ser definidas, a função de perda e o otimizador. A função de perda serve para indicar o quão distante um valor previsto está do valor real, e o otimizador é a função responsável pela atualização dos pesos dos dados de entrada de um neurônio. Mesmo que os dois modelos de redes neurais sejam diferentes e lidem com dados de entrada diferentes, o objetivo deles é o mesmo, e portanto, a compilação de ambos será idêntica.

A função de perda utilizada nas redes neurais criadas neste trabalho é a entropia cruzada binária, que é indicada quando o modelo deve exibir um valor previsto entre 0 e 1, mas é arredondado de acordo com a criação do modelo que utiliza esta função. Para modelos de classificação não-binária, são indicadas outras funções de entropia cruzada, como a entropia cruzada categórica.

Para atualizar os pesos dos dados de entrada dos neurônios das redes neurais criadas, é utilizado o otimizador Adam (*adaptive moment estimation*, ou estimativa de momento adaptativo), que pode adaptar a taxa de aprendizagem de um modelo de acordo com o seu treinamento [Géron 2019].

Finalizada a criação dos modelos, obtém-se uma tabela com as métricas obtidas em cada modelo. Em conjunto com estas métricas, são também geradas matrizes e gráficos que ajudam a identificar o modelo com o melhor desempenho. As métricas capturadas (que serão explicadas posteriormente) são as seguintes:

- **Acurácia**: Contém a acurácia do modelo.
- **Precisão**: Contém a precisão do modelo.
- ***Recall***: Contém o *recall* do modelo.
- **Especificidade**: Contém a especificidade do modelo.
- **F1-Score**: Contém o *F1-Score* do modelo.
- **FN**: Contém os falsos negativos do modelo.
- **FP**: Contém os falsos positivos do modelo.
- **TN**: Contém os verdadeiros negativos do modelo.
- **TP**: Contém os verdadeiros positivos do modelo.

A base das métricas de um modelo preditivo consiste na matriz de confusão. Os modelos desenvolvidos neste trabalho são de classificação binária, ou seja, o modelo busca determinar a qual das duas classes (classe positiva ou negativa) os dados de entrada pertence. No presente trabalho, a classe positiva corresponde aos alunos que concluíram a disciplina com sucesso, enquanto a classe negativa corresponde aos alunos que não concluíram a disciplina, seja por reprovação ou desistência. A matriz de confusão imprime uma matriz quadrada com os seguintes valores:

- **Verdadeiros Positivos (TP)**: Valores que correspondem à classe positiva e foram previstos corretamente.
- **Falsos Positivos (FP)**: Valores que correspondem à classe positiva, mas foram previstos como pertencentes à classe negativa.
- **Falsos Negativos (FN)**: Valores que correspondem à classe negativa, mas foram previstos como correspondentes à classe positiva.
- **Verdadeiros Negativos (TN)**: Valores que correspondem à classe negativa e foram previstos corretamente.

A Tabela 15 exemplifica a impressão de uma matriz de confusão e de suas variáveis, onde suas várias combinações dão origem a métricas capazes de identificar se o desempenho de um modelo preditivo atendeu ou não às expectativas.

Tabela 15. Exemplo de matriz de confusão

		Valor Real	
		Classe Positiva (Sucesso)	**Classe Negativa (Insucesso)**
Valor Previsto	**Classe Positiva (Sucesso)**	TP	FP
	Classe Negativa (Insucesso)	FN	TN

Para exemplificar de forma mais clara o uso da matriz de confusão, serão impressos os dados de dois modelos criados. O primeiro corresponde ao modelo desenvolvido com algoritmo baseado em *Machine Learning*, com uso dos dados de ações isoladas e dos padrões sequenciais, para a disciplina de Matemática para Administração, correspondente

à primeira iteração de semestres (usando o primeiro semestre da base de dados como treinamento e o segundo semestre como teste). O segundo modelo terá como diferença a iteração de semestres, que será a última (o último semestre da base de dados é utilizado para teste enquanto todos os outros são usados para treinamento). A ideia de comparar estes dois modelos é identificar se, com o aumento de dados para evoluir o modelo preditivo, existe um ganho de desempenho.

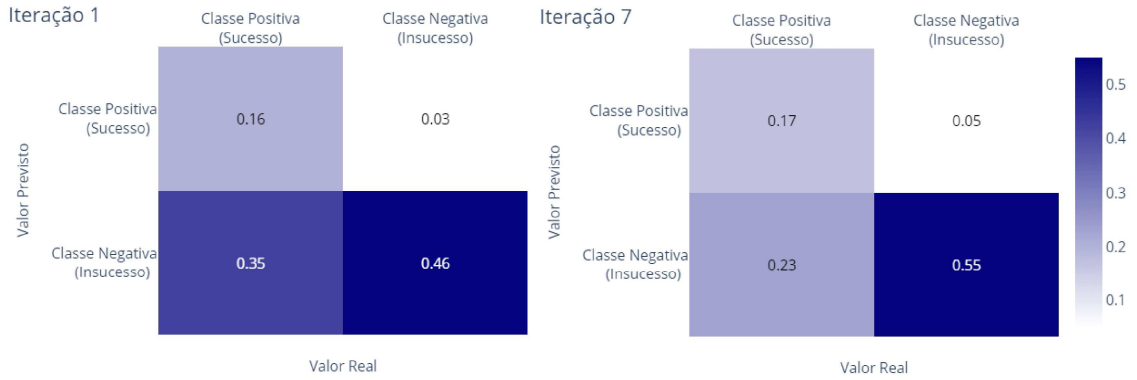


Figura 7. Comparação de matrizes de confusão entre iterações
Fonte: Elaborado pelo autor

Os resultados impressos na Figura 7 são representados de forma proporcional, em função do tamanho dos dados de testes variar entre as interações. Na primeira iteração, aproximadamente dois terços dos alunos que concluíram a disciplina com sucesso foram classificados como insucesso, enquanto conseguiu classificar corretamente quase todos os alunos que tiveram insucesso na disciplina. O modelo da última iteração conseguiu manter o desempenho da classificação dos alunos com insucesso, enquanto mostrou uma leve melhora ao classificar os alunos que concluíram com sucesso a disciplina, ainda que tenha classificado corretamente menos da metade.

Com estes dados, conseguimos calcular métricas como acurácia, precisão, *recall*, especificidade e *F1-Score*, com o objetivo de auxiliar a indicar o desempenho geral dos modelos desenvolvidos. Começando com a acurácia, ela é calculada da seguinte forma:

$$A = \frac{TP + TN}{FP + FN + TP + TN}$$

O primeiro modelo teve uma acurácia de 61,86%, enquanto que o segundo modelo teve uma acurácia de 71,66%. Considerando apenas esta métrica, leva-se à conclusão de que o modelo melhorou seu desempenho com a adição de dados de novos semestres. Porém, a métrica que melhor irá determinar se um modelo preditivo teve um desempenho satisfatório deve ser escolhida de acordo com o objetivo que este modelo deve atender.

O objetivo da criação dos modelos preditivos deste trabalho é identificar alunos que podem não concluir suas disciplinas com sucesso, e assim, intervir de forma que estes alunos consigam manter um bom desempenho e evitar, no pior caso, a sua evasão. Dado este contexto, o modelo com o melhor desempenho será o que possuir o menor FP possível, ou seja, alunos que provavelmente não finalizariam o curso com sucesso, mas foram classificados incorretamente como sucesso. Obter os maiores valores TP e TN é

o que se espera de um modelo com ótimo desempenho, mas isso não significa que os valores de FP e FN virão zerados.

Um valor FN diferente de zero é aceitável para este contexto, pois ele corresponde a alunos com tendência a concluírem determinada disciplina com sucesso, mas com o modelo classificando-os como insucesso. Para estes casos, a instituição de ensino vinculada à disciplina executaria suas ações de intervenção para alunos que provavelmente finalizariam com sucesso. Já um valor FP diferente de zero corresponde a alunos que tendem ao insucesso, sendo que o modelo os classifica como aprovados. Isto fará com que a instituição de ensino não faça nada para auxiliar aquele aluno e ele acabe reprovando, ou até mesmo evadindo do curso, o que gera prejuízo para a instituição e para o próprio aluno.

Portanto, as métricas indicadas para este contexto são as influenciadas pela variação do FP, levando assim ao cálculo da especificidade e precisão do modelo. A precisão de um modelo é calculada através da seguinte fórmula:

$$P = \frac{TP}{FP + TP}$$

Enquanto a especificidade de um modelo é calculada utilizando esta fórmula:

$$E = \frac{TN}{TN + FP}$$

Outra métrica utilizada para a avaliação no desempenho de modelos preditivos é o *recall*, que corresponde à taxa de verdadeiros positivos encontrados em um modelo. O *recall* é obtido através da seguinte fórmula:

$$R = \frac{TP}{FN + TP}$$

Como já mencionado, o objetivo deste modelo é ter o menor valor de FP e o maior valor de TN possíveis, fazendo com que a precisão e a especifidade sejam, para este contexto, as métricas chave para escolha do melhor modelo. A diferença entre as métricas exibidas na Tabela 16 mostra que, em relação ao primeiro modelo, houve uma redução na precisão e um aumento no *recall*, isso mantendo uma especificidade muito alta, sendo esta última métrica a mais importante do contexto dos dados. Levando em conta o contexto dos dados, ainda é positiva a evolução entre estes modelos, em função da redução dos falsos positivos e aumento dos falsos negativos.

Tabela 16. Comparação da precisão, *Recall* e especificidade entre os modelos de exemplo

Métrica	Iteração 1	Iteração 7
Precisão	82,60%	76,92%
Recall	31,66%	41,66%
Especificidade	93,10%	91,66%

Considerando o contexto dos dados analisados neste trabalho, obter um *recall* baixo pode levar a um grande esforço por parte da instituição de ensino para auxílio de

alunos sem necessidade. Como o modelo deve servir para evitar o insucesso de alunos, pecar pelo excesso pode ser uma decisão aceitável para a instituição de ensino que aplicar o modelo. Porém, buscar auxiliar uma grande quantidade de alunos sem necessidade pode levar a reclamações em função do critério usado para esta intervenção.

Os resultados obtidos na Tabela 16 indicam que o segundo modelo mostrou uma leve evolução no *recall*. Porém, este *recall* ainda é muito baixo, fazendo com que muitos alunos com bom desempenho acabem recebendo intervenção da instituição de ensino sem necessidade.

Tendo os valores de precisão e *recall*, é possível calcular o *F1-Score*, que consiste na média harmônica das duas métricas, representada pela seguinte fórmula:

$$F1 = 2 * \frac{P * R}{P + R}$$

Alinhado ao contexto dos dados avaliados pelos modelos, o *F1-Score* pode ser considerada a métrica de entrada para avaliação do desempenho de um modelo preditivo. Um *F1-Score* fora do esperado é o ponto de entrada para identificar através das métricas que o compõem (precisão e *recall*) por que determinado modelo não está se comportando bem. Os valores da Tabela 17 sugerem uma melhoria geral entre o primeiro e segundo modelo.

Tabela 17. Comparação do *F1-Score* entre os modelos de exemplo

Iteração 1	Iteração 7
45,78%	54,05%

Considerando os resultados do exemplo acima feito com dois modelos preditivos, pode-se concluir que o melhor modelo para atender os objetivos de negócio deve atender aos seguintes critérios:

- O modelo deve ter uma especificidade alta, mesmo que isso leve a um *recall* baixo, considerando os objetivos que este modelo deve atender.
- O modelo pode ter *recall* baixo, considerando os objetivos que este modelo deve atender.
- O modelo pode ter uma precisão alta, mas considerando os objetivos primários do modelo, a precisão pode ser menor em detrimento de uma especificidade alta.

Tendo os critérios para seleção do melhor modelo já estabelecidos, pode-se buscar as métricas dos modelos criados a fim de identificar o modelo de melhor desempenho. O primeiro passo é listar as métricas de precisão de todos os modelos. As Tabelas 18 a 22 exibem as métricas por disciplina, onde as linhas correspondem aos modelos, e as colunas correspondem à disciplina, em que:

- **NN** corresponde a modelos de redes neurais que utilizam tanto os dados de ações isoladas quanto os dados de padrões sequenciais.
- **NN-IS** corresponde a modelos de redes neurais que utilizam dados de ações isoladas.
- **NN-SP** corresponde a modelos de redes neurais que utilizam dados de padrões sequenciais.
- **RF** corresponde a modelos de *Random Forest* que utilizam tanto os dados de ações isoladas quanto os dados de padrões sequenciais.

- **RF-IS** corresponde a modelos de *Random Forest* que utilizam dados de ações isoladas.
- **RF-SP** corresponde a modelos de *Random Forest* que utilizam dados de padrões sequenciais.
- **RNN** corresponde a modelos de redes neurais recorrentes.
- **FundProcAdm** corresponde à disciplina de Fundamentos do Processo Administrativo.
- **MatAdm** corresponde à disciplina de Matemática para Administração.
- **OficinaRacLog** corresponde à disciplina de Oficina de Raciocínio Lógico.

A Tabela 18 mostra as precisões encontradas no último semestre de teste para cada disciplina e modelo utilizado. As melhores métricas coletadas para cada disciplina foram:

- 89,74% para Fundamentos do Processo Administrativo, com o Modelo **RNN**.
- 100,00% para Matemática para Administração, com o Modelo **RNN**.
- 79,23% para Oficina de Raciocínio Lógico, com o Modelo **RNN**.

Tabela 18. Tabela de precisões

Modelo	FundProcAdm	MatAdm	OficinaRacLog
NN	82,22%	63,63%	73,58%
NN-IS	85,41%	88,88%	74,63%
NN-SP	86,48%	61,53%	75,32%
RF	83,33%	89,47%	78,14%
RF-IS	82,69%	80,00%	76,19%
RF-SP	82,69%	85,71%	78,00%
RNN	**89,74%**	**100,00%**	**79,23%**

Ainda que cada disciplina tenha um resultado que favoreça um modelo diferente, é interessante verificar como cada modelo melhorou à medida que dados de outros semestres fossem adicionados aos dados de treino. Essa análise pode ser feita através da Figura 8, que leva a algumas conclusões:

- O modelo de redes neurais recorrentes apresentou a melhor precisão para todas as disciplinas, sendo que para Matemática para Administração, apesar da precisão ter chegado a 0% no segundo semestre testado, ela chegou aos 100% a partir do penúltimo semestre testado.
- Analisando cada disciplina, a evolução de todas as precisões ocorreu de maneira bastante semelhante em cada modelo, com exceção do modelo de redes neurais simples usando ações isoladas e do modelo de redes neurais recorrentes para Matemática para Administração, que mostraram uma volatilidade bem maior, apesar do modelo de redes neurais recorrentes ter atingido precisão máxima.
- Para Oficina do Raciocínio Lógico, todos os modelos tiveram uma precisão muito parecida, tanto no resultado final quanto na evolução do modelo com a inserção de dados de mais semestres. Apesar da grande semelhança no resultado de todos os modelos, o modelo de redes neurais recorrentes teve o melhor desempenho, mesmo com tendência de queda. Depois deste modelo, os modelos de rede neural simples e *Random Forest* que utilizaram apenas dados de padrões sequenciais mostraram um resultado melhor do que os modelos que avaliaram apenas ações isoladas.

Figura 8. Precisão dos modelos para todas as disciplinas
Fonte: Elaborado pelo autor

A Tabela 19 mostra os *Recalls* encontrados no último semestre de teste para cada disciplina e modelo utilizado. As melhores métricas coletadas para cada disciplina foram:

- 91,83% para Fundamentos do Processo Administrativo, com o Modelo **RF**.
- 75,00% para Matemática para Administração, com o Modelo **RF-SP**.
- 88,72% para Oficina de Raciocínio Lógico, com o Modelo **RF**.

Tabela 19. Tabela de *Recalls*

Modelo	FundProcAdm	MatAdm	OficinaRacLog
NN	75,51%	58,33%	87,96%
NN-SP	65,30%	66,66%	87,21%
NN-IS	83,67%	33,33%	77,44%
RF	**91,83%**	70,83%	**88,72%**
RF-SP	87,75%	**75,00%**	87,96%
RF-IS	87,75%	66,66%	84,21%
RNN	71,42%	29,16%	77,44%

Analisando a evolução do *Recall* por semestre e por disciplina, na Figura 9, pode-se tirar as seguintes conclusões:

- Para Matemática para Administração, quase todos os modelos tiveram evolução semelhante com o passar dos semestres, com exceção do modelo de redes neurais simples usando ações isoladas e do modelo de redes neurais recorrentes, que, além de mostrarem um *Recall* inferior aos demais, demonstraram uma tendência de desempenho de queda maior do que os outros modelos.
- Para Fundamentos do Processo Administrativo, os modelos de *Random Forest* mantiveram um *Recall* alto no último semestre testado, diferente dos outros modelos, que mostraram uma grande tendência de queda.

Figura 9. *Recall* dos modelos para todas as disciplinas
Fonte: Elaborado pelo autor

- Para Oficina de Raciocínio Lógico, todos os modelos mostraram tendência de alta no *Recall* com o passar dos semestres.

A Tabela 20 mostra as especificidades encontradas no último semestre de teste para cada disciplina e modelo utilizado. As melhores métricas coletadas para cada disciplina foram:

- 68,42% para Fundamentos do Processo Administrativo, com o Modelo **RNN**.
- 94,44% para Matemática para Administração, com o Modelo **RF**.
- 65,06% para Oficina de Raciocínio Lógico, com o Modelo **NN-SP**.

Tabela 20. Tabela de especificidades

Modelo	FundProcAdm	MatAdm	OficinaRacLog
NN	52,63%	00,00%	63,85%
NN-SP	00,00%	55,55%	**65,06%**
NN-IS	63,15%	52,77%	54,21%
RF	52,63%	**94,44%**	60,24%
RF-SP	52,63%	91,66%	60,24%
RF-IS	52,63%	88,88%	57,83%
RNN	**68,42%**	02,77%	63,85%

Analisando a evolução da especificidade por semestre e por disciplina, na Figura 10, pode-se tirar as seguintes conclusões:

- Para Matemática para Administração, os modelos de *Random Forest* mostraram desempenho semelhante, além de resultarem nas melhores especificidades. Todos os outros modelos tiveram uma volatilidade muito grande, apesar dos modelos de redes neurais simples mostrarem uma boa evolução nos últimos semestres. O modelo de redes neurais recorrentes mostrou uma especificidade muito baixa em todos os semestres avaliados.

- Para Fundamentos do Processo Administrativo, os dados de padrões sequenciais impactaram negativamente nos modelos de redes neurais, reduzindo drasticamente a especificidade no último semestre. No modelo que utiliza apenas padrões sequenciais, a especificidade chegou a 0%. O modelo de redes neurais recorrentes, apesar de ter uma especificidade de 0% no primeiro semestre avaliado, apresentou a melhor especificidade para esta disciplina e não apresentou uma queda significativa de desempenho nos últimos semestres.
- Para Oficina de Raciocínio Lógico, os modelos de redes neurais simples e recorrente, independente dos dados utilizados, mostraram melhorias no desempenho de cada modelo.

Figura 10. Especificidade dos modelos para todas as disciplinas
Fonte: Elaborado pelo autor

A Tabela 21 mostra o *F1-Score* encontrado no último semestre de teste para cada disciplina e modelo utilizado. As melhores métricas coletadas para cada disciplina foram:

- 87,37% para Fundamentos do Processo Administrativo, com o Modelo **RF**.
- 80,00% para Matemática para Administração, com o Modelo **RF-SP**.
- 83,09% para Oficina de Raciocínio Lógico, com o Modelo **RF**.

Tabela 21. Tabela de *F1-Score*

Modelo	FundProcAdm	MatAdm	OficinaRacLog
NN	78,72%	60,86%	80,13%
NN-SP	74,41%	64,00%	80,83%
NN-IS	84,53%	48,48%	76,01%
RF	**87,37%**	79,06%	**83,09%**
RF-SP	85,14%	**80,00%**	82,68%
RF-IS	85,14%	72,72%	80,00%
RNN	79,54%	45,16%	78,32%

Analisando a evolução do *F1-Score* por semestre e por disciplina, na Figura 11, pode-se tirar as seguintes conclusões:

- Cada disciplina teve todos os modelos executados apresentando a mesma evolução do *F1-Score*, tendo como única exceção o modelo de redes neurais simples usando ações isoladas para Matemática para Administração, que teve um desempenho bem inferior aos demais.
- Para Oficina do Raciocínio Lógico, os modelos de redes neurais simples e recorrente mostraram melhoria no desempenho do *F1-Score* com o passar dos semestres.
- Para Matemática para Administração, os modelos de redes neurais simples que fazem uso de padrões sequenciais e o modelo de redes neurais recorrentes mostraram tendência de evolução do *F1-Score* nos últimos semestres testados.
- Para Fundamentos do Processo Administrativo, apesar do desempenho bastante semelhante em todos os modelos, os modelos de redes neurais simples e recorrentes mostraram as maiores quedas de desempenho do *F1-Score* nos últimos semestres testados.

Figura 11. *F1-Score* dos modelos para todas as disciplinas
Fonte: Elaborado pelo autor

A Tabela 22 mostra a acurácia encontrada no último semestre de teste para cada disciplina e modelo utilizado. As melhores métricas coletadas para cada disciplina foram:

- 80,88% para Fundamentos do Processo Administrativo, com o Modelo **RF**.
- 85,00% para Matemática para Administração, com os Modelos **RF** e **RF-SP**.
- 77,77% para Oficina de Raciocínio Lógico, com o Modelo **RF**.

Analisando a evolução da acurácia por semestre e por disciplina, na Figura 12, pode-se tirar as seguintes conclusões:

- Para Oficina do Raciocínio Lógico, os modelos de redes neurais simples e recorrente mostraram melhoria no desempenho do *F1-Score* com o passar dos semestres.

Tabela 22. Tabela de acurácias

Modelo	FundProcAdm	MatAdm	OficinaRacLog
NN	70,58%	70,00%	73,14%
NN-SP	67,64%	70,00%	74,53%
NN-IS	77,94%	71,66%	69,90%
RF	**80,88%**	**85,00%**	**77,77%**
RF-SP	77,94%	**85,00%**	77,31%
RF-IS	77,94%	80,00%	74,07%
RNN	73,52%	71,66%	73,61%

- Para Fundamentos do Processo Administrativo, os modelos de redes neurais mostraram as maiores quedas de desempenho nos últimos semestres, enquanto os modelos de *Random Forest* que utilizam dados de ações isoladas, apesar da volatilidade conforme o andamento dos semestres, mostraram tendência de ganho de desempenho nos últimos semestres.
- Para Matemática para Administração, os modelos de redes neurais simples que usam dados de padrões sequenciais e o modelo de redes neurais recorrentes mostraram tendência de crescimento nos últimos semestres.

Figura 12. Acurácia dos modelos para todas as disciplinas
Fonte: Elaborado pelo autor

Além das métricas já mostradas, existem dois gráficos que fazem uso das métricas já obtidas, ajudando a avaliar e interpretar o desempenho de modelos de classificação binária, a curva ROC (*Receiver Operating Characteristic*) e a curva *Precision-Recall*.

A curva ROC é a relação entre a taxa de verdadeiros positivos e a taxa de falsos positivos, sob cada limiar de aceitação que pode ser utilizado no modelo. O limiar de aceitação é um valor entre 0 e 1 que define sob qual circunstância um valor previsto deve ser classificado como positivo ou negativo. A curva ROC exibe a variação da taxa de verdadeiros positivos e falsos positivos com uma variedade de limiares possíveis a serem

empregados no modelo preditivo. Além de servir como avaliação do desempenho do modelo, a curva ajuda a identificar um limiar que consiga atender os objetivos que o modelo deve atender, pois como mencionado antes, faz mais sentido uma determinada métrica ter um valor maior do que as outras dependendo do contexto que o modelo preditivo deve atender. A curva *Precision-Recall* possui a mesma premissa da curva ROC, mas é obtida através da relação entre a precisão e o *Recall* de um modelo sob vários limiares de aceitação.

Para [Géron 2019], a curva *Precision-Recall* é mais indicada para a avaliação do desempenho de um modelo quando existe uma preocupação maior sobre os falsos positivos do que para os falsos negativos, o que corresponde com os objetivos a serem atendidos pelos modelos utilizados. Outro fator que favorece a utilização da curva *Precision-Recall* é exposto por [Saito e Rehmsmeier 2015], sugerindo que a curva *Precision-Recall* é mais informativa para bases desbalanceadas do que a área sobre a curva ROC, em função do desbalanceamento dos dados ser visível na curva *Precision-Recall*, diferente da área sobre a curva ROC, como mostra a Figura 13.

O modelo preditivo mostra um bom desempenho quando a curva *Precision-Recall* é vista dentro da área verde de cada gráfico. Cada gráfico de exemplo possui uma proporção diferente de classes positivas e negativas, o que leva a áreas de tamanhos diferentes. A linha verde de cada gráfico exemplifica um classificador perfeito, onde a precisão máxima é obtida com qualquer *Recall* resultado pelo modelo, enquanto a linha pontilhada serve de referência para uma classificação randômica. Quando a curva *Precision-Recall* vai desta linha pontilhada para a área vermelha, significa que o modelo faz uma classificação pobre dos dados recebidos.

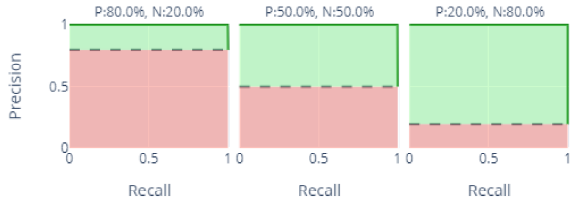


Figura 13. Exemplo da curva *Precision-Recall* sobre dados desbalanceados.
Fonte: [Saito e Rehmsmeier 2015]. Adaptado pelo autor

A Figura 14 mostra a proporção da quantidade de alunos aprovados e reprovados por semestre e por disciplina. Para a disciplina de Fundamentos do Processo Administrativo, a quantidade de alunos aprovados possui uma proporção aproximada entre 70 e 80% das bases de dados de testes, o que sugere que a curva de *Precision-Recall* seja observada com mais atenção neste caso na hora de avaliar o desempenho dos modelos testados. Já para as disciplinas de Matemática para Administração e Oficina do Raciocínio Lógico, essa proporção é reduzida para uma margem entre 60 e 70%, com um caso específico no primeiro semestre de Matemática para Administração, onde a proporção entre alunos aprovados e reprovados é de quase 50%, o que reduz o peso da análise sobre a curva de *Precision-Recall* sobre os modelos, evidentemente, sem descartar os *insights* da métrica.

Com a definição da relação entre as classes positivas e negativas de cada semestre e disciplina, a *baseline* de cada semestre pode ser exibida nos gráficos de *Precision-Recall* de cada modelo. Desta forma, as Figuras 15, 16 e 17 mostram a variação das precisões

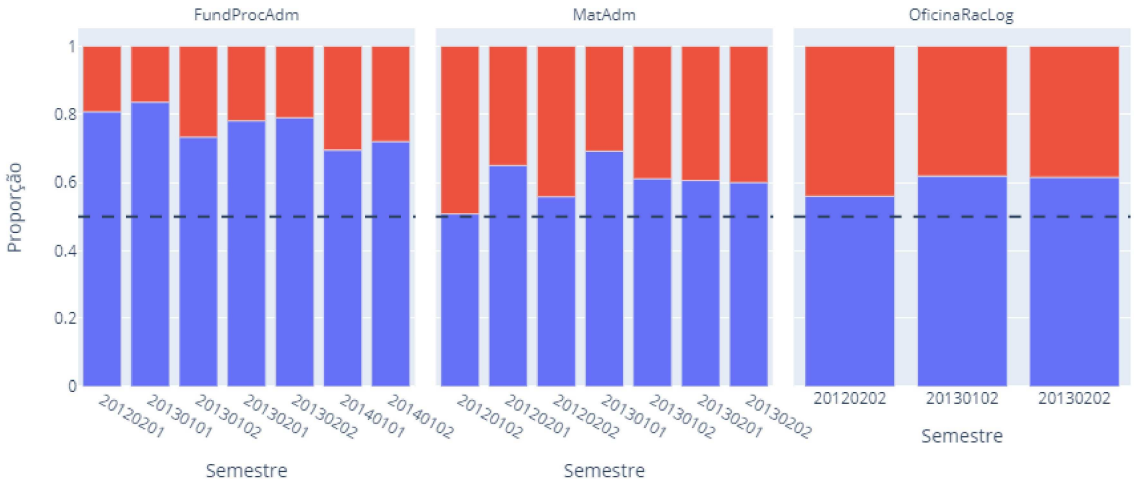


Figura 14. Balanceamento das Bases de Teste
Fonte: Elaborado pelo autor

e *recalls* dos modelos criados, sendo que os gráficos da primeira linha correspondem ao primeiro semestre avaliado e a segunda linha exibe os gráficos do último semestre. Desta forma, pode-se comparar a evolução do desempenho de cada modelo conforme dados de outros semestres são adicionados ao treinamento do modelo.

Começando pela Figura 15, com as curvas de *Precision-Recall* de Fundamentos do Processo Administrativo, disciplina cujas bases de dados são mais desbalanceadas, indicam que a precisão de quase todos os modelos foi reduzida, comparando os dados do primeiro e do último semestre testados. Ou de forma mais condizente com os gráficos, para manter a precisão alta, seria preciso reduzir o *Recall* aceitável de cada modelo.

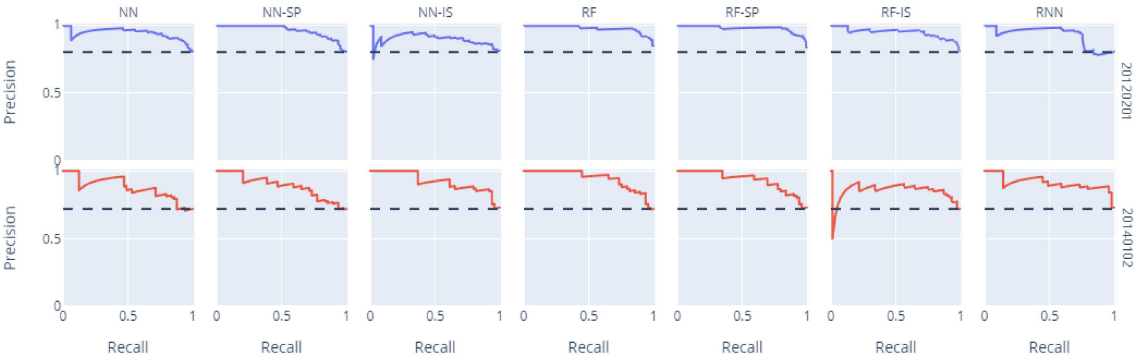


Figura 15. Curvas de Precision-Recall para Fundamentos do Processo Administrativo

Fonte: Elaborado pelo autor

As curvas de *Precision-Recall* exibidas na Figura 16 mostram que os modelos de redes neurais simples realizam uma classificação puramente randômica no primeiro semestre testado. Já com os dados do último semestre, somente a rede neural que usou somente dados de padrões isolados foi capaz de ter uma precisão mais alta, ainda que ao custo de um *Recall* baixo. Este comportamento pode ser visto também no modelo de redes neurais recorrentes, tanto no primeiro quanto no último semestre testados. Já os

modelos *Random Forest* conseguem ter uma precisão relativamente alta sem necessidade de abrir mão de um *Recall* alto.

Figura 16. Curvas de Precision-Recall para Matemática para Administração
Fonte: Elaborado pelo autor

A Figura 17 mostra que apesar de cada modelo ter uma curva bem diferente no primeiro semestre avaliado, todos mostraram uma curva semelhante na avaliação do último semestre.

Figura 17. Curvas de Precision-Recall para Oficina de Raciocínio Lógico
Fonte: Elaborado pelo autor

A Figura 18 exibe a área sobre a curva ROC variando por semestre, para a disciplina de Fundamentos do Processo Administrativo, visando mostrar a evolução da área sobre a curva conforme dados de novos semestres são adicionados aos modelos. O valor total da área sobre a curva ROC varia entre 0 e 1, sendo que uma área igual a 1 representa um classificador perfeito, e uma área igual 0.5 representa uma classificação puramente randômica. Para representar estes 0.5, cada gráfico possui uma linha pontilhada entre os pontos 0 e 1 de ambos os eixos de cada gráfico. Se a área sobre a curva ROC ocupar qualquer espaço depois desta linha pontilhada, significa que a área sobre a curva ROC é superior a 0.5.

As Figuras 19, 20 e 21, sendo cada Figura correspondente a uma disciplina, mostram a evolução da área sobre a curva ROC em cada modelo avaliado, onde cada coluna representa um modelo utilizado e cada linha um semestre testado. Cada Figura exibe os dados primeiro semestre testado na primeira linha e do último semestre testado na segunda linha, com o objetivo de comparar a evolução de cada modelo conforme dados de novos semestres são adicionados ao treinamento dos modelos.

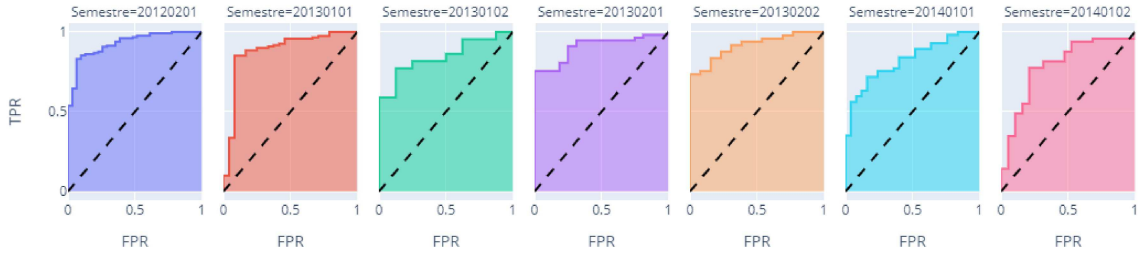


Figura 18. Exemplo de evolução por semestre da curva ROC
Fonte: Elaborado pelo autor

Os gráficos exibidos na Figura 19 mostram que os modelos que utilizaram apenas dados de ações isoladas tiveram uma área sobre a curva ROC inferior aos modelos que utilizaram dados de ações sequenciais, apesar de que quase todos os modelos tiveram uma área inferior entre o primeiro e último semestre testados. Curiosamente, o modelo de redes neurais simples com dados de ações isoladas teve uma área sobre a curva ROC maior no último semestre testado do que o primeiro semestre.

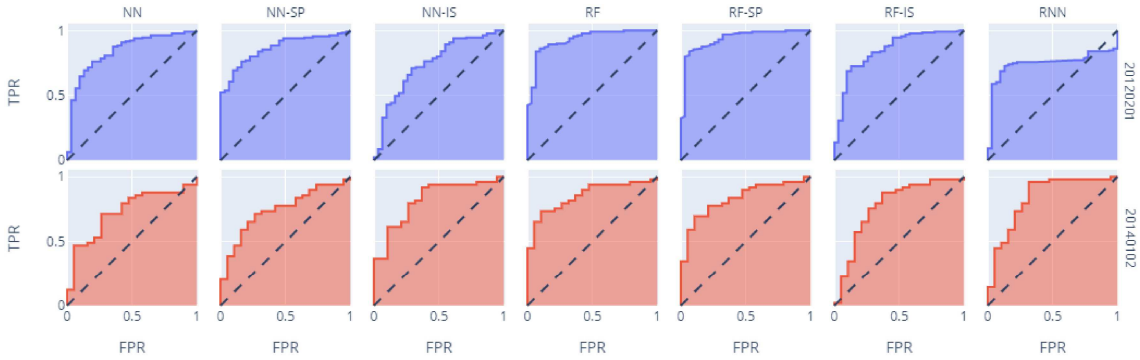


Figura 19. Curvas ROC para Fundamentos do Processo Administrativo
Fonte: Elaborado pelo autor

A Figura 20 mostra que o desempenho dos modelos de redes neurais mostraram uma classificação puramente randômica no primeiro semestre, mas tiveram boa evolução até o último semestre. Porém, este mesmo desempenho pode ser visto nos testes do primeiro semestre para o modelo de rede neural recorrente e os modelos de árvore randômica, sendo estes últimos que mostraram a maior evolução na área sobre a curva ROC comparando o primeiro e último semestres.

A Figura 21 leva a entender que, apesar da área sobre a curva ROC ser diferente para cada modelo utilizado no primeiro semestre testado, todos os modelos, com exceção do modelo de redes neurais recorrentes, mostraram uma área sobre a curva ROC muito semelhante no último semestre testado. Ainda que a área de todos os gráficos tenha sua semelhança, os modelos que utilizaram apenas dados de ações isoladas e o modelo de redes neurais recorrentes mostra uma área levemente menor que os outros modelos.

A Figura 22 mostra a evolução da área sobre a curva ROC conforme a adição de semestres aos modelos utilizados por disciplina, da mesma maneira que as figuras anteriores que avaliaram precisão, *recall*, especificidade, *F1-Score* e acurácia. Da mesma forma que os gráficos que mostram a área sobre as curvas ROC, este gráfico possui uma

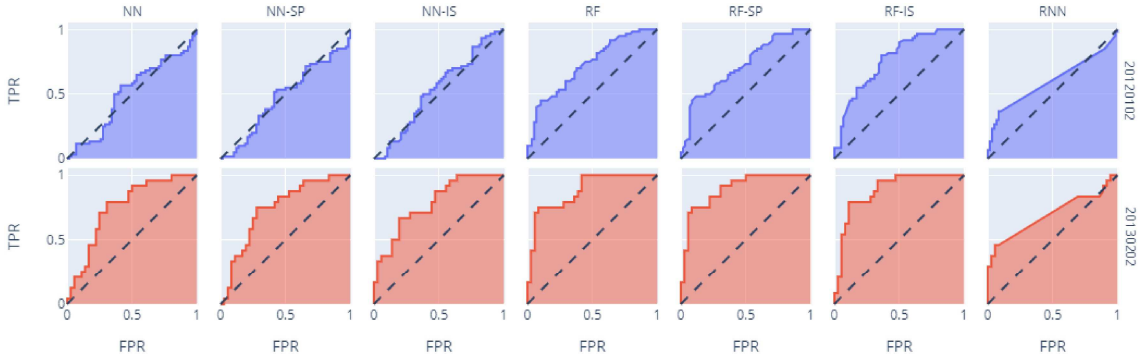


Figura 20. Curvas ROC para Matemática para Administração
Fonte: Elaborado pelo autor

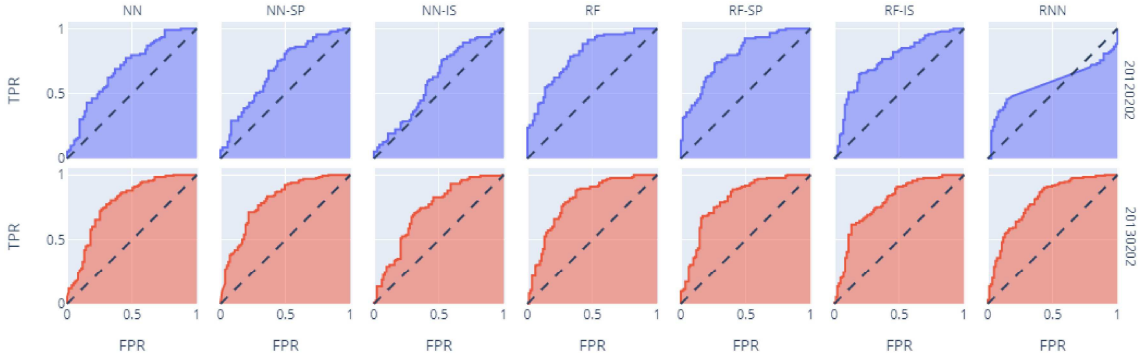


Figura 21. Curvas ROC para Oficina de Raciocínio Lógico
Fonte: Elaborado pelo autor

linha pontilhada de referência, correspondente à área sobre a curva de 0.5, ou 50%, que corresponde a um modelo de classificação puramente randômica. Esta mesma Figura faz com que seja possível prover *insights* semelhantes, como os seguintes:

- Para as três disciplinas, os modelos de *Random Forest* apresentaram os maiores valores absolutos de área sobre a curva ROC em relação aos outros modelos, e também mostram que a adição de dados de padrões sequenciais influenciou positivamente nas classificações.
- O modelo de redes neurais recorrentes, para Fundamentos do Processo Administrativo e Matemática para Administração, além de não se destacar no valor absoluto da área sobre a curva ROC, não mostrou tendência de alta neste valor com o passar dos semestres, o que influencia negativamente na escolha final deste tipo de modelo para ser utilizado neste contexto. Para Oficina do Raciocínio Lógico, este modelo obteve um desempenho semelhante aos demais.
- Para Matemática para Administração, os modelos de redes neurais simples, além de mostrarem uma volatilidade muito grande no valor absoluto da área sobre a curva ROC, tiveram amostras cuja área foi menor do que 0.5, levando a necessidade de uma reavaliação do modelo ou dos dados de entrada desta disciplina antes que se possa cogitar a utilização destes modelos para avaliar o desempenho dos alunos desta disciplina.

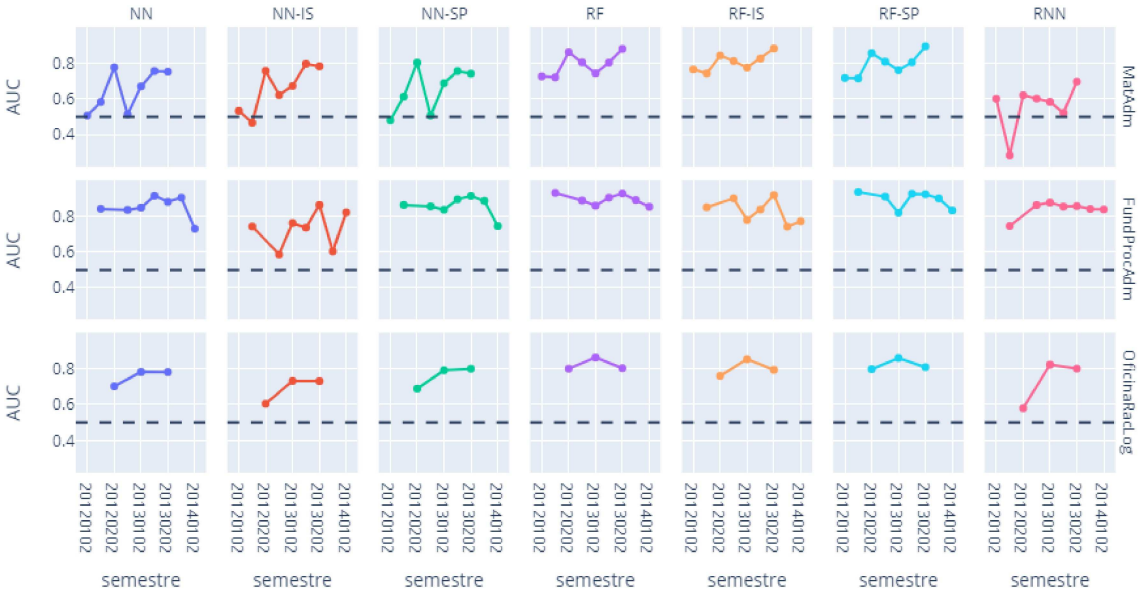


Figura 22. Evolução da área sobre a curva ROC para todas as disciplinas
Fonte: Elaborado pelo autor

5.3. Avaliação

A etapa de avaliação, de acordo com [Ncr et al. 2004], difere do processo de avaliação dos modelos, realizada na etapa de modelagem. Isto porque durante o processo de modelagem, a avaliação consiste na verificação dos números obtidos pelos modelos a fim de identificar qual deles obteve a melhor performance de acordo com as métricas de avaliação previamente estabelecidas. O processo de avaliação está mais aderente a um nível mais alto, visando identificar se os modelos criados atendem aos requisitos de negócio. Esta avaliação se dá pelo cumprimento das seguintes etapas:

1. **Avaliação dos Resultados**: Verifica se os resultados trazidos pelos modelos criados foram capazes de atender aos requisitos de negócio levantados no início de todo o processo, bem como identificar fatores que classifiquem os modelos como ineficientes, caso estes requisitos não sejam atendidos.
2. **Revisão do Processo**: Lista todas as atividades previstas em todo o processo de mineração de dados para conferir se todas as etapas foram cumpridas. Quando os modelos gerados são capazes de atender aos requisitos de negócio, presumidamente todas as etapas foram cumpridas.
3. **Próximos Passos**: Com base nos resultados obtidos, é decidido se o projeto pode ser finalizado e os modelos criados podem ser devidamente implantados em ambientes de produção ou se novas iterações do processo devem ser conduzidas com base nas descobertas feitas durante o processo atual.

Após listadas todas as métricas obtidas de todos os modelos avaliados, são escolhidos os modelos de melhor desempenho para cada disciplina, ou se possível, um modelo que tenha um desempenho destacado para todas as disciplinas. Dentre todos os modelos testados, dois serão destacados nesta seção, o modelo RF[2] e o modelo RNN. As Tabelas

[2]Modelo que utilizou tanto os dados dos padrões sequenciais quanto os dados das ações isoladas

23 e 24 mostram, respectivamente, todas as métricas dos dois modelos destacados para todas as disciplinas testadas.

Tabela 23. Resumo de métricas do modelo *Random Forest*

Métrica	FundProcAdm	MatAdm	OficinaRacLog
Acurácia	80,88%	85,00%	77,77%
Precisão	83,33%	89,47%	78,14%
Recall	91,83%	70,83%	88,72%
F1-Score	87,37%	79,06%	83,09%
Especificidade	52,63%	94,44%	60,24%
Área ROC	85,17%	87,84%	80,16%
Área PR	94,31%	82,43%	82,09%

Tabela 24. Resumo de métricas do modelo de redes neurais recorrentes

Métrica	FundProcAdm	MatAdm	OficinaRacLog
Acurácia	73,52%	71,66%	73,61%
Precisão	89,74%	100,00%	79,23%
Recall	71,42%	29,16%	77,44%
F1-Score	79,54%	45,16%	78,32%
Especificidade	68,42%	02,77%	63,85%
Área ROC	83,67%	69,61%	79,94%
Área PR	91,30%	73,28%	84,10%

Separar dois modelos ajuda a adequar um deles para cada disciplina, dependendo da métrica utilizada como critério-chave. Recapitulando as métricas mais importantes definidas anteriormente, na etapa de Modelagem, o modelo ideal deveria ter uma especificidade alta, além da possibilidade de ter uma precisão alta e um *Recall* baixo, se necessário para uma especificidade alta. Portanto, os modelos ideais por disciplina seriam os seguintes:

- Para Fundamentos do Processo Administrativo, o modelo de redes neurais recorrentes é o ideal por apresentar a maior especificidade e precisão. Apesar do *Recall* ser menor que o do modelo *Random Forest*, uma redução nesta métrica é aceitável para atender os objetivos do modelo.
- Para Matemática para Administração, o modelo *Random Forest* é o ideal pela maior especificidade apresentada. Apesar do modelo de redes neurais recorrentes ter uma precisão de 100% para esta disciplina, sua especificidade é muito baixa para ser considerada para esta disciplina.
- Para Oficina do Raciocínio Lógico, os dois modelos apresentaram desempenho semelhante, porém, o modelo de redes neurais recorrentes apresentou maior especificidade e precisão, apesar da diferença destas métricas para o modelo *Random Forest* ser menor que 4%. O *Recall* do modelo de redes neurais recorrentes é menor do que o do modelo *Random Forest*, mas segue aceitável para os critérios de aceitação do modelo escolhido.

6. Conclusões

É de interesse das instituições de ensino evitar que seus alunos não concluam suas disciplinas como deveriam, seja por mal desempenho ou por evasão. Isso traz prejuízo tanto para a instituição como para o aluno, e o prejuízo de ambas as partes acaba sendo sentido também por toda a sociedade, que lida hoje com um índice crescente na evasão do ensino superior.

Cada disciplina foi capaz de prover diferentes *insights* em relação aos resultados e às técnicas utilizadas para a transformação dos dados em relação à busca dos padrões sequenciais, como os seguintes:

- O modelo *Random Forest* foi o escolhido para Matemática para Administração, mostrando que quando já transformados os dados para uma estrutura tabular, como já muito utilizada em modelos de classificação, um modelo de redes neurais não irá necessariamente proporcionar uma grande diferença no desempenho para qualquer dado. O uso de padrões sequenciais em conjunto com os dados de ações isoladas melhorou o desempenho deste modelo do que utilizando cada tipo de dado separadamente.
- O modelo de redes neurais recorrentes foi o escolhido para Fundamentos do Processo Administrativo e Oficina do Raciocínio Lógico, apesar de não ser utilizada nenhuma etapa de transformação de dados para obtenção dos padrões sequenciais.

Do ponto de vista técnico, a etapa de preparação de dados mostrou que pode-se obter uma economia significativa de tempo ao utilizar DataFrames no formato Parquet, com o tempo de carga dos arquivos pré-transformados reduzido em mais de 99%.

Os resultados obtidos neste trabalho mostram que tanto o uso de padrões sequenciais quanto o uso de redes neurais podem contribuir positivamente para a criação de modelos capazes de identificar alunos que tendem a não concluir suas disciplinas com sucesso.

Além disso, mostraram que a acurácia nem sempre é a métrica ideal para decidir se um modelo é bom ou não, apesar de ser a métrica mais comum (e em muitos exemplos, a única) utilizada na avaliação de desempenho de modelos preditivos. Considerando os modelos estudados neste trabalho, a especificidade, precisão e *Recall* foram as métricas cruciais na escolha do melhor modelo, mostrando a importância de escolher a métrica que classifica um modelo preditivo com base no contexto que estes dados possuem.

No futuro, com base nos resultados dos modelos desenvolvidos neste trabalho, esta pesquisa pode ser estendida através das seguintes formas:

- Criar modelos utilizando dados de disciplinas diferentes das avaliadas neste trabalho, a fim de verificar se dados diferentes conseguem produzir métricas satisfatórias como as desenvolvidas neste trabalho.
- Havendo uma base de dados com muitas disciplinas e muitos semestres, avaliar a criação de um modelo que consegue prever o desempenho de alunos para qualquer disciplina.

Referências

Ayres, J., Flannick, J., Gehrke, J., e Yiu, T. (2002). Sequential pattern mining using a bitmap representation. In *Proceedings of the Eighth ACM SIGKDD International*

Conference on Knowledge Discovery and Data Mining, KDD '02, pages 429–435, New York, NY, USA. ACM.

Baker, R., Isotani, S., e Carvalho, A. (2011). Mineração de Dados Educacionais: Oportunidades para o Brasil. *Revista Brasileira de Informática na Educação*, 19(02):3–13.

Bandeira, M. (2016). Comportamento de estudantes em ambientes virtuais: uma análise de padrões sequenciais.

Castro R., L. F., Espitia P., E., e Montilla, A. F. (2018). Applying CRISP-DM in a KDD Process for the Analysis of Student Attrition. In Serrano C., J. E. e Martínez-Santos, J. C., editors, *Advances in Computing*, pages 386–401, Cham. Springer International Publishing.

Cechinel, C. e Camargo, S. d. S. (2019). *Metodologia de Pesquisa em Informática na Educação: Abordagem Quantitativa de Pesquisa*. SBC.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., e Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation.

Colpani, R. (2018). Educação a Distância: identificação dos fatores que contribuíram para a evasão dos alunos no curso de Gestão Empresarial da Faculdade de Tecnologia de Mococa. *Ead Em Foco*, 8(1):1–13.

de Oliveira, P. R., Oesterreich, S. A., e de Almeida, V. L. (2017). Evasão na pós-graduação a distância: evidências de um estudo no interior do Brasil. *Educação e Pesquisa*, 44(0):1–20.

Fournier-Viger, P., Lin, J. C.-W., Gomariz, A., Gueniche, T., Soltani, A., Deng, Z., e Lam, H. T. (2016). The SPMF Open-Source Data Mining Library Version 2. In Berendt, B., Bringmann, B., Fromont, É., Garriga, G., Miettinen, P., Tatti, N., e Tresp, V., editors, *Machine Learning and Knowledge Discovery in Databases*, pages 36–40, Cham. Springer International Publishing.

Google (2020). Keras — tensorflow core. `https://www.tensorflow.org/guide/keras`. Acessado em: 07/11/2020.

Grus, J. (2016). *Data Science do Zero*. Alta Books.

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, Inc, Sebastopol, CA.

Jaber, M., Wood, P. T., Papapetrou, P., e Gonzalez-Marcos, A. (2016). A Multi-Granularity Pattern-Based Sequence Classification Framework for Educational Data.

Lakatos, E. e de Andrade Marconi, M. (2003). *Fundamentos de metodologia científica*. Atlas.

Makhabel, B. (2015). *Learning Data Mining with R*. Community Experience Distilled. Packt Publishing.

Mansingh, G., Osei-Bryson, K. M., Rao, L., e McNaughton, M. (2017). Data preparation: Art or science? *Proceedings of the 2016 International Conference on Data Science and Engineering, ICDSE 2016*, pages 1–6.

Mattox, J. R. (2016). *Learning Analytics : Measurement Innovations to Support Employee Development.*, volume 1st edition. Kogan Page.

Ncr, P. C., Spss, J. C., Ncr, R. K., Spss, T. K., Daimlerchrysler, T. R., Spss, C. S., e Daimlerchrysler, R. W. (2004). Crisp-Dm 1.0. pages 1–76.

Nicoletti, M. C., Marques, M., e Guimaraes, M. P. (2018). A data mining approach for forecasting students' performance.

Pimentel, M., Filippo, D., e Santoro, F. M. (2019). *Metodologia de Pesquisa Científica em Informática na Educação: Concepção de Pesquisa.* SBC.

Provost, F. e Fawcett, T. (2016). *Data science para negócios.* Alta Books.

Rojanavasu, P. (2019). Educational Data Analytics using Association Rule Mining and Classification.

Saito, T. e Rehmsmeier, M. (2015). The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, 10(3):1–21.

SILVA, L. A. d. S., PERES, S. M. P., e BOSCARIOLI, C. (2016). Introdução à Mineração de Dados 1ED SILVA, Leandro Augusto da Silva,; with PERES, Sarajane Marques Peres,, BOSCARIOLI, Clodis,.

Silva Filho, R. L. L. e., Motejunas, P. R., Hipólito, O., e Lobo, M. B. d. C. M. (2007). A evasão no ensino superior brasileiro / Higher education institutions' evasion. *Cadernos de Pesquisa*, 37(132):641–659.

Squire, M. (2016). *Mastering Data Mining with Python – Find Patterns Hidden in Your Data.* Community Experience Distilled. Packt Publishing.

Srikant, R. e Agrawal, R. (1996). Mining sequential patterns: Generalizations and performance improvements. In Apers, P., Bouzeghoub, M., e Gardarin, G., editors, *Advances in Database Technology — EDBT '96*, pages 1–17, Berlin, Heidelberg. Springer Berlin Heidelberg.

Tang, S., Peterson, J. C., e Pardos, Z. A. (2016). Deep neural networks and how they apply to sequential education data. *L@S 2016 - Proceedings of the 3rd 2016 ACM Conference on Learning at Scale*, pages 321–324.

Tripathi, A. (2017). *Practical Machine Learning Cookbook.*

Zocca, V., Spacagna, G., Slater, D., e Roelants, P. (2017). *Python Deep Learning.* Packt Publishing.