Universidade Federal do Ceará

Campus Quixadá

Curso de Sistemas de Informação

Natália Lionel Moreira

DETECÇÃO DE ATRIBUTOS QUE MELHOR CARACTERIZAM PERFIS DE INSCRITOS DO ENEM UTILIZANDO REDUÇÃO DE DIMENSIONALIDADE

Quixadá, Ceará

2016

Natália Lionel Moreira

DETECÇÃO DE ATRIBUTOS QUE MELHOR CARACTERIZAM PERFIS DE INSCRITOS DO ENEM UTILIZANDO REDUÇÃO DE DIMENSIONALIDADE

Trabalho de Conclusão de Curso submetido à Coordenação do Curso de Sistemas de Informação do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial para obtenção do Título de Bacharel em Sistemas de Informação.

Orientador: Prof[a] MSc. Ticiana Linhares Coelho da Silva

Quixadá, Ceará

2016

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

M838d Moreira, Natália Lionel.
Detecção de atributos que melhor caracterizam perfis de inscritos do ENEM utilizando Redução de Dimensionalidade / Natália Lionel Moreira. – 2016.
65 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Sistemas de Informação, Quixadá, 2016.
Orientação: Profa. Ma. Ticiana Linhares Coelho da Silva.

1. Mineração de dados (Computação). 2. Exame Nacional do Ensino Médio. 3. Análise por agrupamento. I. Título.

CDD 005

Natália Lionel Moreira

DETECÇÃO DE ATRIBUTOS QUE MELHOR CARACTERIZAM PERFIS DE INSCRITOS DO ENEM UTILIZANDO REDUÇÃO DE DIMENSIONALIDADE

Trabalho de Conclusão de Curso submetido à Coordenação do Curso de Sistemas de Informação do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial para obtenção do Título de Bacharel em Sistemas de Informação.

Área de concentração: Computação

Aprovada em: 04 / julho / 2016

BANCA EXAMINADORA

Profª MSc. Ticiana Linhares Coelho da Silva
(Orientadora)
Universidade Federal do Ceará (UFC)

Profº MSc. Regis Pires Magalhães (Membro)
Universidade Federal do Ceará (UFC)

Profº Dr. Flávio R. C. Sousa (Membro)
Universidade Federal do Ceará (UFC)

A minha família, principalmente, a meus pais, meus avós e minha irmã.

Agradecimentos

Agradeço às pessoas que me ajudaram a vencer mais esta etapa da vida. Á Universidade Federal do Ceará-UFC - Campus Quixadá, pela oportunidade de estudos e utilização de suas instalações.

À minha querida orientadora Ticiana Linhares da Silva, pelo conhecimento compartilhado, pela orientação e por mesmo distante geograficamente ter sempre se feito presente, mostrando-se disponível quando alguma questão surgia.

Aos meus pais Aniza e Nicolas, em especial a minha mãe que mesmo distante sempre esteve presente em minha vida, me apoiando e mostrando que obstáculos surgem para mostrar que sou mais forte do que eles, não devendo desistir jamais.

À minha irmã, Nicoly, e meu sobrinho, Ariel, por todo carinho, pela fonte de energia que me proporcionam sempre que chego em casa depois de meses longe.

Ao meu querido Márcio, por toda a paciência e carinho dedicado a mim, durante o processo de construção deste trabalho, sempre me apoiando.

Às minhas "praias", Talhita e Hinessa, por suportarem minha montanha russa de sentimentos, entendendo minha ausência em muitos momentos e, principalmente, por terem me mostrado uma amizade verdadeira durante esta trajetória.

À todos os professores que de alguma forma contribuíram no meu crescimento profissional e pessoal, inclusive os do Ensino Médio que até hoje são fontes de inspiração.

Aos meus amigos Sergio Filho, Lucas Araújo, Adail Carvalho, Larice Lima, Jonas Sousa, João Marcos, Marcelo Gonçalves, Diogo Nazareno, Wendel Maciel, Daniel Farias, George Júnior, Ederson Abreu, Alison Santos e José Gerlan, que estiveram presente nesta trajetória, ajudando a enfrentar as dificuldades encontradas quando se estar longe da família e por proporcionarem momento inesquecíveis.

"Não desista nas primeiras tentativas, a persistência é amiga da conquista. Se você quer chegar a onde a maioria não chega, faça o que a maioria não faz."

'Bill Gates'

Resumo

O Exame Nacional do Ensino Médio - ENEM desde 2010 tem sido utilizado como o principal meio de ingressar em instituições de ensino superior, tornando-se, então, um vestibular nacional. Este fato causou um aumento da quantidade de interessados em realizar o exame, gerando um grande volume de dados sendo a base do ENEM uma base de alta dimensionalidade. A mineração de dados permite extrair conhecimentos a partir de um grande volume de dados, dentre as técnicas de mineração de dados está a clusterização. Utilizando esta técnica é possível identificar perfis de inscritos do ENEM com base em fatores socioeconômicos. É importante utilizar método de seleção de atributos em bases de alta dimensionalidade para identificar os atributos mais caracterizantes de uma base, a escolha aleatória de atributos pode causar resultado incorreto ou inútil. Tendo em vista a existência de diferentes abordagens de seleção de atributos, este trabalho visa comparar as abordagens *filter* e *wrapper*, utilizando diferentes algoritmos de busca, para identificar a que apresenta melhores resultados. Neste estudo foram utilizados dados do ENEM de 2010 para identificar perfis de inscritos, via clusterização, buscando relacionar a média obtida na prova com os dados socioeconômicos informados pelo inscrito no ato da inscrição. Além disso é realizada uma análise entre os *clusters* obtidos neste estudo e no trabalho de (CAMINHA; MOREIRA; SILVA, 2015), que faz um estudo semelhante a este utilizando, também, base do ENEM. Com este estudo foi possível concluir que os fatores socioeconômicos não possuem grande impacto na nota final da prova, não podendo, portanto, influência-la nos dados analisados.

Palavras-chaves: Mineração de Dados. ENEM. Clusterização. Feature Selection. Seleção de Atributos. Filter. Wrapper.

Abstract

Since 2010, the Exame Nacional do Ensino Medio - ENEM has been used as the only exam to apply for a position in some Brazilian. Because of this, the number of candidates interested in the exam has increased and a large volume of data has being collected from them. This has made ENEM a high dimensional dataset. Data mining can extract knownledge from a large volume of data. One of the most important data mining technique is clustering. By using this technique it is possible to identify profiles of members based on socioeconomic questions usually answered by ENEM candidates. It is important to use feature selection method in high dimensional databases to identify the most characterizing attributes of a dataset and improve the efficiency, since high dimensional data is more costly to process. Futhermore, the random choice of attributes can produce incorrect or useless results. By using different approaches to select attributes, this study aims to compare the filter and wrapper approaches with different searching algorithms and then identify the one of the best results. In this work, the ENEM data from 2010 was used to identify profiles via clustering algorithm, trying to relate the average grade obtained in the exam with socioeconomic data provided by each candidate upon his/her registration. In addition, an analysis was made between the clusters obtained in this study and the work of (CAMINHA; MOREIRA; SILVA, 2015) which makes a similar study using the same data. This study was concluded that socioeconomic factors do not have major impact on the final grade and can not therefore influence it (at least for 2010 data).

Key-words: Data Mining. ENEM. Clustering. Feature Selection. Filter. Wrapper. Seleção de Atributos.

Lista de ilustrações

Lista de tabelas

Sumário

1 Introdução

A educação brasileira, ao longo dos anos, passou por diversas reformas objetivando melhorar a qualidade de ensino. Para a implantação de uma nova reforma, o Ministério da Educação e Cultura (MEC) elaborou o Exame Nacional de Ensino Médio (ENEM), (CASTRO; TIEZZI, 2004) afirmam que a elaboração do exame tem o intuito de fazer com que as escolas passassem a ensinar de uma forma que fizesse o aluno desenvolver um maior interesse por aprender. Inicialmente, o exame tinha, como principal intuito, verificar o nível de conhecimento daqueles que estão saindo do ensino médio e era utilizado como critério de seleção para concorrer à bolsas do Programa Universidade para Todos (ProUni).

A partir de 2010, o ENEM passou a ser utilizado como principal meio de ingressar em uma instituição de ensino superior (IES), tornando-se, então, um vestibular nacional. Essa mudança fez com que aumentasse, gradativamente, a quantidade de inscritos a cada ano. No momento da inscrição, o candidato deve preencher formulários fornecendo algumas informações relacionadas a questões socioeconômicas, gerando, assim, um grande volume de dados. Para realizar a seleção de candidatos às vagas de universidades foi adotado o Sistema de Seleção Unificada (SiSU), um sistema no qual o ministério da educação permite que instituições públicas de ensino superior ofertem vagas e selecione mediante uma nota de corte os candidatos.

A mineração de dados consiste em extrair conhecimentos, de forma automática, a partir de um grande volume dados como, por exemplo, identificar padrões e prever resultados futuros. Dentre as técnicas que podem ser utilizadas para a mineração desses dados, está a clusterização, que consiste no agrupamento de um aglomerado de dados multidimensionais num conjunto de classes, denominadas *clusters*, com base no grau de similaridade das observações (JAIN; MURTY; FLYNN, 1999). Devido à alta dimensionalidade dos dados e ao fato de que alguns atributos podem não apresentar valores representativos de acordo com alguma medida estatística (variância, por exemplo) para participarem do processo de clusterização, faz-se necessário a utilização de métodos de recuperação de informação, que permitem determinar os atributos que melhor caracterizam um conjunto de dados.

Em (CAMINHA; MOREIRA; SILVA, 2015), foi realizado um trabalho utilizando as bases de dados do ENEM de 2009, 2010 e 2011. O objetivo era comparar o desempenho dos inscritos no exame nos anos citados anteriormente, antes e depois da adesão do SiSU utilizando técnica de clusterização de dados, além de descobrir quais fatores influenciam no desempenho dos inscritos no exame via clusterização de dados, baseado em características socioeconômicas e na sua média final. Nesse trabalho, os atributos utilizados para descoberta dos perfis foram selecionados de acordo com o que as autoras julgaram que teriam impacto no desempenho do inscrito, não sendo aplicado qualquer técnica de seleção de atributos. A metodologia utilizada de descartar alguns atributos pode acarretar em perda de informações relevantes no resultado

final da análise.

Sabendo que alguns atributos podem ser mais relevantes do que outros no processo de clusterização, este trabalho tem como objetivo aplicar métricas de seleção de atributos para identificar quais são os atributos mais relevantes da base de dados do ENEM de 2010, ou seja, os atributos que sejam capazes de definir melhor os perfis de inscritos desta edição do exame, via clusterização de dados. Além disto, este estudo pretende analisar as métricas existentes, aplicando-as na base em questão com o intuito de identificar a que apresenta melhores resultados para diferentes algoritmos de busca. Também pretende-se comparar os *cluster* obtido com e sem o uso de métricas. Este estudo torna-se relevante, pois com base nele será possível ter uma visão de como foi o desempenho dos inscritos no ano de 2010, sendo possível relacionar as características socioeconômicas dos candidatos com o seu desempenho.

O restante deste trabalho está organizado da seguinte forma. O Capítulo 2 contém a fundamentação teórica, onde serão apresentados os principais conceitos utilizados neste trabalho. No Capítulo 3 serão apresentados os trabalhos relacionados que serviram de embasamento para a construção deste. Logo após, no Capítulo 5 são abordados os procedimentos metodológicos, em seguida no Capítulo 6 serão apresentados os experimentos e seus resultados e por fim, o Capítulo 7 mostrará a conclusão deste estudo e os trabalhos futuros.

2 Fundamentação Teórica

Para uma melhor compreensão deste trabalho, nesta seção serão abordados os principais conceitos utilizados em sua construção.

2.1 Métodos de seleção de atributos

A etapa de seleção de atributos consiste em identificar, dentro de um conjunto de dados, os atributos mais relevantes a serem utilizados no estudo. O processo de selecionar atributos é importante para reduzir a dimensionalidade dos dados. A escolha aleatória de quais atributos utilizar durante o processo de análise pode afetar no conhecimento a ser extraído dos dados, podendo acarretar a descoberta de conhecimento impreciso ou inútil.

Segundo (FREITAS, 2003), a motivação para este tipo de pré-processamento deve-se ao fato de que atributos irrelevantes podem de alguma forma "confundir"o algoritmo de mineração de dados, levando à obtenção de resultados imprecisos ou inúteis. Além disso, trabalhar com uma alta quantidade de atributos significa um maior custo para obter o resultado, através do algoritmo de mineração. Isto aumenta o custo de memória e tempo de execução. (FREITAS, 2003) diz que os métodos de seleção de atributos podem ser divididos em três abordagens, *embedded*, *filter* e *wrapper*. Neste trabalho foram utilizadas apenas a duas últimas abordagens citadas.

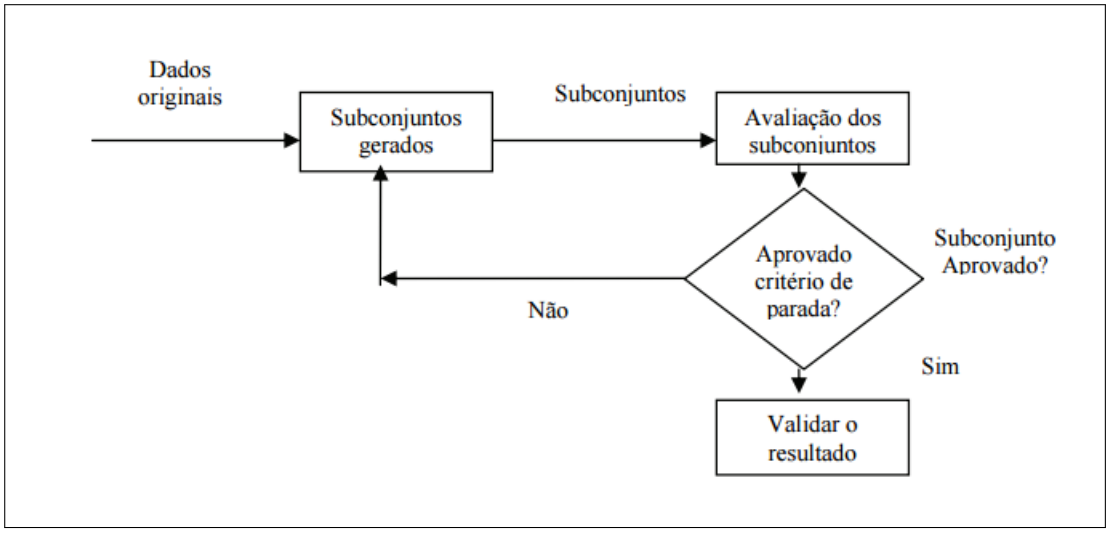
A abordagem *filter* executa o método de seleção de atributos antes da aplicação do algoritmo de mineração de dados, ou seja, de forma independente ao algoritmo de mineração de dados. Ao final da execução, é gerado um *ranking* mostrando uma classificação dos atributos de acordo com alguma métrica (definida de entrada).

A abordagem *wrapper* executa o método de seleção de atributo utilizando o algoritmo de mineração de dados para verificar o “quão bom” é um determinado subconjunto de atributos. O resultado obtido é guardado para que se possa verificar qual o melhor subconjunto. Nesta abordagem são gerados vários subconjuntos até que uma condição de parada seja atendida. A principal característica da abordagem *wrapper* é que a qualidade de um subconjunto de atributos é medida de acordo com a acurácia do algoritmo de mineração de dados aplicado sob os dados utilizando apenas o subconjunto de atributos.

Existem várias etapas no processo de seleção de atributos, conforme pode ser observado na Figura 1. Inicialmente, um conjunto de dados contendo todos os atributos é recebido como entrada. Em seguida, utilizando este conjunto de dados é realizada uma busca por subconjuntos de atributos. Isto é feito usando um algoritmo de busca (mais adiante serão apresentados alguns algoritmos de busca a serem aplicados neste trabalho). Em sequência, uma avaliação é feita em

cada subconjunto encontrado utilizando uma medida que avalia cada subconjunto gerado, para verificar qual o mais adequado. Após isso, acontece uma verificação para saber se o processo deve ser interrompido, caso contrário todos os passos anteriores são repetidos. Segundo (DASH; LIU, 1997), a interrupção pode ocorrer por dois motivos: quando a função de avaliação verifica que os novos subconjuntos gerados não obtêm melhor classificação ou quando o processo de geração encerra após identificar um determinado número de atributos. O segundo caso ocorre quando é informada uma quantidade limite de atributos. Dependendo da forma como a busca é feita, o subconjunto aumenta ou diminui. Quando realizada a partir de um conjunto vazio, aumenta e a partir de um conjunto cheio, diminui. Ao final da seleção de atributos, é realizada a validação do subconjunto, esta tarefa não faz parte do processo de seleção, mas é crucial para garantir que o subconjunto escolhido é ótimo ou próximo disso.

Figura 1: Passos da Seleção de Atributos.



Fonte: Adaptado de (LIU; MOTODA, 1998).

Na literatura existem diversos algoritmos de buscas de subconjuntos. (WITTEN; FRANK, 2011) descreveram alguns que são implementados no software livre Waikato Environment of Knowledge Analysis (WEKA)[1]. A seguir encontra-se a descrição de alguns destes algoritmos.

- *BestFirst:* Este algoritmo realiza buscas gulosas, permitindo realizar buscas tanto *forward* quanto *backward*, ou seja, a busca pode iniciar a partir de um conjunto vazio de atributos para frente ou para trás de um conjunto completo. Além dessas duas formas de busca, ele pode começar a busca em um ponto intermediário determinado por uma lista de índices de atributos. Ele possui facilidade para deslocar-se até um ponto anterior e voltar ao lugar de onde saiu. Isto permite que sejam consideradas todas as adições e exclusões de atributos.

- *ExhaustiveSearch:* Este algoritmo realiza uma busca exaustiva, ou seja, realiza a busca em todo o espaço à procura de subconjuntos de atributos, a partir do conjunto vazio, re-

[1] Disponível em: http://www.cs.waikato.ac.nz/ml/weka/downloading.html

tornando o melhor subconjunto encontrado. Caso seja fornecido um conjunto inicial, o algoritmo leva em consideração que o conjunto fornecido é o ponto de partida, sendo a busca realizada a partir deste ponto. Ao final da execução é retornado o menor subconjunto que possui avaliação melhor ou igual a do conjunto inicial.

- *GeneticSearch*: Este algoritmo é realizado através de um algoritmo genético simples descrito por (GOLDBERG, 1989). Algoritmos genéticos são algoritmos que usam técnicas baseadas na biologia evolutiva para resolver problemas de otimizações e buscas. Para a realização da busca o *GeneticSearch* leva em consideração informações passadas, tendo como parâmetros tamanho da população, número de gerações e probabilidades de *crossover* e mutação. Este algoritmo permite informar uma lista índices de atributos que são tomados como ponto de partida para tornarem-se membro de uma população inicial.

- *GreedStepwise*: Este algoritmo realiza a busca percorrendo todo o espaço de subconjuntos de atributos. Possui funcionamento semelhante ao *BestFirst*, sendo possível realizar buscas tanto *forward* como *backward* diferenciando-se por não poder realizar deslocamento até um ponto anterior e voltar à posição de onde saiu. Esta característica não o impede de sua execução, enquanto não adicionar ou excluir o melhor atributo restante, diminuindo a métrica de avaliação. Este algoritmo permite ainda determinar o número de atributos a serem mantidos ou especificar um limite no qual os atributos devem ser descartados abaixo dele.

- *RandomSearch*: Este algoritmo realiza um busca aleatória procurando o subconjunto de atributos. Caso seja dado como entrada um subconjunto, são realizadas buscas à procura de subconjuntos melhores ou iguais ao subconjunto inicial e que possui uma quantidade menor ou igual de atributos. Caso contrário, a busca inicia a partir de um ponto escolhido aleatoriamente, sendo retornado como resultado o melhor subconjunto encontrado.

- *RankerSearch*: Este algoritmo ordena os atributos fazendo uso de um avaliador de atributo individual, em seguida realiza uma classificação dos subconjuntos candidatos a serem os melhores utilizando um avaliador de subconjunto de atributos. O avaliador é determinado como uma propriedade do algoritmo. Ele inicia classificando um atributo e pega o próximo melhor, os atributos considerados melhores são colocados em um mesmo subconjunto, sendo este retornado como o melhor.

Dentre os algoritmos citados foram escolhidos três para este estudo, sendo eles: *BestFirst*, *GeneticSearch* e *RankerSearch*. A escolha do primeiro deve-se ao fato de ter muitos outros algoritmos que são baseados nele, sendo, portanto, um algoritmo tradicional. O *GeneticSearch* é um algoritmo bastante utilizados na literatura, sendo este o motivo de sua escolha. Já o *RankerSearch* foi escolhido por avaliar cada atributo, sendo capaz de formar um subconjunto apenas com os atributos classificados como melhores de acordo com o *ranking* gerado por ele, onde neste *ranking* encontram-se os melhores atributos.

2.2 Clusterização

O trabalho (JAIN, 2010) define clusterização como uma técnica que, dentro de um conjunto de elementos, encontra grupos organizando elementos de acordo com suas características semelhantes. Dessa forma elementos pertencentes ao mesmo grupo possuem características mais semelhantes entre si do que elementos pertencentes a grupos diferentes. Os grupos formados pela clusterização são chamados de *clusters*. Dentre os diversos algoritmos existentes, foi escolhido o k-means neste trabalho.

O algoritmo k-means é definido por (TAN; STEINBACH; KUMAR, 2009) como uma técnica particional de agrupamento baseada em protótipos que tenta encontrar um número de grupos (K) especificado pelo usuário, que são representados pelos seus centróides. Um centróide representa a média de pontos de determinado grupo de elementos que compartilham características semelhantes entre si.

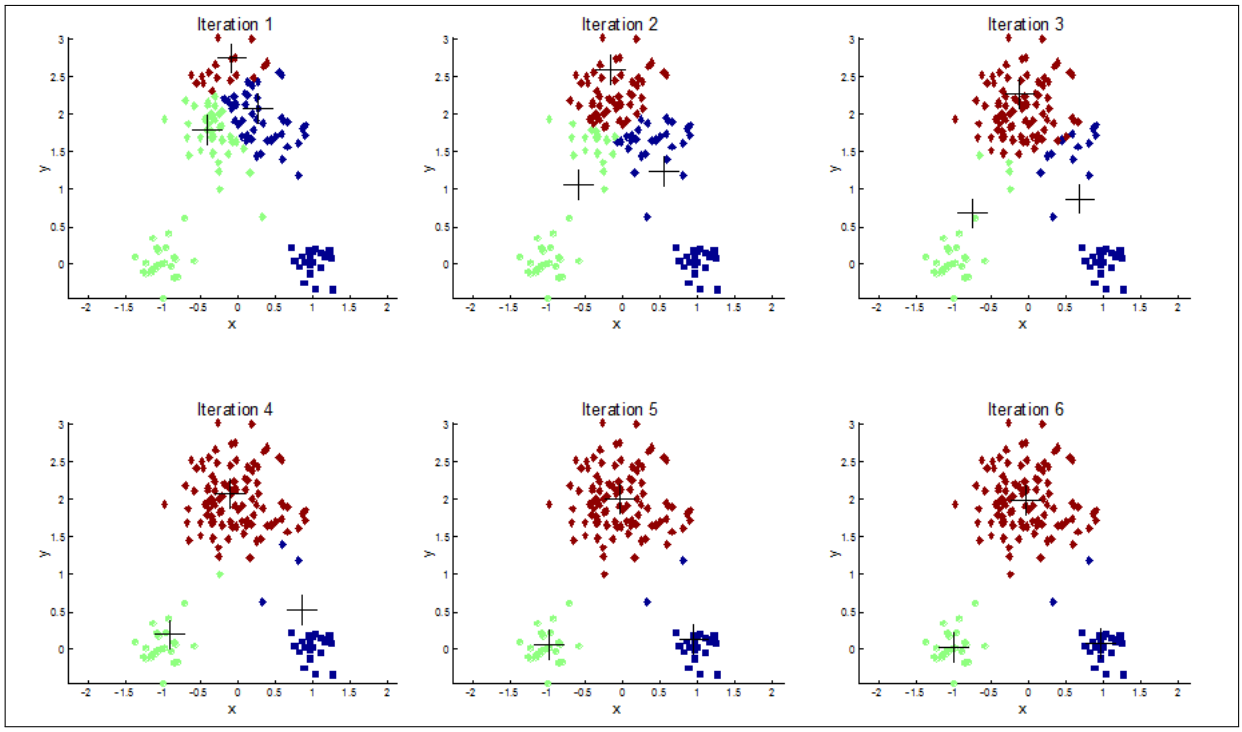
O k-means funciona da seguinte forma. Considere um conjunto de dados. Inicialmente, o usuário informa a quantidade K de grupos desejada, e então, o algoritmo divide esse conjunto de dados em K grupos, conforme pode ser observado na Figura 2, e irá determinar o grupo (*cluster*) ao qual cada elemento deve pertencer. Para gerar os clusters e determinar seus elementos, o algoritmo realiza comparações entre cada elemento e cada centróide por meio de uma função de distância ou de (dis)similaridade. O elemento é designado ao cluster cujo centróide é mais similar (ou de menor distância). A função de distância é calculada utilizando medidas de dissimilaridade, que determinam o quão diferente são dois elementos em um grupo. Exemplos de distâncias que podem ser utilizadas são a Euclidiana, apresentada na Equação 2.1, frequentemente utilizada quando trata-se de pontos de dados em um espaço vetorial real finito, e a distância cosseno, que é mais adequada quando trata-se de documentos. Em seguida, o algoritmo recalcula os centróides para cada um dos grupos baseado nos seus elementos. Esse procedimento se repete várias vezes até que alguma condição de parada seja atingida. Exemplos de condições de paradas podem ser: (i) número máximo de iterações ou (ii) até que a soma do erro quadrático total (entre cada centróide e os elementos do cluster) se estabilize entre uma iteração e outra.

Caso em duas execuções do k-means sejam gerados diferente grupos, será escolhido aquele que apresentar menor erro quadrático, pois isto significa que os centróides deste agrupamento possuem melhor representação do seu grupo. Observe a Equação 2.1, onde p[$p_1, ..., p_n$] e q[$q_1, ...,_n$] representam grupos de elementos diferentes, esta distância calcula a diferença ao quadrado entre os elementos dos dois grupos. Quanto mais próximo de zero maior a semelhança, da mesma forma quanto mais próximo de 1 (um) menor a semelhança.

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + ... + (p_n - q_n)^2} \tag{2.1}$$

A distância cosseno, apresentada na Equação 2.2, é a medida do ângulo entre dois veto-

Figura 2: Exemplo de funciomento do k-means.



Fonte: https://apandre.wordpress.com/visible-data/cluster-analysis/

res **x** e **y** de dimensão n, ou seja, x= p[$p_1, ..., p_n$] e y = q[$q_1, ...,_n$] anteriormente mencionados. Esta distância avalia o grau de similaridade entre eles, dessa forma se a semelhança de cosseno for 1, o ângulo entre os vetores é 0, o que quer dizer que x e y são o mesmo. Se semelhança de cosseno for 0 então o ângulo entre os vetores é 90º, o que quer dizer que x e y não compartilham características em comum.

$$d(x, y) = \frac{x * y}{\|x\|\|y\|} \tag{2.2}$$

O k-means foi escolhido por ser um algoritmo simples e aceitar determinar a quantidade de clusters que o usuário deseja gerar diferente de outros algoritmos como o X-means, que não permite o usuário escolher. A utilização do X-means foi descartada, pois observou-se em estudos anteriores e semelhantes a este que a quantidade de *clusters* gerados muitas vezes não era a considerada ideal. Como será explicado mais adiante, foi medida a soma de erros quadráticos quando para a variação do valor de K. Pelo método de *elbow*, também chamado de regra do cotovelo, é possível avaliar qual valor aproximado de K a ser dado de entrada ao algoritmo k-means, utilizando o conjunto de dados deste trabalho.

A ideia da regra do cotovelo é executar o k-means no conjunto de dados n vezes numa faixa de valores (por exemplo de 1 a 20), e para cada valor de n calcular a soma dos erros quadrados (SSE). Ao final da execução é gerado um gráfico mostrando erro versus o número de *clusters*, ou seja, a soma dos erros quadráticos de acordo com a quantidade de *clusters* gerada.

Se a linha do gráfico é semelhante ao desenho de um braço, o ponto mais baixo representa onde o erro começa a se estabilizar, ou seja, a diferença de erros entre determinado valor x de *clusters* e o seu sucessor $x + 1$ é abaixo de um *threshold*, sendo este o valor ideal.

3 Trabalhos Relacionados

A seguir serão brevemente apresentados alguns trabalhos que serviram de embasamento para a elaboração deste, esclarecendo a forma com que se relacionam.

Em (JORGE, 2010), é realizada uma comparação entre três técnicas de seleção de atributos aplicados em previsão de insolvência de empresas brasileiras não financeiras de capital aberto. Para isso três abordagens de seleção de atributos foram utilizadas, duas de aprendizado de máquina: Filtro e *Wrapper*, e uma de estatística multivalorada: Análise de componentes principais. Em seguida, foi realizada uma avaliação da seleção dos atributos, sendo empregados três algoritmos de classificação, Regressão Logística, Árvore de decisão e Máquina de Vetor Suporte.

Esse trabalho foca bastante a parte de seleção de atributos, mostrando a importância desta etapa no pré-processamento dos dados. Durante o estudo, o autor percebeu que em muitos trabalhos relacionadas a esse tema, as variáveis escolhidas eram as mesmas ou tinham alguma relação com variáveis já utilizadas em outras pesquisas de mesmo propósito. Ao final do estudo, o autor conclui que a abordagem mais eficiente para a base de dados utilizada, entre as adotadas, foi a *Wrapper*. Além de ter apresentado melhores classificações nos três algoritmos abordados.

Levando em consideração que existem diversos métodos de seleção de atributos, assim como o trabalho de (JORGE, 2010), este realiza um estudo sobre as técnicas *Wrapper* e Filtro buscando identificar uma que seja mais eficiente, realizando uma análise sobre os resultados obtidos. Além disso, este trabalho procura, também, destacar a etapa de seleção de atributos. Este trabalho difere-se de (JORGE, 2010) por utilizar dados relacionados a edição do ENEM de 2010. No trabalho referenciado são utilizados dados referentes as empresas classificadas no SERASA e na BOVESPA como “solventes” e “insolventes”, referentes ao período de 2005 a 2007. Os dados do ENEM serão utilizados para identificar perfis de inscritos, baseado em questões socioeconômicas, via clusterização de dados. Além disso, será realizada uma análise entre perfis gerados, tendo os atributos escolhidos a partir de métricas de seleção de atributos e perfis gerados sem o uso de técnicas.

Em (MOLINA; BELANCHE; NEBOT, 2002), é realizada uma revisão de vários algoritmos fundamentais para a escolha de atributos encontrados na literatura, avaliando o desempenho de cada um em determinado cenário. No trabalho de (MOLINA; BELANCHE; NEBOT, 2002) foi proposta uma maneira de avaliar os algoritmos de seleção de atributos, objetivando compreender o seu comportamento geral sobre particularidades de relevância, irrelevância, redundância e tamanho da amostra do conjunto de dados. Para isso foi obtida, a partir de um conjunto de dados gerados artificialmente, uma amostra de dados. Foi determinado um conjunto de soluções ótimas para a escolha dos atributos. Este conjunto é comparado com a saída dos algoritmos de

seleção de atributos. A comparação é feita utilizando uma escala de pontuação para identificar o grau de aproximação entre a solução obtida e a verdadeira solução. Ao final, os autores concluem que cada algoritmo possui comportamentos diferentes para diferentes tipos de dados, ou seja, o algoritmo deve ser escolhido de acordo com o tipo de dado.

Assim como o trabalho de (MOLINA; BELANCHE; NEBOT, 2002), este visa fazer uma comparação entre algoritmos de seleção de atributos buscando identificar o que apresente maior confiabilidade à base de dados utilizada. O trabalho proposto diferencia-se do citado por trabalhar com dados reais e realizar a comparação entre, apenas, dois algoritmos.

Em (CAMINHA; MOREIRA; SILVA, 2015), é realizada uma análise de perfis de inscritos do ENEM de 2009 a 2011 com o intuito de descobrir os principais fatores socioeconômicos que impactam no resultado obtido no exame, via clusterização de dados. Além de fazer uma comparação entre os resultados dos perfis antes e depois da adesão ao Sistema de Seleção Unificada (SiSU). Para gerar os perfis foram selecionados atributos referentes às respostas fornecidas por cada inscrito no ato da inscrição, sendo eles escolaridade dos pais, renda total da família, se concluiu o ensino fundamental em tempo correto, tipo de escola que cursou o ensino médio e se o inscrito trabalhou durante o ensino médio.

Assim como no estudo de (CAMINHA; MOREIRA; SILVA, 2015), este tem o intuito de analisar os perfis de inscritos de uma das bases utilizadas no trabalho citado, visando identificar os fatores que mais influenciam no rendimento do inscrito. Diferenciando-se pela forma como foram selecionados os atributos utilizados no estudo. No trabalho referenciado os atributos foram escolhidos conforme as autoras julgaram importantes. Neste serão aplicadas métricas capazes de determinar os atributos que, de fato, são importantes para caracterizar os perfis. Posteriormente, será feita uma comparação entre os *clusters* gerados neste estudo e no de (CAMINHA; MOREIRA; SILVA, 2015).

Tendo em vista que o prejuízo financeiro gerado por fraudes em comércio eletrônico é um tema motivador de diversas pesquisas, (LIMA; PEREIRA, 2015) realizaram uma revisão sistemática da literatura sobre os trabalhos de detecção de fraude em transações eletrônicas. Além disso, os autores avaliam a eficácia dos modelos de detecção de fraude em dados reais oriundo do sistema mais popular de pagamento eletrônico da América Latina, o PagSeguro. Os autores avaliaram os 30 trabalhos mais citados e os 20 mais relevantes em detecção de fraude, desde o ano de 2011. Inicialmente, observou-se que apenas 53% dos trabalhos descreviam a utilização de alguma técnica de *feature selection*, entretanto, sem tratamento para dados desbalanceados. Para avaliar como o desbalanceamento entre as classes afeta a seleção de atributos foi utilizada a estratégia de *undersampling*, antes da etapa de *feature selection* e construíram modelos de detecção de fraude composto por técnicas de *feature selection* (com ou sem uso de *undersampling*) e técnicas de classificação. Ao final do trabalho (LIMA; PEREIRA, 2015) concluem que o desbalanceamento entre classes reduz a eficácia das técnicas de seleção de atributos para detectar fraudes. É apresentado como uma possível solução a utilização de estratégia

de *undersampling* na etapa de *feature selection*, construindo modelos de detecção de fraude que melhoram em até 61% os ganhos financeiros da empresa.

Assim como no trabalho de (LIMA; PEREIRA, 2015), o estudo aqui proposto utiliza técnicas de *feature selection* para detectar os atributos capazes de representar melhor uma base de dados, visando mostrar o quão importante é a etapa de seleção de atributo. Além disso, assemelham-se por ambos realizar comparações entre estudo com e sem a utilização de técnicas de *feature selection*. Enquanto o trabalho citado utiliza dados oriundos de um sistema de pagamento eletrônico, o PagSeguro, o estudo proposto utiliza uma base de dados abertos referentes à edição do ENEM de 2010.

A seleção de subconjuntos de atributos é muito importante na área de Mineração de Dados e a alta dimensionalidade de dados pode tornar testes e treinamentos de classificação tarefas complicadas. Sabendo disto, (KAREGOWDA; MANJUNATH; JAYARAM, 2010) realizaram um estudo comparativo entre dois filtros de seleção de atributos *Gain Ratio* e *Correlation based Feature Selection*(CFS) para mostrar a importância da seleção de subconjuntos de atributos para a classificação de *Pima Indian Diabetic Database* (PIDD). Para determinar as divisões e para selecionar os atributos mais importantes foi utilizado a árvore C4.5 usando *Gain Ratio*, como método de busca foi utilizado o *GeneticSearch* e como mecanismo de avaliação de subconjuntos o CFS. Ao final da etapa de seleção de atributo cada subconjunto obtido passou por dois métodos de classificação supervisionada, *Back Propagation Neural Network* (BPN) e *Radial Basis Function Network* (RBF network) para verificar qual subconjunto apresenta melhor classificação. Ao final do estudo os autores concluem que os subconjuntos de atributos selecionados pelo CFS obteve melhores resultados tanto para BPN quanto para RBF *network* quando comparado ao subconjunto selecionado pelo *informatio gain*.

Assim como no trabalho de (KAREGOWDA; MANJUNATH; JAYARAM, 2010), este estudo utiliza dados abertos e realiza comparações entre métodos de seleção de atributos com intuito de verificar qual apresenta melhores resultados. Assemelha-se também por utilizar o CFS como avaliador de subconjuntos e árvore C4.5 para determinar as divisões. Diferenciando-se por comparar diversos algoritmos de busca.

Levando em consideração que algumas vezes métodos de seleção não eliminam de forma satisfatória os atributos irrelevantes, o trabalho de (RIBEIRO et al., 2010) propõe um sistema que utiliza ontologias para armazenar o conhecimento prévio sobre um domínio específico, possibilitando uma análise semântica antes não viável pelas metodologias convencionais. Uma ontologia foi elaborada utilizando informações armazenadas em diversos repositórios de ontologias disponíveis na web específica para o domínio médico e com possíveis especificações comuns nas principais áreas da medicina. Com este sistema o usuário poderá selecionar atributos através de categorias semânticas, reduzir a dimensionalidade dos dados e ainda visualizar redundâncias existentes entre atributos correlacionados semanticamente.

Este trabalho assemelha-se ao de (RIBEIRO et al., 2010) por utilizar técnicas de seleção

de atributos e também por realizar uma análise no atributos considerados relevantes. Diferente do trabalho de (RIBEIRO et al., 2010), este tem o intuito de identificar os atributos capazes de representar a Base de Dados do ENEM de 2010, já o trabalho citado utiliza dados referentes a áreas de medicina.

Tendo em vista a extração de conhecimentos de dados brutos tornou-se um diferencial para as organizações, (MENDES, 2011) realiza um estudo sobre as técnicas de Mineração de Dados. Isto é feito com o intuito de apresentar uma forma no qual os métodos de Mineração de Dados possam ser utilizados por instituições bancárias e de crédito a fim de melhorar a qualidade e a eficiência das decisões. (MENDES, 2011) mostra em seu trabalho todas as etapas do processo de Mineração de Dados, procurando mostrar a importância da utilização desta técnica.

O trabalho proposto assemelha-se ao de (MENDES, 2011) por utilizar técnicas de Mineração de Dados, sendo aplicada a clusterização com o intuito de identificar perfis de inscritos do ENEM de 2010. Enquanto (MENDES, 2011) realiza um estudo sobre as técnicas de Mineração de Dados, este realiza um estudo comparativo entre duas técnicas de seleção de atributos, buscando identificar a que apresenta melhores resultados.

A seguir, na Tabela 1 é possível identificar as diferenças e semelhanças entre os trabalhos relacionados e o proposto.

Tabela 1: **Comparação entre os trabalhos relacionados e o proposto**

	Caminha et al. (2015)	**Jorge (2010)**	**Molina (2002)**	**Pereira e Lima (2015)**	**Karegowda et. al (2010)**	**Trabalho Proposto**
Dados Abertos	Sim	Não	Não	N.A	Sim	Sim
Métodos de seleção de atributos	Não	Sim	Sim	Sim	Sim	Sim
Comparação entre abordagens *Filter/ Wrapper*	Não	Sim	Não	N.A.	N.A.	Sim
Mineração de Dados	Clusterização	Classificação	Não	Classificação	Classificação	Classificação e Clusterização
Aplicação de diferentes algoritmos de busca	Não	Não	Sim	N.A.	Sim	Sim

Fonte: Elaborada pela autora.

4 Objetivos

Nas seções seguintes serão apresentados os objetivos, tanto o geral quanto os específicos.

4.1 Objetivo Geral

As bases de dados são multidimensionais e para acelerar o processo de clusterização, é importante aplicar técnicas de seleção de atributos para reduzir a quantidade de atributos do conjunto de dados. Uma base de dados possui diversos atributos, tornando difícil a escolha destes atributos. Tendo em vista isto, este trabalho tem como objetivo identificar quais os atributos mais relevantes, da base do ENEM de 2010, capazes de definir melhor os perfis de inscritos, via clusterização de dados.

4.2 Objetivos Específicos

- Analisar quais são as métricas de recuperação de informação existentes;
- Escolher e aplicar tais métricas nas bases de dados do ENEM de 2010;
- Aplicar a técnica de clusterização de dados utilizando os atributos identificados como importantes;
- Encontrar os perfis de inscritos, a partir dos atributos selecionados;
- Comparar os *clusters* encontrados com e sem uso das métricas.

5 Procedimentos Metodológicos

A seguir serão descritos todos os procedimentos que foram realizados durante este estudo.

5.1 Coleta de Dados

A primeira etapa da execução deste trabalho consistiu na coleta dos dados utilizados neste estudo. Os dados são da base do ENEM correspondentes a edição de 2010, esta encontra-se disponibilizada pelo Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (Inep) e foi obtida por meio de *download* através do Portal Brasileiro de Dados Abertos. Os arquivos obtidos a partir do portal contêm dicionário de variáveis, responsável por esclarecer o papel de cada atributo, dados referentes ao inscrito, como local de prova, gabarito das provas e o questionário socioeconômico, que foi utilizado neste estudo.

5.2 Preparação dos dados

A segunda etapa consistiu em preparar os dados para facilitar o desenvolvimento do estudo e, consequentemente, obter resultados precisos ao final. Os dados obtidos eram persistidos originalmente em arquivos de texto puro de extensão TXT. Estes dados foram organizados e persistidos em um novo arquivo CSV (*Comma-separated values*). Este processo foi feito utilizando o *software* livre *Pentaho Data Integration* (PDI) juntamente com o uso do dicionário das variáveis. O dicionário das variáveis é um arquivo no qual se informa o significado de cada atributo existente na base de dados. Ele auxiliou na separação dos atributos, ajudando a identificar cada atributo. Um dos arquivos contém as notas de cada prova, sendo necessário realizar o cálculo da média. Isto foi feito utilizando o Pentaho. Como foram utilizados dois arquivos diferentes, foi necessário unir em um só os dados referentes às respostas do questionário socioeconômico e a média final obtida no exame.

Os dados, mesmo após organizados, ainda possuíam inconsistências (HAN; KAMBER, 2001) como, por exemplo, em campos numéricos contendo letras. Dados inconsistentes dificultam o trabalho em um conjunto de dados, de modo que se utilizados podem gerar resultados incorretos. Portanto, foi necessário remover inconsistências.

Após a remoção das inconsistências, foram atribuídos pesos aos diversos atributos para que fossem melhor compreendidos pelos algoritmos que neles iriam ser aplicados. Este processo foi feito, também, utilizando o Pentaho.

5.3 Escolha de métodos de Seleção de Atributos a serem aplicados sobre os dados

Tendo em vista que existem diversas métricas de seleção de atributo, esta etapa consistiu na escolha de métodos de seleção de atributos a serem aplicados sobre os dados. Foi realizado um estudo sobre as abordagens *filter* e *wrapper* para identificar a melhor em selecionar atributos para aplicação do algoritmo de clusterização (considerando os dados utilizados neste trabalho). Nesta etapa foi utilizado o *software* livre *Waikato Environment of Knowledge Analysis* (WEKA) para a execução de cada abordagem, sendo utilizada a implementação de cada uma existente no WEKA, para *filter* foi usada a implementação chamada CfsSubsetEval e para *wrapper* a chamada WrapperSubsetEval.

5.4 Avaliação da Redução da quantidade de atributos da Base de Dados

Nesta etapa foi realizada uma avaliação sobre os atributos obtidos a partir da utilização dos métodos de seleção. Os resultados obtidos, a partir da redução de dimensionalidade, em cada uma das abordagens foram comparados. Foi escolhida aquela abordagem que apresentou melhores resultados. Os atributos selecionados foram utilizados no processo de clusterização que será descrito na seção 5.5. Para a avaliação foi utilizada uma medida para identificar se os atributos escolhidos são capazes de representar aqueles que foram descartados.

5.5 Aplicação de algoritmo de clusterização de dados

Nesta etapa foi realizado o cálculo da soma do erro quadrático para descobrir a quantidade ideal de *clusters* a serem gerados. Este procedimento foi realizado executando um algoritmo simples já explicado na seção 2.2. Nele foi passada uma matriz com os dados, obtidos após a aplicação de *feature selection*, e um número representando uma quantidade máxima de clusters. Ao final foi gerado um gráfico mostrando a soma do erro quadrático, ou seja a quantidade de erros de acordo com a quantidade de cluster. Após isto, foi aplicado o algoritmo de clusterização nos dados já pré - processados, com o intuito de gerar clusters com os perfis de inscritos. Considere dados pré-processados como os dados selecionados a partir da seleção de atributos que passaram pelo processo de normalização, que será explicado mais adiante, e encontram-se numa faixa de valores entre 0 e 1.

5.6 Avaliação dos clusters obtidos

Nesta etapa foi realizada uma comparação entre os clusters obtidos neste estudo e os apresentados no trabalho de (CAMINHA; MOREIRA; SILVA, 2015), com o intuito de verificar se os perfis encontrados foram diferentes e analisar o impacto socioeconômico no desempenho do inscrito.

Foram comparados os clusters obtidos por meio da abordagem *filter*, os *clusters* obtidos da *wrapper* e os clusters obtidos no trabalho de (CAMINHA; MOREIRA; SILVA, 2015), a fim de verificar qual oferece melhor perfil de inscritos relacionando desempenho na prova e características socioeconômicas.

6 Experimentos e Resultados

Este capítulo tem por objetivo relatar, detalhadamente, a execução dos procedimentos citados na seção 6.1 e os resultados obtidos em cada procedimento.

6.1 Experimentos

Esta seção apresenta, em detalhes, todo os experimentos executados ao longo deste estudo.

6.1.1 Coleta de Dados

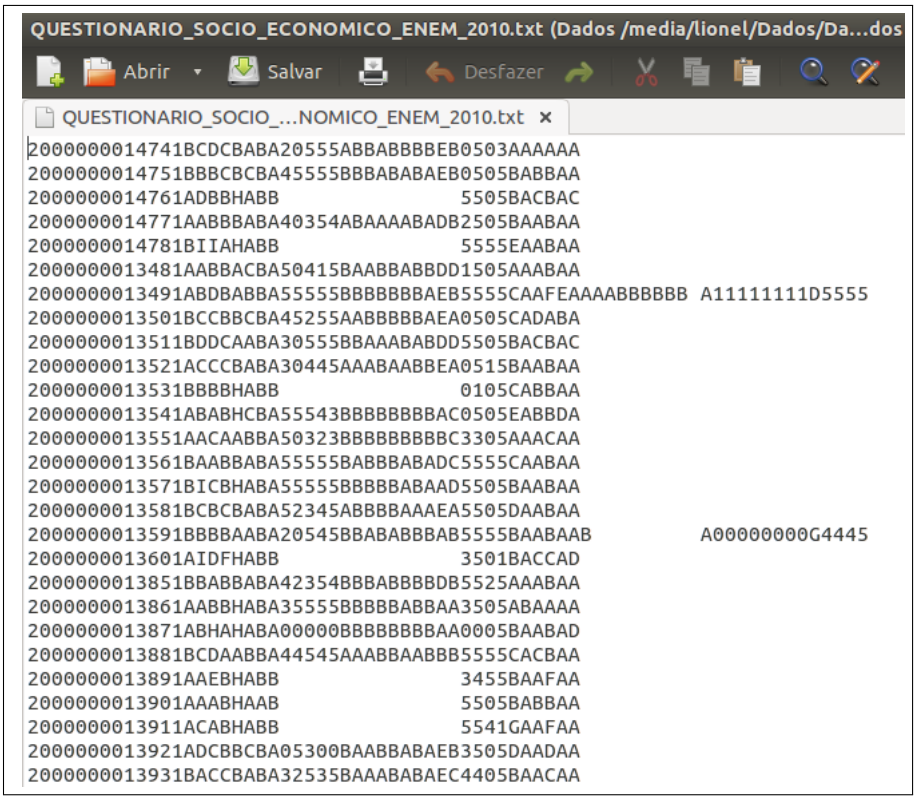
A primeira etapa da execução deste trabalho consistiu na coleta dos dados. A base coletada foi a do ENEM de 2010, cujo tamanho era 4.9 GB, contendo 12 itens, dentre eles encontravam-se o dicionário de variáveis, dados da prova incluindo as notas de cada competência, respostas do questionário socioeconômico e o próprio questionário. Dois arquivos foram utilizados para análise, um deles continha os dados das provas e o outro as respostas referentes ao questionário. O primeiro arquivo possuía 4.611.616 tuplas, e o segundo 4.626.094 tuplas. A diferença na quantidade de tuplas entre eles ocorre devido ao fato que nem todos os inscritos que responderam ao questionário participaram do exame.

6.1.2 Preparação dos dados

A segunda etapa consistiu em preparar os dados para facilitar o desenvolvimento do estudo e, consequentemente, obter resultados precisos ao final. Os dados persistidos originalmente em arquivos de texto puro de extensão TXT, conforme pode ser visto na Figura 3. Foram organizados e persistidos em novo arquivo em formato *Comma-separated values* (CSV). Neste processo foi utilizado o *software* livre *Pentaho Data Integration* (PDI), juntamente com o uso do dicionário das variáveis. O dicionário das variáveis foi utilizado para identificar o tamanho de cada campo, onde inicia e termina, e seu respectivo significado. A Figura 4 mostra como este processo ocorreu no Pentaho. Note que as linhas verticais em vermelho representam onde cada campo inicia e termina. A Figura 5 mostra o resultado final deste processo, ou seja, os dados persistidos em arquivo CSV.

O arquivo referente a prova contém informações relacionadas ao inscrito como o local de prova, gabarito preenchido pelo mesmo, escola onde estuda, notas obtidas em cada prova e nota final da redação, etc. Como as notas encontram-se separadas, foi preciso realizar o cálculo da média final. Isto foi feito também utilizando o Pentaho. Foi criado um *step* no pentaho para somar todas as notas e ao final calcular a média delas e as armazenar em um arquivo CSV. *Step*

Figura 3: Arquivo original com dados persistidos em arquivo de texto puro



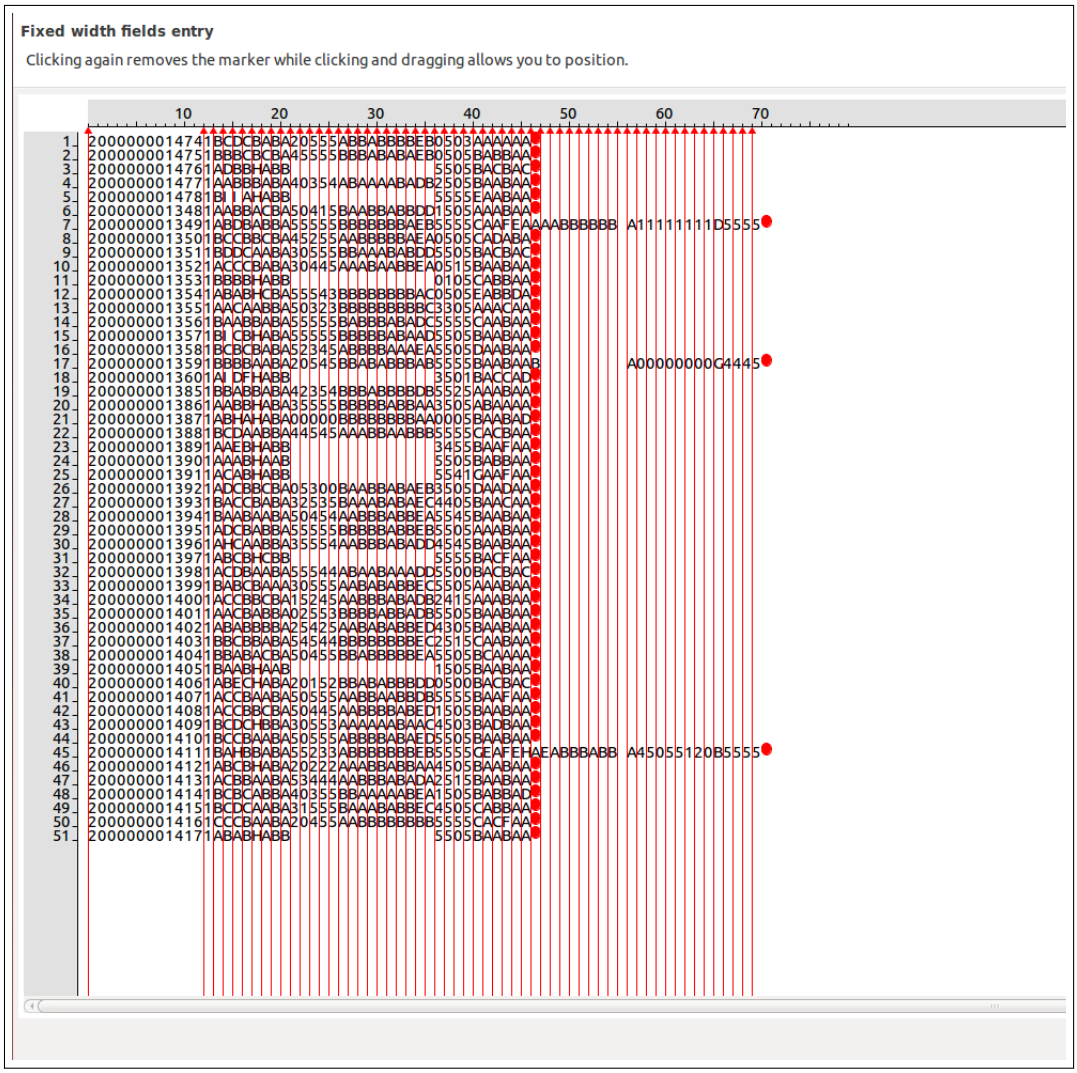
Fonte: Elaborada pela autora

é como o pentaho chama uma sequência de passos para a execução de uma transformação. Cada inscrito foi identificado pelo seu número de inscrição. Tendo a inscrição como identificador, foi possível criar um novo *step* para unir em um mesmo arquivo os dados referentes às respostas do questionário socioeconômico e a média final do exame, conforme pode ser visto na Figura 6.

Os dados, mesmo após organizados, ainda possuíam inconsistências como, por exemplo, campos numéricos continham letras ou o campo vazio continha asterisco (*). Dados inconsistentes dificultam o trabalho em um conjunto de dados, de modo que se utilizados podem gerar resultados incorretos sendo, portanto, necessário removê-los. Para isto foi criado mais um *step* no Pentaho, onde sempre que fosse encontrado um * toda a tupla era removida. Após todo o processo de limpeza dos dados restaram 3.105.939 de tuplas.

O questionário socioeconômico da edição do ENEM de 2010 possui 57 questões. No entanto algumas questões não puderam ser utilizadas. Existiam questões destinadas apenas às pessoas que trabalhavam ou já tinham trabalhado, mas nem todos os inscritos as haviam respondido. O mesmo ocorreu com questões que eram destinadas exclusivamente àqueles que estavam prestando o exame com o intuito de obter certificado. As questões citadas apresentavam inconsistências devido ao fato de que nem todos as haviam respondido, sendo necessário descartá-las pelo motivo já explicado no parágrafo anterior. Após esta etapa sobraram 18 questões a serem utilizadas conforme pode ser observado no Quadro 6.1.

Figura 4: Separação de campos de atributos para persistir os dados em arquivo CSV

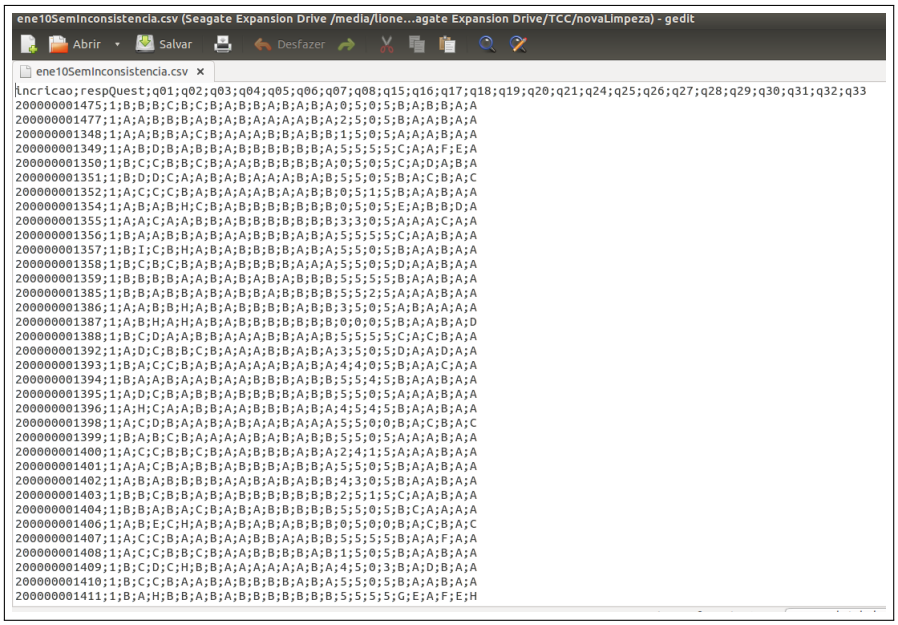


Fonte: Elaborada pela autora

Após a remoção das inconsistências, foram atribuídos pesos aos diversos atributos para que fossem melhor compreendidos pelos algoritmos que neles iriam ser aplicados. Este processo foi realizado, também, utilizando o Pentaho com a criação de mais um *step*(Apresentado na Figura 8), onde era selecionada uma questão e cada item era substituído por um valor numérico, por exemplo a letra A era substituída pelo valor 1. Os pesos foram definidos através da atribuição de categorias para as respostas, recebendo peso 1 as respostas consideradas negativas, 3 as intermediárias e 5 as positivas. A polaridade das respostas está relacionada às condições satisfatórias ou não de qualidade de vida e estudo do candidato para obter sucesso no exame. Os pesos com suas respectivas questões são apresentados no Quadro 6.2.

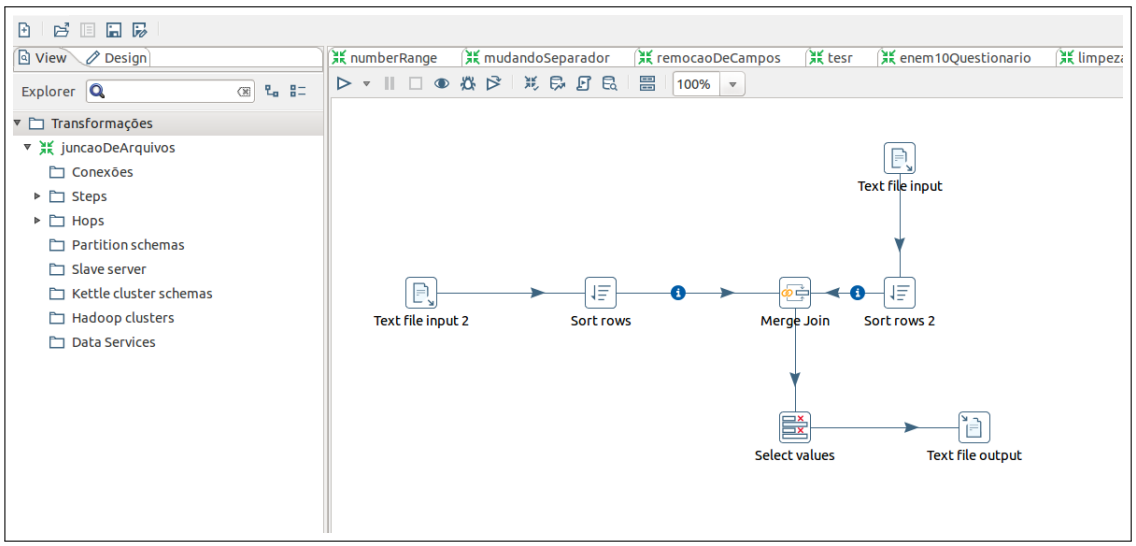
Como desejava-se utilizar como algoritmo de classificação a árvore de decisão C4.8 (QUINLAN, 1993) implementada pelo WEKA como J4.8, fez-se necessário classificar os dados de acordo com a média, sendo classificada como Baixa a nota abaixo de 500, como Intermediária a presente no intervalo de 500 a 650, e como Alta as notas acima de 650. Para isto

Figura 5: Dados persistidos em arquivo CSV



Fonte: Elaborada pela autora

Figura 6: Unindo dados do questionário com a média



Fonte: Elaborada pela autora

foi utilizado o Pentaho. O algoritmo C4.8 trabalha apenas utilizando valores de classes. Ele não reconhece valores numéricos. Isso quer dizer, ele é adequado para valores de classe que sejam discretos e não contínuos.

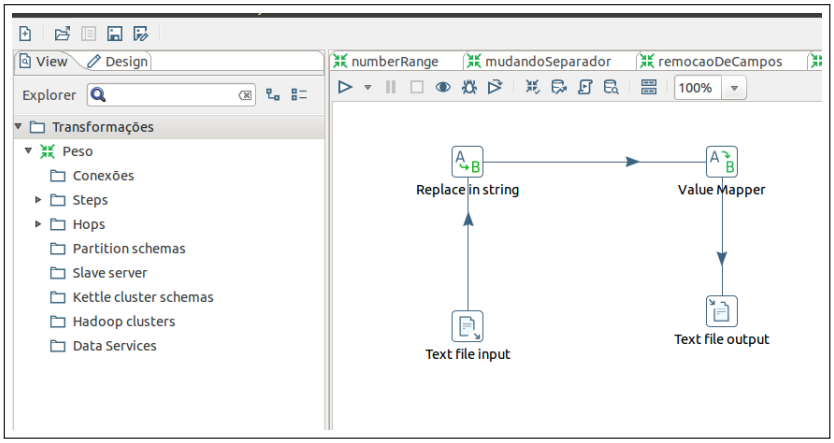
6.1.3 Aplicação dos métodos de Seleção de Atributos

Tendo em vista a existência de diferentes abordagens para a seleção de atributos, esta etapa consistiu na aplicação tanto da abordagem *filter* quanto da *wrapper* sobre os dados. Isto foi feito para realizar um estudo sobre as abordagens citadas, com o intuito de identificar a

QUESTÕES SELECIONADAS
Quantas pessoas moram com você?
Qual é o nível de escolaridade do seu pai?
Qual é o nível de escolaridade da sua mãe?
Somando a sua renda com a renda das pessoas que moram com você, quanto é, aproximadamente, a renda familiar mensal?
Qual a sua renda mensal, aproximadamente?
A casa onde você mora é?(Cedida, Alugada, Própria)
Sua casa está localizada em?
Você trabalha ou já trabalhou?
Testar meus conhecimentos
Prosseguir os estudos no Ensino Superior
Obter a certificação do Ensino Médio ou acelerar meus estudos
Conseguir uma bolsa de estudos (ProUni, outras)
Quantos anos você levou para concluir o ensino fundamental?
Você deixou de estudar durante o Ensino Fundamental?
Em que tipo de escola você cursou o Ensino Fundamental?
Quantos anos você levou para concluir o Ensino Médio?
Você deixou de estudar durante o Ensino Médio?
Em que tipo de escola você cursou o Ensino Médio?

Quadro 6.1: Questões selecionadas do questionário socioeconômico

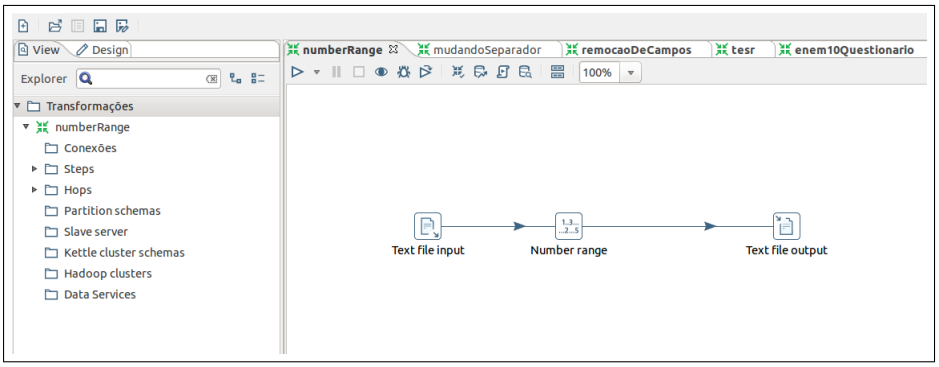
Figura 7: Atribuindo pesos às respostas de cada questão, através do Pentaho.



Fonte: Elaborada pela autora

melhor em selecionar atributos para aplicação do algoritmo de clusterização (considerando os dados utilizados neste trabalho).

Figura 8: Dividindo a média nas classes: Baixa, Intermediária e Alta



Fonte: Elaborada pela autora

Nesta etapa foi utilizado o *software* livre (WEKA) para a execução de cada abordagem, sendo utilizada a implementação de cada abordagem existente no WEKA, para *filter* através do algoritmo *CFsSubsetEval*(CFs) e para *wrapper* a implementação chamada *WrapperSubsetEval*. Estes algoritmos foram utilizados como avaliador de atributos.

Como os dados encontravam-se com valores bem variantes de 1 a 5, devido aos pesos atribuídos anteriormente, foi necessário normalizar os dados em um intervalo de 0 a 1 antes de iniciar a seleção de atributos, portanto, mesmo que eles sejam alterados, a modificação não afetará o resultado final. Para tal etapa foi necessária a aplicação do algoritmo *Normalize* implementado através do WEKA. A normalização permite avaliar melhor a variação de valores pelo algoritmo.

Nesta etapa foram executados quatro algoritmos de busca, sendo eles *BestFirst*, *GeneticSearch* e *RankerSearch* em cada uma das abordagens. Todos os resultados obtidos através do WEKA nesta etapa são mostrados nos Apêncides. No Apêndice A são apresentados os resultados da abordagem *filter* e no Apêndice B os da abordagem *wrapper*. A partir dos resultados obtidos foi possível perceber que a abordagem *filter* obteve melhores resultados, selecionando os mesmos atributos em todos os algoritmos. Enquanto que a abordagem *wrapper* selecionou apenas um atributo igual em todos os algoritmos, o atributo média, sendo este importante pois é o que informa o desempenho do inscrito. A Quadro 6.3 mostra os atributos selecionados de acordo com cada abordagem e algoritmo utilizado.

6.1.4 Avaliação da Redução da quantidade de atributos da Base dos Dados

Nesta etapa foi realizada uma avaliação sobre os atributos obtidos a partir da utilização dos métodos de seleção. A comparação foi realizada entre os resultados obtidos após a aplicação da técnica de seleção de atributos, tanto pela abordagem *filter* quanto pela abordagem *wrapper*, utilizando os diferentes algoritmos de busca apresentados na seção de fundamentação teórica.

Realizou-se a avaliação utilizando o coeficiente de correlação de Pearson, por meio da qual foi possível analisar se os atributos selecionados realmente são capazes de representar aque-

QUESTÕES	PESOS		
Quantas pessoas moram com você?	De uma a três ou sozinho=5	Quatro a sete = 3	Oito a mais de dez=1
Qual é o nível de escolaridade do seu pai?	Baixo = 1	Intermediário = 3	Alto = 5
Qual é o nível de escolaridade da sua mãe?	Baixo = 1	Intermediário = 3	Alto = 5
Somando a sua renda com a renda das pessoas que moram com você, quanto é, aproximadamente, a renda familiar mensal?	Renda Baixa (até 3 salários mínimos) = 1	Renda Intermediária (3 a 9 salários mínimos) = 3	Renda Alta (a partir de 9 salários mínimos) = 5
Qual a sua renda mensal, aproximadamente?	Renda Baixa (até 3 salários mínimos) = 1	Renda Intermediária (3 a 9 salários mínimos) = 3	Renda Alta (a partir de 9 salários mínimos) = 5
A casa onde você mora é?	Cedida=1	Alugada=3	Própria=5
Sua casa está localizada em?	Comunidade quilombola ou indígena=1	Zona rural=3	Zona urbana=5
Você trabalha ou já trabalhou?	Não = 1	Sim = 3	
Testar meus conhecimentos	Grau de interesse Baixo=1	Grau de interesse Intermediário=3	Grau de interesse Alto=5
Prosseguir os estudos no Ensino Superior	Grau de interesse Baixo=1	Grau de interesse Intermediário=3	Grau de interesse Alto=5
Obter a certificação do Ensino Médio ou acelerar meus estudos	Grau de interesse Baixo=1	Grau de interesse Intermediário=3	Grau de interesse Alto=5
Conseguir uma bolsa de estudos (ProUni, outras)	Grau de interesse Baixo=1	Grau de interesse Intermediário=3	Grau de interesse Alto=5
Quantos anos você levou para concluir o ensino fundamental?	Mais de 11 anos ou não concluir = 1	De 10 a 11 anos = 3	Até 9 anos = 5
Você deixou de estudar durante o Ensino Fundamental?	Por três anos ou mais=1	Por um ano ou dois anos=3	Não=5
Em que tipo de escola você cursou o Ensino Fundamental?	Não frequentou a escola = 1	Escola Pública = 3	Escola Privada = 5
Quantos anos você levou para concluir o Ensino Médio?	Acima de 5 anos ou não concluiu=1	De 4 a 5 anos = 3	Até 3 anos= 5
Você deixou de estudar durante o Ensino Médio?	Por 3 anos ou mais = 1	Por 1 ou 2 anos = 3	Não = 5
Em que tipo de escola você cursou o Ensino Médio?	Não frequentou a escola = 1	Escola Pública = 3	Escola Privada = 5

Quadro 6.2: Questões selecionadas e seus respectivos pesos

les que foram descartados. A medida de correlação mede o quão próximos são dois atributos. Neste passo foi utilizada a linguagem de programação R[1] através da IDE Rstudio [2].

Utilizando a linguagem R foram passados como entrada dois vetores de dados para realizar o cálculo da correlação, um contendo os atributos selecionados e outro contendo os descartados. Ao final da execução foi gerada como resultado uma matriz exibindo o valor da correlação entre cada um dos atributos. Esta matriz é apresentada na Figura 11, através da qual é possível perceber que as variáveis selecionadas possuem correlação com as descartadas.

[1] Disponível em https://www.r-project.org/about.html
[2] Disponível em https://www.rstudio.com/

6.1.5 Aplicação do algoritmo de clusterização

Nesta etapa ocorreu a geração dos clusters, para tanto foi realizado o cálculo da soma do erro quadrático, também chamado de *elbow*, com o intuito de descobrir a quantidade ideal de *clusters* a serem gerados. Este procedimento foi realizado executando um algoritmo simples explicado na seção seção 2.2, sendo passados como entrada uma matriz com os dados pré-processados e um número que representava uma quantidade máxima de *clusters*. Ao final foi gerado um gráfico mostrando a soma do erro quadrático. Após isto, foi aplicado o algoritmo de clusterização, k-means, utilizando a linguagem de programação R nos dados já pré-processados, com o intuito de gerar *clusters* com os perfis de inscritos. Para esta etapa foi removida a coluna que possuía a classificação da média de acordo com classes (Alta, Intermediária e Baixa), pois o algoritmo k-means reconhece apenas valores numéricos. No entanto, os valores numéricos em si da média de cada inscrito permaneceu.

6.2 Avaliação dos clusters obtidos

Nesta etapa foi realizada uma comparação entre os clusters obtidos neste estudo e os apresentados no trabalho de (CAMINHA; MOREIRA; SILVA, 2015), com o intuito de verificar se os perfis encontrados foram diferentes, analisando o impacto socioeconômico no desempenho do inscrito. Além disso, foram analisados os atributos utilizados nos dois trabalhos.

6.3 Resultados

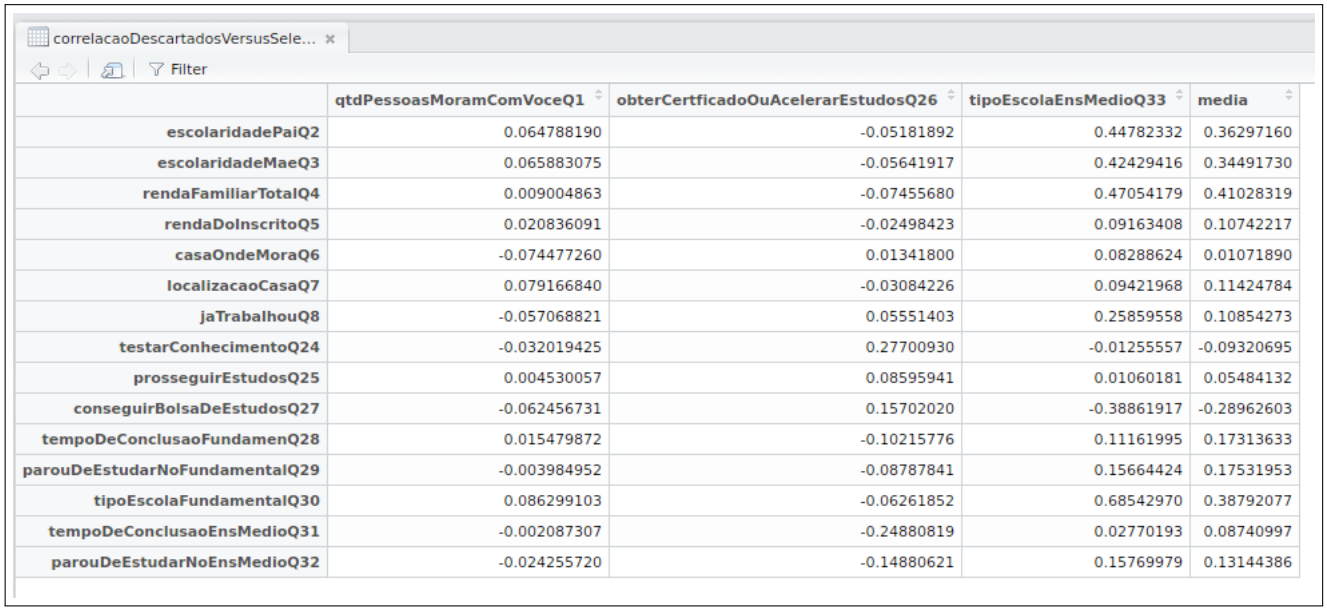
A seguir serão apresentados os resultados obtidos com a realização deste estudo.

Utilizando o método de seleção de atributos, foi possível identificar aqueles capazes de representar toda a base. No Apêndice é possível visualizar os resultados quanto a comparação entre cada abordagem, no Apêndice A são apresentados os resultados da abordagem *filter* e no Apêndice B os resultados da abordagem *wrapper*. No Quadro 6.3 encontram-se os atributos selecionados de acordo com a abordagem. Com base nos resultados obtidos verificou-se através do coeficiente de correlação de Pearson que a abordagem *filter* obteve melhores resultados selecionando os mesmos atributos em todos os algoritmos, reduzindo a quantidade de atributos que antes era 18 e passou para 4. Para uma melhor compreensão observe a Figura 11, que apresenta a correlação entre os atributos selecionado e os descartados. A medida de correlação mede o quão próximos são duas variáveis, quanto mais próximo de 1 indica que os valores das variáveis possuem correlação perfeita positiva e quanto mais próximo de -1 as varáveis possuem correlação negativa, isto é se uma aumenta a outra sempre diminui[3]. Analisando a Figura 11 é possível perceber que todos os atributos estão correlacionados, ou seja, todos os atributos selecionados conseguem de alguma forma representar aqueles que foram descartados, embora

[3] http://www.aurea.uac.pt/pdf_MBA/coef_correl_Pearson.pdf

muitas destas correlações sejam fracas. Vale ressaltar que nem sempre todos os quatro atributos vão determinar juntos os outros descartados, pois o grau de correlação entre as variáveis é diferente.

Figura 9: Correlação entre atributos descartados e selecionados



correlacaoDescartadosVersusSele... ×

Filter

	qtdPessoasMoramComVoceQ1	obterCertficadoOuAcelerarEstudosQ26	tipoEscolaEnsMedioQ33	media
escolaridadePaiQ2	0.064788190	-0.05181892	0.44782332	0.36297160
escolaridadeMaeQ3	0.065883075	-0.05641917	0.42429416	0.34491730
rendaFamiliarTotalQ4	0.009004863	-0.07455680	0.47054179	0.41028319
rendaDoInscritoQ5	0.020836091	-0.02498423	0.09163408	0.10742217
casaOndeMoraQ6	-0.074477260	0.01341800	0.08288624	0.01071890
localizacaoCasaQ7	0.079166840	-0.03084226	0.09421968	0.11424784
jaTrabalhouQ8	-0.057068821	0.05551403	0.25859558	0.10854273
testarConhecimentoQ24	-0.032019425	0.27700930	-0.01255557	-0.09320695
prosseguirEstudosQ25	0.004530057	0.08595941	0.01060181	0.05484132
conseguirBolsaDeEstudosQ27	-0.062456731	0.15702020	-0.38861917	-0.28962603
tempoDeConclusaoFundamenQ28	0.015479872	-0.10215776	0.11161995	0.17313633
parouDeEstudarNoFundamentalQ29	-0.003984952	-0.08787841	0.15664424	0.17531953
tipoEscolaFundamentalQ30	0.086299103	-0.06261852	0.68542970	0.38792077
tempoDeConclusaoEnsMedioQ31	-0.002087307	-0.24880819	0.02770193	0.08740997
parouDeEstudarNoEnsMedioQ32	-0.024255720	-0.14880621	0.15769979	0.13144386

Fonte: Elaborada pela autora

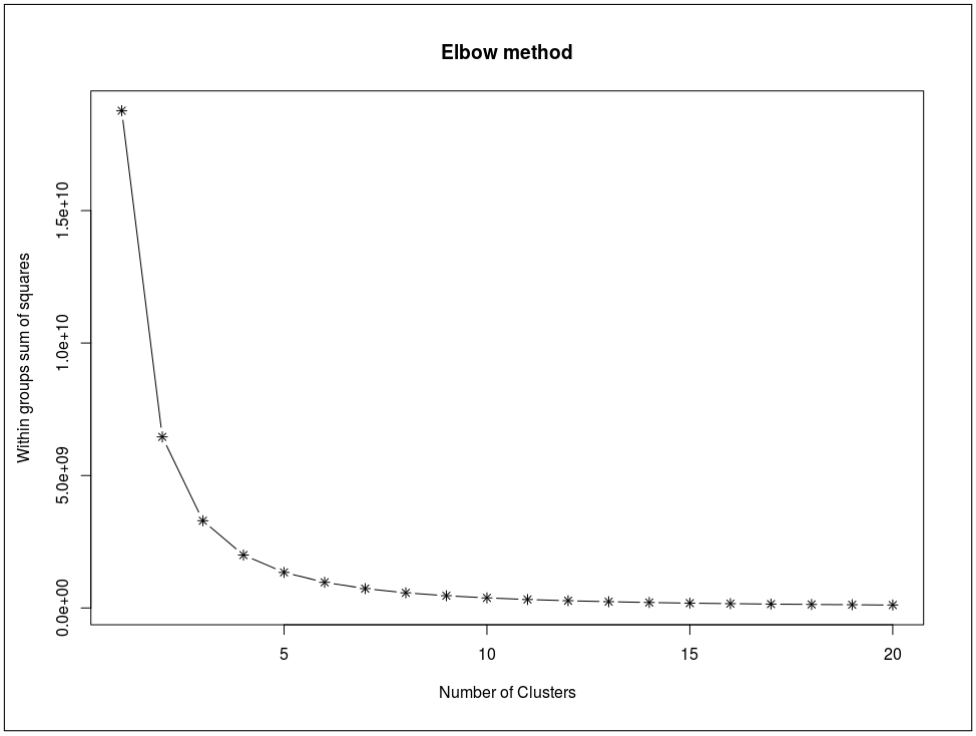
Algoritmo de busca	Abordagem Wrapper	Abordagem Filter
GeneticSearch	rendaDoInscritoQ5 e média	qtdPessoasMoramComVoceQ1 obterCertificadoOuAcelerarEstudosQ26, tipoEscolaEnsMedioQ33 e média
BestFirst	localizacaoCasaQ7 e média	qtdPessoasMoramComVoceQ1, obterCertificadoOuAcelerarEstudosQ26, tipoEscolaEnsMedioQ33 e média
RankerSearch	média	qtdPessoasMoramComVoceQ1, obterCertificadoOuAcelerarEstudosQ26, tipoEscolaEnsMedioQ33 e media

Quadro 6.3: **Atributos Selecionados por cada abordagem**

Como já dito anteriormente, a regra do cotovelo executa o k-means no conjunto de dados, com o intuito de identificar a quantidade ideal de *clusters*. A Figura 10 mostra o gráfico gerado por este método apresentando o erro quadrático de acordo com o número de *clusters*. É considerada a quantidade ideal, o ponto onde os erros começam a se estabilizar, observe que

este ponto estável é identificado pelo valor 15. Portanto 15 é a quantidade quantidade ideal de clusters a ser gerada. Para a execução da regra do cotovelo, foram utilizados apenas os atributos escolhidos na etapa de seleção.

Figura 10: Quantidade ideal de *clusters* a serem gerados utilizando os atributos considerados relevantes



Fonte: Elaborada pela autora

Os centróides de um cluster representam a média gerada pelo k-means em cada atributo pertencente a ele. Através dessas médias, foi possível analisar as características dos perfis, de acordo com os pesos atribuídos anteriormente. Cada centróide corresponde a um perfil de inscrito, conforme pode ser visualizado na Figura 12. Analisando estes centróides é possível perceber, de modo geral, que o inscrito mora com muitas pessoas, não realizou o exame com o intuito de obter certificado ou acelerar os estudos, frequentou escola pública durante seu ensino médio, obtendo um baixo rendimento.

Figura 11: Correlação entre atributos selecionados

	qtdPessoasMoramComVoceQ1	obterCertficadoOuAcelerarEstudosQ26	tipoEscolaEnsMedioQ33	media
qtdPessoasMoramComVoceQ1	1.00000000	-0.03182895	0.07946737	0.1220538
obterCertficadoOuAcelerarEstudosQ26	-0.03182895	1.00000000	-0.09237048	-0.2265971
tipoEscolaEnsMedioQ33	0.07946737	-0.09237048	1.00000000	0.3979045
media	0.12205379	-0.22659712	0.39790452	1.0000000

Fonte: Elaborada pela autora

Figura 12: *Clusters* gerados

	qtdPessoasMoramComVoceQ1	obterCertficadoOuAcelerarEstudosQ26	tipoEscolaEnsMedioQ33	media
1	1.0000000	1.0000000000	0.4803520	0.3652960
2	0.5000000	0.0027159502	0.5000000	0.6246806
3	0.7810363	0.7982698735	1.0000000	0.4991950
4	0.4736662	0.0000000000	1.0000000	0.4235196
5	1.0000000	0.5000000000	0.5000000	0.3952270
6	0.4709135	0.8490059096	0.4922030	0.3590696
7	0.7830474	0.1533804238	0.0000000	0.3746425
8	1.0000000	0.0000000000	1.0000000	0.5601661
9	0.0000000	0.1196339827	0.5087449	0.3701133
10	0.5000000	0.0000000000	0.5000000	0.5091310
11	0.4875936	0.0007065743	1.0000000	0.6388436
12	0.5000000	0.0000000000	0.5000000	0.4167085
13	0.5000000	0.0000000000	0.5000000	0.2213814
14	1.0000000	0.0000000000	0.5000000	0.4284913
15	0.5000000	0.0000000000	0.5000000	0.3255821

Fonte: Elaborada pela autora

Analisando os *clusters* obtidos neste estudo e os apresentados no trabalho de (CAMINHA; MOREIRA; SILVA, 2015) é possível observar que a maioria dos atributos utilizados são diferentes, possuindo apenas um igual que é o referente ao tipo de escola frequentada durante o ensino médio. O trabalho de (CAMINHA; MOREIRA; SILVA, 2015) apresenta perfis, de modo geral, no qual os pais do inscrito possuem escolaridade baixa, consequentemente uma renda familiar baixa, onde o inscrito não exerceu nenhuma atividade remunerada até a data do exame, sendo possível o inscrito ter cursado o ensino fundamental parte em escola pública e parte em particular, tendo sido transferido para a rede pública no ensino médio, obtendo um resultado considerado insatisfatório. Nos clusters obtidos neste trabalho, como já dito anteriormente, a maioria dos centróides representam um perfil padrão de inscrito que mora com muitas pessoas, não fez o ENEM para obter certificado ou acelerar o estudos, tendo estudado durante o ensino médio em escola pública obtendo baixo rendimento. Entretanto, existem centróides capazes de representar um perfil diferente do padrão, onde o inscrito divide a casa com muitas pessoas, seu grau de interesse em realizar o ENEM para obter certificado ou acelerar os estudos foi baixo e obteve uma média intermediária, mas que é maior de todos os centróides. Assim como há casos no qual o inscrito mora com poucas pessoas, não tinha intenção em obter certificado ao realizar o exame, cursou o ensino médio na rede pública, obtendo um baixo rendimento. Também existem casos onde o inscrito morava com poucas pessoas, tinha grau de interesse em obter certificado alto, tendo estudado durante o ensino médio na rede particular, entretanto, obteve baixo rendimento. Mostrando que quase todos os cenários de variação dos atributos selecionados estão presentes em algum dos clusters.

Figura 13: *Clusters* apresentado no trabalho de (CAMINHA; MOREIRA; SILVA, 2015)

```
> cEnem10$centers
   EscolaridadePai EscolaridadeMae  rendaTotal JaTrabalhou tempoFundamental TipoEscolaMedio     media
1      0.500000000      0.00000000 0.000000000   0.0000000        0.9919770     0.000000000 0.5294602
2      0.999192735      0.99861882 0.026437941   0.7197906        0.8139442     0.051715439 0.3782271
3      0.000000000      0.00000000 0.000000000   0.0000000        1.0000000     0.000000000 0.4456878
4      0.011060342      0.04327132 0.000000000   0.0000000        0.5000000     0.000000000 0.3808005
5      0.773136545      0.00000000 0.002831152   1.0000000        0.9111390     0.000000000 0.4499727
6      0.000000000      0.00000000 0.000000000   0.0000000        1.0000000     0.000000000 0.2973616
7      0.491273501      0.47182216 1.000000000   0.0000000        0.9735604     0.002018052 0.6124085
8      0.000000000      0.00000000 0.000000000   1.0000000        1.0000000     0.000000000 0.4411161
9      0.000000000      0.00000000 0.000000000   1.0000000        1.0000000     0.000000000 0.2975657
10     0.533637612      0.51881474 0.000000000   1.0000000        0.9806416     0.000000000 0.5259503
11     0.000000000      0.66034432 0.000000000   1.0000000        0.9658158     0.000000000 0.4643820
12     0.000000000      0.00000000 0.000000000   0.0000000        1.0000000     0.000000000 0.5992486
13     0.064748067      0.06943374 0.021477044   1.0000000        0.8337207     1.000000000 0.4027256
14     0.112290014      0.11676607 1.000000000   0.5498925        0.9661411     0.000000000 0.4961108
15     0.005445095      0.04104130 0.008867807   1.0000000        0.2946685     0.000000000 0.3839108
16     0.045865201      0.10278211 0.063884557   0.0000000        0.0000000     0.000000000 0.3890482
17     0.000000000      0.00000000 0.000000000   1.0000000        1.0000000     0.000000000 0.5786653
18     0.171004988      0.54470885 0.000000000   0.0000000        0.9915223     0.000000000 0.5228021
19     1.000000000      0.06566936 0.000000000   0.0000000        0.9178456     0.000000000 0.4171904
20     0.076836735      0.07806122 0.048027211   0.0000000        0.8517687     1.000000000 0.3752235
```

Fonte: (CAMINHA; MOREIRA; SILVA, 2015)

7 Conclusão e Trabalhos Futuros

Foi apresentada neste trabalho a importância da etapa de seleção de atributos quando é necessário trabalhar com base de alta dimensionalidade, ressaltando que escolher atributos, a serem utilizados em um estudo, de forma aleatória pode causar resultado impreciso ou inútil.

Diferentes perfis encontrados neste estudo apresentam características semelhantes entre si, onde o inscrito mora com muitas pessoas, fez o ENEM sem o intuito de obter certificado ou acelerar o estudos, cursou o ensino médio em escola pública obtendo como resultado um baixo rendimento. Entretanto, existem casos que fogem do padrão, como o cluster 11 que representa um perfil de inscrito que mora com muitas pessoas tendo realizado o ENEM sem o intuito de obter certificado e frequentou o ensino médio em escola particular obtendo média intermediária. Este caso contrasta com outro perfil onde o inscrito mora com poucas pessoas, frequentou escola particular durante o ensino médio e obteve um baixo rendimento. (CAMINHA; MOREIRA; SILVA, 2015) concluem o trabalho afirmando que fatores socioeconômicos não são capazes de determinar o rendimento do inscrito. Diante dos resultados apresentados na análise entre os clusters obtidos neste estudo e os apresentados no trabalho de (CAMINHA; MOREIRA; SILVA, 2015), pode-se concluir o mesmo.

Este trabalho propôs analisar métricas de recuperação de informação existentes, escolher e aplica-las na Base de Dados do ENEM de 2010, aplicar clusterização nos atributos selecionados para identificar perfis de inscritos e realizar uma comparação entre os *cluster* encontrados neste trabalho e no de (CAMINHA; MOREIRA; SILVA, 2015). Ao final deste trabalho podemos afirmar que todos os objetivos foram alcançados.

Como trabalhos futuros, pretende-se estender o estudo de modo que consiga abranger mais bases de outros anos, verificando se são selecionados os mesmos atributos em todos os anos. Além disto, pretende-se aplicar a clusterização evolutiva para verificar se a adesão ao SiSU influenciou na melhora do rendimento na prova do ENEM, e identificar se existem alterações nos perfis ao longo dos anos.

Referências

CAMINHA, H. D.; MOREIRA, N. L.; SILVA, T. L. C. da. Detecção e análise dos perfis de inscritos do enem via mineração de dados. *VIII Congresso Tecnológico*, 2015. No prelo.

CASTRO, M. H. G. de; TIEZZI, S. A reforma do ensino médio e a implantação do enem no brasil1. *Desafios*, v. 65, n. 11, p. 46–115, 2004.

DASH, M.; LIU, H. Feature selection for classification. *Intelligent data analysis*, IOS Press, v. 1, n. 3, p. 131–156, 1997.

FREITAS, A. A. A survey of evolutionary algorithms for data mining and knowledge discovery. In: *Advances in evolutionary computing*. [S.l.]: Springer, 2003. p. 819–845.

GOLDBERG, D. E. *Genetic algorithms in search, optimization and machine learning*. [S.l.]: Addison-Wesley, 1989. ISBN 0201157675.

HAN, J.; KAMBER, M. *Data mining: concepts and techniques. Copyright by Morgan Kaufmann Publishers*. [S.l.]: Inc, 2001.

JAIN, A. K. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, Elsevier, v. 31, n. 8, p. 651–666, 2010.

JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data clustering: a review. *ACM computing surveys (CSUR)*, Acm, v. 31, n. 3, p. 264–323, 1999.

JORGE, M. J. Comparação de técnicas de seleção de atributos para previsão de insolvência de empresas brasileiras no período 2005-2007. *Anais do Encontro da ANPAD*, n. 34, 2010.

KAREGOWDA, A. G.; MANJUNATH, A.; JAYARAM, M. Comparative study of attribute selection using gain ratio and correlation based feature selection. *International Journal of Information Technology and Knowledge Management*, v. 2, n. 2, p. 271–277, 2010.

LIMA, R. F.; PEREIRA, A. C. M. Modelos computacionais baseados em feature selection e undersampling para detecção de fraudes eletrônicas. *BRAZILIAN SYMPOSIUM ON DATABASES*, v. 30, p. 87–92, 2015.

LIU, H.; MOTODA, H. Feature extraction, construction and selection: A data mining perspective. Kluwer Academic Publishers, 1998.

MENDES, L. *Data Mining – Estudo de Técnicas e Aplicações na Área Bancária*. Dissertação (Monografia) — FACULDADE DE TECNOLOGIA DE SÃO PAULO, São Paulo, 2011.

MOLINA, L. C.; BELANCHE, L.; NEBOT, À. Feature selection algorithms: a survey and experimental evaluation. In: IEEE. *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*. [S.l.], 2002. p. 306–313.

QUINLAN, R. *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann Publishers, 1993.

RIBEIRO, L. d. S. et al. Uma abordagem semântica para seleção de atributos no processo de kdd. Universidade Federal da Paraíba, 2010.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. *Introdução ao DATAMINING Mineração de Dados*. [S.l.]: Editora Ciência Moderna Ltda., 2009. ISBN 917-85-7393-761-9.

WITTEN, I. H.; FRANK, E. *Data Mining: Practical machine learning tools and techniques*. [S.l.]: Elsevier, 2011.

Apêndices

APÊNDICE A – Resultados obtidos utilizando a abordagem *filter*

Neste apêndice serão apresentados os resultados obtidos na etapa de seleção de atributos utilizando a abordagem *filter*. Para estes resultados foi utilizado como avaliador de subconjunto o *CFsSubsetEval*, sendo utilizado três algoritmos de busca, *geneticSearch*, *RankerSearch* e *bestFirst*.

Testes utilizando como algoritmo de busca o *geneticSearch*

A seguir encontra-se o primeiro teste realizado com a base do ENEM de 2010 utilizando como algoritmo de busca o *geneticSearch*.

Antes de executar o *geneticSearch* foi utilizado o algoritmo de classificação C4.4 implementado pelo weka como J4.8. Observe o resultado gerado por ele a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-
           weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===
```

```
J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 65.77 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances     3105938               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

O resultado a seguir foi obtido a partir da execução da seleção de atributos com *geneticSearch*.

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval
Search:weka.attributeSelection.GeneticSearch -Z 20 -G 20 -C 0.6 -M 0.033 -R 20 -S 1
Relation: enem2010-weka.filters.unsupervised.attribute.Remove-R1-
         weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
```

```
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Evaluation mode:10-fold cross-validation

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

number of folds (%)  attribute
          10(100 %)    1 qtdPessoasMoramComVoceQ1
           0(  0 %)    2 escolaridadePaiQ2
           0(  0 %)    3 escolaridadeMaeQ3
           0(  0 %)    4 rendaFamiliarTotalQ4
           0(  0 %)    5 rendaDoInscritoQ5
           0(  0 %)    6 casaOndeMoraQ6
           0(  0 %)    7 localizacaoCasaQ7
           0(  0 %)    8 jaTrabalhouQ8
           0(  0 %)    9 testarConhecimentoQ24
           0(  0 %)   10 prosseguirEstudosQ25
          10(100 %)   11 obterCertficadoOuAcelerarEstudosQ26
           0(  0 %)   12 conseguirBolsaDeEstudosQ27
           0(  0 %)   13 tempoDeConclusaoFundamenQ28
           0(  0 %)   14 parouDeEstudarNoFundamentalQ29
           0(  0 %)   15 tipoEscolaFundamentalQ30
           0(  0 %)   16 tempoDeConclusaoEnsMedioQ31
           0(  0 %)   17 parouDeEstudarNoEnsMedioQ32
          10(100 %)   18 tipoEscolaEnsMedioQ33
          10(100 %)   19 media
```

Note que foram selecionados apenas quatro atributos: qtdPessoasMoramComVoceQ1, obterCertficadoOuAcelerarEstudosQ26, tipoEscolaEnsMedioQ33 e media

Após a seleção de atributos foi executada novamente o algoritmo J4.8. Observe a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-
           weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0-weka.filters.unsupervised.attribute.Remove
Instances:    3105938
```

```
Attributes:   5
              qtdPessoasMoramComVoceQ1
              obterCertficadoOuAcelerarEstudosQ26
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 13.59 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances     3105938               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

Teste utilizando como algoritmo de busca o *rankerSearch*

A seguir encontra-se o segundo teste realizado utilizando como algoritmo de busca o *rankerSearch*.

Antes de executar o *rankerSearch* foi utilizado o algoritmo de classificação C4.4 implementado pelo weka como J4.8. Observe o resultado gerado por ele a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-
           weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 65.77 seconds

=== Stratified cross-validation ===
=== Summary ===
```

```
Correctly Classified Instances     3105938              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

O resultado a seguir foi obtido a partir da execução da seleção de atributos com *ranker-Search*.

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval
Search:weka.attributeSelection.RankSearch -S 1 -R 0 -A weka.attributeSelection.GainRatioAttributeEval --
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-
          weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
```

```
              media
              classificacaoDaMedia
Evaluation mode:10-fold cross-validation



=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

number of folds (%)  attribute
          10(100 %)    1 qtdPessoasMoramComVoceQ1
           0(  0 %)    2 escolaridadePaiQ2
           0(  0 %)    3 escolaridadeMaeQ3
           0(  0 %)    4 rendaFamiliarTotalQ4
           0(  0 %)    5 rendaDoInscritoQ5
           0(  0 %)    6 casaOndeMoraQ6
           0(  0 %)    7 localizacaoCasaQ7
           0(  0 %)    8 jaTrabalhouQ8
           0(  0 %)    9 testarConhecimentoQ24
           0(  0 %)   10 prosseguirEstudosQ25
          10(100 %)   11 obterCertficadoOuAcelerarEstudosQ26
           0(  0 %)   12 conseguirBolsaDeEstudosQ27
           0(  0 %)   13 tempoDeConclusaoFundamenQ28
           0(  0 %)   14 parouDeEstudarNoFundamentalQ29
           0(  0 %)   15 tipoEscolaFundamentalQ30
           0(  0 %)   16 tempoDeConclusaoEnsMedioQ31
           0(  0 %)   17 parouDeEstudarNoEnsMedioQ32
          10(100 %)   18 tipoEscolaEnsMedioQ33
          10(100 %)   19 media
```

Note que foram selecionados os mesmos atributos do algoritmo anterior: qtdPessoasMoramComVoceQ1, obterCertficadoOuAcelerarEstudosQ26, tipoEscolaEnsMedioQ33 e media

Após a seleção de atributos foi executada novamente o algoritmo J4.8. Observe a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-
           weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0-weka.filters.unsupervised.attribute.Remove
Instances:    3105938
Attributes:   5
              qtdPessoasMoramComVoceQ1
              obterCertficadoOuAcelerarEstudosQ26
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
```

```
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 13.59 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances     3105938               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

Teste utilizando como algoritmo de busca o *bestFirst*

A seguir encontra-se o segundo teste da abordagem *wrapper* realizado utilizando como algoritmo de busca o *bestFirst*.

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval
Search:weka.attributeSelection.BestFirst -D 1 -N 5
Relation:     enem2010-weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0-
             weka.filters.unsupervised.attribute.Remove-R1
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
```

```
                rendaFamiliarTotalQ4
                rendaDoInscritoQ5
                casaOndeMoraQ6
                localizacaoCasaQ7
                jaTrabalhouQ8
                testarConhecimentoQ24
                prosseguirEstudosQ25
                obterCertficadoOuAcelerarEstudosQ26
                conseguirBolsaDeEstudosQ27
                tempoDeConclusaoFundamenQ28
                parouDeEstudarNoFundamentalQ29
                tipoEscolaFundamentalQ30
                tempoDeConclusaoEnsMedioQ31
                parouDeEstudarNoEnsMedioQ32
                tipoEscolaEnsMedioQ33
                media
                classificacaoDaMedia
Evaluation mode:10-fold cross-validation

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

number of folds (%)  attribute
          10(100 %)    1 qtdPessoasMoramComVoceQ1
           0(  0 %)    2 escolaridadePaiQ2
           0(  0 %)    3 escolaridadeMaeQ3
           0(  0 %)    4 rendaFamiliarTotalQ4
           0(  0 %)    5 rendaDoInscritoQ5
           0(  0 %)    6 casaOndeMoraQ6
           0(  0 %)    7 localizacaoCasaQ7
           0(  0 %)    8 jaTrabalhouQ8
           0(  0 %)    9 testarConhecimentoQ24
           0(  0 %)   10 prosseguirEstudosQ25
          10(100 %)   11 obterCertficadoOuAcelerarEstudosQ26
           0(  0 %)   12 conseguirBolsaDeEstudosQ27
           0(  0 %)   13 tempoDeConclusaoFundamenQ28
           0(  0 %)   14 parouDeEstudarNoFundamentalQ29
           0(  0 %)   15 tipoEscolaFundamentalQ30
           0(  0 %)   16 tempoDeConclusaoEnsMedioQ31
           0(  0 %)   17 parouDeEstudarNoEnsMedioQ32
          10(100 %)   18 tipoEscolaEnsMedioQ33
          10(100 %)   19 media
```

Note que foram selecionados os mesmos atributos selecionados pelos algoritmos anteriores: qtdPessoasMoramComVoceQ1, obterCertficadoOuAcelerarEstudosQ26, tipoEscolaEnsMedioQ33 e media

APÊNDICE B – Resultados obtidos utilizando a abordagem *wrapper*

A seguir encontra-se o primeiro teste realizado com a abordagem *wrapper*, utilizando como algoritmo de busca o *geneticSearch*. Antes de executar o *geneticSearch* foi utilizado o algoritmo de classificação C4.4 implementado pelo weka como J4.8. Observe o resultado gerado por ele a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-
       weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5
```

```
Time taken to build model: 65.77 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances     3105938              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

A seguir o resultado obtido a partir da seleção de atributos.

```
=== Run information ===

Evaluator:    weka.attributeSelection.WrapperSubsetEval -B weka.classifiers.trees.J48 -F 5 -T 0.01 -R 1 -- -C
Search:weka.attributeSelection.GeneticSearch -Z 20 -G 20 -C 0.6 -M 0.033 -R 20 -S 1
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Norm
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
```

```
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Evaluation mode:10-fold cross-validation

=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

number of folds (%)  attribute
           0(  0 %)    1 qtdPessoasMoramComVoceQ1
           0(  0 %)    2 escolaridadePaiQ2
           0(  0 %)    3 escolaridadeMaeQ3
           0(  0 %)    4 rendaFamiliarTotalQ4
          10(100 %)    5 rendaDoInscritoQ5
           0(  0 %)    6 casaOndeMoraQ6
           0(  0 %)    7 localizacaoCasaQ7
           0(  0 %)    8 jaTrabalhouQ8
           0(  0 %)    9 testarConhecimentoQ24
           0(  0 %)   10 prosseguirEstudosQ25
           0(  0 %)   11 obterCertficadoOuAcelerarEstudosQ26
           0(  0 %)   12 conseguirBolsaDeEstudosQ27
           0(  0 %)   13 tempoDeConclusaoFundamenQ28
           0(  0 %)   14 parouDeEstudarNoFundamentalQ29
           0(  0 %)   15 tipoEscolaFundamentalQ30
           0(  0 %)   16 tempoDeConclusaoEnsMedioQ31
           0(  0 %)   17 parouDeEstudarNoEnsMedioQ32
           0(  0 %)   18 tipoEscolaEnsMedioQ33
          10(100 %)   19 media
```

Note que foram selecionados apenas dois atributos, rendaDoInscritoQ5 e media, estes atributos são diferentes dos selecionados com a abordagem anterior.

Após a seleção de atributos foi executada novamente o algoritmo J4.8. Observe a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Norm
    -weka.filters.unsupervised.attribute.Remove-R1,3-4,6-18-weka.filters.unsupervised.attribute.Remove-R1
Instances:    3105938
Attributes:   3
              rendaDoInscritoQ5
              media
              classificacaoDaMedia
Test mode:evaluate on training data

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)
```

```
Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 6.61 seconds

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances     3105938              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

Teste utilizando como algoritmo de busca o *rankerSearch*

A seguir encontra-se o segundo da abordagem *wrapper* teste realizado, utilizando como algoritmo de busca o *rankerSearch*.

Antes de executar o *rankerSearch* foi utilizado o algoritmo de classificação C4.4 implementado pelo weka como J4.8. Observe o resultado gerado por ele a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-
       weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
```

```
               escolaridadeMaeQ3
               rendaFamiliarTotalQ4
               rendaDoInscritoQ5
               casaOndeMoraQ6
               localizacaoCasaQ7
               jaTrabalhouQ8
               testarConhecimentoQ24
               prosseguirEstudosQ25
               obterCertficadoOuAcelerarEstudosQ26
               conseguirBolsaDeEstudosQ27
               tempoDeConclusaoFundamenQ28
               parouDeEstudarNoFundamentalQ29
               tipoEscolaFundamentalQ30
               tempoDeConclusaoEnsMedioQ31
               parouDeEstudarNoEnsMedioQ32
               tipoEscolaEnsMedioQ33
               media
               classificacaoDaMedia
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 65.77 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances     3105938               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1
```

```
=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

A seguir o resultado obtido a partir da seleção de atributos.

```
=== Run information ===

Evaluator:    weka.attributeSelection.WrapperSubsetEval -B
            weka.classifiers.meta.ClassificationViaClustering -F 5 -T 0.01 -R 1 -- -W
            weka.clusterers.SimpleKMeans -- -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10
Search:weka.attributeSelection.RankSearch -S 1 -R 0 -A weka.attributeSelection.GainRatioAttributeEval --
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Norm
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Evaluation mode:evaluate on all training data



=== Attribute Selection on all input data ===

Search Method:
RankSearch :
Attribute evaluator : weka.attributeSelection.GainRatioAttributeEval
Attribute ranking :
 19 media
 15 tipoEscolaFundamentalQ30
 18 tipoEscolaEnsMedioQ33
  4 rendaFamiliarTotalQ4
 12 conseguirBolsaDeEstudosQ27
  2 escolaridadePaiQ2
  3 escolaridadeMaeQ3
```

```
 14 parouDeEstudarNoFundamentalQ29
 13 tempoDeConclusaoFundamenQ28
  5 rendaDoInscritoQ5
 11 obterCertficadoOuAcelerarEstudosQ26
 17 parouDeEstudarNoEnsMedioQ32
  7 localizacaoCasaQ7
  8 jaTrabalhouQ8
  1 qtdPessoasMoramComVoceQ1
 16 tempoDeConclusaoEnsMedioQ31
 10 prosseguirEstudosQ25
  9 testarConhecimentoQ24
  6 casaOndeMoraQ6
Merit of best subset found : 0.744

Attribute Subset Evaluator (supervised, Class (nominal): 20 classificacaoDaMedia):
Wrapper Subset Evaluator
Learning scheme: weka.classifiers.meta.ClassificationViaClustering
Scheme options: -W weka.clusterers.SimpleKMeans -- -N 2 -A weka.core.EuclideanDistance -R first-last -I 500 -
Subset evaluation: classification accuracy
Number of folds for accuracy estimation: 5

Selected attributes: 19 : 1
                     media
```

Note que foram selecionados apenas um atributo, media, este atributo é diferente dos selecionados com algoritmo anterior.

Após a seleção de atributos foi executada novamente o algoritmo J4.8. Observe a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Norm
      weka.filters.unsupervised.attribute.Remove-R1,3-4,6-18-weka.filters.unsupervised.attribute.Remove-R1-we
Instances:    3105938
Attributes:   2
              media
              classificacaoDaMedia
Test mode:evaluate on training data

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 5.26 seconds
```

```
=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances     3105938              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

Teste utilizando como algoritmo de busca o *bestFirst*

A seguir encontra-se o terceiro teste realizado com a abordagem *wrapper*, utilizando como algoritmo de busca o *bestFirst*.

Antes de executar o *bestFirst* foi utilizado o algoritmo de classificação C4.4 implementado pelo weka como J4.8. Observe o resultado gerado por ele a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-
       weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
```

```
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Test mode:10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 65.77 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances     3105938               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

A seguir o resultado obtido a partir da seleção de atributos.

```
=== Run information ===

Evaluator:    weka.attributeSelection.WrapperSubsetEval -B weka.classifiers.meta.ClassificationViaClustering
Search:weka.attributeSelection.BestFirst -D 1 -N 5
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Norm
Instances:    3105938
Attributes:   20
              qtdPessoasMoramComVoceQ1
              escolaridadePaiQ2
              escolaridadeMaeQ3
              rendaFamiliarTotalQ4
              rendaDoInscritoQ5
              casaOndeMoraQ6
              localizacaoCasaQ7
              jaTrabalhouQ8
              testarConhecimentoQ24
              prosseguirEstudosQ25
              obterCertficadoOuAcelerarEstudosQ26
              conseguirBolsaDeEstudosQ27
              tempoDeConclusaoFundamenQ28
              parouDeEstudarNoFundamentalQ29
              tipoEscolaFundamentalQ30
              tempoDeConclusaoEnsMedioQ31
              parouDeEstudarNoEnsMedioQ32
              tipoEscolaEnsMedioQ33
              media
              classificacaoDaMedia
Evaluation mode:evaluate on all training data

=== Attribute Selection on all input data ===

Search Method:
Best first.
Start set: no attributes
Search direction: forward
Stale search after 5 node expansions
Total number of subsets evaluated: 119
Merit of best subset found:    0.822

Attribute Subset Evaluator (supervised, Class (nominal): 20 classificacaoDaMedia):
Wrapper Subset Evaluator
Learning scheme: weka.classifiers.meta.ClassificationViaClustering
Scheme options: -W weka.clusterers.SimpleKMeans -- -N 2 -A weka.core.EuclideanDistance -R first-last -I 500 -
Subset evaluation: classification accuracy
Number of folds for accuracy estimation: 5

Selected attributes: 7,19 : 2
                     localizacaoCasaQ7
                     media
```

Note que foram selecionados dois atributo,localizacaoCasaQ7 e media, estes atributos são diferentes dos selecionados com algoritmos anteriores .

Após a seleção de atributos foi executada novamente o algoritmo J4.8. Observe a seguir.

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     enem2010-weka.filters.unsupervised.attribute.Remove-R1-7,9-19-
            weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0
Instances:    3105938
Attributes:   3
              localizacaoCasaQ7
              media
              classificacaoDaMedia
Test mode:evaluate on training data

=== Classifier model (full training set) ===

J48 pruned tree
------------------

media <= 0.363934: baixa (1142390.0)
media > 0.363934
|   media <= 0.622287: intermediaria (1738039.0)
|   media > 0.622287: alta (225509.0)

Number of Leaves  :  3

Size of the tree :  5


Time taken to build model: 6.57 seconds

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances     3105938              100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances          3105938

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        baixa
                 1         0          1         1         1          1        intermediaria
                 1         0          1         1         1          1        alta
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

       a       b       c   <-- classified as
 1142390       0       0 |       a = baixa
       0 1738039       0 |       b = intermediaria
       0       0  225509 |       c = alta
```

Realizando uma comparação nos resultados obtidos entre as duas abordagens é possível concluir que a abordagem *filter* obteve melhores resultados.