Capítulo I

* 1. **INTRODUÇÃO**

As escolas têm armazenado grande massa de dados, obtidas através dos rastreamentos de notas, frequência, grade escolar, compras de livros texto, dados sócios econômicos dos pais, preocupação com a saúde (física e mental) dos alunos, a escolaridade dos pais, dados governamentais e similares. Mas pouco tem sido feito com essas informações - devido a questões de privacidade ou capacidade técnica - para melhorar a aprendizagem dos alunos.

No entanto, com a adoção de mais tecnologias nas escolas e com o acesso mais fácil aos dados governamentais, há claramente uma maior oportunidade para uma melhor coleta de dados e análises desses dados em prol da educação. Infelizmente, muitas escolas e universidades fazem muito pouco com essa riqueza de dados, uma ou outra que possivelmente produz um relatório anual de seus perfis. Mesmo uma simples análise dos dados institucionais poderia levantar perfis dos alunos para um potencial atendimento padronizado ou indicar um atendimento individual ao aluno.

Segundo Rezende (2003), os avanços em hardware e software tem permitido que os computadores tenham aplicações em áreas não convencionais. Como por exemplo, os Sistemas Inteligentes, utilizam a tecnologia da informação para manipular conhecimentos especializados com benefícios qualitativo e quantitativo. Permitindo desta forma que, um maior número de pessoas tenha acesso ao conhecimento a partir da aquisição, sistematização, representação e processamento desse conhecimento.

Os Sistemas Inteligentes podem manipular símbolos que representam entidades do mundo real, e dessa forma, são capazes de trabalhar eficazmente com conhecimento.

A Inteligência Artificial, tem em suas pesquisas, o objetivo de capacitar o computador a executar funções que os seres humanos desempenham usando conhecimento e raciocínio. Então, para que possamos aspirar à ação inteligente, é preciso analisar todos os aspectos relativos ao desenvolvimento e uso da inteligência (Rezende, 2003). Dentro deste contexto, torna-se evidente que a incorporação de conhecimento é um requisito fundamental para a construção de sistemas computacionais inteligentes

Neste trabalho pretende-se desenvolver uma solução tecnológica que permita ao professor fazer o acompanhamento e a avaliação do aluno a partir das interações destes com um ambiente de ensino. O foco principal do trabalho é usar o processo de descoberta de conhecimento em bases de dados, também conhecido *Knowledge Discovery in Databases* (**KDD**), afim de observar, a viabilidade e, aplicabilidade de um caso real de apoio a decisão. O estudo inclui também o uso de tecnologias de análise e recuperação de dados úteis ao processo decisório, conhecidas como **OLAP** e, da aplicação de técnicas e algoritmos de Mineração de Dados (em Inglês *Data Mining*) para descoberta de novos conhecimentos e padrões nos dados. Este conhecimento adquirido poderá então ser usado na melhoria da educação.

* 1. **OBJETIVOS**
     1. **Objetivos Gerais**

Nesta pesquisa pretende-se desenvolver uma ferramenta para auxiliar o professor ou os gestores a diagnosticar problemas relacionados ao ensino aprendizagem, tais como: alto índices de evasão e repetências escolar e o mau desempenho dos alunos nas respectivas disciplinas e sugerir soluções para os problemas encontrados. Para isto, será desenvolvida uma solução tecnológica que permita aos gestores da educação no Instituto Federal de Educação, Ciências e Tecnologia do Rio Grande do Norte – **IFRN,** detectar através de relatórios e gráficos, a real situação dos alunos, em relação a evasão e a repetência escolar. Além disso, o professor poderá através desta ferramenta acompanhar o desempenho dos alunos, diagnosticando aqueles alunos que estão em situação de risco e, dessa forma, orientar esses alunos em tempo hábil na tentativa de recuperar o desempenho dos mesmos, sugerindo a esses alunos, atividades engajadoras (baseadas em ***Gamification***), tais como jogos, e também um sistema de aconselhamento ao aluno. Esta pesquisa tem também como foco, o uso de técnicas de Mineração de Dados, para descobrir relações entre os atributos da base de dados, que indique o perfil dos alunos evadidos ou retidos nos diversos cursos ministrados pelo instituto.

* + 1. **Objetivos Específicos**
* Coletar os dados dos diversos sistema de informação (acadêmico, formulários de pesquisas do aluno, do professor e da instituição);
* Modelar uma estrutura multidimensional;
* Extração, transformação e carga dos dados para o sistema multidimensional;
* Criar um *Data Warehouse* para consolidar os dados extraídos das diversas fontes de dados disponíveis;
* Aplicar as técnicas de Mineração de Dados para traçar o perfil da evasão e da repetência escolar no IFRN;
* Aplicar as técnicas da Mineração de Dados para fazer previsões futuras sobre a evasão e a repetência escolar no IFRN;
* Interpretar os resultados obtidos;
* Desenvolver o Sistema de Aconselhamento ao aluno (Caso Baseado em Raciocínio – em Inglês *Case-Based Racionary* – **CRB**);
* Desenvolver um Visualizador (protótipo) de dados que será utilizado pelo usuário final.
  1. **MOTIVAÇÃO**

Desde 2006, a rede federal de Educação Profissional, Científica e Tecnológica (Rede Federal), tem passado por uma expansão e a sua interiorização. Essa expansão tem proporcionado uma expansão física e a democratização da oferta de vagas. Com isso, o desafio cresce no sentido de manter a qualidade do ensino, o atendimento à diversidade, a permanência e o êxito dos estudantes no processo educativo.

Nesta direção, o Ministério da Educação (MEC), por meio da Secretaria de Educação Profissional e Tecnológica (SETEC), elaborou um plano de ação, para o entendimento e o combate à evasão e a repetência escolar na Rede Federal de Ensino. Este plano é composto por sete dimensões que abrangem: o entendimento dos fenômenos da evasão e retenção e medidas para o seu combate, a formação de parcerias, o desenvolvimento de pesquisa e inovação, a inserção profissional dos estudantes, a distribuição de cargos e funções às instituições, a capacitação dos servidores e a avaliação de cursos de educação profissional e tecnológica.

Portanto, a motivação deste trabalho é atacar de perto um problema que está em pauta na Rede Federal de Educação, no sentido de entender os fenômenos de evasão e retenção/repetência na Rede Federal e à adoção de medidas para o seu combate.

* 1. **METODOLOGIA**

Esta sessão tem como finalidade descrever o modo como será desenvolvida esta pesquisa, em relação aos objetivos definidos. Primeiramente será feita uma pesquisa bibliográfica e, em seguida o desenvolvimento do projeto, que compreende, a coleta de dados, construção do modelo multidimensionais do sistema, extração, transformações e carga dos dados das fontes de dados de origem para o modelo multidimensional, aplica algoritmos de Mineração de Dados para descoberta de novos conhecimentos, analisar o conhecimento obtido e finalmente desenvolver a ferramenta de visualização de dados.

* 1. **ESTRUTURAÇÃO DA TESE**

Esta pesquisa está estruturada em cinco capítulos, introdução, referencial teórico, desenvolvimento do projeto, conclusões e trabalhos futuros e referencial bibliográfico, que serão descriminados a seguir.

O primeiro capítulo trata de uma introdução, onde é dada uma visão ampla da pesquisa, os objetivos que se almeja alcançar, a motivação que deu incentivo ao desenvolvimento dessa pesquisa e a metodologia a ser aplicada no desenvolvimento da pesquisa. No segundo capítulo é feita uma pesquisa bibliográfica, onde será pesquisado o estado da arte em relação aos conceitos utilizados na pesquisa em curso. No terceiro capítulo será desenvolvido e implantado o projeto, desde a coleta de dados até o sistema de visualização de dados para o usuário final. No capítulo quarto serão apresentadas as conclusões e os trabalhos futuros relacionados e finalmente, no capítulo 5, temos o referencial bibliográfico e bibliografia consultada.

Capítulo 2

2. **Fundamentação Teórica**

2.1. **Business Intelligence**

Segundo Rob (2011), o termo *Business intelligence* (**BI**) é utilizado para descrever um conjunto amplo, coeso e integrado de ferramentas e processos utilizados para captar, coletar, integrar, armazenar e analisar dados para a geração e a apresentação de informações que deem suporte à tomada de decisões de negócio. Como o próprio nome diz, **BI** trata da criação de inteligência sobre o negócio. Portanto, o **BI**, é um modelo que permite à empresa transformar dado em informação, informação em conhecimento e conhecimento em sabedoria.

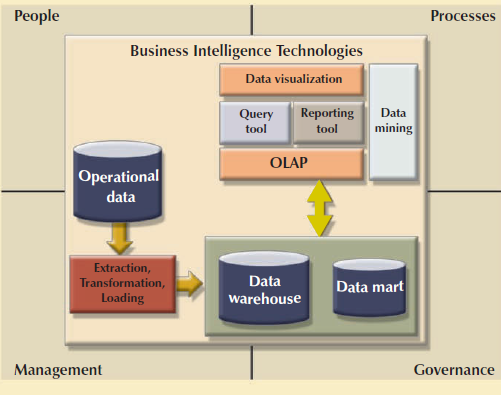
O **BI** não é, por si só, um produto, mas um modelo de conceitos, práticas, ferramentas e tecnologias (*data warehouse*, *data mart*, **OLAP** e/ou ferramentas de mineração de dados) que auxiliam uma empresa a compreender melhor seus recursos centrais, identificando oportunidades fundamentais para criar competitividade (Rob, 2011). Em geral, o **BI** envolve as seguintes etapas:

* Coleta e armazenamento de dados operacionais.
* Agregação de dados operacionais em dados de suporte a decisões.
* Análise de dados de suporte a decisões para gerar informações.
* Apresentação dessas informações ao usuário final para dar suporte a decisões de negócios.
* Tomada de decisões de negócio, o que, por sua vez, gera mais dados que são coletados, armazenados etc. (reiniciando o processo).
* Monitoramento para avaliar os resultados das decisões de negócio (Rob, 2011).

**2.1.1. Arquitetura de Business Intelligence**

Segundo Rob (2011), o **BI** utiliza-se de tecnologias e aplicações para o gerenciamento de todo o ciclo de vida dos dados, da aquisição ao armazenamento, transformação, integração, análise, monitoramento e apresentação. Não existe uma arquitetura única de **BI**, no entanto, há alguns tipos gerais de recursos, que são compartilhados por todas as implementações de **BI**.

Uma arquitetura de BI deve ser composta de dados, pessoas, processos, tecnologias e gerenciamento desses componentes. A Figura 2.1 ilustra um modelo de BI.



**Figura 2.1**. Modelo de *business intelligence*. **Fonte**: Rob (2011).

Para compreender a arquitetura de **BI**, será feita uma descrição dos componentes básicos que fazem parte de sua infraestrutura. Alguns desses componentes, possuem recursos adicionais. Porém, há quatro componentes básicos que todos os ambientes de **BI** devem fornecer, descritos as seguir (Rob, 2011):

* **Extração, transformação e carregamento (ETL) de dados**: esse componente é encarregado de coletar, filtrar, integrar e agregar dados operacionais a serem salvos em um armazém de dados otimizado para o suporte a decisões.
* **Armazenamento de dados (Data warehouse)**: o armazém de dados é otimizado para o suporte a decisões e costuma ser representado por um *data warehouse* ou data mart. Ele contém dados de negócios extraídos de bancos de dados operacionais e de fontes externas. Esses dados são armazenados em estruturas otimizadas, com foco na velocidade de análise e consulta.
* **Processamento analítico online (OLAP)**: esse componente executa as tarefas de recuperação, análise e mineração, utilizando os dados no armazém de dados e os modelos de análise de dados de negócio. Tal componente é utilizado pelo analista de dados para criar as consultas que acessam o banco de dados. Essa ferramenta orienta o usuário sobre quais dados selecionar e como construir um modelo de dados confiáveis.
* **Ferramentas de apresentação e visualização de dados**: esse componente é encarregado de apresentar os dados ao usuário final de várias formas. É utilizado pelo analista de dados para organizar e apesentar os dados. Essa ferramenta ajuda o usuário final a selecionar o formato de apresentação mais adequado, como relatório resumido, mapa ou gráfico.
* **Mineração de dados** (em inglês **Data Mining**):

**2.1.2. DATA WAREHOUSE**

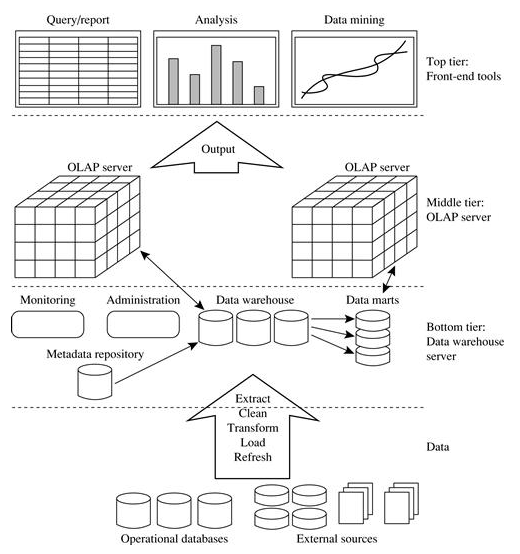
Segundo Inmon (1994), o termo ***data warehouse*** é “um conjunto de dados integrado, orientado por assunto, variável no tempo e não volátil, que fornece suporte a tomada de decisões”.

* **Integrado**. O *data warehouse* é um banco de dados consolidado e centralizado, que integra dados proveniente de toda a organização e de várias fontes de dados, com diversos formatos.
* **Orientado por assunto**. Os dados do *data warehouse* são dispostos e otimizados de modo a fornecerem respostas a perguntas provenientes de diversas áreas funcionais da empresa. São organizados e resumidos por temas, contendo assuntos de interesse especifico – produtos, clientes, departamentos, regiões, promoções, e assim por diante.
* **Variável no tempo**. Os dados são carregados periodicamente no *data warehouse*, e quando isso acontece, todas as agregações dependentes do tempo, são recalculadas. Por exemplo, se os dados de vendas da semana, são carregados no *data warehouse*, serão atualizadas todas as agregações dependentes dessa carga.
* **Não volátil**. Uma vez inserido um dado no *data warehouse*, ele nunca será removido. Uma vez que ele representa o histórico da empresa.

Resumindo, o *data warehouse* é um repositório de dados semanticamente consistente, que serve como uma implementação física de um modelo de dados de apoio a decisões. Ele armazena as informações que uma empresa necessita para tomar decisões (Han & Kamber, 2011). Normalmente é um banco de dados apenas de leitura, otimizado para processamento de análises e consultas. Em geral, os dados são extraídos de diversas fontes e, em seguida, transformados e integrados, antes de serem carregados no *data warehouse* (Inmon, 1994).

**2.1.2.1. Arquitetura de Data Warehouse**

Segundo Han & Kamber (2011), um **data warehouse** adota uma arquitetura em três camadas, como ilustra a Figura 2.2.



**Figura 2.2**. Uma arquitetura de data warehouse em três camadas.

**Fonte**: Han & Kamber (2011, 3nd)

1. **A camada inferior** é um servidor de *data warehouse*, que quase sempre, é um sistema de banco de dados relacional. Segundo Han & Kamber (2011), são usadas ferramentas de back-end e utilitários para extrair dados dessa camada e alimentar a camada superior.
2. **A camada intermediária**, segundo Han & Kamber (2011), é um servidor OLAP que geralmente é implementado usando (1) um modelo relacional **OLAP** (**ROLAP**) (fornece recursos de OLAP utilizando bancos de dados relacionais e ferramentas familiares de consulta relacional para armazenar dados multidimensionais; ou (2) um modelo multidimensional **OLAP** (**MOLAP**) (amplia os recursos de OLAP para sistemas de gerenciamento de banco de dados multidimensionais (**SGBDM**s).
3. **A terceira camada**, segundo Han & Kamber (2011), é o fronte-end do cliente, a qual contém as ferramentas de consulta, de relatório, de análise e mineração de dados (por exemplo, análise de tendência, previsão, e assim por diante.).

Segundo Han & Kamber, do ponto de vista da arquitetura, há três modelos de data warehouse: o warehouse empresarial, o data mart e warehouse virtual.

**2.1.2.1. Data Warehouse Empresarial**

Um data warehouse empresarial, segundo Han & Kamber (2011), coleta todas as informações sobre todos os assuntos da organização. Ele fornece informações de dados de toda empresa, geralmente, oriundas de um ou mais sistemas operacionais ou informações obtidas externamente. Ele contém dados detalhados e/ou sumarizados, e pode variar em tamanho de poucos gigabytes para centenas de gigabytes, terabytes ou superiores. A sua implementação pode ser em um mainframe, supercomputador, em uma plataforma de arquitetura paralela. Requer uma modelagem comercial extensiva e pode levar muitos anos para ser projetado e construído.

**2.1.2.2. Data Mart**

De acordo com Inmon (1994), embora o *data warehouse*, seja uma proposta muito atraente, que traga muitos benefícios, os gerentes podem relutar em adotar essa estratégia, pelo fato de que, a criação de um *data warehouse* exige tempo, dinheiro e considerável esforço gerencial. Estes fatos, fazem com que muitas empresas iniciem na criação de *data warehouse*, focando em conjuntos de dados gerenciais, orientados a atender pequenas áreas de negócio, dentro da empresa. Esses armazenamentos menores são chamados de ***data marts***. Um ***data mart*** é portanto, segundo Inmon (1994), um pequeno subconjunto de um *data warehouse*, sobre um único assunto, que fornece suporte às decisões de um pequeno grupo de pessoas. No entanto, pode-se criar um *data mart* a partir de dados extraídos de um *data warehouse*, com a finalidade especifica de dar suporte a um acesso mais rápido a determinado grupo ou função. Dessa forma, os *data marts* e o *data warehouse* podem coexistir em um ambiente de business *intelligence*.

**2.1.2.3. Virtual Data Warehouse**

De acordo com Han & Kamber (2011), um warehouse virtual é um conjunto de visões sobre bases de dados operacionais. Você pode materializar algumas visões operacionais, para obter um processamento de consultas eficientes. O warehouse virtual é o estado de visibilidade global de recursos, com base na aquisição e processamento de dados operacionais em tempo real. Informações disponíveis no armazém virtual tem o potencial de reduzir custos e melhorar o serviço ao cliente. A infraestrutura já está disponível para captura de dados em tempo real, e o custo de aquisição de dados continuará a reduzir.

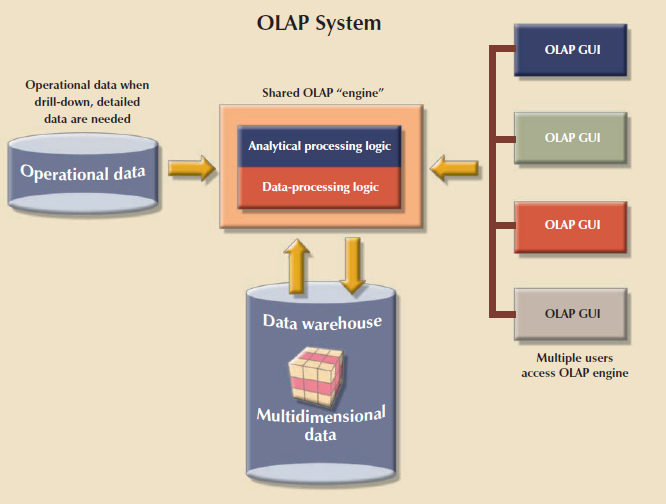
**2.1.3. Processamento Analítico On-Line**

De acordo com Rob (2011), a necessidade de suporte a decisões mais intensivo, levou à introdução de uma nova geração de ferramentas. Tais ferramentas, foram denominadas de **processamento analítico on-line** (**OLAP** – *Online Analytical Processing*). Essa nova ferramenta cria um ambiente avançado de análise de dados que dá suporte à tomada de decisões, modelagem comercial e pesquisa operacional. Ainda segundo Rob (2011), esses sistemas comportam quatro características principais:

* **Utilizam técnicas de análise de dados multidimensionais**: De acordo com Rob (2011), a característica mais evidente das modernas ferramentas **OLAP**, é a capacidade de análise multidimensional, onde, os dados são processados e visualizados como parte de uma estrutura multidimensional. Essas técnicas de análise de dados multidimensionais, utilizam as seguintes funções: apresentação de dados em gráfico 3F, pivô, tabulações cruzadas, cubos dimensionais, e assim por diante. Além da capacidade de utilizar funções financeiras, estatísticas e de previsão.
* **Proporcionam suporte avançado a bancos de dados**: As ferramentas **OLAP**, para apresentar suporte eficiente a decisões, deve ter recursos avançados de acesso a dados. Tais recursos incluem (Rob, 2011): acesso a fontes de dados variadas, recursos de *drill down* e *roll up* e particionamento de bases de dados.
* **Fornecem interface fácil de utilizar para o usuário final**: diversas ferramentas de geração de relatório, planilhas e de visualização de dados, fornecem acesso aos cubos **OLAP,** facilitando, dessa forma a interação do usuário final, inclusive com interfaces gráficas fáceis de utilizar.
* **Dão suporte a arquitetura cliente/servidor**: Um ambiente cliente/servidor possibilita que um sistema OLAP seja dividido em vários componentes que definem sua arquitetura. Esses componentes podem, então, ser colocados no mesmo computador ou distribuídos entre diversas máquinas. Assim, segundo Rob (2011), o OLAP é projetado para atender a exigências de facilidades de utilização, ao mesmo tempo em que mantém a flexibilidade do sistema.

**2.1.3.1. Arquitetura OLAP**

Segundo Rob (2011), os sistemas OLAP são projetados para utilizar, tanto dados operacionais, quanto **data warehouse**. Para isso, existem várias arquiteturas de instalação de sistema OLAP, baseados nas regras de negócio de cada empresa. O fato é que, um sistema OLAP pode acessar ambos tipos de armazenamento de dados (operacional ou *data warehouse*) ou apenas um, dependendo da implementação que se deseje configurar. Em todo caso, a análise multidimensional de dados exige algum tipo de representação de dados multidimensionais, o que normalmente é fornecido pelo mecanismo OLAP. A Figura 2.3 ilustra uma arquitetura OLAP com organização de armazenagem de dados multidimensionais.



**Figura 2.3** Servidor OLAP com organização de armazenagem de dados multidimensionais. **Fonte**: Rob (2011).

De acordo com Rob (2011), há diversas formas de gerenciar e armazenar os dados em um sistema OLAP, como já foi visto anteriormente, o OLAP relacional (**ROLAP**) e o OLAP multidimensional (**MOLAP**):

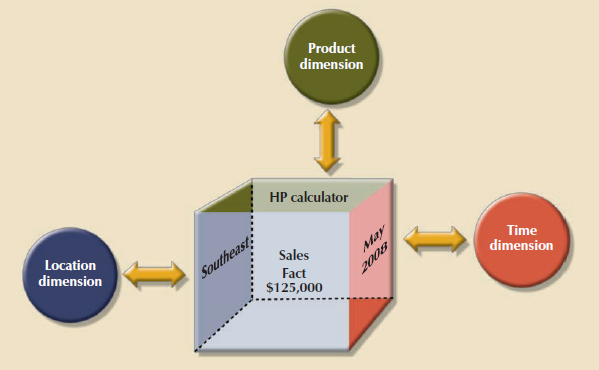
* **O processamento analítico on-line relacional** (**ROLAP**, sigla em inglês para *Relational Online Analytical Processing*): Essa abordagem se estrutura a partir de tecnologias relacionais existentes e representa uma extensão natural para todas as empresas que já utilizem sistemas de gerenciamento de banco de dados relacionais. O ROLAP utiliza uma técnica especial de projeto que permite à tecnologia SGBDR dar suporte a representações de dados multidimensionais, conhecida como “Esquema estrela”.
* **O processamento analítico on-line multidimensional** (**MOLAP**, sigla em inglês para Multidimensional *Online Analytical Processing*): O MOLAP amplia os recursos de OLAP para sistemas de gerenciamento de banco de dados multidimensionais (**SGBDMs**). O pressuposto do MOLAP é que os bancos de dados multidimensionais são os mais adequados para gerenciar, armazenar e analisar dados multidimensionais. Nessa arquitetura os usuários visualizam os dados armazenados como um **cubo de dados**.

**2.1.3.2 Modelagem Multidimensional**

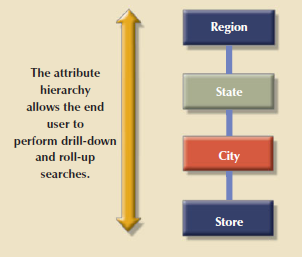
A modelagem multidimensional é uma forma de Modelagem de Dados voltada para concepção e visualização de conjunto de medidas que descrevem aspectos comuns de um determinado assunto. É utilizada especialmente para sumarizar e reestruturar dados, apresentando-os em visões que suportem a análise dos dados envolvidos (Passos & Goldschimdt, 2005).

De acordo com Han & Kamber (2011), o *data warehouse* e as ferramentas **OLAP** são baseadas em um **modelo de dados multidimensional**. Nesses modelos, os dados são vistos na forma de um cubo de dados. Um modelo multidimensional possui três componentes básicos: Fatos (***facts tables***), Dimensões (***dimensions***) e Medidas (***measures***). E existem diversas formas de modelagem física de um data warehouse, incluindo esquema estrela (***star schema***), esquema floco de neves (***snowflake***) e constelação de fatos (***fact constellation***). Estes conceitos serão discutidos a seguir:

* **Esquema Estrela:**  O esquema estrela, segundo Rob (2011), é uma técnica de modelagem de dados multidimensionais de suporte a decisões em um banco de dados relacional. O esquema estrela básico possui quatro componentes: **fatos**, **dimensões**, **atributos** e **hierarquias de atributos**.
  + **Fatos:** Um **fato** é uma coleção de itens de dados, composta de dados de medidas e de contexto. Representa um item, ou uma transação ou um evento associado ao tema da modelagem. São medidas numéricas (valores) que representam um aspecto ou atividade específica dos negócios. Os fatos são armazenados em tabelas de fatos que constituem o centro do esquema estrela. A **tabela de fatos** (***fact table***) contém fatos vinculados por meio de suas dimensões (Kimball, 2002, Passos & Goldschimdt, 2005).
  + **Dimensões:** Uma dimensão é um tipo de informação que participa da definição de um fato. As dimensões determinam o contexto do assunto. As **dimensões** são características de qualificação que fornecem perspectivas adicionais a um determinado fato. Essas dimensões são armazenadas em **tabelas de dimensões**. A Figura 2.4 ilustra um esquema estrela para vendas com as dimensões de produto, localização e tempo.
  + **Medidas:** Uma medida é um atributo ou variável numérica que representa um fato. Exemplos: valor da ação, número de evasões escolares, quantidade de produtos vendidos, valor total de venda, e assim por diante.
  + **Atributos:** De acordo com Kimball (2002), cada tabela de dimensão contém atributos. Os atributos costumam ser utilizados para buscar, filtrar e classificar fatos. As dimensões fornecem características descritivas sobre os fatos por meio de seus atributos.
  + **Hierarquias de Atributos:** De acordo com Kimball (2002), os atributos no interior de dimensões podem ser ordenados em hierarquias bem definidas. A hierarquia de atributos, fornecem uma organização vertical utilizada para duas finalidades principais: agregação e análise de dados por ***drill down*** e ***roll up***. A Figura 2.5 ilustra como os atributos da dimensão de localização podem ser organizados em uma hierarquia por região, estado, cidade e loja.

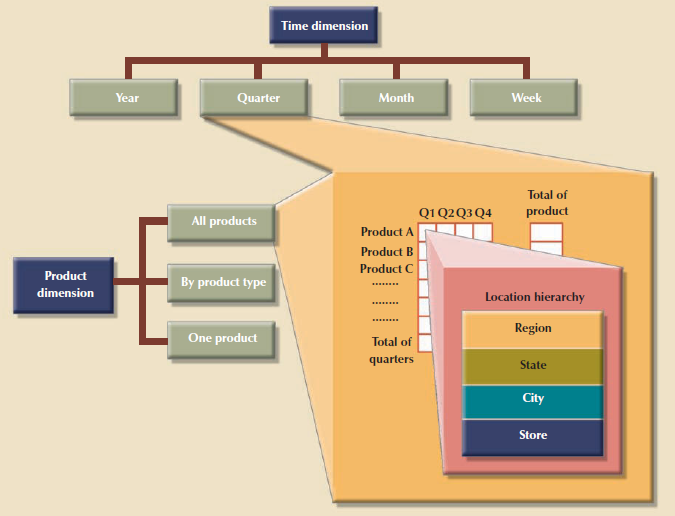


**Figura 2.4** **Esquema estrela**. **Fonte:** Rob (2011)



**Figura 2.5** Hierarquia de atributo de localização. **Fonte**: Rob (2011).

De acordo com Rob (2011), a hierarquia de atributos fornece a possibilidade de executar buscas de *drill down* e *roll up* no *data warehouse*. A Figura 2.6 ilustra um cenário em que o analista de dados estuda fatos de vendas utilizando as dimensões de produto, tempo e localização. Nesse exemplo, a dimensão de produto é estabelecida como “Todos os produtos”, o que significa que o analista de dados verá todos os produtos no eixo y. a dimensão de tempo (eixo x) é estabelecida com “Trimestre”, indicando que os dados estão agregados por trimestre (por exemplo, vendas totais dos produtos A, B e C em T1, T2, T3 e T4). Finalmente, a dimensão de localização é inicialmente estabelecida para “Região”, garantindo, assim, que cada célula contenha as vendas regionais totais de determinado produto em determinado trimestre.



**Figura 2.6** Hierarquias de atributos em análise multidimensional. **Fonte**: Rob (2011).

* **Esquema Floco de Neves:** Segundo Rob (2011), para facilitar a navegação do usuário final, utiliza-se a técnica de normalização das tabelas dimensionais. Esse esquema normalizado é conhecido como esquema floco de neves.

Resumindo, projetar um *data warehouse* significa receber a oportunidade de ajudar a desenvolver um modelo integrado que capture os dados considerados essenciais para a organização, tanto da perspectiva do usuário final, como da perspectiva dos negócios. Para tanto, um projeto de *data warehouse*, deve satisfazer:

* Critérios de integração e carregamento de dados.
* Recursos de análises de dados com desempenho aceitável de consulta.
* Necessidades de análises de dados do usuário final

Segundo Rob (2011), a preocupação técnica mais evidente na implementação de um *data warehouse* é fornecer ao usuário final suporte a decisões com recursos avançados de análise de dados – no momento certo, no formato certo, com os dados certos e ao custo certo.

**2.1.4. Conclusões**

Nessa seção foi apresentada uma visão geral sobre *Business Intellingence* (**BI**), onde deu para perceber que, o BI é um conjunto amplo, coeso e integrado de ferramentas e processos utilizados para captar, coletar, integrar, armazenar e analisar dados para a geração e a apresentação de informações que deem suporte à tomada de decisões para os gestores da empresa.

2.2. **O processo de descoberta de conhecimento em bases de dados**

2.2.1. **Introdução**

Os constantes avanços na área da tecnologia da informação têm viabilizado o armazenamento de grandes e múltiplas bases de dados. Tecnologias como a Internet, sistemas gerenciadores de banco de dados, dispositivos de armazenamento de dados de maior capacidade e de menor custo e sistemas de informação em geral são alguns dos exemplos que têm viabilizado a proliferação de inúmeras bases de dados de natureza comercial, administrativa, governamental e cientifica (Han & Kamber & Pei, 2011).

Pesquisas científicas, tais como missões espaciais da NASA, monitoramento temporal, em redes sociais, como Google, Facebook, e assim por diante, tem manipulado grandes massas de dados, muitas vezes, têm alcançado proporções gigantescas, na ordem de zetabytes e ou petabytes de informações.

Diante desse cenário, a análise de grandes quantidades de dados pelo homem é inviável sem o auxílio de ferramentas computacionais apropriadas. Portanto, torna-se imprescindível o desenvolvimento de ferramentas que auxiliem o homem, de forma automática e inteligente, na tarefa de analisar, interpretar e relacionar esses dados para que se possa desenvolver estratégias de ação em cada contexto de aplicação (Han; Kamber & Pei, 2011).

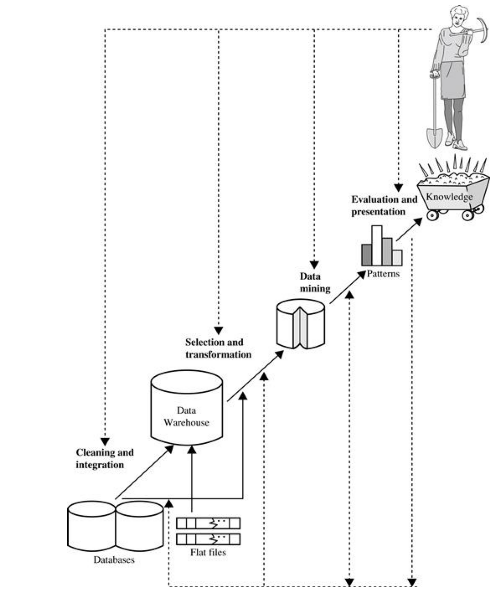
Então, para atender a este novo contexto, surge segundo Han; Kamber & Pei (2011), uma nova área denominada a mineração de dados (em inglês ***Data Mining -* DM**), que vem despertando grande interesse junto às comunidades cientificas e industrial. No entanto, a mineração de dados é apenas uma fase em um processo maior chamado de Processo de descoberta de em bases de dados (em inglês **Knowledge Discovery in Databases – KDD).**

2.2.2. **Caracterização do Processo de KDD**

Basicamente, uma aplicação de **KDD** é composta por três tipos de componentes: o problema em que será aplicado o processo de **KDD**, os recursos disponíveis para a solução do problema e os resultados obtidos a partir da aplicação dos recursos disponíveis em busca da solução do problema (Passos & Goldschmidt, 2005).

1. **O problema a ser submetido ao processo de KDD:** Este componente pode ser caracterizado por três elementos: conjunto de dados, o especialista do domínio da aplicação e objetivos da aplicação.
2. **Os recursos disponíveis para solução do problema em questão** - Entre eles podem ser destacados: o especialista em KDD, as ferramentas de KDD e plataforma computacional disponível (Passos & Goldschmidt, 2005).
3. **Os resultados obtidos a partir da aplicação dos recursos no problema** - Compreende, fundamentalmente, os modelos de conhecimento descobertos ao longo da aplicação de KDD e o histórico das ações realizadas (Passos & Goldschmidt, 2005).

Segundo Han et al. (2011); Witten & Frank (2005); Elmasri (2005), o processo de KDD é mostrado na Figura 2.7 e consiste da sequência iterativa dos seguintes passos: Seleção de dados, Limpeza dos dados, Enriquecimento, Transformação de dados, Data Mining, Avaliação dos padrões e representação do conhecimento.



**Figura 2.7** Mineração de Dados como um passo no processo de Descoberta de Conhecimento.

**Fonte**: Han; Kamber & Pei (2011).

* **Seleção dos dados:** Essa função, também denominada Redução de Dados, compreende, em essência, a identificação de quais informações, dentre as bases de dados existentes, devem ser efetivamente consideradas durante o processo de KDD. A seleção dos dados pode ter dois enfoques distintos: a escolha de atributos ou a escolha de registros que devem ser considerados no processo de KDD (Passos & Goldschmidt, 2005).
* **Limpeza dos Dados:** Abrange qualquer tratamento realizado sobre os dados selecionados de forma a assegurar a qualidade (completude, veracidade e integridade) dos fatos por eles representados. Informações ausentes ou inconsistentes nas bases de dados devem ser corrigidas de forma a não comprometer a qualidade dos modelos de conhecimento a serem extraídos ao final do processo de KDD (Passos & Goldschmidt, 2005).
* **Transformação dos Dados:** Segundo Passos & Goldschmidt (2005), codificação de dados é a operação de pré-processamento responsável pela forma como os dados serão representados durante o processo de KDD. A maneira como a informação é codificada tem forte influência sobre o tipo de conhecimento a ser encontrado. Em essência, a codificação pode ser: Numérica – Categórica, que transforma valores reais em categorias ou intervalos; Categórica – Numérica, que representa numericamente valores de atributos categóricos.
* **Enriquecimento dos dados:** De acordo com Passos & Goldschmidt (2005), a função de enriquecimento consiste em conseguir de alguma forma mais informações que possa ser empregada aos registros existentes, enriquecendo os dados, para que esses forneçam mais informações para o processo de descoberta de conhecimento. A seguir será comentada algumas das operações mais usualmente utilizadas no processo de enriquecimento das bases de dados.
* **Mineração de Dados:** De acordo com Han; Kamber & Pei (2011); Passos & Goldschmidt (2005), a execução da etapa de Mineração de Dados (Data Mining) compreende a aplicação de algoritmos sobre os dados procurando abstrair conhecimento. Estes algoritmos são fundamentados em técnicas que procuram, segundo determinados paradigmas, explorar os dados de forma a produzir modelos de conhecimento.
* **Avaliação dos padrões**: Segundo Liu & Hsu (1996), a obtenção do conhecimento não é o passo final do processo de Extração de Conhecimento de Bases de dados. O conhecimento extraído pode ser utilizado na resolução de problemas da vida real, seja por meio de um Sistema Inteligente ou de um ser humano como apoio a algum processo de tomada de decisão. Para isso é importante que algumas questões sejam respondidas aos usuários:
  + O conhecimento extraído representa o conhecimento do especialista?
  + De que maneira o conhecimento do especialista difere do conhecimento extraído?
  + Em que parte o conhecimento do especialista está correto?

2.2.3. **Conclusões**

Nessa seção foi mostrado de uma forma sucinta, o processo de descoberta de conhecimento em bases de dados e, na seção a seguir, será detalhado o processo de mineração de dados. Como foi mostrado, o processo de mineração de dados é uma das fases do processo maior, o processo KDD.

2.3. **Mineração de dados**

2.3.1. **Definição**

Existem, segundo Han; Kamber & Pei (2011), várias definições para Mineração de dados (em inglês **Data Mining** – **DM**), por ser um assunto verdadeiramente interdisciplinar, ela envolve um extenso campo de pesquisa, que associa técnicas e conceitos de diversas áreas como sistemas de banco de dados, sistemas baseados em conhecimento, inteligência artificial, aprendizado de máquina, aquisição do conhecimento, estatística, bancos de dados espaciais e visualização de dados. As várias tarefas desenvolvidas em Mineração de Dados têm como objetivos primário a predição e/ou a descrição. A predição usa atributos para predizer valores futuros de uma ou mais variáveis (atributos) de interesse. A descrição contempla o que foi descoberto nos dados de vista da interpretação humana.

Durante a etapa de Mineração de Dados é realizada a busca efetiva por conhecimentos úteis no contexto da aplicação de KDD. É por tanto, na Mineração de Dados, onde são definidas, as técnicas e os algoritmos a serem utilizados no problema em questão. A escolha da técnica depende, muitas vezes, do tipo de tarefa de KDD a ser realizada. São muitas as tarefas de Mineração de dados, dentre elas as tarefas de **Associação**, **Classificação**, **Árvores de Decisão** e **Agrupamento**.

De acordo com Han; Kamber & Pei (2011); Passos & Goldschmidt (2005), a execução da etapa de Mineração de Dados (Data Mining) compreende a aplicação de algoritmos sobre os dados procurando abstrair conhecimento. Estes algoritmos são fundamentados em técnicas que procuram, segundo determinados paradigmas, explorar os dados de forma a produzir modelos de conhecimento.

2.3.2. **Tarefas de Mineração de Dados**

As metas primárias que podem ser alcançadas através da Mineração de Dados, são as seguintes (Fayyad, et al, 1996):

* **Previsão** - Nesse caso busca-se um modelo de conhecimento que permita, a partir de um histórico de casos anteriores, prever os valores de determinados atributos em novas situações.
* **Descrição** - Nesse caso busca-se por um modelo que descreva, de forma compreensível pelo homem, o conhecimento existente em um conjunto de dados.

Ainda, segundo Fayyad, et al (1996), a mineração preditiva consiste na generalização de exemplos ou experiências passadas com respostas conhecidas ou regras de negócios estabelecidas por especialistas. Enquanto, a mineração descritiva, consiste na identificação de comportamentos intrínsecos do conjunto de dados, sendo que estes dados não possuem uma classe especifica.

2.3.2.1. **Descoberta de Associações**

De uma forma geral, a tarefa clássica de busca por regras de associação, também denominada regras associativas, foi introduzida em (Agrawal et al., 1993). Intuitivamente essa tarefa consiste em encontrar conjunto de registros de itens que ocorram simultaneamente e de forma frequente em um banco de dados.

Uma **regra de associação** é uma implicação da forma X => Y, onde X = {x1, x2, ..., xn} e Y = {y1, y2, ..., ym} são conjunto de itens, com xi e yj sendo itens distintos para todo i e todo j. Essa associação estabelece que se um cliente comprar X, ele também estará propenso a comprar Y. Para que uma regra de associação seja de interesse para a mineração de dados, a regra precisa satisfazer algumas medidas. Duas medidas de interesse comuns fornecem suporte e confiança. Segue a seguir, as formulas para cálculo do fator de suporte e de confiança (Agrawal, et al., 1996; Elmasri, 2005).

**, onde N é o número total de tuplas. (2.1)**

**(2.2)**

O fator de **suporte** pode ser descrito como a probabilidade de uma transação qualquer satisfazer tanto X como Y, ao passo que o fator de **confiança** é a probabilidade de que uma transação satisfaça Y, dado que ela satisfaça X. a tarefa de descobrir regras de associação consiste em extrair do banco de dados todas as regras com “Fs” e “Fc” maiores ou iguais a um “Fs” e “Fc” especificado pelo analista de dados (Agrawal, et al., 1996).

Uma associação é considerada frequente se o número de vezes em que a união de conjuntos de itens (x ocorrer em relação ao número total de transações do banco de dados for superior a uma frequência mínima (denominada suporte mínimo) que é estabelecida em cada aplicação. Busca-se por meio do suporte, identificar que associações surgem em uma quantidade expressiva a ponto de ser destacada das demais existentes (Agrawal, et al., 1996).

Ainda segundo Agrawal et al. (1996), uma associação é considerada válida se o número de vezes em que X Y ocorrer em relação ao número de vezes que X ocorrer for superior a um valor denominado confiança mínima, que também é estabelecida em cada aplicação. A medida de confiança procura expressar a qualidade de uma regra, indicando o quanto a ocorrência do antecedente da regra pode assegurar a ocorrência do consequente desta regra.

Desta forma, a tarefa de Descoberta de Associações ou Descoberta de Regras de Associações, pode ser definida formalmente como a busca por **regras de associação frequentes e válidas** em um banco de dados, a partir da especificação dos parâmetros de suporte e confiança mínima (Agrawal et al., 1996).

Os valores dos parâmetros de suporte e confiança mínima devem ser especificados pelo especialista em KDD em conjunto com o especialista no domínio da aplicação (Agrawal et al., 1996).

Segundo Agrawal et al. (1996), existem diversos algoritmos desenvolvidos especificamente para aplicação na tarefa de descoberta de associações, dentre eles: Apriori, DHP (Direct Hashing and Pruning), Parttion, DIC (Dynamic ItemSet Counting), Eclat, MaxEclat, Clique, MaxClique, etc. além dos mais, existem versões destes algoritmos para sistemas distribuídos.

2.3.2.2. **Classificação**

Essa tarefa pode ser compreendida como a busca por uma função que permita associar corretamente cada entrada **X**i de um conjunto de dados de entrada, a um único rótulo **Y**i, denominado classe. Uma vez identificada, essa função pode ser aplicada a novas entradas de forma a prever a classe em que tais dados se enquadram. Dados podem ser associados a classes ou a conceitos através de um processo de discriminação ou caracterização (Han; Kamber & Pei, 2011).

De acordo com Han; Kamber & Pei (2011), a tarefa de classificação é um processo de dois passos. No primeiro passo, constrói-se um modelo com base nos dados. No segundo passo, determina se a acurácia desse modelo é aceitável, se assim for, usa-se esse modelo para classificar novos dados.

Para formalizar a tarefa de classificação, consideremos um par ordenado da forma **(x, ƒ(x))**, onde **x** é um vetor de entradas n-dimensional e **ƒ(x)** a saída de uma função **ƒ**, desconhecida, aplicada a **x**. A tarefa de inferência indutiva consiste em, dada uma coleção de exemplos de **ƒ*,*** obter uma função ***h*** que se aproxime de **ƒ**. A função ***h*** é chamada de hipótese ou modelo de ***ƒ*** (Passos, Goldschmidt, 2005).

Nos casos em que a imagem de **ƒ** é formada por rótulos de classes, a tarefa de inferência indutiva é denominada classificação e toda hipótese ***h*** chamada de classificador. A identificação da função ***h*** consiste em um processo de busca no espaço de hipóteses **H**, pela função que mais se aproxime da função original **ƒ**. Esse processo é denominado **aprendizado** (Russel e Norving, 1955, citado por Passos & Goldschmidth, 2005). Todo algoritmo que possa ser utilizado nesse processo é denominado de **algoritmo de aprendizado**. O conjunto de todas as hipóteses que podem ser obtidas a partir de um algoritmo de aprendizado **L** é representado por **H**L. Cada hipótese pertencente ao **H**L, é representado por ***h***L.

A acurácia de um classificador em um dado conjunto de teste é a percentagem do conjunto de tuplas de testes que são corretamente classificadas pelo classificador Han; Kamber & Pei (2011), ou seja, a acurácia da hipótese ***h*** em mapear corretamente cada vetor de entradas x em **ƒ(x)**. O conjunto de pares **(x, ƒ(x))** utilizado na identificação da função ***h*** é denominada conjunto de treinamento. Por outro lado, o conjunto de pares **(x, ƒ(x))** utilizados para avaliar a acurácia de ***h*** é denominado conjunto de testes. Dessa forma, o algoritmo **L** pode ser interpretado como uma função tal que:

(2.3)

Onde,

T é o espaço composto por todos os conjuntos de treinamento possíveis para **L**.

Segundo Utgolff (1996), cada algoritmo possui *bias* indutivo que direciona o processo de construção dos classificadores. O *bias* indutivo de um algoritmo, pode ser definido como o conjunto de fatores que, coletivamente influenciam na seleção de hipótese.

Em termos práticos, o *bias* de um algoritmo de aprendizado **L** afeta o processo de aprendizado de duas formas: restringem o tamanho de espaço de hipóteses **HL**, e impõem uma ordem de preferência sobres as hipóteses em **HL** (Bensusan, 1999, citado por Utgolff, 1996).

Conforme mencionado anteriormente, uma medida de desempenho de um classificador comumente utilizada é a acurácia (Acc(*h*)), também conhecida como precisão do classificador.

**Acc(*h*) = 1 – Err(*h*)** (2.4)

Onde,

Err(h) é denominada taxa de erro ou taxa de classificação incorreta:

(2.5)

Onde,

O operador ||E|| retorna 1 se a expressão E for verdadeira e 0, caso contrário;

**n** é o número de exemplos (registros da base de dados, por exemplo);

**yi** é a classe real associada ai i-ésimo exemplo;

***h*(i)** é a classe indicada pelo classificador para o i-ésimo exemplo.

O modelo derivado pode ser apresentado de várias formas, tais como regras de classificação (***IF-THEN***), árvores de decisão, fórmulas matemáticas, ou redes neurais. Existem muitos outros métodos de construção de modelos de classificação, tais como classificação ***naïve Bayesian***, máquina de vetor de suporte (***support vector machines***), e classificação do vizinho mais próximo (***k-nearest neighbor***), Backpropagation, classificação usando Padrões frequentes, etc (Han; Kamber & Pei, 2011).

2.3.2.3. **Agrupamento (Clustering)**

Técnicas de agrupamento e classificação objetivam realizar uma separação ótima entre objetos de uma coleção, permitindo a descoberta de novos padrões, previamente desconhecidos. O resultado da segmentação, independentemente da ferramenta utilizada, pode ser interpretado eficientemente por um especialista na área de origem dos dados sob análise. A facilidade de visualização resultante do agrupamento, favorece a análise (Han; Kamber & Pei, 2011).

A tarefa de agrupamento, é usada para segmentar os dados de entrada em subconjuntos ou clusters, de tal forma que elementos de um cluster compartilhem um conjunto de propriedades comuns que os distingam dos elementos de outros clusters. O objetivo desta tarefa é maximizar similaridades intra-cluster e minimizar similaridades inter-cluster. Diferente da classificação que tem rótulos predefinidos, o agrupamento precisa automaticamente identificar os rótulos. Por essa razão, o agrupamento também é chamado de indução não supervisionada (Passos & Goldschmidt, 2005).

Segundo Han; Kamber & Pei (2011), em geral, a classe não é representada nos dados de treinamento, simplesmente porque elas não são conhecidas no começo. Entretanto, os agrupamentos (*clusters*) podem ser usados para gerar tais rótulos. Os objetos são agrupados baseados no princípio de máxima similaridade intra-classe e mínima similaridade inter-classe. Isto é, o agrupamento de objetos, é formado, de modo que os objetos dentro do agrupamento, tenham alta similaridade em comparação com um outro objeto, mas são muitos diferentes dos objetos em outro agrupamento. Cada agrupamento formado pode ser visto como uma classe de objetos, da qual pode derivar regras.

Formalmente para o processo de agrupamento, supõe-se a existência de **n** pontos de dados **x1**, **x2**, ..., **xn** tais que cada ponto pertença a um espaço d dimensional **Rd**. A tarefa de agrupamento desses pontos de dados, separando-os em **k** clusters consiste em encontrar **k** pontos **mj** em **Rd** de tal forma que a expressão (Passos & Goldschmidt, 2005):

(2.6)

Seja minimizada, onde **d2(xi, mj)** denota uma distância entre **xi** e **mj**. Os pontos **mj** são denominados centroides ou médias dos clusters.

De forma resumida, o problema descrito acima consiste em encontrar k centroides de clusters de tal maneira que a distância entre cada ponto de dado e o centroide do cluster mais próximo seja minimizada (Passos & Goldschmidt, 2005).

2.3.3. **Métodos de Mineração de Dados**

De acordo com Passos & Goldschmidt (2005), cada método de Mineração de Dados requer diferentes necessidades de pré-processamento. Tais necessidades variam em função do aspecto extensional da base de dados em que o método será utilizado. Em decorrência da diversidade de métodos de pré-processamento de dados, são muitas as alternativas possíveis de combinações entre métodos. A escolha dentre estas alternativas pode influenciar na qualidade do resultado do processo de KDD (Morik, 2000; Engels, 1996; Engels et al., 1997).

Os métodos de Mineração de Dados, sendo um caso particular de um método KDD, podem ser considerados operadores definidos a partir de precondições e efeitos. Uma precondição de um método de KDD é um predicado que estabelece um requisito que deve ser cumprido antes da execução do método. Um efeito de um método de KDD também é um predicado que descreve uma situação gerada após a aplicação do método. Um plano de ação de KDD válido é toda sequência de métodos de KDD onde as precondições para execução de cada um dos métodos da sequência, sejam devidamente atendidas Passos & Goldschmidt (2005).

Dentre os métodos de Mineração de dados, pode-se citar os baseados em redes neurais, os estatísticos, os métodos específicos Apriori, indução de árvore de decisão, lógica nebulosa, hierárquicos e os métodos baseados em densidade.

* **Métodos Baseados em Redes Neurais:** Segundo Han; Kember & Pei (2011); Passos & Goldschmidt (2005); Rezende (2003), diversos modelos de Redes Neurais podem ser utilizados na implementação de métodos de Mineração de Dados. Classificação, Regressão, Previsão de Séries Temporais e Agrupamento (Clusters) são exemplos de tarefas de Mineração de Dados que podem ser implementadas por métodos de Redes Neurais. Além do mais, alguns modelos de Redes Neurais podem ser aplicados em mais de um tipo de tarefa de Mineração de dados. Alguns dos algoritmos de aprendizado indicados na tarefa de Mineração de dados são:
  + **Back-Propagation:** O algoritmo Back-Propagation, também conhecido como algoritmo de retro-propagação do erro, é um algoritmo de aprendizado supervisionado, cuja aplicação é adequada a tarefa de Mineração de Dados tais como Classificação, Regressão ou Previsão. Esse algoritmo tem como objetivo minimizar a função de erro entre a saída gerada pela rede neural e a saída real desejada, utilizando o método do gradiente descendente.
  + **Kohonen:** O mapa Kohonen pertence à classe das Redes Neurais Auto organizáveis. Em uma Rede Neural Auto organizável o treinamento é não supervisionado, geralmente baseado em uma forma de competição entre os elementos processados. Entre as principais aplicações das Redes Auto organizáveis estão:
    - **Tarefas de Clusterização –** Tarefa na qual os dados de entrada devem ser agrupados em conjuntos que agregam padrões semelhantes.
    - **Detecção de Regualaridades** – Modelo em que o sistema deve extrair as características relevantes dos padrões de entrada.
* **Métodos Estatísticos:** De acordo com Passos & Goldschmidt (2005); Han; Kamber & Pei (2011), diversos algoritmos de Mineração de Dados são fundamentados em princípios e teorias da Estatística. A seguir serão apresentados alguns deles:
  + **Classificador Bayeasiano Ingênuo:** O classificador Bayeasiano Ingênuo baseia-se no Teorema de Bayes, estando relacionado ao cálculo de probabilidades condicionais. É aplicável, conforme o próprio nome sugere, em tarefas de classificação.
  + **K-Means:** O algoritmo k-means é um método popular da tarefa de agrupamento. Toma-se, randomicamente, k pontos de dados (dados numéricos) como sendo os centroides (elementos centrais) dos clusters. Em seguida, cada ponto (ou registro da base de dados) é atribuído ao cluster cuja distância deste ponto em relação ao centroide de cada cluster é a menor dentre todas as distâncias calculadas. Um novo centroide para cada cluster é computado pela média dos pontos do cluster, caracterizando a configuração dos clusters para a iteração seguinte. O processo termina quando os centroides dos clusters param de se modificar, ou após um número limitado de iterações que tenham sidos especificados pelo usuário (Han, Kamber & Pei, 2011; Passos & Goldschmidt, 2005).
  + **K-Modes** – O algoritmo k-modes é uma variação do método k-means, só que utilizado para agrupamento de dados categóricos (nominais). Em geral, no lugar do cálculo da média, calcula-se a moda dos objetos, usando medidas de similaridades para tratar objetos categóricos, e usando métodos baseados em frequência para atualizar as modas dos clusters.
  + **K-Prototypes** – O método *k-prototypes* é a integração dos métodos *k-means* e *k-modes*. Esse método pode ser aplicado a bases de dados que contenham tanto atributos numéricos quanto atributos categóricos.
  + **K-Medoids** - O algoritmo k-medoids baseia-se, primeiramente, em encontrar os medoids (objetos mais centralmente localizado em um cluster). Os objetos restantes são então agrupados com o medoid ao qual ele é mais similar. Há então uma troca iterativa, de um medoid por um não medoid, visando à melhoria do agrupamento. O método então, é realizado baseado no princípio de minimizar a soma das dissimilaridades entre cada objeto **p** e seu correspondente objeto representativo.
* **Método Específico – Apriori:** O *Apriori* é um algoritmo clássico de Mineração de Regras de Associação (Agrawal, 1993). Diversos algoritmos tais como **GSP**, **DHP**, **Partition**, **DIC**, etc., foram inspirados no funcionamento do *Apriori* e se baseiam no princípio da antimonotonicidade do suporte. Segundo este princípio, “Um *k-itemset* somente pode ser frequente se todos os seus (k-1)-*itemsets* forem frequentes”. Assim sendo, a combinação de *itemsets* para gerar um novo *itemset* somente ocorre quando estes são frequentes.
* **Métodos Baseados em Indução de Árvores de Decisão:** Segundo Passos & Goldschmidt (2005), alguns dos principais métodos de Mineração de Dados são baseados na construção de árvores de decisão a partir da base de dados. Em geral a construção de uma árvore de decisão é realizada segundo alguma abordagem recursiva de particionamento da base de dados. Um exemplo clássico de método baseado na indução de árvores de decisão é o algoritmo C4.5.
  + **C4.5** – O C4.5, procura abstrair árvores de decisão a partir de uma abordagem recursiva de particionamento das bases de dados. Utiliza, para tanto, conceitos e medidas da Teoria da Informação.
* **Métodos Baseados em Lógica Nebulosa:** Diversos métodos de Mineração de dados foram adaptados de forma a incorporar a flexibilidade proporcionada pela Lógica Nebulosa. Entre eles pode-se citar as versões nebulosas K-Means e do C4.5. Nessas versões, os registros da base de dados podem pertencer a diversos clusters e classe simultaneamente, com diferentes graus de pertinência. Um exemplo de algoritmo nebuloso é o algoritmo de Wang-Mendel, concebido para aplicação na tarefa de Previsão de Séries Temporais (Passos & Goldschmidt, 2005).
* **Métodos Hierárquicos:** Agrupamento hierárquico produz uma fusão sequencial aninhada das variáveis em estudo, baseado em algumas métricas de correlação ou semelhança, como a correlação de Pearson ou a distância Euclidiana. A fusão aninhada é representada por um “dendrograma". No nível mais baixo do dendrograma, cada variável é um membro de um cluster (grupo) individual. No primeiro passo, as variáveis mais semelhantes são fundidas em um cluster. Em seguida, as próximas duas variáveis mais similares são unidas em outro cluster. Em cada passo, os dois grupos mais semelhantes (incluindo conjuntos unitários) são unidos para formar um grande grupo. No nível mais alto do dendrograma, existe um conjunto contendo todas as variáveis (Han; Kamber & Pei, 2011).
* **Métodos Baseados em Densidade:** Os métodos baseados em densidade permitem descobrir grupos de formatos arbitrários. Estes métodos consideram grupos como sendo regiões densas de objetos no espaço de dados separados por regiões de baixa densidade, que geralmente representam ruídos, onde esses ruídos estão incluídos nos clusters (Han; Kamber & Pei, 2011). Pode-se citar como exemplos de algoritmos baseados em densidade **DBSCAN**, **OPTICS** e **DENCLUE**, que serão discutidos em seguida.
  + **DBASCAN** - O algoritmo **DBSCAN** (*Density-based Spatial Clustering of Applications with Noise*) cria grupos de objetos em regiões de alta densidade e é capaz de descobrir grupos de formatos arbitrários em bases de dados espaciais e com ruído. Para o **DBSCAN** um cluster é definido como um conjunto máximo de pontos densamente conectados. Ele encontra clusters com formatos (***shape***) arbitrários em bancos de dados espaciais, contendo ruídos (*outliers*) (Han; Kamber & Pei, 2011).
  + **OPTICS** - Embora o **DBSACN** possa encontrar clusters dado os parâmetros de entrada tais como **ε-vizinhança** (o raio máximo de uma vizinhança) e MintPts (o número de pontos requeridos na vizinhança de um *core objects*), isso transfere para os usuários a responsabilidade de selecionar os valores dos parâmetros que irão levar a descoberta de clusters aceitáveis. Isto é um problema, que também estar associados a muitos outros algoritmos de clusters. A configuração de tais parâmetros, são usualmente definidos empiricamente, especialmente para parâmetros do mundo real, conjunto de dados de alta dimensionalidade. Além do mais, muitos algoritmos são sensíveis para esses valores de parâmetros: configurações, levemente diferentes pode levar a muitos diferentes agrupamentos de dados (Han; Kamber & Pei, 2011). Para superar a dificuldade em usar os parâmetros na análise de clusters, foi proposto um método chamado **OPTICS**.
  + **DENCLUE** (*DENsity-based CLUstEring*) - É um método de agrupamento baseado num conjunto de funções de distribuição de densidade. Primeiro é dado uma estimativa da densidade, e então descreve o algoritmo DENCLUE.

2.3.4. **Tecnologia de suporte a Mineração de Dados**

O processo de **KDD** é realizado incorporando-se várias técnicas de diferentes áreas como Aprendizado de Máquina, Data Warehousing, Banco de Dados, Estatísticas, Visualização de Dados, dentre outras. A etapa de Extração de Conhecimento, por exemplo, utiliza muitos recursos da área de Aprendizado de Máquina, e as demais são consideradas como áreas de apoio ao processo de KDD (Han; Kamber & Pei, 2011).

**2.3.4.1. Aprendizado de Máquina**

De acordo com Han; Kamber & Pei (2011), aprendizagem de máquina, investiga como os computadores pode aprender, com base em dados.

Segundo Rezende (2003), aprendizado de máquina, é uma área de Inteligência Artificial (**IA**), cujo objetivo, é o desenvolvimento de técnicas computacionais sobre o aprendizado, bem como a construção de sistemas capazes de adquirir conhecimento de forma automática.

A principal área de pesquisa está voltada para programas de computador, aprenderem automaticamente reconhecer padrões complexos, com base em dados, e a partir daí, tomar decisões inteligentes. Por exemplo, um problema típico de aprendizado de máquina, é programar um computador para que ele possa reconhecer automaticamente os códigos postais escritos, depois de aprender a partir com um conjunto de exemplos Han; Kamber & Pei (2011).

De acordo com Mitchell (1997), aprendizado de máquina, é uma subárea da Inteligência Artificial **(IA**), cujo objetivo é desenvolver métodos, técnicas e ferramentas para construir máquinas inteligentes, que se modificam para realizar cada vez melhor suas tarefas.

Para aprender, os sistemas, bem como os seres humanos, podem se valer de estratégias de aprendizado. A seguir, será apresentado, algumas dessas estratégicas clássicas de aprendizagem de máquina relacionado a mineração de dados (*data mining*) (Han; Kamber & Pei, 2011):

* **Aprendizado supervisionado:** é basicamente um sinônimo para classificação. No aprendizado supervisionado o objetivo é induzir conceitos a partir de exemplos que estão pré-classificados, ou seja, exemplos que estão rotulados com uma classe conhecida. Se as classes possuírem valores discretos, o problema é caracterizado como classificação. Caso as classes possuam valores contínuos, o problema é caracterizado, como regressão. Segundo (Bigus, 1999), o aprendizado supervisionado é usado para a realização de treinamento de redes neurais na obtenção de classificação, funções de aproximação ou modelagem e previsões baseadas no tempo.
* **Aprendizado não-supervisionado**: é essencialmente, um sinônimo para agrupamento (*cluster*). No aprendizado não-supervisionado, o indutor analisa os exemplos fornecidos e tenta determinar se alguns deles podem ser agrupados de alguma maneira, formando agrupamentos ou *clusters*. A tarefa do algoritmo é agrupara exemplos não rotulados, i.e., exemplos que não possuem o atributo classe especificado. Nesse caso, é possível utilizar algoritmos de aprendizado para descobrir padrões nos dados a partir de alguma caracterização de regularidade, sendo esses padrões denominados *clusters* (Decker & Focardi, 1995; McCallum, Nigan, & Ungar, 200). Tipicamente, pode-se usar agrupamento para descobrir classes dentro de dados (Han; Kamber & Pei, 2011).
* **Aprendizado semi**-**supervisionado**: o aprendizado semi-supervisionado, consiste em utilizar algoritmos que aprendam apartir de exemplos rotulados e não rotulados. Ou seja, o aprendizado semi-supervisionado, pode ser aplicável tanto em tarefas de classificação quanto em tarefas de agrupamento (Han; kamber & Pei, 2011).
* **Aprendizado ativo:** é uma abordagem de aprendizado de máquina, que permite que o usuário execute um papel ativo no processo de aprendizagem. Uma abordagem de aprendizado ativo, pode pedir ao usuário (i.e., o especialista do domínio) para rotular um exemplo, que pode ser a partir de um conjunto de exemplos não-rotulados, ou sintetizado pelo programa de aprendizagem. O objetivo é otimizar a qualidade do modelo, através de aquisição de conhecimento ativamente do usuário humano, a partir da quantidade exemplos a serem rotulados (Han; Kamber & Pei, 2011).

2.3.4.2. **Banco de dados** e **Data Warehousing**

Um Banco de dados é uma coleção integrada de dados, organizada de tal forma a facilitar o armazenamento eficiente, assim como sua modificação e recuperação (DATE, 2003). Um Sistema Gerenciador de Banco de Dados (SGBD) é uma coleção de procedimentos e mecanismos para recuperação, armazenamento e manipulação de dados.

O Data Warehousing, se refere ao processo de coleta de dados e pré-processamento dos dados armazenados em um ou mais banco de dados operacionais, com o objetivo de servir de fonte para os Sistema de Suporte a Decisão. O resultado desse processo é a criação de um Depósito de Dados (tradução da forma em inglês Data Warehouse), uma coleção de dados integrados, consolidados e possivelmente estruturado no tempo (dados históricos) (GOLDSCHMIDT, 2015).

No entanto, para o processo de Mineração de Dados não é necessário ser implementado um **DW**. No entanto, se a empresa já possui um DW, o tempo gasto na etapa de pré-processamento será reduzido drasticamente (Inmon, 1996).

2.3.4.3. **Estatística**

A estatística estuda as coleções, análise, interpretação ou explicação, e apresentação dos dados. A mineração de dados tem conexão própria com a estatísticas (Han; Kamber & Pei, 2011).

Segundo Han; Kamber & Pei (2011), um modelo estatístico, é um conjunto de funções matemáticas, que descrevem as ações dos objetos, em uma classe alvo, em termos de suas variáveis aleatórias e sua distribuição de probabilidades associadas. Ainda segundo autores, os modelos estatísticos são muito usados para modelar dados e classes de dados. Por exemplo, pode-se utilizar os modelos estatísticos para caracterização e classificação de dados em tarefas de Mineração de Dados. Em outras palavras, tais modelos estatísticos podem ser o resultado de uma tarefa de Mineração de Dados. Alternativamente, tarefas de Mineração de Dados podem ser construídas em cima dos modelos estatísticos.

A Estatística, junto com a área de Aprendizado de Máquina, é considerada ancestral da área de KDD. Técnicas de reconhecimento de padrões e de análises exploratória de dados provenientes da estatística são muito utilizados em algoritmos de Mineração de Dados (GOLDSCHMIDT, 2015). Seleção de dados e amostragem, pré-processamento, transformação dos dados e avaliação de padrões extraídos são apenas alguns exemplos de métodos há muito tempo utilizados em estatística e que são aplicados durante o processo de KDD (GOLDSCHMIDT, 2015).

Segundo Goldschmidt (2015), o conhecimento que se infere a partir dos dados tem um componente estatístico fundamental, que é o grau de certeza com o qual se espera que este conhecimento descreva ou faça predições sobre os dados. A estatística fornece uma linguagem para quantificação da incerteza resultante quando se tenta inferir padrões a partir de uma amostra de uma coleção de dados.

2.3.4.4. **Visualização de Dados**

As técnicas e ferramentas para Visualização de Dados são indispensáveis ao processo de Mineração de dados. Elas podem ser usadas durante a execução das etapas do processo de extração de conhecimento melhorando a compreensão dos resultados obtidos e a comunicação entre os usuários (Rezende et al., 1998).

As técnicas de Visualização de Dados estimulam naturalmente a percepção e a inteligência humana, aumentando a capacidade de entendimento e associação de novos padrões. Logo, a Visualização de Dados utiliza a percepção humana como um primeiro método para descobrir valores. Poderosas ferramentas de visualização que consigam gerar diversas formas de visualização (árvores, regras, gráficos 3D/2D, espectro) combinadas com técnicas de Mineração de Dados podem melhorar o processo de Mineração de dados (Fayyad, Grinstein, & Wierse, 2002).

Segundo Han, Kamber & Pei (2011), a visualização de dados, utiliza várias abordagens, incluindo técnicas orientada a pixel, técnicas de projeção geométrica, técnicas baseada em ícone, hierárquica e técnicas baseada em gráfico.

2.3.4. **Conclusões**

Nesta seção foi feita uma abordagem geral, mais sucinta, do processo de Mineração de dados, apresentando alguns de seus algoritmos e métodos utilizados na extração de conhecimento em grande massa de dados.

2.4. **Casos Baseados em Raciocínio**

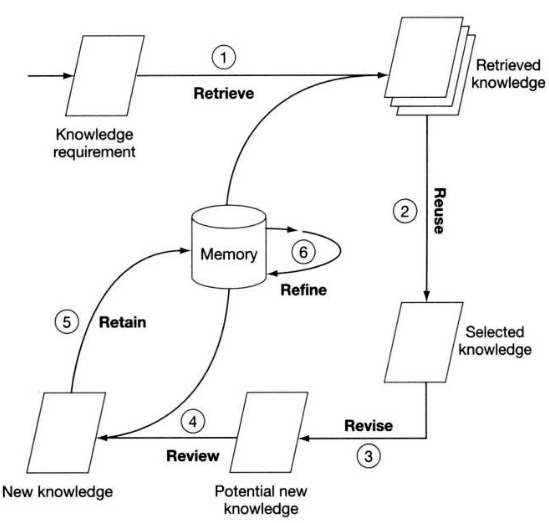
2.4.1 **Introdução**

De acordo com Watson (2003), o Caso Baseado em Raciocínio (**CBR)** usa o conceito de similaridade para recuperar coisas (casos) de uma biblioteca (uma base de casos). Casos são usados em muitas situações; por exemplo, para fornecer informações de produtos para um cliente, resolver problemas em uma central de informações aos clientes, configurar equipamentos de manufatura, ou resolver problemas financeiros complexos.

De acordo com Watson (2003), nós resolvemos problemas usando experiências adquiridas e que podemos aprender novas experiências. O Raciocínio baseado em Casos pode ser descrito por seis atividades ocorrendo em ciclo, como mostra a Figura 2.8.

Este ciclo é constituído de seis processos:

1. Recuperar
2. Reusar
3. Revisar
4. Avaliar
5. Manter
6. Refinar



**Figura 2.8 O funcionamento do Raciocínio Baseado em Casos** (*CBR-cycle*). **Fonte**: Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003)

2.4.2. **Definição**

Para Watson (2003), a ideia básica em um sistema CBR é que, para um domínio particular, os problemas a serem resolvidos tendem a ser recorrentes e repetir-se com pequenas alterações em relação a sua versão original. Dessa forma, soluções anteriores podem ser reutilizadas também com pequenas alterações.

Riesbeck e Schank (1996), definem **CBR** como “Um sistema de CBR resolve problemas por adaptar soluções que foram utilizadas para resolver problemas anteriores”.

2.4.3. **Representação de Casos**

De acordo com Watson (2003), casos são registros de experiências que contém conhecimento, que pode ser ambos explicito e tácito. Por exemplo, ele pode ser casos de históricos de pacientes no sentido médico, detalhes de empréstimos bancários, ou descrição de situações de erros de equipamentos. Cada um desses registros de casos compreende:

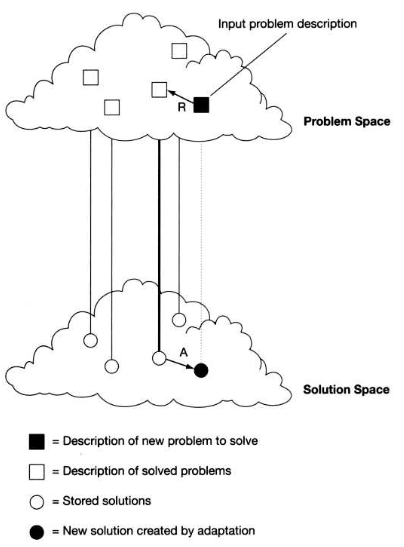
* Uma descrição
* O respectivo resultado ou solução

Assim, um caso tipicamente compreende um par problema e solução. Uma coleção de casos, é chamado de uma base de casos, justamente como uma base de registros é chamado de banco de dados (Watson, 2003).

Uma forma de visualizar é em função do espaço do problema e do espaço de solução. Na Figura 2.9 pode-se ver que, um caso individual é composto de dois componentes: uma descrição do problema e o armazenamento da solução. Estes residem respectivamente no espaço do problema e no espeço de solução. A descrição do problema a ser resolvido, é colocado no espaço de problema. Recupera-se o caso mais similar a descrição do problema, e sua solução, armazenada encontrada. Se necessário, ocorrem adaptações, e uma nova solução é armazenada. Este modelo conceitual de CBR, assume que há um mapeamento direto de um-para-um entre o problema e os espaços de solução. (Watson, 1999).

Bases de Casos divide-se em duas grandes categorias (Watson, 2003):

* **Em bases de Casos homogêneas**: todos os casos compartilham os mesmos dados ou estrutura de registros; isto é, casos têm os mesmos atributos, mas variando os valores.
* **Em Bases de Casos Heterogêneas**: casos têm estrutura de registros variados; isto é, casos podem ter diferentes atributos e valores variados.



**Figura 2.9 Os espaços de problema e de solução**.

**Fonte**: Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003).

Um exemplo de caso homogêneo pode ser o caso de venda de casa, onde na base de casos de casas tem os mesmos atributos que são suficientes para descrever uma casa. Então, um corretor, que tenha acesso a essa base de casos, pode assumir que já tenha todas as informações para realizar a transação. Embora, se o corretor ainda não tiver esta propriedade na base de casos, ele pode facilmente criar um registro nessa base de casos.

Um exemplo de base de casos heterogênea poderia ser uma base de casos de diagnósticos de pacientes. Registros de pacientes contêm um lote de informações em comum, tais como idade, tipo sanguíneo, pressão sanguínea, mas também muitas informações que são únicas para cada paciente, por exemplo, histórico médico, tratamento e prognósticos.

Quando se desenvolve uma base de casos heterogênea, os desenvolvedores, nunca pode assegurar que ele tenha um conjunto completo de características (Watson, 2003). Por exemplo, uma base de dados de diagnósticos de pacientes, os desenvolvedores poderiam não listar todas as possíveis condições médicas, sintomas, e testes que uma pessoa poderia ter.

Dentro de um **Caso** pode-se armazenar muitos tipos de dados, tais como nomes, identificação de produto, valores como custo, temperatura, e notas textuais. Algumas ferramentas de CBR também suportam dados com características de multimídia, tais com imagens, sons e vídeo (Watson, 1999, 2003).

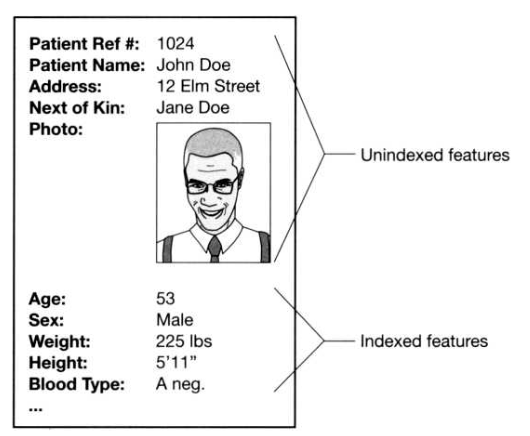
Não há um consenso por parte da comunidade de CBR, que informações exatamente, poderia ser um Caso. Embora, duas medidas pragmáticas poderiam ser tomadas para se decidir o que poderia ser representada em Casos: a funcionalidade da informação e a facilidade de aquisição da informação (Watson, 1999).

2.4.3. **Indexação**

Muitos sistemas de Banco de Dados utilizam-se de índices para agilizar a recuperação de dados. Um índice é computacionalmente, uma estrutura de dados que pode ser realizada em memória, tornando a localização da informação, muito rápida, sem ter que fazer a busca dos registros no disco. O CBR também faz uso de índice para agilizar a recuperação de Casos. A informação dentro de um Caso, é de dois tipos (Watson, 1999):

1. Informação Indexadas, que á usada para recuperar um Caso.
2. Informação não indexada, que fornecem informações contextuais para o usuário, mas que não são usadas diretamente para recuperação de Caso.

Por exemplo, em um sistema médico, pode-se usar as informações do paciente, tais como idade, sexo, altura, e peso como características a serem indexadas, que pode ser usada para recuperação do Caso e, outras informações, tais como nome, endereço e fotografia como informações contextuais, ou seja, não indexadas, que não podem ser usadas para recuperação de Casos. A Figura 2.10 ilustra este exemplo.



**Figura 2.10 Informações indexadas e não-indexadas**.

**Fonte**: Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003).

Como diretrizes, os índices devem:

* Ser preditivo.
* Indicar o propósito em que o Caso será usado.
* Ser abstrato o suficiente para permitir ampliar a base de casos e seu uso no futuro.
* Ser concreto o suficiente para ser reconhecido em futuras situações.

Se tomarmos como exemplo um sistema bancário. Informações dos clientes, tais como nome e telefone, são claramente não preditivas, você não decidir emprestar dinheiro a um cliente como base em seu nome e telefone. No entanto, informações tais como renda, e seus compromissos financeiros, tais como empréstimos habitacionais, pagamentos de carros, e seguro de vida, e assim por diante, são claramente preditivos. Dessa forma, informações como renda e compromissos financeiros podem ser escolhidos como índices e nome e telefone como informações contextuais (Watson, 1999).

A escolha do índice, tanto pode ser manual com automatizada. Escolher um índice manualmente, envolve decidir o propósito do Caso em respeito ao objetivo do sistema e decidir em que circunstancias o Caso vai ser útil (Watson, 1999).

Existem um número crescente de métodos de indexação automática na literatura, incluindo: MEDIATOR, CHEF e CYRUS, etc.

Várias ferramentas de CBR presentes no mercado suportam a identificação de índices de casos automaticamente, para aplicações práticas, índices pode ser escolhido automaticamente, manualmente ou ambas técnicas (Watson, 1999, 2003).

2.4.4. **Aquisição**

A representação do Caso é um importante aspecto no projeto de sistemas CBR, no que se refere a visão conceitual do que é representado no Caso e, levando-se em conta os índices que caracterizam os casos. A base de Casos poderia ser organizada em uma estrutura gerenciável que suporte pesquisas e métodos de recuperação eficientes. Deve-se ser encontrado um equilíbrio entre os métodos de armazenamento que preserve a riqueza de casos e seus índices e métodos que simplifique o acesso e recuperação de casos relevantes (Watson, 1999). Estes métodos são usualmente chamados de modelos de memória de casos (*case-memory models*). Os dois modelos de memória de casos mais influentes na academia são o modelo de memória dinâmica de Schank e Kolodner, e o modelo categoria exemplares de Porter e Bareiss (Watson, 1999).

2.4.4.1. **Modelos de Memória Dinâmica**

O modelo de memória dinâmica é composto principalmente de pacotes de organização de memória (**MOP**s), que são frames que compõem uma unidade básica da memória dinâmica. Eles representam conhecimento sobre classes de eventos de duas maneiras:

* Instâncias, que representam casos, eventos ou objetos.
* Abstrações, que representam versões generalizadas de instâncias ou de outras abstrações.

O desenvolvimento da teoria mais geral de Schank levou aos pacotes de organização de memória episódicos (**E-MOP**s), implementados no sistema **CYRUS** (Kolodner, 1993). A ideia básica é organizar casos específicos que partilham propriedades similares sob uma estrutura mais geral, ou seja, um **E-MOP**. Uma **E-MOP** contém os casos, as propriedades comuns entre eles e as características que os diferenciam.

2.4.4.2. **Modelos de Categoria Exemplares**

Esse modelo, considera que, os casos no mundo real, podem ser vistos como exemplares de acontecimentos. Neste caso, uma memória de casos é uma rede semântica de categorias de casos ligados por relações semânticas de hierarquias, de semelhança ou de diferenças. Cada caso é associado a uma categoria e suas características têm importância distintas para enquadrá-lo ou não na categoria. Características similares de um caso apontam para as de outro caso ou categoria. Dessa forma, compõe-se uma rede de conhecimento genérico do domínio que permite alguma recuperação do raciocínio do sistema para gerar explicações. Para gerar um novo caso, é buscado um caso semelhante no banco de casos. Se houver pequenas diferenças entre os dois, apenas um deles é retido, ou é armazenado uma única combinação dos dois (Watson, 2003).

2.4.5. **Recuperação**

Dada uma descrição de um problema, um algoritmo de recuperação deveria encontrar os casos mais similares à situação atual, utilizando-se dos índices da memória de casos. Os algoritmos baseiam-se nos índices e na organização de memória para guiar a busca dos casos potencialmente úteis.

De acordo com Watson (2003), a recuperação de casos, está diretamente relacionado e dependente ao método de indexação usado. Em geral, duas técnicas são correntemente usadas pelas ferramentas de **CBR** comerciais: algoritmo de vizinhança (***Nearest-Neighbor***) e indutivo.

2.4.5.1. **Algoritmo de Vizinhança**

Esse método, segundo Watson (2003), baseia-se na comparação entre um novo caso e aqueles armazenados na base de casos, utilizando uma soma ponderada das suas características. Para isso é necessário atribuir um peso a cada uma das características que descrevem o caso e que serão utilizadas na recuperação.

Na prática, a similaridade (isto é, a proximidade) do caso destino para o caso fonte para cada atributo é determinado. Esta medida é multiplicada por um fator peso. Então a soma da similaridade de todos os atributos é calculada. Esta pode ser representada por uma equação relativamente simples

**(2.6)**

Onde,

T é o caso destino

S é o caso fonte

n é o número de atributos em cada caso

i é um atributo individual de 1 até n

é a função de similaridade para atributo i nos casos T e S

w é o peso do atributo i

Algoritmos de similares a este são usados por muitas ferramentas CBR para realizar recuperação do caso mais similar. Similaridade são normalmente para cair dentro da faixa de 0 para 1 (onde o significa totalmente dissimilar e 1 exatamente similar) ou usando um percentual, onde 100% é totalmente similar (Watson, 1999; 2003).

2.4.5.2. **Algoritmo de Indução**

Indução é uma técnica desenvolvida por pesquisadores de Aprendizado de Máquinas para extrair regras ou construir de dados passados. Em sistema CBR, a base de casos é analisada por algoritmo de indução para produzir uma árvore de decisão que classifica (ou indexa) os casos. O algoritmo de indução foi amplamente usado pela ferramenta CBR chamada ID3.

2.4.6. **Adaptação**

A tarefa final do Sistema CBR é adaptar a solução associada a um caso recuperado para as necessidades do problema corrente. Quando uma situação é fornecida, o algoritmo de recuperação traz o melhor caso que ele encontrar para a memória. Normalmente, o caso selecionado não atende perfeitamente coma descrição do problema do usuário. Ou seja, existem diferenças entre o problema do usuário e o caso contido no banco de casos que devem ser levadas em conta. Então, o processo de adaptação procura por diferenças salientes entre as duas descrições e aplica regras de forma a compensá-las. Em geral, existem dois tipos de adaptação em CBR (Watson, 2003):

* **Adaptação Estrutural** - as regras de adaptação são aplicadas sobre a solução armazenada junto aos casos.
* **Adaptação Derivacional** – o algoritmo reusa os algoritmos, métodos ou regras que geraram a solução que consta no banco de casos para gerar uma nova solução para o problema corrente. Neste método, a sequência que construiu a solução original deve ser armazenada juntamente com o caso na memória de casos. O algoritmo de adaptação derivacional exige uma perfeita compreensão dos casos armazenados e da forma como as soluções foram geradas.

Segundo Watson (2003), várias técnicas têm sido usadas em sistema CBR. Incluindo as seguintes:

* **Adaptação nula** – ele simplesmente aplica a solução recuperada ao problema corrente sem modificação. Adaptação nula é útil para problemas envolvendo raciocínio complexo mais com solução simples. Por exemplo, em um sistema para concessão de crédito, embora seja necessário coletar muitas informações do cliente, a solução final de conceder ou rejeitar o crédito é direta.
* **Ajuste por parâmetros** – é uma técnica de adaptação estrutural que compara parâmetros específicos entre o caso recuperado e o novo para modificar a solução armazenada na direção apropriada. Esta técnica foi usada no sistema CBR chamado JUDGE, que recomenda sentenças mais curtas para crimes menos violentos.
* **Reinstanciação** – instancia uma nova solução para um caso recuperado do banco de casos com novas características adequadas ao problema do usuário. Por exemplo, o sistema CBR CHEF, que a partir de uma receita existente, criar uma nova receita.
* **Substituição derivacional** – repete o método, ou parte do método que gerou uma solução armazenada em um caso similar de forma a obter a solução para o novo caso, substituindo os atributos distintos. Como no sistema BOGART que reaplica os planos de geração de projetos para novos problemas.
* **Repara guiado por modelos** – utiliza um modelo casual para adaptar as soluções armazenadas aos problemas do usuário. O sistema CBR, CELIA, utiliza-o para aprendizado e diagnóstico de problemas mecânicos de automóveis.

De acordo com Watson (2003), a adaptação é útil em muitas situações. Mais não significa que seja essencial. Muitos dos sistemas CBR comerciais não usam adaptação para tudo. Eles simplesmente reusam a solução sugerida para o melhor caso correspondente (i.e., adaptação nula) ou eles deixam a adaptação para as pessoas.

2.4.7. **Conclusões**

Pode-se concluir que, O **Raciocínio Baseado em Casos** é um método em que problemas novos são resolvidos através de soluções adaptadas que foram usadas para resolver problemas mais antigos.

Um **Caso** é um pedaço contextualizado de conhecimento que representa uma experiência. Ao se analisar cada caso, se tem a descrição do problema e a solução armazenada. Caso já exista um problema semelhante já anteriormente armazenado no banco de dados, a solução será recuperada. Porém, se não existir um caso similar, a descrição desse novo problema, será enviado ao espaço de problemas, recuperando o caso com o problema mais similar possível, criando uma nova solução (Watson,1997).

Devido a essas características, o **RBC** é indicado para tarefas de segmentação e categorização.

A recuperação de casos é profundamente relacionada e dependente do método de categorização utilizado. As duas técnicas utilizadas atualmente são a de **recuperação por vizinho mais próximo** e **recuperação indutiva**. Na recuperação por vizinho mais próximo, deverão ser definidos os índices dos casos e os seus respectivos pesos, dependendo do problema a ser resolvido. Esses índices deverão ser previsíveis, identificar o propósito em que os casos serão utilizados, serem abstratos o bastante para permitir o uso posterior da base de dados, e serem concretos o bastante para serem reconhecidos no futuro (Watson, 1997). Como explicitado anteriormente, o caso similar será utilizado como solução do problema. No caso de recuperação por indução, uma árvore de decisão é utilizada, classificando os casos.

2.5 **Big Data**

2.5.1. **Introdução**

Big Data é um termo que vem chamando a atenção pela acelerada escalada em que, volumes cada vez maiores de dados são criados pela sociedade. Fala-se comumente em petabytes de dados gerados a cada dia, e zetabytes começa a ser uma escala real e não mais futurista. A uma década atrás terabyres era uma quantidade futurista, agora temos em nossos próprios computadores. Muito tem sido escrito sobre Big Data e como ele pode servir como base para a inovação, diferenciação e crescimento da análise de dados em grandes massas de dados (Kolb, 2013).

De acordo com Raj (2013), as tecnologias que sustentam o Big Data, podem ser analisadas sob duas óticas: as envolvidas com análise de dados, tendo Hadoop e Map-Reduce como as principais e as tecnologias de infraestrutura, que armazenam e processam os dados. Neste aspecto, destacam-se os bancos de dados NoSQL (Not Only SQL).

O termo Big Date está diretamente ligado a questões como volume, variedade, velocidade, complexidade e valor (Mayer-Schönberger, 2013):

* **Volume** – o volume está claro. Geramos petabytes de dados a cada dia. E estima-se que este volume dobre a cada 18 meses.
* **Variedade** - Variedade também, pois estes dados vêm de sistemas estruturados (hoje minoria) e não estruturados (a imensa maioria), gerados por e-mails, mídias sociais (Facebook, Twitter, YouTube e outros), documentos eletrônicos, apresentações estilo Powerpoint, mensagens instantâneas, sensores, etiquetas RFID, câmeras de vídeo, e assim por diante.
* **Velocidade** - De acordo com o Gartner, velocidade significa tanto o quão rápido os dados estão sendo produzidos quanto o quão rápido os dados devem ser tratados para atender a demanda. Etiquetas RFID e contadores inteligentes estão impulsionando uma necessidade crescente de lidar com torrentes de dados em tempo quase real. Reagir rápido o suficiente para lidar com a velocidade é um desafio para a maioria das organizações.
* **Valor** - E valor porque é absolutamente necessário que a organização que implementa projetos de Big Data obtenha retorno destes investimentos. Um exemplo poderia ser a área de seguros, onde a análise de fraudes poderia ser imensamente melhorada, minimizando-se os riscos, utilizando-se, por exemplo, de análise de dados que estão fora das bases estruturadas das seguradoras, como os dados que estão circulando diariamente nas mídias sociais.
* **Complexidade** - Quando você lida com grandes volumes de dados, eles vêm de diversas fontes. É um grande desafio vincular, correlacionar, limpar e transformar os dados de um sistema. No entanto, é necessário conectar e correlacionar interações, hierarquias e vínculos múltiplos de informação ou então os dados podem rapidamente sair de controle. Governança de dados pode ajudar a determinar como os dados díspares se relacionam com definições comuns e como integrar sistematicamente os ativos de dados estruturados e não estruturados para produzir informações de alta qualidade, uteis, adequadas e atualizadas.

2.5.2. **O Uso do Big Data**

Os modelos relacionais, quando proposto por Edgar F. Codd, atenderam muito bem, a demanda era acessar dados estruturados, de acordo com (Elmasri & Navathe (2005), gerados pelos sistemas internos das corporações. Estes modelos não foram desenhados para tratar dados não estruturados e nem para volumes de dados na casa dos petabytes de dados.

Para tratar dados na escala de volume, variedade e velocidade do Big Data precisamos de outros modelos. Surgem os softwares de banco de dados NoSQL, desenhados para tratar imensos volumes de dados estruturados e não estruturados. Existem diversos modelos como sistemas colunares como o [*Big Table*](http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/pt-BR/archive/bigtable-osdi06.pdf)(De uso interno pelo Google),o modelo ***Key/value*** como [*DynamoDB da Amazon*](http://aws.amazon.com/pt/dynamodb/), o modelo “*document database*” baseado no conceito proposto pelo Lotus Notes da IBM e aplicado em softwares como MongoDB, e o modelo baseado em grafos como o [Neo4j](http://neo4j.org/), e assim por diante (Kolb, 2013).

Aplicações modernas de mineração de dados, frequentemente chamada "Big-Data Analytics", exigir-nos gerenciar grande quantidade de dados rapidamente e em muitas dessas aplicações, exige-se um amplo paralelismo (Kolb, 2013).

Para lidar com aplicações tais como essas, novos tipos de software têm surgido. Estes sistemas de programação são projetados para obter o máximo do paralelismo. O novo tipo de software começa com uma nova forma de sistema de arquivos, chamada "Sistema de arquivos distribuídos", que contam com unidades muito maiores do que os blocos de disco dos sistemas operacionais convencionais. Além do mais, os sistemas distribuídos também fornecem replicação de dados ou redundância para proteger os dados, contra falhas frequentes de mídias, que ocorrem quando o dado é distribuído para milhões de nós de computadores (Kolb, 2013).

No topo destes sistemas de arquivos, diversos sistema de alto nível de programação foram desenvolvidos. No centro do novo software está o sistema de programação chamada ***Map-Reduce***. Implementações de ***Map-Reduce*** permite que os cálculos sob os dados em grande escala, sejam executados em clusters de computação de forma eficiente e tolerante a falhas de hardware (kolb, 2013).

2.5.3. **Map-Reduce**

Map-reduce não é um produto ou um software especifico, mas sim uma tecnologia desenvolvida pelo Google para lidar com grande quantidade de dados, cortando-os e combinando-os no final (Kolb, 2013).

A ideia básica é que os dados que precisam ser processados, entram no sistema, e é cortado em pedaços chamados de chunks. Essas peças de software, são responsáveis em fazer esses cortes é chamado de “***Mapper***”. Os chunks, são então enviados para outras peças de software para fazer o processamento requerido sobre eles, e então eles são ainda enviados para outra peça de software chamado “***Reducers***” que combina o resultado final para a saída. A Figura 2.11 ilustrado esse processo (Kolb, 2013).

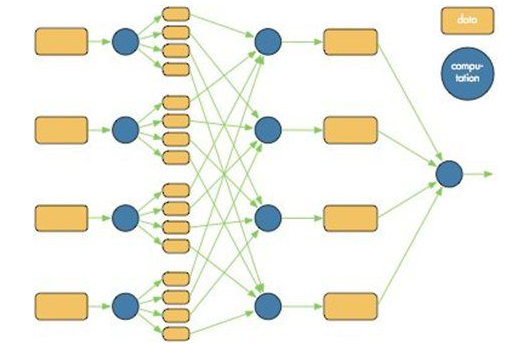
CombinedOutput

Reduce (tasks)

Group by keys

Map (tasks)

Input (chuncks)



**Figura 4.1 Um esquema da função Map-Reduce**. **Fonte**: The Big Data Revolution, Kolb, 2013

Todas essas peças de software - mapeador, processador, e redutor – tipicamente rodam no mesmo servidor de uma só vez. Dessa forma, a carga de processamento dos dados mapeados, podem ser espalhados por todo servir disponível, e mais servidores podem ser adicionado em tempo real, se for necessário um resultado mais rápido (Kolb, 2013)

O importante aqui, é você entender que, a tecnologia de Map-Reduce é capaz de pegar uma grande quantidade de dados, que seria muito dispendioso rodar em apenas um servidor, e poder distribui-lo por vários servidores. Este é um novo paradigma de programação. Existem algumas ferramentas que implementam esse novo paradigma, entre elas cito, o **Hadoop** do *Apache Foundation* (Kolb, 2013).

2.5.4. **Sistemas Distribuídos**

A maioria dos computadores são feitos com um único processador, com sua memória principal, cache, e disco local (um nó de computação). No passado, grandes cálculos científicos, eram realizadas em aplicações chamada de processamento paralelo, feitos com propósito especial em computadores de uso paralelo com muitos processadores e hardware especializado. No entanto, a prevalência de serviços Web de grande escala tem levado a computação a ser realizada em instalações com milhares de nós de computação operando mais ou menos de forma independente. Nestas instalações, os nós de computação são hardware de commodity, que reduz os custos comparado com as máquinas paralelas (Stonebraker et Al., 2010).

Esta nova arquitetura de computação tem dado origem a uma nova geração de sistemas de programação. Estes sistemas tiram máximo proveito do paralelismo e são tolerantes a falhas, quer seja de software ou de hardware (Stonebraker et al., 2010).

2.5.4.1. **Organização Física de Nós de Computação**

A nova arquitetura de computação paralela, algumas vezes chamada de computação em cluster, é organizada como segue. Os nós de computação são armazenados em racks, talvez de 8-64 nós em um rack. Os nós em um simples rack são conectados por uma rede, tipicamente uma Ethernet gigabits. Pode haver muitos racks de nós de computador, e racks são interconectados outros níveis de rede ou a um switch (Dean & Ghemawat, 2008).

2.5.4.2. **Organização de Sistemas de Arquivos em Grande Escala**

Para explorar computação em cluster, o arquivo deve parecer e se comportar um pouco diferente de sistemas de arquivos convencionais encontrados em computadores simples. Este novo sistema de arquivos, geralmente chamado de *distributed file system* ou **DFS** (embora estes termos tenham tido outro significado no passado), é tipicamente usado como segue (Dean & Ghemawat, 2008):

* Os arquivos são divididos em chunks, que são tipicamente de 64 megabytes de tamanho. Esses chunks são replicados, talvez três vezes, para três diferentes nós de computação. Além disso, os nós que armazenam um chunk deve estar localizado em racks diferentes, dessa forma não se perde todas as cópias devido uma falha de rack. Normalmente, o tamanho do chunk e grau de replicação, pode ser decidido pelo usuário.
* Para encontrar os chunks de um arquivo, existe um outro pequeno arquivo chamado nó **master** ou nó de nome para o arquivo. O nó master é ele mesmo replicado, e o diretório do sistema de arquivos como um todo sabe onde encontrar suas cópias. O diretório pode ser replicado e todos os participantes usando o DFS sabem onde estão suas cópias no diretório.

2.5.5. **As Tarefas de Mapeamento**

De acordo com Kolb (2013), o arquivo de entrada para uma tarefa Mapeamento, consiste de elementos, que podem ser de qualquer tipo: uma tupla ou um documento, por exemplo. Um chunk é uma coleção de elementos, e nenhum elemento é armazenado em dois chunks. Tecnicamente, todas as entradas para as tarefas de mapeamento (The Map Tasks) e saídas para as tarefas Redução (The Reduce Tasks) são os pares na forma chave-valor (key-value), geradas por uma função hash. Essa forma de entradas e saídas são motivadas pelo desejo de permitir a composição de vários processos Map-Reduce (Kolb, 2013).

A função de mapeamento (Map) recebe um elemento com seus argumentos e produz zero ou mais pares chave-valor. Os tipos de chaves e valores são arbitrários. Mais, as chaves não são "chaves" no sentido usual; elas não precisam ser únicas. Mais uma tarefa de mapeamento pode produzir vários pares chave-valor com a mesma chave, mesmo a partir do mesmo elemento (Kolb, 2013).

**Exemplo 2.5.1**: Suponha que se deseje contar o número de ocorrências para cada palavra em uma coleção de documentos. Neste exemplo, o arquivo de entrada é um repositório de documentos, e cada documento é um elemento. A função de mapeamento para este exemplo usa chaves que são do tipo String (a palavra) e valores que são inteiros. A tarefa de mapeamento lê um documento e quebra ele em uma sequência de palavras w1, w2, w3, ..., wn. Ela então emite uma sequência de pares de chave-valor onde o valor é sempre 1. Isto é, a saída da tarefa de mapeamento para este documento é a sequência de pares chave-valor (w1,1),(w2,1),....,(wn,1).

Note que uma simples tarefa de mapeamento irá processar muitos documentos - todos os documentos em um ou mais chunks. Assim, a saída produzida será mais do que a sequência para o documento sugerida acima. Note também que se uma palavra w aparece m vezes entre todos os documentos atribuídos a esse processo, então haverá m pares chave-valor (w,1) entre sua saída. Uma opção para resolver esse problema é usar agrupamento e agregação, que é combinar esses m pares em um simples par (w, m), isso só é possível porque as tarefas Redução, aplica uma operação associativa e comutativa, para os valores.

2.5.6. **Agrupamento e Agregação**

O processo controlador mestre sabe quantas tarefas Reduce haverá, digamos r tarefas, pois o usuário normalmente informa ao sistema map-reduce quais são as r tarefas. Então o controlador mestre aplica uma função hash e produz uma tabela de chaves de números (códigos) de 0 até r-1. Cada chave produzida pela tarefa Map é um hash e seus pares chave-valor são colocados em um arquivo local. Cada arquivo é destinado para uma das tarefas Reduce (Kolb, 2013).

Após todas as tarefas Map terem completadas com sucesso, o controlador mestre junta os arquivos de cada tarefa Map que são destinados para uma particular tarefa e alimenta o arquivo resultante com uma lista de pares chave-valor. Isto é, para chave k, a entrada para a tarefa Reduce que manipula a chave k é um par da forma (k, [v1, v2, ..., vn]), onde (k, v1), (k, v2), ..., (k, vn) são todos pares chave-valor e k, vindo de todas as tarefas Map.

2.5.7. **As Tarefas de Redução**

Os argumentos da função Reduce é um par consistindo de uma chave e sua lista de valores associados. A saída da função Reduce é uma sequência de zero ou mais pares chave-valor. Esses pares chave-valor podem ser de tipo diferente daqueles enviados das tarefas map para as tarefas Reduce, mais normalmente elas são do mesmo tipo. Referimo-nos a aplicação da função Reduce que reduz para uma simples chave e seus valores associados de redutor (Kolb, 2013).

Uma tarefa reduce recebe uma ou mais chaves e sua lista de valores associados. Isto é, uma tarefa reduce executa um ou mais redutores. As saídas de todas as tarefas reduce são juntas em um simples arquivo. Redutores podem ser divididos em tarefas reduce menores e a função hash associa cada chave com um dos códigos da tabela hash (Kolb, 2013).

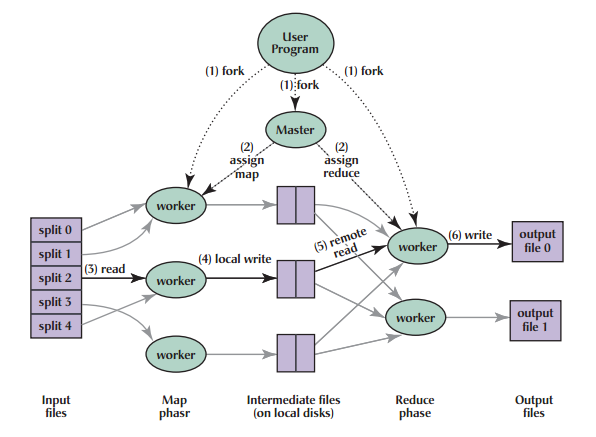
**Exemplo 2.5.2**: Vamos continuar com o exemplo conta palavras do Exemplo 4.1. A função Reduce simplesmente agrega todos os valores. A saída de um redutor consiste da palavra e da soma. Isto é, a saída de todos as tarefas Reduce é uma sequência de pares (w,m), onde *w* é uma palavra que aparece pelo menos uma vez entre todos os documentos e *m* é o total de ocorrências de *w* em todos os documentos.

2.5.8. **Detalhes de Execução de Map-Reduce**

A Figura 2.12 oferece um esboço de como processo, tarefas, e arquivos interagem. Aproveitando uma biblioteca fornecida por um sistema map-reduce tal como Hadoop, o programa do usuário bifurca o processo controlador mestre e alguns dos processos Worker para diferentes nós de computação. Normalmente, um Worker manipula suas tarefas Map (um Map worker) ou tarefas Reduce (um Reduce worker), mas não ambos (Kolb, 2013).

O mestre tem muitas responsabilidades. Uma é criar um certo número de tarefas Map e algumas tarefas Reduce, este número sendo selecionado pelo programa do usuário. Estas tarefas serão atribuídas para o processo Worker pelo Mestre. É razoável criar uma tarefa Map para cada chunk de arquivo de entrada, mas pode-se desejar criar poucas tarefas Reduce. A razão para limitar o número de tarefas Reduce é que é necessário para cada tarefa Map criar um arquivo intermediário para cada tarefa Reduce, e se existe muitas tarefas Reduce o número de arquivos intermediários aumenta bastante (Kolb, 2013).

O Mestre (Master) se matem informado do estado de cada tarefa Map e Reduce (ocioso, executando um particular worker, ou concluído). Um processo Worker relata para o Mestre quando ele termina uma tarefa, e uma nova tarefa é agendada pelo Mestre para esse processo Worker (Kolb, 2013).



**Figura 4.2** **Esboço de Iteração de um Processo Map-Reduce. Fonte:** Dean & Ghemawat (2008)

2.5.9. **Lindando com Falha de Nós**

A pior coisa que pode acontecer é quando o nó de computação que está executando o Mestre falha. Neste caso, toda carga de entrada do map-reduce deve ser reiniciado. Mas somente este nó pode derrubar um processo inteiro; outras falhas iram ser gerenciadas pelo Mestre, e o trabalho do map-reduce irá completar eventualmente (Kolb, 2013).

Suponha que o nó de computação no qual reside o Map worker falha. Esta falha irá ser detectada pelo Mestre, porque ele periodicamente pings o processo Worker. Todas as tarefas Map que foram atribuídas para este Worker terão que ser refeitas, mesmo que tivessem concluído. A razão para refazer completamente as tarefas Map é que sua saída destinada as tarefas Reduce residem no nó de computação, e devido falha, está indisponível para as tarefas Reduce. O Mestre configura o estado de cada uma das tarefas Map para ociosa e reprograma-o para um Worker quando se tornar disponível. O Mestre também informa cada tarefa Reduce que a localização de suas entradas, que são as tarefas Map, mudaram (Kolb, 2013).

Lidar com uma falha para um Reduce Worker é simples. O Mestre, simplesmente, configura os estados de suas correntes tarefas Reduce em execução, para ociosa. Estas serão agendadas para outro reduce worker mais tarde (Kolb, 2013).

2.5.10. **Conclusões**

Concluindo, Map-Reduce é um modelo de programação, e framework introduzido pelo Google para suportar computações paralelas em grandes coleções de dados em clusters de computadores. Agora Map-Reduce é considerado um novo modelo computacional distribuído, inspirado pelas funções map e reduce usadas comumente em programação funcional. Map-Reduce é um “Data-Oriented” que processa dados em duas fases primárias: Map e Reduce. A filosofia por trás do Map-Reduce é: Diferentemente de data-stores centrais, como um banco de dados, você não pode assumir que todos os dados residem em um lugar central, portanto você não pode executar uma query e esperar obter os resultados em uma operação síncrona. Em vez disso, você precisa executar a query em cada fonte de dados simultaneamente. O processo de mapear a requisição do originador para o data source é chamado de ‘Map’, e o processo de agregação do resultado em um resultado consolidado é chamado de ‘Reduce’.

Hoje existem diversas implementações de Map-Reduce, como: Hadoop, Disco, Skynet, FileMap e Greenplum. Hadoop é a implementação mais famosa.

A tecnologia de big data não apenas suporta a habilidade de coletar grandes volumes de dados como também provê a habilidade de compreendê-los e tirar proveito de seu valor.

2.6 **Bases Conceituais sobre a Evasão e Retenção**

A Constituição Federal (**CF**) de 1988, em seu art. 6º, define a educação como um direito social, ao lado de outros, como: saúde, trabalho, moradia, lazer, segurança, previdência social, proteção à maternidade e à infância, assistência aos desamparados. Como dever do Estado e da família, o direito à educação deve consolidar-se na promoção do pleno desenvolvimento da pessoa, no preparo para o exercício da cidadania e na qualificação para o trabalho [BRASIL, 1988, art. 205].

O direito à educação pode ser considerado como um dos alicerces da República Federativa do brasil na medida em que é instrumento necessário

Á construção de uma sociedade livre, justa e solidária; à garantia do desenvolvimento nacional; à erradicação da pobreza e da marginalização, com a redução das desigualdades sociais e regionais; e à promoção do bem de todos, sem preconceitos de origem, raça, sexo, cor, idade e quaisquer outras formas de discriminação [Garcia, 2004, s.p.].

Portanto, à educação, é parte da matriz que constitui em larga escala o respeito à dignidade humana. Esse preceito vem sendo expresso em inúmeros documentos, tratados, acordos nacionais e internacionais na LDB [MEC, 2014].

Assim, à educação é considerada como [MEC, 2014]:

* **Direito social e dever do Estado**, para corresponder às aspirações da sociedade por um país democrático, justo e isonômico, traduzindo-se em ações que visem dar respostas públicas aos compromissos socialmente assumidos em cada uma das instituições, tanto fortalecendo o processo de inserção cidadã como contribuindo para o desenvolvimento pessoal e profissional dos sujeitos e para o desenvolvimento local, regional e nacional do país;
* **Direito de cidadania**, para formar pessoas críticas, autônomas, emancipadas e competentes tecnicamente, ativas na dinâmica do convívio social e partícipes na definição dos projetos de desenvolvimento nos âmbitos público e privado, pessoal e coletivo;
* **Bem público**, na perspectiva da inclusão e valorização da educação profissional e tecnológica como política pública, comprometendo-se o Estado com a qualidade social; e
* **Questão de soberania conjunta Estado-cidadão**, para cumprir a função social e os compromissos firmados a expansão do direito e a universalização do acesso.

Portanto, o conceito de educação para cidadania impõe-se como requisito político e pedagógico para que as instituições cumpram sua função social.

No entanto, não basta admitir a educação como direito fundamental. É essencial concretizar e prover as ações que permitam a garantia desse direito. Nesse sentido, tanto a CF, em seu art. 206, quanto a **LDB** (Lei de Diretrizes e base), em seu art. 3º, indicam os seguintes princípios, com relação direta com o sucesso escolar, para que o processo educacional ocorra de forma efetiva: a igualdade de condição para o acesso e permanência na escola, a garantia do padrão de qualidade, a valorização do profissional da educação escolar e a vinculação entre a educação escolar, o trabalho e as práticas sociais [MEC, 2014].

2.6.1. **Evasão e a Retenção Escolar**

Dentre as questões conflitantes, que envolve a relação entre educação, instituições de ensino e sociedade, s retenção e a evasão merecem destaque. Entre todas as modalidades de ensino, da educação básica à educação superior, esses problemas estão presentes [MEC, 2014].

Apesar das pesquisas relativas à evasão não identificar um conceito homogêneo, a partir de 1970, autores como Tinto (1995), professor da Syracuse University, passaram a abordar o modelo de integração do estudante, destacando que a decisão de evadir-se é tomada em função da falta de integração com o ambiente acadêmico e social da instituição, sendo esta integração influenciada pelas características individuais, pelas expectativas para a carreira ou curso e, por último, pelas intenções/objetivos e compromissos assumidos antes do início do curso [MEC, 2014].

O modelo desenvolvido por Tinto (1995) sugere seis conjuntos de variáveis:

* Os atributos de pré-entrada, entendidos como habilidades do aluno, escolaridade anterior e antecedentes familiares;
* Os comprometimentos iniciais ou metas traçadas pelo próprio estudante;
* A integração acadêmica, tida como o vínculo entre o estudante e a estrutura da instituição de ensino;
* A integração social entre os grupos de estudantes e docentes como variável;
* Os comprometimentos subsequentes ou influencias das dimensões acadêmicas e sociais da integração no vínculo com a instituição e na intenção de alcançar o objetivo de conclusão de curso; e
* Os aspectos externos.

Tinto (1995) finalmente, descreve os resultados, constituídos pela decisão, persistência ou deserção do curso ou sistema, como variável. Após o embasamento teórico de Tinto (1995) pode-se pensar em explicações sociológicas e políticas no estudo da evasão. Considerando que a evasão escolar, entendida como interrupção no ciclo de estudos, deve ser vista como um fenômeno complexo e não um problema comum, uma vez que compromete o efetivo do direito à educação de qualidade para todos [MEC, 2014].

Segundo Dore (2011), no Brasil a evasão pode se referir à retenção e repetência do aluno na escola; à saída do aluno da instituição, do sistema de ensino, da escola e posterior retorno; ou à não conclusão de um determinado nível de ensino. Portanto, para a pesquisadora a evasão ou abandono escolar é um processo que tem natureza multiforme.

A escolha de abandonar ou permanecer na escola é fortemente condicionada por características individuais, por fatores sociais e familiares, por características do sistema escolar e pelo grau de atração que outras modalidades de socialização, fora do ambiente escolar, exercem sobre o estudante [DORE, 2013, p.5].

2.6.2. **Categorização das causas da evasão e da retenção escolar**

A classificação proposta por Brasil (1996), para categorizar as causas da evasão e da retenção, em função de um plano estratégico e monitoramento desses problemas, mapeou os seguintes fatores ou categorias motivadores, da evasão e da retenção, adaptados às especificidades da contemporaneidade e das próprias instituições de ensino da Rede Federa:

* **Fatores individuais** – são problemas relacionados a aspectos peculiares às características do estudante, tais como: adaptação à vida acadêmica, capacidade de aprendizagem e habilidade de estudo, escolha precoce da profissão, qualidade de formação escolar anterior, questões financeiras do estudante ou da família, descoberta de novos interesses ou novo processo de seleção, dentre outros;
* **Fatores internos às instituições** – são problemas relacionados à infraestrutura, ao currículo, a gestão administrativa e didático-pedagógica da instituição, bem como outros fatores que desmotivam e conduzem o aluno a evadir do curso. Aqui se destacam os fatores: atualização, estrutura e flexibilidade curricular, formação do professor, infraestrutura física, material, tecnológica e de pessoal para ensino, motivação do professor, questões didático-pedagógica e relação escola-família;
* Fatores externos – relacionam-se às dificuldades financeiras do estudante permanecer no curso e às questões inerentes à futura profissão. Dentre estes fatores destacam-se: avanços tecnológicos, econômicos e sociais, oportunidade de trabalho para egressos do curso, políticas governamentais para a educação profissional e tecnológica e para a educação superior, reconhecimento social do curso e valorização da profissão.

Dessa forma, o conceito de evasão adotado aproxima-se dos conceitos propostos por Brasil (1996) e Dore (2013), sendo definido como a interrupção do aluno no ciclo do curso. Em tal situação, o estudante pode ter abandonado o curso, não ter realizado a renovação da matrícula ou formalizado o desligamento do curso. Por outro lado, a retenção consiste da não conclusão do curso no período previsto, fator concorrente para o aumento da propensão em relação à evasão escolar [MEC, 2014].

2.6.3. **Indicadores de Evasão, Retenção e Conclusão**

Os conceitos de evasão e de retenção servem de base para a construção de indicadores que relacionam esses conceitos ao número de estudantes ingressantes e matriculados nas instituições, fornecendo subsídios para identificação de necessidades de ações específicas [MEC, 2014].

O Sistema Nacional de Informações da Educação Profissional e Tecnológica (SISTEC), tem o registro efetivo da vida do estudante ou de um conjunto de estudantes (ciclo de matrículas) na instituição, desde seu ingresso até sua saída, e as mudanças que ocorrem durante esse período. Isso permite o acompanhamento dos indicadores de conclusão, evasão e retenção dentro de um mesmo ciclo.

Portanto, o SISTEC, a partir das situações de matrícula, estabelece conceitos de total de abandono, retenção e conclusão que serão utilizados no cálculo dos indicadores de evasão, retenção e conclusão. Nesse sentido, estabelecem-se os seguintes conceitos [MEC, 2014]:

* **Total de matrículas ativas**: número de matrículas que permanecem ativas com situação “em curso” ou “integralizado”.
* **Total de retenção**: número de matrículas que permanecem ativas com situação “em curso” ou “integralizado” após a data para o término do ciclo de matrícula do curso.
* **Total de saída sem êxito**: número de matrículas finalizadas com situação “transferido interno”, “transferido externo”, “desligado/desistente” ou “evadido”.
* **Total de saída com êxito**: número de matrículas finalizadas com situação “concluído”.

Para o cálculo das taxas de evasão, retenção e conclusão, pode ser realizado considerando a amostra escolhida como sendo os estudantes matriculados no período em análise ou sobre os estudantes matriculados em um ciclo, a partir dos dados de matrículas ativas ou finalizadas.

* **Análise no período**: relativo ao conjunto total de estudantes matriculados no período. De forma simplificada, as taxas de conclusão, evasão e retenção no período podem ser definidas como:

**Taxa de conclusão no período** = estudantes que concluíram no curso (saída com êxito) / estudantes com matrículas ativas

**Taxa de evasão no período** = estudantes que tiveram a matrícula finalizada sem êxito / estudantes com matrículas ativa

**Taxa de retenção no período** = estudante com tempo de matrícula maior do que o tempo previsto de duração do ciclo (retenção) / estudantes com matrículas ativas

Os indicadores relativos ao período são relevantes para a avaliação sistêmica das instituições e das redes de ensino, inclusive para relatórios anuais de gestão, porque medem o resultado obtido com os recursos destinados ao trabalho institucional e apresentam o desempenho do conjunto geral de matriculados em relação à expectativa de sucesso daqueles que ingressaram. Estes indicadores são indicados também para comparação entre instituições [MEC, 2014].

* **Análise de ciclo**: a análise de ciclo é focada na situação atual de um universo de estudantes que ingressaram em um dado ciclo. As taxas de conclusão, evasão e retenção do ciclo também podem ser definidas como:

**Taxa de conclusão do ciclo** = estudantes que concluíram o curso (saída com êxito) / ingressantes no ciclo

**Taxa de evasão do ciclo** = estudantes que tiveram a matrícula finalizada sem êxito no ciclo / ingressantes no ciclo

**Taxa de retenção do ciclo** = estudantes com tempo de matrícula maior do que o tempo previsto de duração do ciclo (retenção) / ingressantes no ciclo

As taxas no ciclo permitem observar o desempenho de um determinado grupo de ingressantes de um determinado curso e com isso associar outros fatores à análise (econômico, sociais, etc.). Por outro lado, não permitem a avaliação sistêmica, pois para realizar a análise dos ciclos, se faz necessário definir o conjunto a partir do critério “período de estimativa de finalização”. E deixando de considerar todas as matrículas que eventualmente concluíram ou abandonaram no período, mas são ingressantes de outros ciclos que não se enquadram no critério [MEC, 2014].

2.6.4. **Conclusões**

Nesta seção foi dada uma visão geral sobre os conceitos relacionados a evasão e a retenção escola, bem como os índices indicadores de evasão, retenção e concussão. Sem, no entanto, ter a pretensão de esgotar o assunto, uma vez que o mesmo envolve muitas varáveis em seu entorno.

Para a pesquisa realizada nessa seção, foi tomado como base o “Documento orientador para a superação da evasão e retenção na Rede Federal”, realizado pelo Ministério da Educação em 2014.

2.6.5. **Referência bibliográfica do capítulo 2**

|  |
| --- |
| Agrawal, R.; IMIELINSKI, T.; SWAMI, A. Mining Association Rules Between Sets of Itens in Large Databases. ACM SIGMOD Conference Management of Data, 1993.  Bassan, Amarpreet Singh; Sarkar, Debarchsan. Mastering SQL Server 2014 Data Mining. Packt Publishing Ltd. Birminggham B3 2PB, UK, Dez 2014.  Dean, Jeffrey; Ghemawat, Sanjay. MapReduce: Simplified Data Processing On Large Cluster. CUMMUNICATIONS OF THE ACM, janeiro de 2008.  Dong, G. & J. LI. Interestingness of discovered association rules in terms of neighbordhood-based unexpectedness. Lecture Notes in Artificial Intelligence, pp. 72-86, 1998.  ELMAN, Jeffrey L. *Learning and Development in Neural Networks: The Importance of Starting Small*. Cognition, 48(1993), pp.71-99. 1993. Web: http://crl.ucsd.edu/~elman/  Ftp: <ftp://crl.ucsd.edu/pub/neuralnets/cognition.ps.Z>  Elmasri, Ramez; Navathe, Shamkant B. Sistemas de Banco de Dados. São Paulo: Addison Wesley, 2005.  Engels. R. Planning tasks for knowledge discovery in databases: Performing Task-Oriented User-Guidance. Proceeding of the International Conference on Knowledge Discovery and Data Mining. Portland: AAAI Press, 1996.  Engels, R.; LINDNER, G.; STUDER, R. A Guided Tour Through the Data Mining Jungle. Proceeding of the Third International Conference on Knowledge Discovery in Databases. Newport Beach, 1997.  Fayyad, Usama; PIATETSKI-SHAPIRO, Gregory; SMYTH, Padhraic (1996). The KDD Process for Extracting Useful Knowledge from Volumes of Data. In: Communications of the ACM, pp.27-34, Nov.1996.  Fiesler, E. *Neural Networks Formalization and Classification.* Computer Standard & Interfaces, Special Issue on Neural Networks Standards, John Fulcher (Ed.). V.16, N.3. Elsevier Sciences Publishers, Amsterdam, June, 1994. Web: http://www.idiap.ch/idiap-networks.html.  Freitas A. A. A multi-criteria approach for the evaluation of rule interestingness. Em Proceedings of the International Conference on Data Mining. Rio de Janeiro, RJ, pp. 7-20, 1998.  Freitas A. A. On rule interestingness measures. Knowledge-Based Systems 12(5-6), 309-315, 1999.  Goldschmidt, R.; Passos, E.; Vellasco, M.; Pacheco, M. Task Definition Assistence in KDD Applications. CLEI’03 – XXIX Conferência Latino Americana de Informática. La Paz, 2003.  Goldschmidt, Ronaldo. Data mining: conceitos, técnicas, algoritmos, orientações e aplicações/Ronaldo Goldschmidt, Eduardo Bezerra 2ª, ed. -Rio de Janeiro: Elsevier, 2015.    Han, Jiawei; Kamber, Micheline. Data Mining: Concepts and Techniques. Second Edition. Elsevier. San Francisco, CA, 2006.  Han, Jiawei; Kamber, Micheline; Pei, Jian. Data Mining: Concepts and Techniques. Third Edition. Elsevier. San Francisco, CA, 2011.  Haykin, Simon. Redes neurais: princípios e práticas/Simon Haykin; trad. Paulo Martins Engel. – 2.ed. – Porto Alegre: Bookman, 2001  Harinath, Sivakumar; Pihlgren, Ronald; Lee, Denny Guang-Yeu; Sirmon, John,  Bruckner, Robert M. Professional Microsoft SQL Server 2012 Analysis Services with MDX and DAX. Published by John Wiley & Sons, Inc. Indianapolis, Indiana, 2012.  Hussain F.; Liu H.; Suzuki E.; Lu H. EXCEPTION RULE MINING WITH  RELATIVE INTERESTINGNESS MEASURE. PAKDD, 2000; pg 86-97.  Inmon, Bill & Chuck Kelly. The Twelve Rules of Data Warehouse for a Client/Server World, Data Management Review, 1994. (\*\*\*\*-)  Kimball, Ralph. Data Warehouse toolkit: o guia completo para modelagem multidimensional /Ralph Kimball, Margy Ross; tradução Ana Beatriz Tavares, Daniela Lacerda. Rio de Janeiro: Campus, 2002.  Knight, Brian; Veerman, Erik; Moss, Jessica M.; Davis, Mike; Rock, Chris. Professional Microsoft SQL Server 2012 Intregration Services. John Wiley & Sons, Inc, 2012.  Kolb, Jason; KOLB. Jeremy. The Big Data Revolution. The Tricks Tour Competitors Don’t Want You To Know By Jason Kolb and Jeremy Kolb. AppliedData Labs. Plainfield, IL, 2013.  Kohonen, Teuvo. *Self-Organization and Associative Memory*. Springer-Verlag Series in  Information Science. 1987.  Kolodner, J. L. Proceedings of the DARPA Case-Based Reasoning Workshop. San Francisco: Morgan Kaufmann Publishers, 1998  Liu, B. & W. Hsu. Post-analysis of learned rules. AAAI 1, 828-834, 1996.  Mayer-Schönberger, Viktor; Cukier, Kenneth. Big Data. A Revolution That Will Transform How We Live, Work and Things. First published in Greta Britain. John Murray (Publishers) an Hachette UK Compnay, 2013.  MEC [Ministério da Educação]. Documento Orientador para a Superação da Evasão e Retenção Na Rede Federal de Educação Profissional, Científica e Tecnológica, 2014.  Morik, K. The Representation Race- Preprocessing for Handling Time Phenomena. Proceedings of the European Conference on Machine Learning 2000, Lecture Notes in Artificial Intelligence 1810. Berlin: Springer Verlag, 2000  Negnevitsky, Michael. Artificial Intelligence: a guide to intelligent Systems/Michael Negvitsky. Pearson Education Limited. Edinburgh Gate, 2005.  Oliveira, C., EDACLUSTER: Um Algoritmo Evolucionário para Análise de agrupamentos Baseados em Densidade e Grade, Dissertação (Mestrado em Engenharia Elétrica), Universidade Federal do Pará, 2007.  Osório, Fernando. Redes Neurais – Aprendizado Artificial. Forum de I.A. “99 – pg.13”. Rosa, João Luís Garcia. Fundamentos da Inteligência artificial /João Luís Garcia Rosa. Rio de Janeiro: LTC, 2011.  Passos, Emanuel; GOLDSCHMIDT, Ronaldo. Data Mining: Um guia prático. Editora Campos. Rio de Janeiro, 2005.  Pazzini, M. J. Knowledge discovery from data? IEEE Intelligent Systems, 10-13, 2000.  Piatetsky-Shapiro, G & C. J. Matheus. The Interestingness of deviations. Em Proceedings of the International Conference on Knowledge Discovery and Data Mining, pp. 23-36, 1994.  Projeto Político-Pedagógico do IFRN. Uma construção Coletiva. Documento – Base, 2012  Raj, Subu. BIG DATA – AN INTRODUCTION. Kindle Ver 1.1, 2013.  REZENDE, Solange Oliveira. Sistemas inteligentes: fundamentos e aplicações. Editora Manole Ltda. Barueri, SP. 2003.  Riesbeck, C. K., and Schank, R. Inside Case-Based Reasoning. Northvale, NJ: Lawrence Erlbaum Associates, 1996.  Rob, Peter; Coronel, Carlos. Database Systems: Design, Implementation, and Management by Peter Rob and Carlos Coronel 8th Edtion. Thomson Place, Boston, Massachusetts, 2009.  Robt, Peter. Sistemas de Banco de Dados: Projeto, implantação e gerenciamento / Peter Rob, Carlos Coronel. São Paulo: Cengage Learning, 2011.  Rossum, Joost Van; Baccaro, Régis. Extending SSIS With .NET Scripting. A Tool kit For SQL Server Integration Services. Apress Media, LLC is a California LLC, 2015  Savjani, Parikshit. BI Solutions Using SSAS Tabular Model Succinctly. By Syncfusion Inc. Morrisville, NC 27560 USA, 2014.  Silberschatz, A. & Tuzhilin. On subjective measures of interestingness in knowledge discovery. Proceeding of the First International Conference on Knowledge Discovery and Data Mining 1, 275-281, 1995.  Stonebraker, Michael, Abadi; Daniel, DeWitt; David J.; Madden, Sam; Paulson, Erik; Pavlo, Andrew; Rasin, Alexander.MapReduce complements DBMSs since databases are not designed for extracttransform-load tasks, a MapReduce specialty. COMMUNICATIONS OF THE ACM, pp 71. Publicado em Janeiro de 2101.  UTGOFF, P. Shift of Bias for Inductive Concept Learning. Machine Learning: an Artificial Intelligence Approach, v.3, São Francisco: Morgan Kaufmann, 1996.  Watson, Ian D. Applying case-based reasoning: techniques for enterprise systems. San Francisco, CA: Morgan Kaufmann Publishers, Inc, 1997.  Watson, Ian D. Applying Knowledge Management: Techniques for Building Corpoarte Memory. San Francisco, CA: Morgan Kaufmann Publishers, Inc, 2003.  Witten, Ian H.; Frank, Eibe; Hall, Mark A. Data Mining. Practical Machine Learning Tools and Techiniques. 2nd ed. Morgan Kaufmann Publishers is an imprint of Elsevier, 2005.  Witten, Ian H.; Frank, Eibe; Hall, Mark A. Data Mining. Practical Machine Learning Tools and Techiniques. Third Edition. Morgan Kaufmann Publishers is an imprint of Elsevier, 2011. |
| **Sites**  <http://educacao.uol.com.br/noticias/2013/03/14/brasil-tem-3-maior-taxa-de-evasao-escolar-entre-100-paises-diz-pnud.htm>, 14/03/2013 21h28. Acessado em 05/01/2016  <http://ec.europa.eu/education/schooleducation/leaving_en.htm>, novembro 2013. Acessado em 10/02/2016  G1 (31 de agosto de 2012). [Brasil tem 3,7 milhões de crianças e jovens fora da escola, aponta Unicef](http://g1.globo.com/educacao/noticia/2012/08/brasil-tem-37-milhoes-de-criancas-e-jovens-fora-da-escola-aponta-unicef.html). Visitado em 16 de maio de 2013. [Cópia arquivada em 16 de maio de 2013](http://www.webcitation.org/6GfbZZF7L) |

**Capítulo 3 – Desenvolvimento dos Módulos**

* 1. **Introdução**

O Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte (IFRN), nova institucionalidade dada pelos termos da Lei 11.829, de 29 de dezembro de 2008, faz parte da rede federal de educação profissional e tecnológica, vincula-se ao Ministério da Educação, possui natureza jurídica de autarquia e detém autonomia administrativa, patrimonial, financeira, didático-pedagógica e disciplinar. Trata-se de uma instituição de educação superior, básica e profissional, especializada na oferta de educação profissional e tecnológica nas diferentes modalidades de ensino, conjugando conhecimento científico, técnico e tecnológico a ideais pedagógicos de fundamentação histórico-crítica.

Com estrutura multicampis, o IFRN está sediado na Reitoria, localizada na Rua Dr. Nilo Bezerra Ramalho, nº 1692, Bairro Tirol, Natal-RN. É composto, tomando-se, o ano de 2014 como referência, por vinte e um campus.

De organização pluricurricular, o IFRN oferece um ensino público laico, gratuito e de qualidade. Oferta, nesse sentido, cursos em sintonia com a função social que desempenha, visando a consolidação e, o fortalecimento, dos arranjos produtivos, culturais e sociais locais. Apresenta, para tanto, um currículo organizado a partir de quatro eixos – ciência, trabalho, cultura e tecnologia – que atuam, de modo entrelaçado e Inter complementar, como princípios norteadores da prática educativa. O Instituto desenvolve à pesquisa e a extensão, na perspectiva da produção, socialização e difusão de conhecimentos. Estimula a produção cultural e realiza processos pedagógicos que levem à geração de trabalho e renda. Em um contexto mais amplo, a Instituição visa contribuir para as transformações da sociedade, visto que esses processos educacionais são construídos nas relações sociais.

No que concerne à comunidade acadêmica, há os sujeitos sociais diretamente envolvidos com os processos pedagógicos e administrativos do IFRN. Essa comunidade é constituída por três seguimentos: estudantes, professores e técnicos-administrativos. Numa perspectiva mais abrangente, acrescenta-se, a esse coletivo, a comunidade local, composta tanto por pais dos estudantes e/ou responsáveis pelos estudantes quanto representantes da sociedade civil.

Para efeito de regulação, avaliação e supervisão da Instituição e dos cursos de educação superior, equipara-se às universidades federais. Além de se submeter à legislação federal específica, rege-se pelos seguintes instrumentos normativos: estatuto; regimento geral; regimento interno dos campis e dos demais órgãos componentes da estrutura organizacional dos institutos federais; resolução do Conselho Superior (CONSUP); deliberações do Colégio de Diretrizes (CODIR) e do Conselho de Ensino, Pesquisa e Extensão (CONSEPEX); e atos da Reitoria.

A expansão do IFRN amplia, significativamente, a atuação nas áreas de ensino, de pesquisa e de extensão; contribui, de modo mais extensivo, para a formação humana e cidadã; e estimula o desenvolvimento socioeconômico, à medida que potencializa soluções científicas, técnicas e tecnológicas, com compromisso de estender benefícios à comunidade.

Essa ampla abrangência em todo território norte-rio-grandense contribui para posicionar tanto o IFRN como uma instituição de educação, ciência e tecnologia quanto os seus campis como elos de produção de conhecimento e de desenvolvimento social. Garante, assim, a manutenção da respeitabilidade junto às comunidades nas quais os campis se inserem e da credibilidade construída ao longo da história da Instituição [Projeto Político-Pedagógico do IFRN, 2012].

* 1. **Histórico**

Para uma melhor compreensão do perfil institucional, na perspectiva de consolidar a função social, os objetivos e os princípios orientadores do instituto, faz-se necessário uma análise histórica, políticas e sociais do IFRN.

Criada pelo Decreto 7.566, de 23 de setembro de 1909, como Escola de Aprendizes Artífices, a instituição passou por diversas mudanças e recebeu várias denominações ao longo do tempo. Instalada, inicialmente, em janeiro de 1910, no prédio do antigo Hospital da Caridade, na Praça Coronel Lins Caldas, nº 678, Cidade Alta, onde hoje funciona a casa de estudantes de Natal, a Escola de Aprendizes Artífices oferecia cursos primários de desenho e oficinas de trabalhos manuais. Em 1914, o estabelecimento foi transferido para a Avenida Rio Branco, nº 743, ocupando, durante cinquenta e três anos, um edifício construído no início do século XX.

Mais tarde, ocorreu a mudança de denominação para Liceu Industrial de Natal, orientada pela reforma instituída pela Lei 378, de 13 de janeiro de 1937, do Ministério da Educação e Saúde, órgão a que a Instituição estava subordinada desde 1930. Na época, eram oferecidas oficinas de desenho, de sapataria, de funilaria, de marcenaria e de alfaiataria, inspiradas, segundo Meireles (2006, p.55), em “modelos exteriores ao Brasil, o que evidencia a influência de outros formatos culturais, educacionais, tecnológicos e produtivos na realidade brasileira do século XX”.

Designada como Escola Industrial de Natal (**EIN**), no ano de 1942, após a promulgação da Lei Orgânica do Ensino Industrial, a Instituição transformou as oficinas em cursos básicos de primeiro ciclo, organizados em quatro seções: Trabalhos de Metal, Indústria Mecânica, Eletrotécnica e Artes Industriais. Ademais, a Escola também estava autorizada a oferecer cursos de mestria para os professores atuantes nessas áreas.

Transformada em autarquia pela Lei Federal 3.552, de 16 de fevereiro de 1959, todas as Escolas Industriais do Brasil conseguiram autonomia administrativa, didática e financeira, transformando-se em instituições federais destinadas a ministrar cursos técnicos de nível médio. Porém, somente em 1963, EIN implantou seus primeiros cursos técnicos de nível médio, com as ofertas de Mineração e de Estradas. O novo modelo, tinha equivalência ao ensino de 2º grau, o que permitia a continuidade de estudos no ensino superior para os egressos que assim o desejassem.

Em 1965, o Estabelecimento passou a nomear-se Escola Industrial Federal do Rio Grande do Norte **(EIFRN**). Nessa década, no dia 11 de março de 1967, ocorreu a inauguração da “nova” Escola Industrial nas recém-construídas instalações do prédio situado na Avenida Salgado Filho nº 1559, no bairro Tirol, atendendo a uma comunidade escolar de 233 servidores e cerca de 1.100 estudantes.

Na condição de Escola Técnica Federal do Rio Grande do Norte (ETFRN), mudança impetrada pela Portaria Ministerial 331, de 16 de junho de 1968, o Conselho de Representantes deliberou a extinção gradativa dos cursos industriais básicos, passando-se a ministrar somente o ensino profissional de nível técnico. Em consequência, foram criados, em 1969 e 1973, os cursos técnicos de nível médio em Eletrotécnica, em Mecânica, em Edificações, em Saneamento e em Geologia, sob a orientação da Lei 5.692/71, a qual definia a estrutura do ensino de 2º grau como ensino profissionalizante obrigatório. A partir de então, a ETFRN passou a dedicar-se, exclusivamente, ao ensino técnico profissionalizante de 2º grau.

No ano de 1994, iniciou-se outro processo de transição das escolas técnicas federais. No entanto, somente no dia 18 de janeiro de 1999, efetivou-se a mudança de ETFRN para Centro Federal de Educação Tecnológica do Rio Grande do Norte (CEFETRN). De acordo com os documentos oficiais, os centros de educação tecnológica foram implantados com a finalidade de formar e qualificar profissionais no âmbito da educação tecnológica, em diferentes níveis e modalidades de ensino, para os diversos setores da economia. Também objetivavam realizar pesquisas aplicadas e promover o desenvolvimento tecnológico de novos processos, produtos e serviços, em estreita articulação com os setores produtivos e com a sociedade.

Em 2006, o Governo Federal lançou um arrojado plano de expansão da rede federal. Com características de interiorização de educação profissional e tecnológica para todo o País, e foram implantadas mais três unidades de ensino vinculadas ao CEFET-RN: na Zona Norte da cidade de Natal, na cidade de Ipanguaçu e Currais Novos. Ainda nesse mesmo ano, devido ao lançamento do Programa Nacional de Educação de Jovens e Adultos (PROEJA), o CEFET-RN começou a atuar na educação profissional técnica de nível médio, na modalidade de educação de jovens e adultos, oferecendo também, para educadores que atuam nessa modalidade, cursos de formação em nível de pós-graduação lato sensu.

Ao limiar de um século de existência, a Instituição adquiriu nova configuração, transformando-se em Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte (IFRN), nos termos da Lei 11.892, de 29 de dezembro de 2008. De acordo com o estabelecido oficialmente, os institutos federais são instituições, pluricurriculares e multicampis, de educação superior, básica e profissional. São especializadas, na oferta de educação profissional e tecnológica nas diferentes modalidades de ensino, ancorando-se na conjuntura de conhecimentos técnicos e tecnológicos com as práticas pedagógicas. Esse processo, uma vez criada a rede federal de educação profissional, cientifica e tecnológica, constituiu-se em elemento de redefinição do sistema de ensino brasileiro. Reforçou, por outro lado, a autonomia administrativa, patrimonial, financeira, didático-pedagógica e disciplinar dessas instituições educativas, além de imprimir a equivalência às universidades federais, no que refere às disposições que regem a regulação, a avaliação e a supervisão das instituições e dos cursos de educação superior.

A expansão da rede federal de educação profissional e tecnológica está pautada na interiorização da educação profissional, com o compromisso de contribuir, significativamente, para o desenvolvimento socioeconômico do País. Nessa perspectiva, a criação dos institutos federais responde à necessidade da institucionalização definitiva da educação profissional e tecnológica como política pública permanente de Estado.

Esse processo de interiorização da educação profissional e tecnológica contribui para o combate às desigualdades estruturais de diversas ordens, proporcionando o desenvolvimento social por meio da formação humana integral dos sujeitos atendidos. Propicia, ainda, o desenvolvimento econômico, a partir da articulação das ofertas educacionais e das ações de pesquisas e de extensão. Tal articulação vincula-se aos arranjos produtivos sociais e culturais, com possibilidades de permanência e de emancipação dos cidadãos assim como de desenvolvimento das diversas regiões do Rio Grande do Norte.

* 1. **Visão do problema**

O Instituto Federal de Educação, Ciência e tecnologia do Rio Grande do Norte, ministra diversos cursos em diferentes níveis de ensino: médio, técnico, técnico-subsequente, tecnológico, PROEJA e cursos de pós-graduação, latu e estrito sensu. Com um corpo docente amplo e uma grande massa de alunos, como já foi falado, nos parágrafos anteriores. A movimentação diária, e constantes, do sistema acadêmico, gera uma grande massa de dados e de diversos tipos, como por exemplo, dados demográficos **(**idade, sexo, situação financeira dos pais, nível de escolaridade dos pais, região de procedência, escola de origem, dentre outros), e dados comportamentais (conclusão, repetência, cancelamento de matrícula, evasão, transferência e assim por diante) e por fim, os dados sobre avaliações(Notas e faltas). Essa grande massa de dados está armazenada em um sistema de banco de dados relacional e, em algumas situações em planilhas eletrônicas, como no caso das pesquisas semestrais que são realizadas no IFRN, pela gestão e pela equipe pedagógica, entre alunos e servidores, com a finalidade de se mapear os problemas relacionados a infraestrutura e ensino aprendizagem.

Nos últimos anos, têm-se observado, por parte dos educadores e gestores educacionais, um alto índice de repetência e evasão escolar, em alguns cursos e até mesmo, nota-se um certo desestímulos por parte dos alunos, em determinados cursos, ministrados nos diversos campus da instituição. É fato, por exemplo, que alguns cursos, as turmas de 3º e 4º períodos, chegam com média muita baixa de alunos, em torno de 20%, ou seja, essas turmas iniciaram com 40 alunos e, no 4º período estão com no máximo 8 a 10 alunos. Isso tem como consequência, vários professores, ministrando aulas para poucos alunos, diminuindo em muito, a relação aluno professor, um dos índices, que é utilizado pelo governo federal, para avaliar o Instituto. Diante desse quadro, surgiu a necessidade de fazer uma análise mais aprofundada de tais problemas. Ou seja, deseja-se analisar o **índice de reprovação**, o **índice de evasão**, o **índice de aprovação**, o **índice de conclusão** etambém o **índice de matrículas canceladas**. Tendo como finalidade, tentar mapear quais são as causas que implicam, diretamente ou indiretamente nesses índices e identificadas essas causas, e se possível, apresentar as soluções para os problemas.

Para tanto, o projeto será desenvolvido em vários módulos. Um Armazém de dados (em inglês **Data Warehouse - DW)**, onde será consolidado todos os dados das diversas fontes de dados, dentre elas o sistema acadêmico e as planilhas de pesquisas realizadas semestralmente, entre os alunos, professores/servidores. Um outro módulo é o sistema de **Mineração de Dados**, que irá mapear as causas relacionadas a evasão e a repetição dos alunos e, também fazer previsões futuras sobre a evasão e a repetência. Um terceiro módulo é um sistema de aconselhamento pedagógico (será utilizado no desenvolvimento desse módulo o Raciocínio Baseado em Casos – **RBC**). O módulo do **RBC**, será usado pela equipe pedagógica, para resolver os problemas relacionados a pedagogia, tais como: relacionamento dos alunos com os colegas e ou como os pais, alunos desestimulados e assim por diante. O último módulo é um jogo (baseado na técnica de **Gamification**), que terá como proposito engajar/estimular o aluno, a continuar e terminar seu curso.

* 1. **Arquitetura de Desenvolvimento da Pesquisa**

Para o desenvolvimento do projeto proposto nesta pesquisa, será necessário acessar e consumir dados de diversas fontes de dados, tais como bancos de dados relacionais, modelos multidimensionais, planilhas eletrônicas, arquivos simples, dentre outras. Dessa forma, será adotada a arquitetura do *Business Intelligence Semantic Model* (**BISM**), pois a mesma permite acessar e consumir dados de bases de dados relacionais, tais como Oracle, SQL Server, Teradata e DB2. Além do modelo tabular, planilhas eletrônicas, arquivos OData, arquivos simples e também de serviços na nuvem, como Windows Azure SQL Database.

Além do mais, o BISM disponibiliza ferramentas de interface intuitiva para manipular dados, criação de relatórios, Dashboards e KPI’s.

Conceitualmente o modelo pode ser apresentado em três camadas: modelo de dados, camada lógica e de consulta e a camada de acesso a dados. A Figura 3.1 ilustra esse modelo.

**Figura 3.1 As camadas conceituais do modelo BISM**

**Fonte**: adaptado de Harinath.

O modelo de dados (***Data model***), tanto pode ser um modelo multidimensional (com dimensões, medidas e cubos), quanto um modelo tabular (com tabelas, colunas e relações). Isso na perspectiva do desenvolvedor da aplicação.

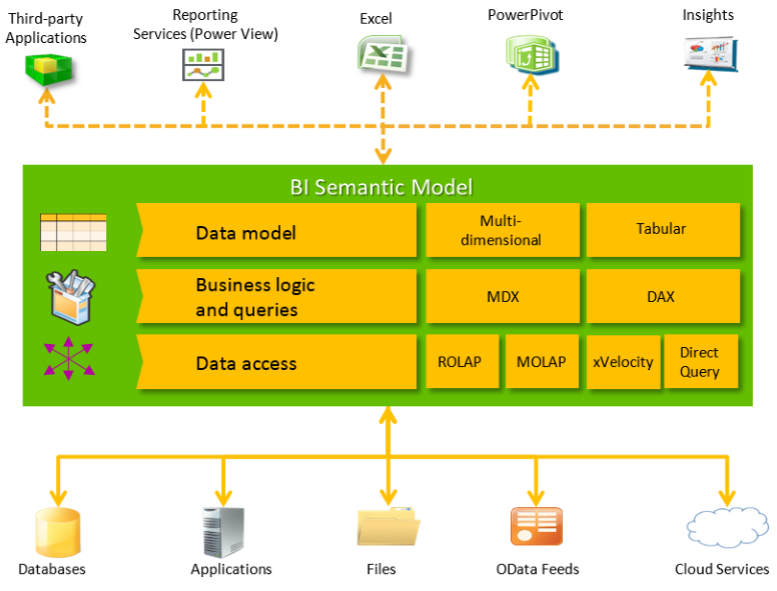
Para a camada lógica de negócio e consultas (Business logic and queries), o modelo BISM oferece a linguagem *Multi-Dimensional eXpressions* (**MDX**) para o modelo multidimensional e a linguagem **DAX**, um acrônimo para *Data Analysis Expressions*, para o modelo tabular.

Para a camada de acesso a dados (*Data Access*), o BISM dá suporte a todas as funcionalidades para ambos modelos multidimensionais **MOLAP** e **ROLAP**. Por outro lado, para o modelo tabular, o BISM usa um mecanismo de armazenamento de colunas em cache, organizando os dados em colunas, e utiliza um mecanismo de acesso a dados muito rápido (***Vertipaq***) que por manipular os dados em memória, dispensa a necessidade de indexação ou agregação, pelo fato do mesmo simplesmente realizar um escaneamento na memória.

Portanto, de um lado tenho a complexidade do modelo multidimensional, que suporta a construção de *Data warehousing*, com dimensões, factos e cubos, onde os dados podem ser analisados através do **OLAP** e técnicas de mineração de dados e são consumidos pelos utilizadores finais (analistas de negócio ou tomadores de decisão). E por outro lado, temos o modelo Tabular composto de tabelas e relacionamentos e, dessa forma, mais simples e mais intuitivo de trabalhar.

Então, para enfatizar ou justificar a escolha pela arquitetura **BISM**, para o desenvolvimento deste trabalho, considere o seguinte cenário.

**Cenário**: tenho a necessidade de acessar e consumir dados de bases de dados relacionais e multidimensionais, modelos tabulares, planilhas eletrônicas, e de diversos outros tipos de arquivos. Além dos mais, preciso apresentar para os usuários finais, os resultados obtidos através de relatórios, ***Dashboards***, **KPI**’s, **planilhas** ou **tabelas** e assim por diante. Então, diante dessa perspectiva, apresento na Figura 3.2 a arquitetura BISM que será utilizada neste trabalho.



**Figura 3.2** Arquitetura BISM adotada para o desenvolvimento do projeto.

**Fonte**: Harinath,2012

Como se pode ver na Figura 3.2, a arquitetura **BISM** permite acessar diversas fontes de dados, desde bases de dados relacionais, arquivos simples ou mesmo dados na nuvem. Gerar modelos multidimensional ou tabular, manipular esses dados e disponibilizá-los os utilizadores finais. Além do mais, esta arquitetura tem a flexibilidade, de utilizar várias ferramentas na criação de relatório, tais como Excel, SQL Server Report Services, Power BI, PowerPivot, Power View, e assim por diante).

3.5. **O Processo Business Intelligence**

O **BI** é um processo de converter dados em informações para que os envolvidos na tomada de decisões da empresa, possam fazê-las de forma rápida e precisa.

No contexto desse trabalho o primeiro módulo a ser desenvolvido é o armazém de dados (DW). Então, vai ser realizado um estudo de viabilidade para a construção do mesmo. Em seguida, vai ser feita a extração, limpeza e carga dos dados das fontes de dados disponíveis para o armazém de dados. A Figura 3.3 mostra o processo da solução adotado.

Solução BI

Relatórios

Fonte de dados

**Figura 3.3 O processo BI da solução**.

**Fonte**: Autor (adaptado de Savjani,2014)

3.5.1. **Módulo Data Warehouse**

Neste módulo, será desenvolvido o *Data Warehouse* do projeto de pesquisa. Nesse trabalho será desenvolvido não apenas um Data Warehouse, mas sim vários Data mart e a constelação de data marts formaram o data warehouse do IFRN. Antes de tudo será feito um estudo de viabilidade da construção do data warehouse.

3.5.1.1. **O Ciclo de vida**

Inicialmente, será definido o ciclo de vida de desenvolvimento do sistema, em inglês system development life cycle (SDLC) que, alguns autores chamam simplesmente de metodologia de desenvolvimento de sistema, em inglês system development methodology. Há duas variantes principais: o modelo em cascata e o iterativo. O modelo em cascata é também conhecido como a metodologia sequencial. A metodologia iterativa é conhecida como espiral ou incremental. Para esse projeto será utilizado a metodologia em cascata, seguindo o diagrama da Figura 3.4 [Rainardi, 2008].

Estudo de viabilidade

Requisitos

Arquitetura

Modelagem

Desenvolvimento

Teste

Implantação

Operação

**Figura 3.4**. **Metodologia em Cascata. Fonte: adaptado de Rainardi, 2008**

3.5.1.2. **Estudo de viabilidade**

O instituto tem um sistema de controle acadêmico que é responsável em gerenciar toda a vida acadêmica dos alunos. Neste sistema, temos dados acadêmicos dos alunos (seus dados pessoais, notas e faltas), cadastro dos cursos e suas matrizes curriculares, cadastro dos docentes e suas capacitações e áreas de atuação, inclusive pesquisas e extensão. Temos os dados socioeconômicos dos alunos, e situação financeira dos pais. Resumindo, todas essas informações estão em uma base de dados, em um servidor, localizado no setor administrativo do instituto, mais precisamente no prédio da reitoria, em Natal, capital do estado. Além desses dados, temos os dados de formulários de pesquisas que, são aplicadas semestralmente, no instituto, para avaliação do desempenho escolar dos alunos e turmas, avaliação do corpo docente e avaliação da instituição, no que se diz respeito a infraestrutura, bibliotecas, e outros assuntos administrativos relacionados a educação. Temos também, dados do serviço social da escola, que constam da assistência social que o instituto presta aos alunos, como por exemplos, bolsas de estudo, alimentação, assistência medica, dentária dentre outras. Estes dados, geralmente, estão em planilhas eletrônicas, em seus respectivos setores. Temos ainda, os dados de acesso a biblioteca, onde são registradas todas as consultas que os alunos fizeram, em forma de requisições de livros. Estes dados, estão armazenados em bases de dados e, não em planilhas. A tabela 3.1 a seguir, mostra um resumo dessas fontes de dados.

**Tabela 3.1** Bases de Dados do Instituto Natal Central - RN

|  |  |  |
| --- | --- | --- |
| Fonte de dados | Modelo | Descrição |
| Controle acadêmico | Base de dados SQL Server | Vida acadêmica dos alunos, professores, cursos e matrizes curriculares. |
| Pedagogia | Planilhas eletrônicas | Dados de pesquisas aplicadas semestralmente, para avaliação do desempenho escolar dos alunos. Para avaliação do desempenho dos professores e pesquisas avaliativas do instituto, como por exemplo, infraestrutura. |
| Serviço social | Planilhas eletrônicas | Dados referentes a assistência social do instituto para com os alunos carentes. |
| Biblioteca | Base de dado MySql | Informações de acesso a biblioteca pelos alunos (requisições de livros). |

Diante desse cenário, pode-se perceber facilmente que, essas diversas bases de dados não estão interligadas, ou seja, elas não se comunicam entre si, dificultando se fazer uma análise estatística com esses dados. Portanto, a proposta apresentada nesse projeto é de consolidar essas diversas fontes de dados, afim de propiciar uma maneira mais palpável, de se fazer cruzamentos com esses dados, afim de tentar encontrar relacionamentos entre os mesmos, que possam ser úteis para os gestores educacionais, tomarem decisões mais precisas, em prol do ensino aprendizagem e, como isso, desenvolver ações que possam engajar ou incentivar os alunos em nossos cursos, e dessa forma, minimizar os problemas da evasão e reprovação nos institutos. Portanto, a justificativa para implementar um sistema de *Data Warehouse*, é propiciar aos gestores educacionais, uma forma deles poderem analisa essas informações de diversas maneira, através de relatórios extraídos do modelo multidimensional e dos cubos **OLAP** gerados.

3.5.1.3. **Requisitos**

Para assegurar que os objetivos dos usuários serão alcançados, é de fundamental importância que sejam definidos os requisitos funcionais e não funcionais do sistema *Data Warehouse*, antes de começar seu desenvolvimento. Os requisitos funcionais, definem o que o sistema irá fazer, ou seja, os recursos que o sistema irá ter. Os requisitos não funcionais definem a arquitetura que será usada na construção do sistema.

3.5.1.3.1. **Requisitos Funcionais**

Nesta fase, foram levantadas, de forma sucinta, todas as características que cercam a educação no instituto federal. Essas informações foram coletadas de forma a elucidar os principais índices de desempenho que se queira analisar, de forma a facilitar a escolha do processo educativo, que se queira modelar. Um processo educativo, é por exemplo, analisar a evasão escolar. No geral, foram realizadas as seguintes atividades:

* Identificar os fatores “humanos” relacionados ao ambiente educacional, tais como o interesse e expectativas dos gestores educacionais em relação ao desenvolvimento e implantação do projeto em questão;
* Reuniões para se identificar os requisitos necessários para a definição do processo a modelar;
* Levantamento de dados das diversas fontes de dados, através do estudo de documentos, planilhas e relatórios e da própria base de dados do acadêmico;
* Levantamento de informações analíticas, ou seja, dos índices indicadores de desempenho escolar.

Em consequência dos levantamentos realizados nessa fase, se pode constatar os seguintes fatores, em relacionados ao projeto:

* Com a divulgação do projeto junto aos usuários da gestão educacional, criou-se uma grande expectativa dos potenciais usuários, especialmente aos pedagogos, que puderam enxergar os benefícios que o sistema trará para os mesmos;
* Os dados coletados para testes dos modelos, serão os dados do sistema controle acadêmico, dados do sistema social e dados do sistema da biblioteca do campus Natal central.
* O levantamento junto aos gestores educacionais, revelaram os seguintes índices a serem analisados: **reprovação escolar**, **evasão escolar**, **aprovação escolar**, **conclusão de cursos**, relação **aluno x professor**, **alunos desligados** e **alunos diplomados**.

Após a análise sucinta das fontes de dados, pode-se identificar os seguintes requisitos funcionais, mostrados na Tabela 3.2.

**Tabela 3.2** Requisitos Funcionais do Sistema Data Warehouse do IFRN.

|  |  |  |
| --- | --- | --- |
| Nº | Requisito | Prioridade |
| 1 | Os gestores educacionais, necessitam analisar o **índice de evasão escolar** nos campis do IFRN. Isso poderá ser feito, analisando o seu desempenho escolar, suas faltas, sua participação em projetos, seu índice de satisfação com o curso, e assim por diante. | Alta |
| 2 | Os gestores educacionais, necessitam analisar o **índice de reprovação escolar** nos campis do IFRN. Isso poderá ser feito, analisando o seu desempenho escolar, suas faltas, acessos a biblioteca, relacionamentos com professores e colegas. Situação financeira dos pais, dentre outras coisas. Essa análise deve ser feita por curso, turmas, disciplinas, períodos. | Alta |
| 3 | Precisa-se também, analisar o índice de aprovados, parasse ter, por exemplo, uma melhor projeção dos concluintes por turma e curso. Isso, implicará na formação de novas turmas e para poder definir com mais precisão as entradas nos cursos. | Alta |
| 4 | Precisa-se também, analisar o índice da relação aluno x professor, para se poder ter uma maior precisão na definição de quantos professores serão necessários para se manter um curso de qualidade, em uma visão ampla de vários anos à frente. Dessa forma, pode-se predizer com antecedência, quantos professores serão necessários contratar, para cada curso e disciplinas relacionadas. | Média |
| 5 | Analisar também a relação em alunos que concluíram o curso e alunos diplomados. Uma vez que, temos constantemente, situações onde, o aluno conclui todas as disciplinas e não se diploma, por vários motivos: doença, desiste sem explicação, não faz o trabalho de conclusão do curso, começa a trabalhar, etc. | Alta |
|  |  |  |

3.5.1.3.2. **Requisitos não funcionais**

Os requisitos não funcionais fornecem um guia e restrições para a arquitetura do sistema. Algumas dessas restrições, é devido os padrões de TI, outras restrições são impostas pela equipe de arquitetura de TI e, algumas outras são impostas pelos administradores de banco de dados (**DBA**s) e administradores do sistema operacional. Após reuniões com os arquitetos de TI, administradores do sistema operacional e usuários finais, foram definidos os seguintes requisitos não funcionais, como mostra a Tabela 3.3.

**Tabela 3.3 Requisitos Não Funcionais do Data Warehouse do IFRN**

|  |  |
| --- | --- |
| **Nº** | **REQUISITO** |
| 1 | Todos os usuários do data warehouse poderão acessar a aplicação front-end do data warehouse (relatórios e cubos OLAP), logando-se apenas uma vez no sistema. |
| 2 | A aplicação front-end do data warehouse não deve ser acessível fora do IFRN. |
| 3 | A aplicação front-end do data warehouse deve ser Windows form ou web. |
| 4 | O acesso a aplicação front-end do data warehouse deve ser através do portal do IFRN ou através de aplicações móveis. |
| 5 | Alguns usuários terão permissão de executarem consultas SQL diretamente no sistema. |
| 6 | Alguns dados sensíveis serão vistos somente por alguns usuários. |
| 7 | O tempo máximo de resposta para um relatório ou consulta é de 30 segundos. |
| 8 | O data warehouse deve estar disponível 24 horas ao dia, 7 dias da semana. |
| 9 | Para construir o projeto do data warehouse, será usado o Microsoft SQL Server, incluindo a ferramenta de ETL, Analysis Services, reporting e OLAP. |
| 10 | Deve-se ser feito o backup do data warehouse e testado o restore pelo menos a cada 6 meses. |
| 11 | Estima-se um número de usuários entre 300 e 1000. Cerca de 10% desses usuários, acessarão o sistema mais frequentemente; o resto ocasionalmente. |

3.5.1.4 **Modelagem do Data Warehouse**

De acordo com Rainardi [2008], um sistema *Data Warehouse* tem duas arquiteturas principais: a arquitetura de fluxo de dados e a arquitetura do sistema. A arquitetura de fluxo de dados define como os dados do Data Warehouse são armazenados e como os dados fluem do sistema fonte para o *Data Warehouse*. A arquitetura de sistema define a configuração física de servidores, rede, software, *storage* e cliente.

3.5.1.4.1. **Arquitetura de Fluxo de Dados**

Em um *Data Warehousing*, a arquitetura de fluxo de dados é a configuração dos armazéns de dados dentro do sistema Data Warehouse, e como os dados fluem das fontes de dados, para os armazéns de dados e desses para os usuários finais do sistema. Isto inclui, como esses dados são controlados, monitorado, bem como os mecanismos para assegurar a qualidade desses dados. Existem diversas arquitetura para fluxo de dados em data warehouse, dentre elas pode-se citar: um simples *Dimensional Data Store*(**DDS**), *Normalized Data Store* (**NDS)** + **DDS**, *Operational Data Store* (**ODS)** + **DDS**, e ***Data Warehouse*** federado.

Para este trabalho vou utilizar a arquitetura simples DDS, ilustrada pela Figura 3.5.

Sistema

Acadêmico

DDS

**Aplicação**

ETL

Planilhas

...

MDB

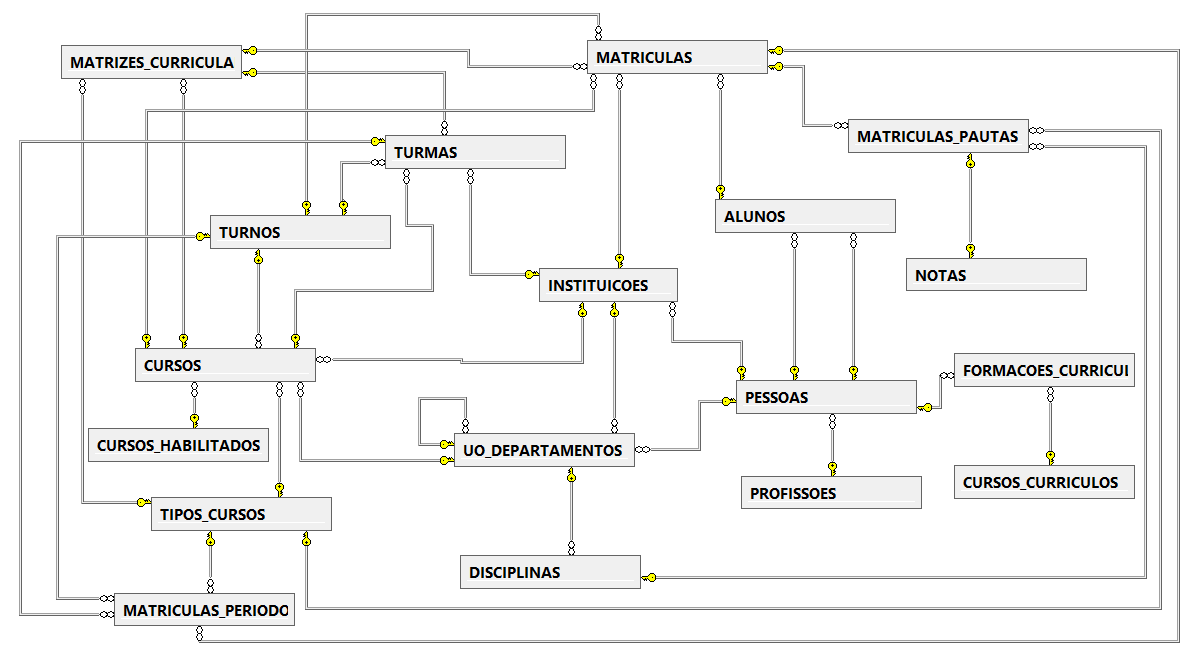
**Aplicação**

MySQL

**Figura 3.5. Arquitetura Simples DDS combinado com o ETL dos dados das fontes de dados disponíveis**. Fonte: Autor (adaptado de Rainardi, 2008).

3.5.1.4.2. **Fontes de Dados**

A principal fonte de dados para esse trabalho é sem dúvida nenhuma, a base de dados do sistema acadêmico do IFRN, onde estão armazenados todos os dados, referentes a vida acadêmica dos alunos, cursos, professores, dados sociais e assim por diante. Portanto, antes de modelar a base de dados dimensional do sistema, irei fazer um detalhamento da base de dados do sistema acadêmico, eliminando as informações desnecessárias ao projeto em desenvolvimento. Isto é necessário, pois a base de dados do sistema acadêmico, contêm informações que não podem ser exibidas ao público em geral, por motivo de segurança e sigilo, como por exemplo os nomes dos alunos e nomes dos pais dos alunos, telefones e assim por diante. Também é fundamental fazer uma seleção das informações que são realmente necessárias, uma vez que, a base de dados do sistema acadêmico, contêm um número de grande de tabelas (exatamente 478 tabelas) e, nem todas essas tabelas contêm informações relevantes para o projeto ora proposto. Como por exemplo, informações sobre recibos, pagamentos e protocolos, que não são de interesse para o projeto. A Figura 3.6 mostra o diagrama do sistema acadêmico, onde foram selecionadas apenas as tabelas que contêm as informações relevantes para o nosso projeto.



**Figura 3.6**. Diagrama com as tabelas selecionadas do sistema acadêmico.

**Fonte**: Autor

Com a fonte de dados modelada, o primeiro passo seguinte, é construir o modelo dimensional (**DDS**), ou seja, modelar as bases de dados dimensional do sistema, onde serão definidos os fatos e suas dimensões correspondentes.

Então, temos que construir um modelo dimensional que contemple informações relacionadas aos fatos da evasão e da repetência escolar, fatores relevantes para o nosso projeto. Dessa forma, o *data warehouse* para essa solução será composto de vários data marts. Cada data mart irá tratar de um fato em especifico e, isso será determinado pela granularidade de cada assunto. Por exemplo, quando estiver analisando dados sobre boletim escolar, temos um fato, pois a granularidade aqui, é baseada nos lançamentos dos dados no boletim do aluno. Neste caso, será analisado o índice da evasão e da repetência em função dos dados contidos no boletim escolar dos alunos. Por outro lado, quando estivermos analisando os dados sociais dos alunos, teremos um outro fato, com outra granularidade, pois temos outras informações diferentes das informações do boletim do aluno. Portanto, para cada tipo de informações lançada no sistema teremos um fato (Boletim, histórico, dados sociais, forma de ingresso nos cursos e assim por diante). Um data mart é composto de tabela de fato e dimensões. Os fatos contêm as medidas, ou seja, os totais, e as dimensões discriminam essas medidas [Kimball, 2002].

3.5.1.4.3. **Modelo Dimensional (DDS)**

O primeiro fato a ser trabalhado é para analisar a situação dos alunos em relação a evasão, o cancelamento de matrículas e o total de alunos que concluíram seus respectivos cursos, totalizados por campus, curso, ano letivo e período letivo. Para isto, foram criadas as dimensões e tabela de fatos, mostrada na Tabela 3.4

**Tabela 3.4. Dimensões e tabela de fatos**.

|  |  |
| --- | --- |
| Dimensão | Comentários |
| dimCursos | Dados referentes aos cursos ministrados no IFRN. |
| dimInstituicao | Cadastro de todos os campi. |
| dimTempo | Dados referentes ao ano ou semestre letivo. |
| dimEvaJubPorCampusCurso | Dados referente aos alunos que evadiram ou foram jubilados dos cursos. |
| DimCanceladoPorCampusCurso | Dados referentes aos alunos que cancelaram suas matrículas nos cursos. |
| dimConcluidosPorCampusCurso | Dados referentes aos alunos que concluíram seus cursos. |
| FatoEvaCanCon | Tabela fatos com os totais de alunos evadidos, que cancelaram matrícula e total de concluintes. |

E as tabelas 3.5 até a tabela 3.11 mostram as estruturas de cada dimensão e da tabela de fatos.

**Tabela 3.5 Dimensão Cursos**

|  |  |  |
| --- | --- | --- |
| Coluna | Tipo de dados | Descrição |
| COD\_CURSO | Int | Código do curso |
| COD\_TURNO | Int | Código do turno |
| COD\_INSTITUICAO | Int | Código da instituição |
| DESC\_CURSO | String | Descrição do curso |
| SIGLA\_CURSO | String | Sigla do curso |
| DATA\_INICIO | Smalldatetime | Data que o curso iniciou |

**Tabela 3.6 Dimensão Instituição**

|  |  |  |
| --- | --- | --- |
| Coluna | Tipo de dado | Descrição |
| COD\_INSTITUICAO | Int | Código da instituição |
| DESC\_INSTITUICAO | String | Descrição por extenso da instituição |
| SIGLA\_INSTITUICAO | String | Sigla da instituição |
| ENDERECO | String | Endereço da instituição |
| CIDADE | String | Cidade onde está localizada a instituição |
| BAIRRO | String | Bairro da cidade onde está localizada a instituição. |
| CEP | String | CEP do endereço onde está localizada a instituição. |

**Tabela 3.7 Dimensão Tempo**

|  |  |  |
| --- | --- | --- |
| Coluna | Tipo de dado | Descrição |
| Ano | Smallint | Ano letivo da ocorrência do fato. |
| Período | Tinyint | Período do ano letivo da ocorrência do fato. |

**Tabela 3.8 Dimensão Evadidos por campus e curso**

|  |  |  |
| --- | --- | --- |
| Coluna | Tipo de dado | Descrição |
| Campus | String | Sigla do campus onde está sendo totalizado os evadidos/jubilados. |
| Curso | String | Sigla do curso onde está sendo totalizado os evadidos/jubilados. |
| Ano Letivo | Smallint | Ano letivo onde está sendo totalizado os evadidos/jubilados. |
| Período letivo | Tinyint | Período letivo onde está sendo totalizado os evadidos/jubilados. |
| TotalEvadidos | Int | Total de alunos evadidos por campus, curso, ano letivo e período letivo. |

**Tabela 3.9 Dimensão Cancelados por campus e curso**

|  |  |  |
| --- | --- | --- |
| Coluna | Tipo de dado | Descrição |
| Campus | String | Sigla do campus onde está sendo totalizado os que cancelaram o curso. |
| Curso | String | Sigla do curso onde está sendo totalizado os alunos que cancelaram o curso. |
| Ano Letivo | Smallint | Ano letivo onde está sendo totalizado os que cancelaram o curso. |
| Período letivo | Tinyint | Período letivo onde está sendo totalizado os que cancelaram o curso. |
| TotalCancelados | Int | Total de alunos que cancelaram o curso, agrupados por campus, curso, ano letivo e período letivo. |

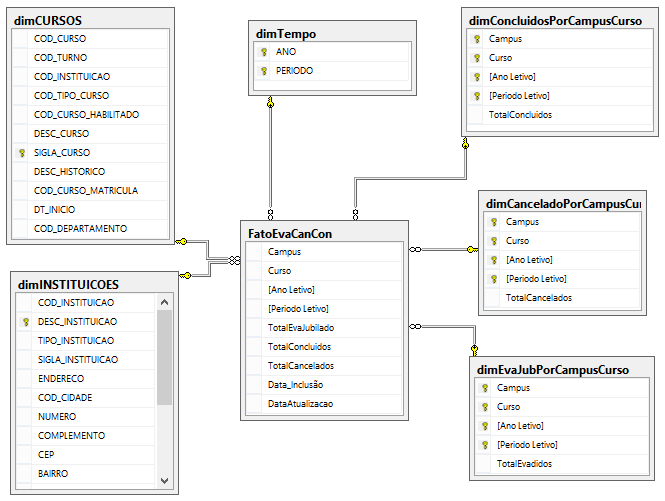
**Tabela 3.10 Dimensão Concluídos por campus e curso**

|  |  |  |  |
| --- | --- | --- | --- |
| Coluna | Tipo de dado | | Descrição |
| Campus | | String | Sigla do campus onde está sendo totalizado os que concluiram o curso. |
| Curso | | String | Sigla do curso onde está sendo totalizado os concluíram. |
| Ano Letivo | | Smallint | Ano letivo onde está sendo totalizado os que concluíram o curso. |
| Período letivo | | Tinyint | Período letivo onde está sendo totalizado os que concluíram o curso. |
| TotalConcluidos | | Int | Total de alunos que concluíram o curso, agrupados por campus, curso, ano letivo e período letivo. |

**Tabela 3.11. Tabela de medidas – Fato Evadidos Cancelados e Concluídos**

|  |  |  |
| --- | --- | --- |
| Coluna | Tipo de dado | Descrição |
| Campus | String | Campus onde ocorreu o fato. |
| Curso | String | Curso onde ocorreu o fato. |
| Ano letivo | Smallint | Ano letivo da ocorrência do fato. |
| Período letivo | Tinyint | Período letivo da ocorrência do fato. |
| TotalEvadidos | Int | Total de alunos evadidos por campus, curso, ano letivo e período letivo. |
| TotalCancelados | Int | Total de alunos que cancelaram matricula por campus, curso, ano letivo e período letivo. |
| TotalConcluidos | Int | Total de alunos que concluíram o curso por campus, curso, ano letivo e período letivo. |
| Data\_Inclusao | Date | Data que ocorreu a inserção dos dados na tabela de fatos. |
| Data\_Atualizacao | Date | Data que ocorreu a atualização da tabela de fatos. |

Observação: a base de dados dimensional foi nomeada de **DDSEducacional**.



**Figura 3.6 Data mart para as medidas de Evasão, cancelado e concluídos**.

**Fonte**: Autor

Concluído o modelo dimensional, o próximo passo, de acordo com a arquitetura da Figura 3.5, é a extração, a transformação e a carga dos dados da fonte de dados para o modelo dimensional. Para esta tarefa, será utilizada consultas SQL, para extrair os dados da base de dados acadêmica para as dimensões e em seguida, será feita a carga na tabela de fatos, também usando uma consulta SQL.

**Listagem 1**: SQL para popular a dimensão dimEvaJubPorCampusCurso

INSERT INTO dimEvaJubPorCampusCurso

(Campus,

Curso,

[Ano Letivo],

[Periodo Letivo],

TotalEvadidos)

SELECT desc\_instituicao AS [Campus],

sigla\_curso AS [Curso],

ano\_let AS [Ano Letivo],

periodo\_Let AS [Periodo Letivo],

sum(QTD\_ALUNO) AS TotalEvadidos

FROM vJuntandoEvaDeslCon

WHERE Situacao\_Da\_Matricula = 'Evasão' OR

Situacao\_Da\_Matricula ='Jubilado'

GROUP BY DESC\_INSTITUICAO, Sigla\_Curso,ANO\_LET, PERIODO\_LET

**Listagem 2**: SQL para popular a dimensão dimCanceladoPorCampusCurso

INSERT INTO dimCanceladoPorCampusCurso

(Campus,

Curso,

[Ano Letivo],

[Periodo Letivo],

TotalCancelados)

SELECT desc\_instituicao AS [Campus],

sigla\_curso AS [Curso],

ano\_let AS [Ano Letivo],

periodo\_Let AS [Periodo Letivo],

sum(QTD\_ALUNO) AS TotalCancelados

FROM vJuntandoEvaDeslCon

WHERE Situacao\_Da\_Matricula = 'Cancelado'

GROUP BY DESC\_INSTITUICAO, Sigla\_Curso,ANO\_LET, PERIODO\_LET

**Listagem 3**: SQL para popular a dimensão dimConcluidosPorCampusCurso

INSERT INTO dimConcluidosPorCampusCurso

(Campus,

Curso,

[Ano Letivo],

[Periodo Letivo],

TotalConcluidos)

SELECT desc\_instituicao AS [Campus],

sigla\_curso AS [Curso],

ano\_let AS [Ano Letivo],

periodo\_Let AS [Periodo Letivo],

sum(QTD\_ALUNO) AS TotalConcluido

FROM vJuntandoEvaDeslCon

WHERE Situacao\_Da\_Matricula = 'Concluido' OR

Situacao\_Da\_Matricula = 'Formadodo'

GROUP BY DESC\_INSTITUICAO, Sigla\_Curso,ANO\_LET, PERIODO\_LET

**Listagem 4**: SQL para popular a tabela de fatos FatoEvaCanCon

INSERT INTO FatoEvaCanCon (Campus,Curso, [Ano Letivo], [Periodo Letivo],

TotalEvaJubilado,

TotalConcluidos,

TotalCancelados,

Data\_Inclusão,

DataAtualizacao)

SELECT coalesce(EJ.Campus,null), coalesce(EJ.Curso,null),

coalesce(EJ.[Ano Letivo],null), coalesce(EJ.[Periodo Letivo],null),

EJ.TotalEvadidos, CO.TotalConcluidos,

CA.TotalCancelados, getdate(), getdate()

FROM dimEvaJubPorCampusCurso EJ inner Join dimConcluidosPorCampusCurso CO

ON EJ.campus = CO.Campus and EJ.Curso = CO.Curso AND

EJ.[Ano Letivo] = CO.[Ano Letivo] AND EJ.[Periodo Letivo] = CO.[Periodo letivo inner Join dimCanceladoPorCampusCurso CA

ON CO.campus = CO.Campus and EJ.Curso = CA.Curso AND

CO.[Ano Letivo] = CA.[Ano Letivo] AND

CO.[Periodo Letivo] = CA.[Periodo letivo] inner Join dimTempo T

ON CA.[Ano Letivo] = T.[ANO] AND

CA.[Periodo Letivo] = T.[PERIODO]

Como se pode observar, em todos os scripts SQL para popular as dimensões, foi utilizada na cláusula FROM, uma exibição de dados. Essa exibição de dados tem a função de juntar os dados relativos aos alunos evadidos/jubilados, que cancelaram suas matrículas em seus respectivos cursos e os alunos que concluíram e se formaram, cuja fonte de dados é o sistema acadêmico. O script dessa exibição de dados é ilustrado na Listagem 5.

**Listagem 5**: SQL para criação da Exibição de dados vJuntandoEvaDeslCon

USE [Sistema\_Academico]

GO

CREATE VIEW [dbo].[vJuntandoEvaDeslCon]

AS

SELECT \* FROM CA\_TOTAL\_ALUNOS\_EVADIDOS

UNION ALL

SELECT \* FROM CA\_TOTAL\_ALUNOS\_DESLIGADOS

UNION ALL

SELECT \* FROM CA\_TOTAL\_ALUNOS\_CONCLUIDOS

GO

As dimensões dimCursos e dimInstituicoes foram populadas, fazendo-se uma cópia, dos dados das tabelas do sistema acadêmico, diretamente para as dimensões, uma vez que não era necessário se fazer nenhuma mudança nos dados.

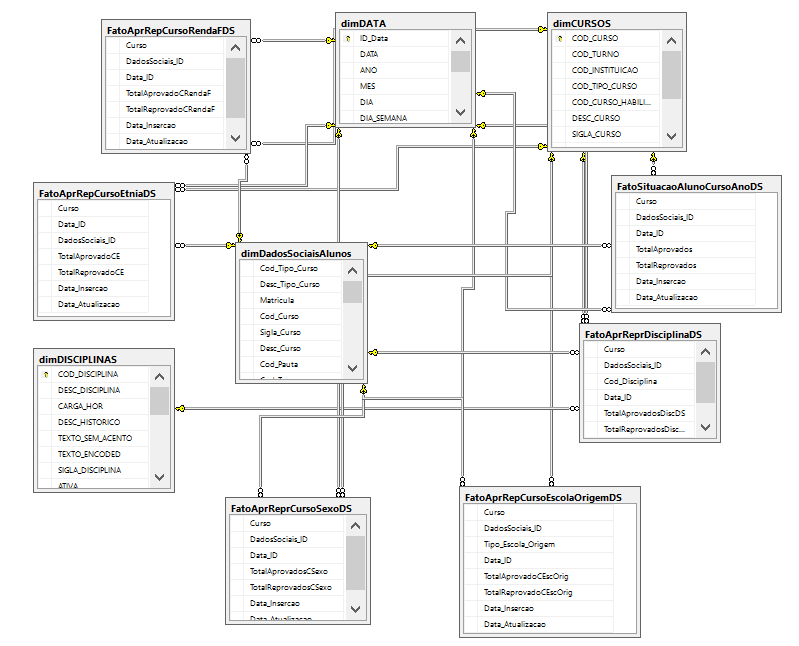
O segundo, data mart, a ser modelado é para analisar repetência e a evasão escolar, em função dos dados sociais dos alunos, contidos no sistema acadêmico. As medidas de fatos irão totalizar a quantidade de alunos reprovados em função dos dados sociais dos alunos. Por exemplo, desejo saber o total de alunos reprovados por curso, renda familiar, escola de origem, sexo, etnia, disciplina e assim por diante e, ainda agrupados por anos e semestre letivo. A tabela 3.12 mostra as dimensão e fatos que foram definidos.

**Tabela 3.12** Lista de dimensões e tabelas de fatos

|  |  |
| --- | --- |
| Dimensão ou tabela de fato | Comentários |
| dimData | Identifica do tempo onde ocorreu o fato. |
| dimCursos | Identifica o curso onde ocorreu o fato. |
| dimDisciplinas | Identifica a disciplina na qual ocorreu o fato. |
| dimDadosSociaisAlunos | Esta tabela contém o cadastro do aluno e seus dados sociais (Renda familiar, escola de origem, etnia, sexo, e assim por diante). |
| FatoAprRepCursoRendaFDS | Total de Aprovados e reprovados, agrupados por curso e renda familiar. |
| FatoAprRepCursoEtniaDS | Total de Aprovados e reprovados, agrupados por curso e etnia. |
| FatoAprRepCursoSexoDS | Total de Aprovados e reprovados, agrupados por curso e sexo. |
| FatoAprRepCursoEscolaOrigemDS | Total de Aprovados e reprovados, agrupados por curso e escola de origem. |
| FatoAprRepCursoDisiciplinaDS | Total de Aprovados e reprovados, agrupados por curso e disciplina. |
| FatoAprRepCursoAnoDS | Total de Aprovados e reprovados, agrupados por curso e ano letivo. |

A Figura 3.9 mostra o diagrama do modelo dimensional ou *data mart* para este cenário. A Tabela 3.13 mostra o detalhamento da dimensão dimDadosSociaisAlunos.

Para popular as dimensões e as tabelas de fatos o data mart da Figura 3.9, foi utilizada o script SQL, disponíveis no Anexo A.



**Figura 3.9**. Modelo dimensional em função dos dados sociais dos alunos.

**Fonte**: Autor

**Tabela 3.13** Dimensão Dados Sociais dos Alunos

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| Código\_Tipo\_Curso | Smallint | Indica o tipo do curso |
| Desc\_Tipo\_Curso | String | Descreve o tipo do curso |
| Matrícula | String | Matrícula do aluno no curso |
| Cod\_Curso | Int | Código do curso |
| Sigla\_Curso | String | Sigla do curso |
| Desc\_Curso | String | Descrição completa do curso |
| Cod\_Pauta | Int | Código da pauta está relacionada a disciplina em pauta |
| Cod\_Turma | Int | Código da turma a qual o aluno pertence |
| Sigla\_Disciplina | String | Sigla da disciplina |
| Ano\_Let | Smallint | Ano letivo onde ocorreu o fato |
| Período\_Let | Tinyint | Período letivo onde ocorreu o fato |
| N\_faltas | Int | Total de faltas |
| Situação | String | Situação do aluno no momento (Aprovado, reprovado, jubilado, evadido e assim por diante) |
| Media\_final | Decimal | Indica a média final do aluno no ano/período letivo. |
| Forma\_de\_Ingresso | String | Indica a forma de ingresso do aluno no curso (Exame de seleção, ENEM, transferido, e assim por diante) |
| Sexo | String | Indica o sexo do aluno |
| Renda\_Familiar | Decimal | Indica a renda familiar do aluno |
| Procedencia\_Escola\_Origem | String | Informa a área de procedência da escola de origem (urbana ou rural). |
| Tipo\_Escola\_Origem | String | Informa se a escola de origem é pública estadual ou pública federal ou privada. |
| Etnia | String | Informa a etnia do aluno (branca, parda, negra, e assim por diante) |
| Reside | String | Informa se o aluno reside ou não com os pais. |

O terceiro fato, está relacionado, aos dados de entrada e saída dos alunos nos cursos. Pretende-se com esse fato, fazer uma projeção da evasão escolar, de cada curso em cada campus, ao longo dos anos. Tenho também, através de fato, a possibilidade de calcular quantos alunos terminam os seus cursos no prazo determinado. Isso me dá a possibilidade de calcular o gasto extra que foi feito com esses alunos repetentes. A Tabela 3.14, lista as dimensões e tabela de fato.

**Tabela 3.14** Dimensões e fato para o “FatoEvaEntradaSaida”

|  |  |
| --- | --- |
| Dimensão ou fato | Descrição |
| dimCursos | Cadastro de curso |
| dimData | Cadastro de datas |
| dimInstituicoes | Cadastrado de campus |
| dimAlunosSituacaoDadoIngresso | Esta dimensão contêm os dados referentes a sua situação no curso. |
| FatoEvaEntradaSaida | Esta tabela de fato contêm as medidas total de evadidos e concluídos. |

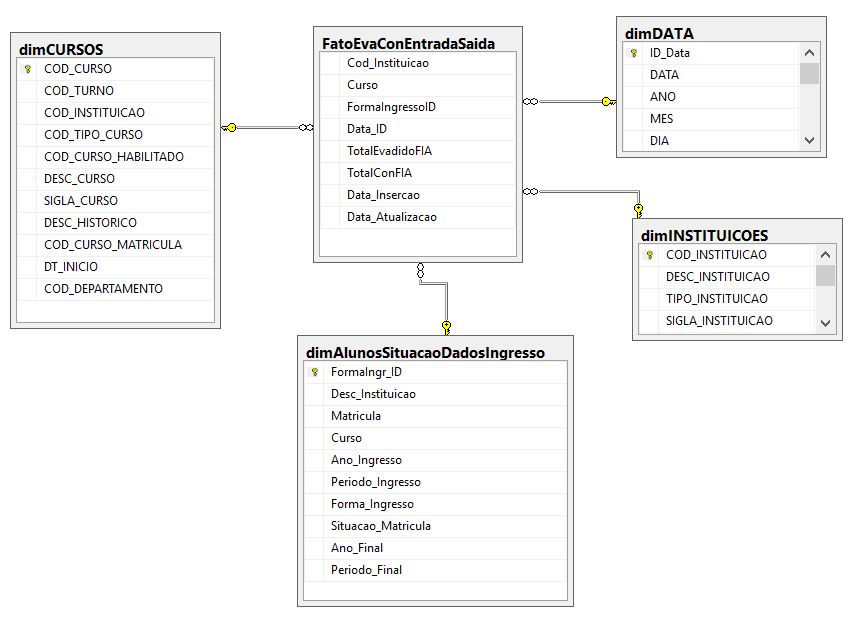
A Tabela 3.15 mostra o detalhamento da dimensão dimAlunosSituacaoDadoIngresso.

**Tabela 3.15** Detalhamento da dimensão “dimAlunosSituacaoDadoIngresso”.

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| FormaIngr\_ID | Int | Chave primaria da tabela |
| Desc\_Instituicao | String | Nome do campus. |
| Matricula | String | Matricula do aluno. |
| Curso | String | Nome do curso. |
| Ano\_Ingresso | Smallint | Ano em que o aluno ingressou no curso. |
| Perido\_Ingresso | Tinyint | Semestre em que o aluno ingressou no curso |
| Forma\_Ingresso | String | Como o aluno ingressou no curso (exame de seleção, ENEM, transferido, etc.) |
| Situacao\_Matricula | String | A situação do aluno no curso no momento (concluído, evadido, matriculado, etc.) |
| Ano\_Final | Smallint | Ano de saída do aluno no curso. |
| Período\_Final | Tinyint | Semestre de saída do aluno do curso. |

A Figura 3.10 mostra o diagrama do modelo dimensional ou *data mart* para este cenário. A Tabela 3.13 mostra o detalhamento da dimensão dimDadosSociaisAlunos.

Para popular as dimensões e as tabelas de fatos o data mart da Figura 3.10, foi utilizada o script SQL, disponíveis no Anexo A.



**Figura 3.10** Data mart para o fato “FatoEvaEntradaSaida”.

**Fonte**: Autor

Uma outra fonte de dados, são as planilhas eletrônica. O sistema acadêmico fornece a opção de extrair dados para planilhas eletrônicas. Uma dessas opções é a possibilidade de extrair o boletim e o histórico escolar dos alunos.

Então, será criada uma tabela com os dados dos boletins escolares para se analisar a taxa de repetência escolar, também em função desses dados.

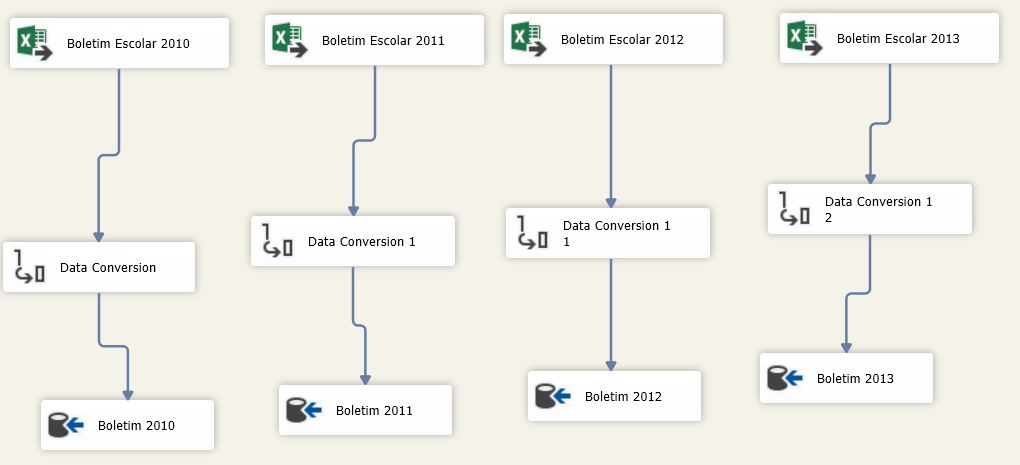
Com os dados dos boletins escolares, será possível fazer um acompanhamento, bimestre a bimestre, da taxa de aprovação e reprovação dos alunos e também analisar a situação da turma. A tabela 3.16 mostra a estrutura da tabela “BoletimEscolar”.

**Tabela 3.16** Detalhamento da dimensão BoletimEscolar

|  |  |  |
| --- | --- | --- |
| Atrinuto | Tipo | Descrição |
| BoletimID | Int | Chave primária da tabela |
| Matricula | String | Matricula do aluno |
| NomeAluno | String | Nome do aluno |
| AnoLetivo | Smaillint | Ano letivo do lançamento do fato |
| PeriodoLetivo | Tinyint | Período letivo do lançamento do fato |
| Curso | String | Curso que o aluno está matriculado |
| Turma | String | Turma a que o aluno pertence |
| Campus | String | O campus onde é ministrado o curso |
| Situação | String | Situação do aluno no momento do lançamento do fato (aprovado, reprovado, e assim por diante) |
| PercentualFrequencia | Decimal | Percentual de frequência do aluno |
| SiglaCurso | String | Sigla do curso |
| CoeficienteRendimento | Decimal | Coeficiente de rendimento do aluno no curso |
| Media | Decimal | A média obtida pelo aluno no período |

Para popular esta tabela, foi utilizada a ferramenta de **ETL** do Microsoft SQL Server, chamada de *SQL Server Integration Service* (**SSIS**). O SSIS é uma ferramenta para integração, consolidação e transformação de dados e usada para ETL por desenvolvedores, administradores de bases de dados (DBA) e, analistas de dados, no processo de extração, transformação e carga dos dados, das fontes de dados para a base consolidada de dados [Rossum, 2015][Knight, 2012]. A Figura 3.11 mostra o diagrama do pacote **ETL** para extrair os dados das planilhas Excel para o banco de dados dimensional (base consolidada). A base de dados consolidada é um banco de dados relacional do Microsoft SQL Server. Na Figura 3.11, temos os dados do boletim escolar sendo extraído de planilhas Excel e em seguida, esses dados passam por conversão de tipos de dado e sendo, então, gravados no banco de dados consolidada.

**A Figura 3.11** Pacote ETL para extrair os dados das planilhas Excel para o banco de dados dimensional (DDSEDucacional).



**Fonte**: Autor

A figura 3.12 mostra o modelo dimensional em função dos dados do boletim escolar dos alunos.



**Figura 3.12** Modelo dimensional para os dados relacionados ao boletim escolar.

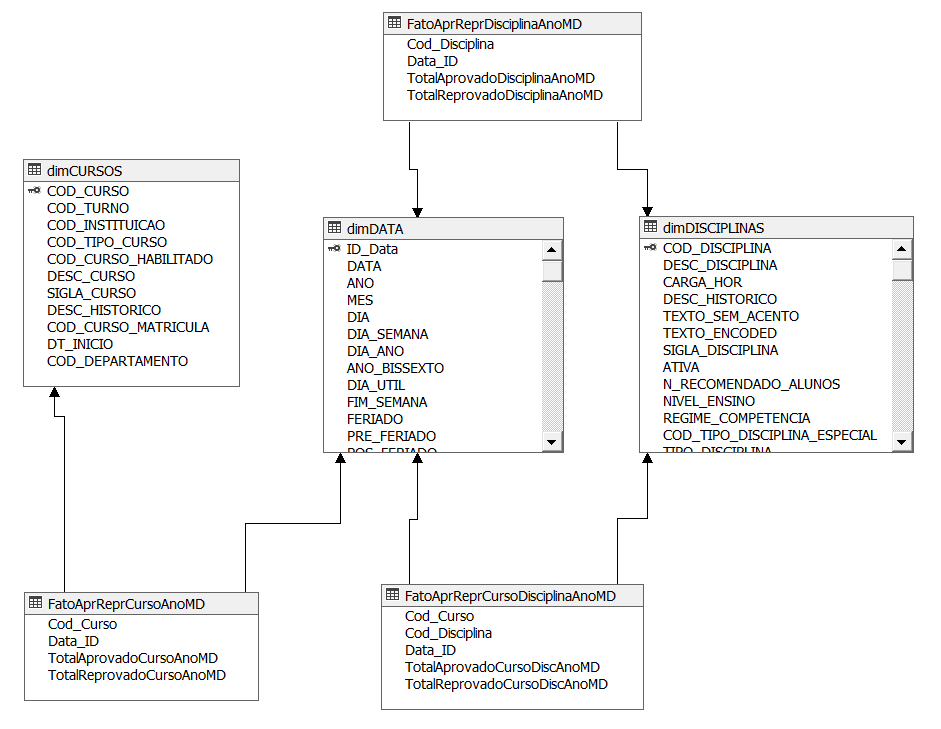
**Fonte**: Autor

A dimensão cursos já foi detalhada anteriormente. Então, a tabela 3.17 mostra o detalhamento da dimensão dimBoletimEscolar e da tabela de fato, desse modelo dimensional.

**Tabela 3.17** Detalando os atributos da dimensão dimBoletim

|  |  |  |
| --- | --- | --- |
| Tabela | Atributo | Descrição |
| dimBoletimEscolar | Boletim\_ID | Chave primária |
| Matricula | Matrícula do aluno |
| AnoLetivo | Ano letivo em curso |
| PeriodoLetivo | Semestre em curso |
| Curso | Código do curso em que o aluno está matriculado. |
| Turma | Turma do aluno no curso |
| Campus | Nome do campus onde o aluno está matriculado |
| Situação | Aprovado, reprovado ou trancado |
| PercentualFrequencia | Percentual de frequência do aluno no semestre |
| SiglaCurso | Sigla do curso |
| Frequência | Frequencia total do aluno no semestre |
| Rendimento | Desempenho do aluno no semestre |
| MediaFinal | Nota final do aluno no semestre. |

No próximo cenário, será analisado a taxa de repetência escolar por curso, disciplina e ano, ou seja, através desses dados, será possível extrair relatório que mostre a taxa de repetência escolar por Curso e/ou disciplina agrupados por ano letivo. A figura 3.13 mostra o modelo dimensional para esse cenário.



**Figura 3.13** Modelo dimensional para análise de repetência escolar agrupado por curso, disciplina e ano letivo **Fonte**: Autor

As tabelas 3.18, 319 e 3.20 faz um detalhamento de cada tabela de fato do modelo 3.13 respectivamente.

**Tabela 3.18** Detalhamento das tabelas de fato FatoAprReprDisciplinaAnoMD do modelo 3.13.

|  |  |
| --- | --- |
| Atributo | Descrição |
| Cod\_disciplina | Chave estrangeira para a dimensão dimDisciplina |
| Data\_ID | Chave estrangeira para a dimensão dimData |
| TotalAprovadoDisciplinaAnoMD | Total de aprovado agrupados por disciplina e ano |
| TotalReprovadoDisciplinaAnoMD | Total de Reprovado agrupados por disciplina e ano |

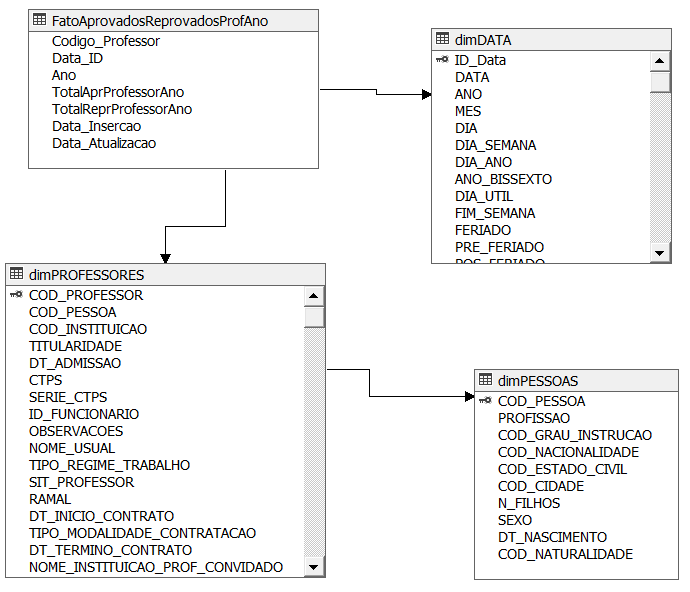
**Tabela 3.19** Detalhamento das tabelas de fato FatoAprReprCursoDisciplinaAnoMD do modelo 3.13.

|  |  |
| --- | --- |
| Atributo | Descrição |
| Cod\_Curso | Chave estrangeira para a dimensão dimCursos |
| Cod\_Disciplina | Chave estrangeira para a dimensão dimDisciplina |
| TotalAprovadoCursoDiscAnoMD | Total de aprovados agrupados por curso, disciplina e ano |
| TotalReprovadoCursoDiscAnoMD | Total de reprovados agrupados por curso, disciplina e ano |

**Tabela 3.20** Detalhamento das tabelas de fato FatoAprReprCursoAnoMD do modelo 3.13.

|  |  |
| --- | --- |
| Atributo | Descrição |
| Cod\_Curso | Chave estrangeira para a dimensão dimCursos |
| Data\_ID | Chave estrangeira para a dimensão dimData |
| TotalAprovadoCursoAnoMD | Total de aprovados agrupados por curso e ano |
| TotalReprovadoCursoAnoMD | Total de reprovados agrupados por curso e ano |

No próximo cenário, será analisado a taxa de repetência escolar por professor. Através desse estudo de caso, será possível extrair relatório que mostre a taxa de repetência escolar por professor. Este relatório poderá ser utilizado pela equipe pedagógica, juntamente com os professores, sobre as reprovações e, com isso resolver alguns problemas relacionados a repetência escolar, na tentativa de diminuir essa taxa. A figura 3.14 mostra o modelo dimensional para esse cenário.



**Figura 3.14** Modelo dimensional para dados relacionados a repetência escolar agrupados por professor.

**Fonte**: Autor

A tabela 3.20 detalha a tabela FatoAprovadosReprovadosProfAno, seus atributos e a descrição de cada um deles.

**Tabela 3.20** Detalhamento da tabela de fato FatoAprovadosReprovadosProfAno

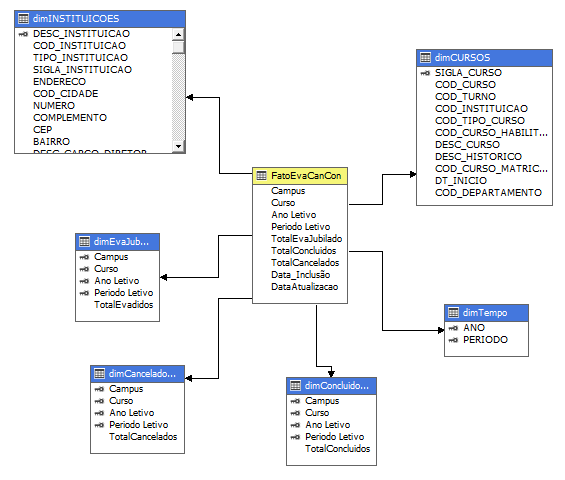
|  |  |
| --- | --- |
| Atributo | Descrição |
| Código\_Professor | Chave estrangeira para a dimensão dimPorfessores |
| Data\_ID | Chave estrangeira para a dimensão dimData |
| TotalAprovadoProfAno | Total de alunos aprovados agrupados por professor e ano |
| TotalReprovadoProfAno | Total de alunos reprovados agrupados por professor e ano |

3.5.1.4.4. **Modelo Multidimensional (MDB)**

Nesta etapa será desenvolvido o banco de dados multidimensional. Um banco de dados multidimensional é uma forma de banco de dados, onde os dados são armazenados em células e a localização de cada célula é definida por hierarquias chamadas de dimensões. Cada célula representa um evento de negócio, e os valores das dimensões mostram quando e onde este evento ocorreu [Rainardi, 2008].

A estrutura multidimensional, armazena as agregações geradas, bem como os dados da base de dados, em formato de matriz, em vez de tabelas relacionais. Os valores das agregações são pré-calculadas, sumarizadas em função dos dados da base de dados [Rainardi, 2008].

Banco de dados multidimensionais são tipicamente usados para **BI**, especialmente para **OLAP** e mineração de dados. A Figura 3.12 mostra a estrutura de cubo o fato FatoEvaCanCon. Este cubo será utilizado para análise ds evasão escolar em função dos dados obtidos da base de dados fonte (sistema acadêmico), onde foram feitas consultas a essa base de dados e foram extraídos os dados sobre evasão e jubilamento escolar (dimEvaJubilmento), dados sobre matrículas canceladas pelos alunos (dimCancelados), dados sobre os alunos que concluíram os seus respectivos cursos (dimConcluidos).

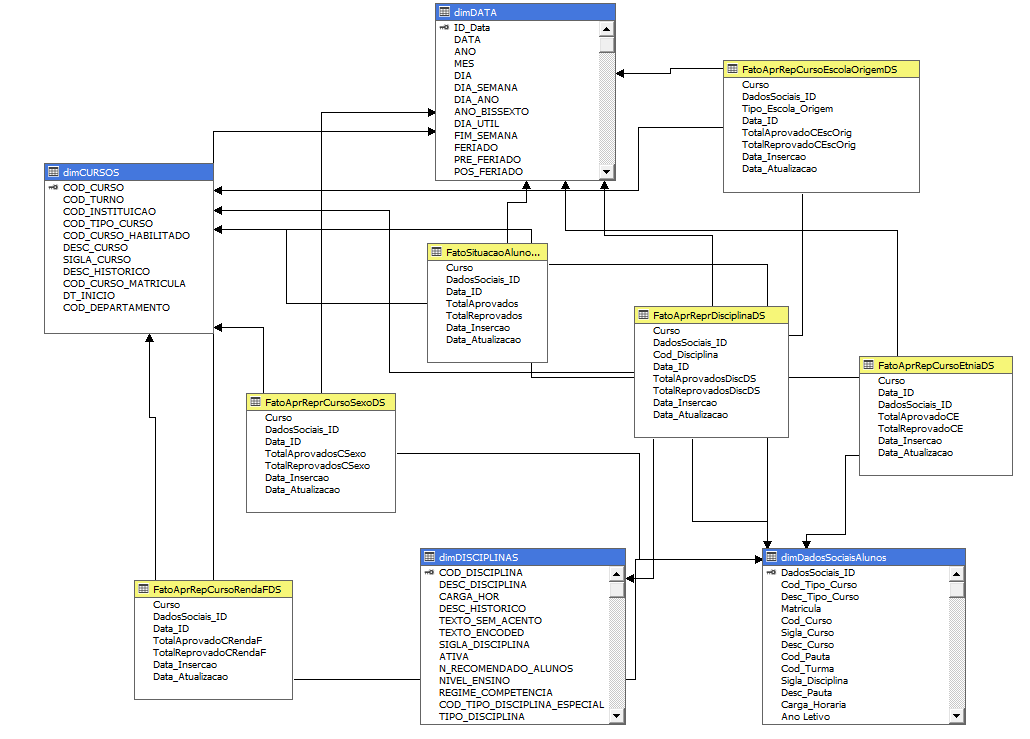


**Figura 3.12**: **Estrutura do Cubo para o fatoEvaCanCon**.

**Fonte**: Autor

A Figura 3.13 mostra o modelo multidimensional para os fatos relacionados aos dados sociais dos alunis, tais renda familiar, tipo de escola de origem, sexo e etnia. Este cubo servirá para a análise da repetência escolar. Como se pode observar, este modelo é uma constelação de fatos [Kimball, 2002].

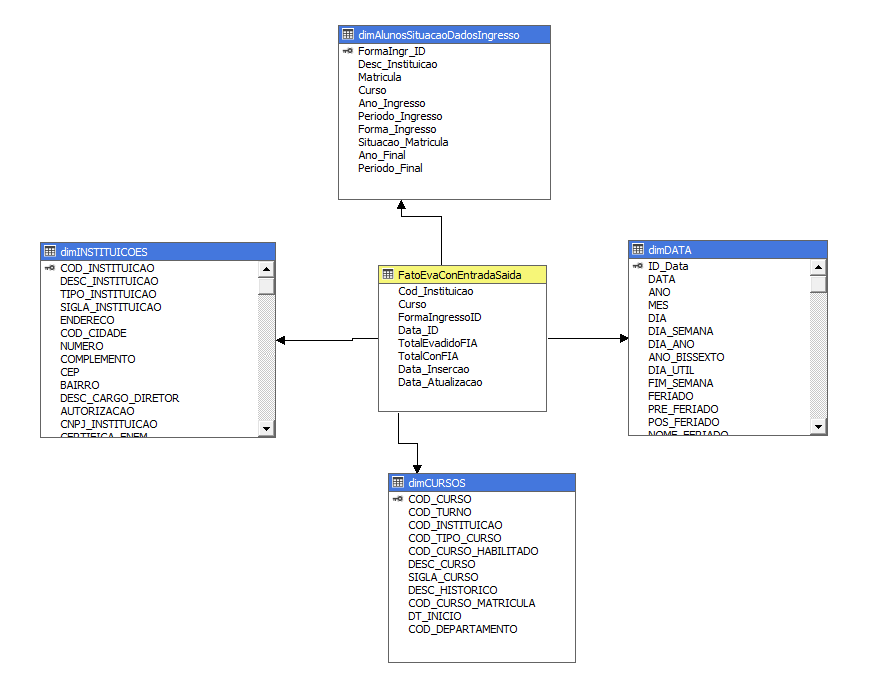
A principal dimensão, nesse cubo é a dimensão dimDadosSociaisAlunos, pois a mesma contém os lançamentos de onde serão calculados os totais de alunos reprovados e ou aprovados, agrupados ora por etnia, ora por renda familiar e assim por diante. Esta tabela, contém aproximadamente 1 milhão de registros, se for considerado apenas os anos de 2010 até 2014.



**Figura 3.13** Estrutura de cubo criada para os dados sociais dos alunos.

**Fonte**: Autor.

A Figura 3.14 mostra o modelo multidimensional para o fato “FatoEvaEntradaSaida”. Este cubo será utilizado tanto para geração de relatórios administrativos, exibindo relatórios com informações sobre a evasão escolar, com também será utilizado pelos algoritmos de mineração de dados para a descoberta de relacionamentos entre os atributos que estão relacionados com a evasão escolar, mostrando o grau de influência de cada atributo nesse relacionamento.

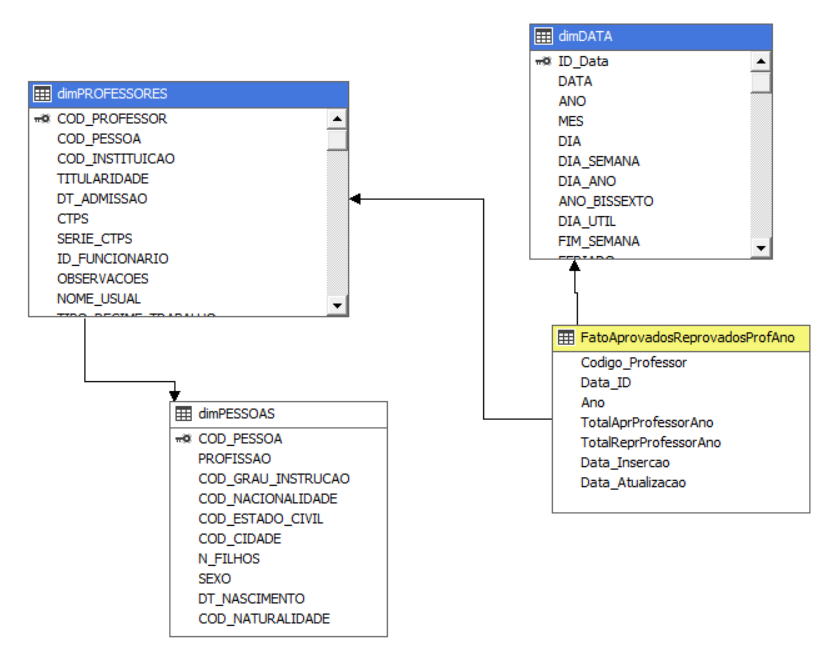


**Figura 3.14** Estrutura de cubo criada para o fato “FatoEvaEntradaSaida”.

**Fonte**: Autor.

A Figura 3.15 mostra o modelo multidimensional para o fato “FatoAprovadosReprovadosProfAno”. Este cubo será utilizado para geração de relatórios administrativos, exibindo relatórios com informações sobre a repetência escolar por professor, podendo ser representado por semestre e ano.

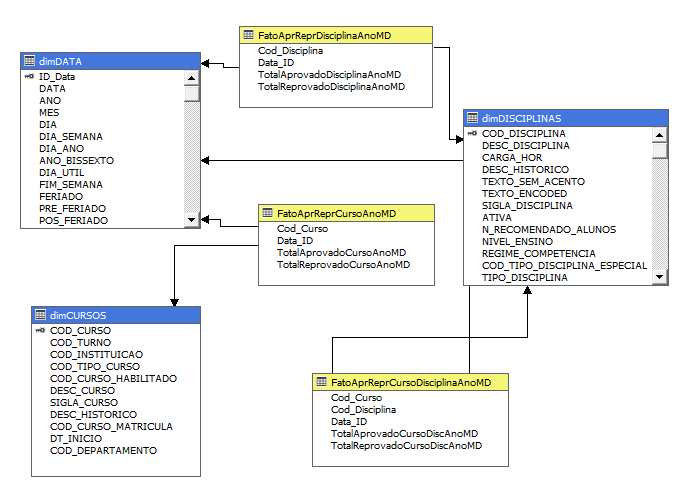
Devo lembrar que, muitos dos relatórios administrativos, serão gerados a partir dos dados do modelo dimensional. Também, alguns modelos de mineração de dados serão aplicados diretamente sobre os dados de tabelas do modelo dimensional. Isso, deve-se pelo fato de que algumas tabelas do modelo dimensional, já reúnem em seus dados todas as informações necessárias para a execução do algoritmo de mineração de dado, ou seja, essas tabelas já foram previamente preparadas para os devidos algoritmos.



**Figura 3.15** Estrutura de cubo para o fato “FatoAprovadosReprovadoProfAno”.

**Fonte**: Autor

A Figura 3.16 mostra o modelo multidimensional para os fatos relacionados a repetência escolar em função das médias anuais das disciplinas nos repectivos cursos. Este cubo será utilizado tanto para geração de relatórios administrativos, exibindo relatórios com informações sobre a repetência escolar, com também será utilizado pelos algoritmos de mineração de dados, na tentaiva de identificar a influencia dos atributos de entrada sobre a repetência escolar.



**Figura 3.16** Modelo multimdimensional para as medidas aprovados e reprovados por disciplina e curso

**Fonte**: Autor.

**3.5.2 Projetando Agregações e Hierarquias**

Até o momento foram projetadas as medidas e as dimensões dos cubos criados. Vamos, a partir desse ponto, considerar a possibilidade de otimização desses cubos, criando agregações e hierarquias de usuário.

As agregações melhoram o desempenho das consultas sem aumentar excessivamente o espaço de disco necessário para armazenar dados no cubo. Por outro lado, as hierarquias permitem que os usuários naveguem por um cubo com mais eficiência, e também contribui para aperfeiçoar o desempenho da consulta [Harinath, 2012].

Um cubo de processamento analítico online (**OLAP**) parece conter cada valor totalizado possível por cada atributo de cada dimensão. Por exemplo, o cubo da Figura 3.14, contém as medidas TotalEvadidoFIA e TotalConFIA e, as dimensões dimCursos, dimAlunosSituacaoDadosIngresso, dimInstituicoes e dimTempo. Se você fizer uma consulta no cubo em um nível de detalhe mais baixo, por exemplo, qual é o total de evadidos do curso 01025 do campus CNAT em 2010, o cubo retornará um valor, como se estivesse armazenado diretamente nele. Ou seja, parece que o cubo contém todos os valores totalizados possíveis em todos os níveis de detalhes para cada dimensão. Considerando que a dimensão dimAlunosSituacaoDadosIngresso tem 104060 registros e, que a dimensão dimCursos tem 696 registros e, que a dimensão dimInstituicoes tem 21 registros e que dimensão dimData tem 10959 registros. Isto nos dá um total de combinações da ordem de 10406 x 696 x 21 x 10959 = 16.667.991.980.640 combinações possíveis e, isso é denominado de explosão de dados. A explosão de dados é um dos principais problemas com os cubos OLAP e todos os produtos **OLAP** precisão lidar com isso de alguma maneira [Jacobson, 2005][Harinath, 2012].

De acordo com Jacobson (2005), a maneira mais fácil de evitar a explosão de dados é evitar armazenar as agregações juntas, calculando-as sob demanda. No entanto, quando o *data warehouse* é realmente grande, essa opção afeta rapidamente o desempenho, porque solicitar um único valor totalizado de alto nível do cubo requer recuperar e somar centenas ou milhares de valores da fonte de dados. E como já foi falado, o desafio do OLAP é tornar as consultas mais rápidas possível evitando a explosão de dados.

A ferramenta utilizada neste trabalho, o *Analysis Services do Microsoft SQL Server* fornece vários recursos que permitem ao administrador de banco de dados controlar e fazer o ajuste fino no relacionamento entre o tamanho físico do cubo e a velocidade das consultas. Dentre esses recursos pode-se citar as agregações e as hierarquias.

As agregações são totalizações pré-calculadas de dados detalhados que habilitam o servidor Analysis a responder consultas rapidamente. Quando for solicitado um valor de um cubo, o Analysis Services usa qualquer agregação que esteja disponível para recuperar o valor o mais rápido possível [Jacobson, 2005].

Uma partição no Analysis Services é um local físico de dados armazenados do cubo. Todo cubo possui pelo menos uma partição padrão. Sempre que se cria um grupo de medidas, outra partição é criada. As consultas são executadas rapidamente em um cubo particionado porque o Analysis Services precisa apenas ler os dados das partições que contêm as respostas das consultas. Além dos mais, as consultas podem ser executadas ainda mais rápido quando a partição também armazenar agregações [Jacobson, 2005].

3.5.2.1. **Projetando agregações para a partição do cubo da Figura 3.14**

Por padrão o cubo tem apenas uma partição, como mostra a Figura 3.16. As informações sobre a partição são exibidas em uma tabela: Partition Name, Source, Aggregation e Description.

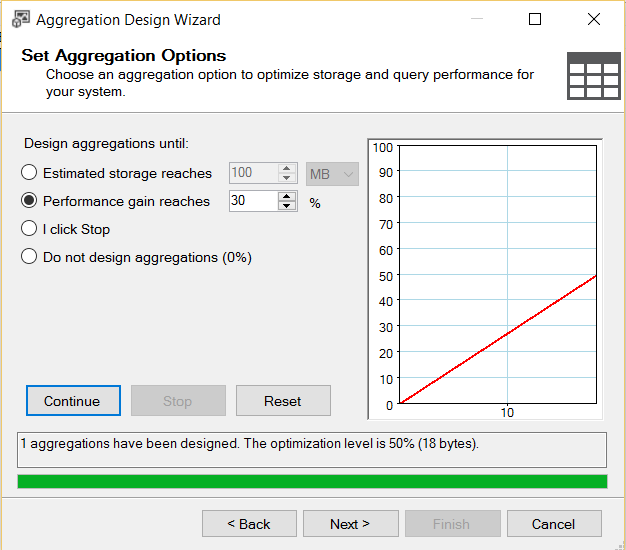
Será usado o assistente de agregações para calcular as agregações possíveis até esgotar todas as possibilidades para ganho de desempenho especificado, ou seja, desejo uma performance de 30% em função do espaço em disco. A Figura 3.17 mostra o resultado obtido, onde foi gerado uma agregação para uma performance em torno de 50% e 18kbytes de espaço em disco.



**Figura 3.16** Partição padrão do cubo Fato Evadidos Entrada Saída.

**Fonte**: Autor.

**Figura 3.17** Calculando agregações para o cubo Fata Evadidos Entrada Saida.

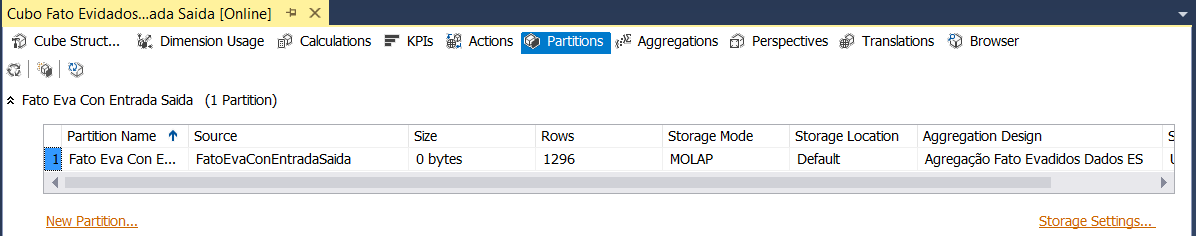


**Fonte**: Autor.

Ao selecionar a opção mostrada na Figura 3.17, para projetar agregações, o assistente calculará as agregações possíveis até esgotar todas as possibilidades para o ganho de desempenho especificado. Se você depois achar que as consultas estão sendo executadas de forma muito lenta, será possível adicionar agregações, basta para isso modificar os valores de ganho de desempenho.

No gráfico, pode-se ver a relação entre o aumento de desempenho (no eixo y) e a quantidade de espaço em disco consumido (no eixo x) como resultado do projeto de agregação atual. Deve ser lembrado que o objetivo da otimização da agregação é obter o melhor ganho de desempenho sem consumir espaço em disco desnecessários.

Agora o cubo, apesar de continuar com apenas a mesma partição padrão, ele tem agora, uma agregação, como mostra a Figura 3.18.



**Figura 3.18** Partição padrão com a agregação calculada pelo assistente.

**Fonte**: Autor.

Veja na Figura 3.18 que a contagem de linhas (coluna Rows), calculada pelo assistente de agregações, foi estimado em 1296 linhas. Este valor reflete a contagem de linhas da tabela de fatos e, esse valor é um valor estimado pelo assistente e, possivelmente difere da contagem de linhas real. Entretanto, muitas vezes, para se obter um projeto de agregação mais preciso, você pode informar um valor próximo do valor real. Deve ser observado que, as contagens são usadas apenas para o projeto de agregações, mas pode resultar em um projeto de agregação que esteja abaixo do ideal se você subestimar a contagem de linhas.

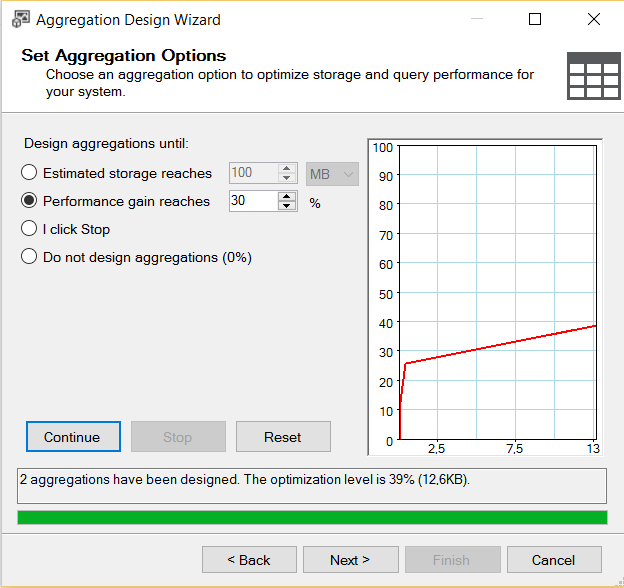
Através do assistente, você tem a possibilidade de modificar o valor da contagem de linhas para obter um projeto de agregação mais preciso. A Figura 3.18 mostra o valor alterado para um milhão de linhas. Veja também que você pode alterar a contagem de linhas para cada tabela que faz parte do cubo. Observe que o assistente estimou a contagem de linhas para cada tabela do cubo. Quando a tabela tem poucas linhas ele estimou o valor real de linhas, mais quando a tabela tem muitas linhas, geralmente o valor estimado é um valor bem menor do que o real. Então, cabe ao administrador do sistema fazer os ajustes das contagens de linhas para que o projeto de agregações seja mais preciso.



**Figura 3.18** Alterando o valor da contagem de linhas.

**Fonte**: Autor.

Após ser feita a alteração do valor das contagens de linha, deve-se fazer mais uma vez o ajuste do projeto de agregações. A Figura 3.20 mostra o novo projeto de agregações gerardo após as mudanças realizadas.



**Figura 3.20** Otimização do projeto de agregações.

**Fonte**: Autor.

Pode-se observar que, após a mudança da contagem de linhas, mostrada na Figura 3.20, o assistente gerou duas agregações, para uma performance de 39% e ocupação de espaço em disco de 12,6 KB. Observe que, para as contagens padrão do assistente, a performance era de 50%, para espaço em disco de 18KB. Portanto, par esse ajuste, houve uma diminuição da performance, mas um ganho de espaço em disco. Então, a regra de negócio é quem vai determinar qual das configurações é a melhor para o sistema. Devo lembrar que esses ajustes vão ocorrer durante todo o ciclo de vida do sistema.

3.5.2.2. **Adicionando atributos ao projeto de agregação**

Apenas os atributos de mais alto nível de hierarquia e de granularidade estão disponíveis para consideração de agregação pelo assistente de projeto de agregação. O padrão atende a maioria dos projetos de agregação. No entanto, quando se tem certos atributos que são usados com frequência e possui um número de membros relativamente pequena, é possível pensar em adicioná-lo ao pool de possíveis candidatos para a agregação. No nosso caso, tem um atributo da dimensão “dimAlunosSituacaodadosIngresso” denominado Situacao\_Matricula que tem poucos membros e é bastante consultado. Portanto, esse atributo será configurado para ser um atributo de agregação. Para isso, basta alterar a propriedade AggregationUsage para Full e depois recalcular as agregações. A Figura 3.21 mostra a alteração feita para o atributo “Situacao\_Matricula”.



**Figura 3.21** Configurando a propriedade AggregationUsage do atributo Situacao\_Matricula para Full.

**Fonte**: Autor.

Após a execução do assistente de agregações, o mesmo faz a contagem de linha para o atributo “Situacao\_Matricula”, pois agora o mesmo é um atributo de agregação e atribue ao mesmo o valor 4. Isso porquer o atributo só possui quantro membros, a saber Matriculado, Cancelado, Concluído e Evadido. Como mostra a Figura 3.22.



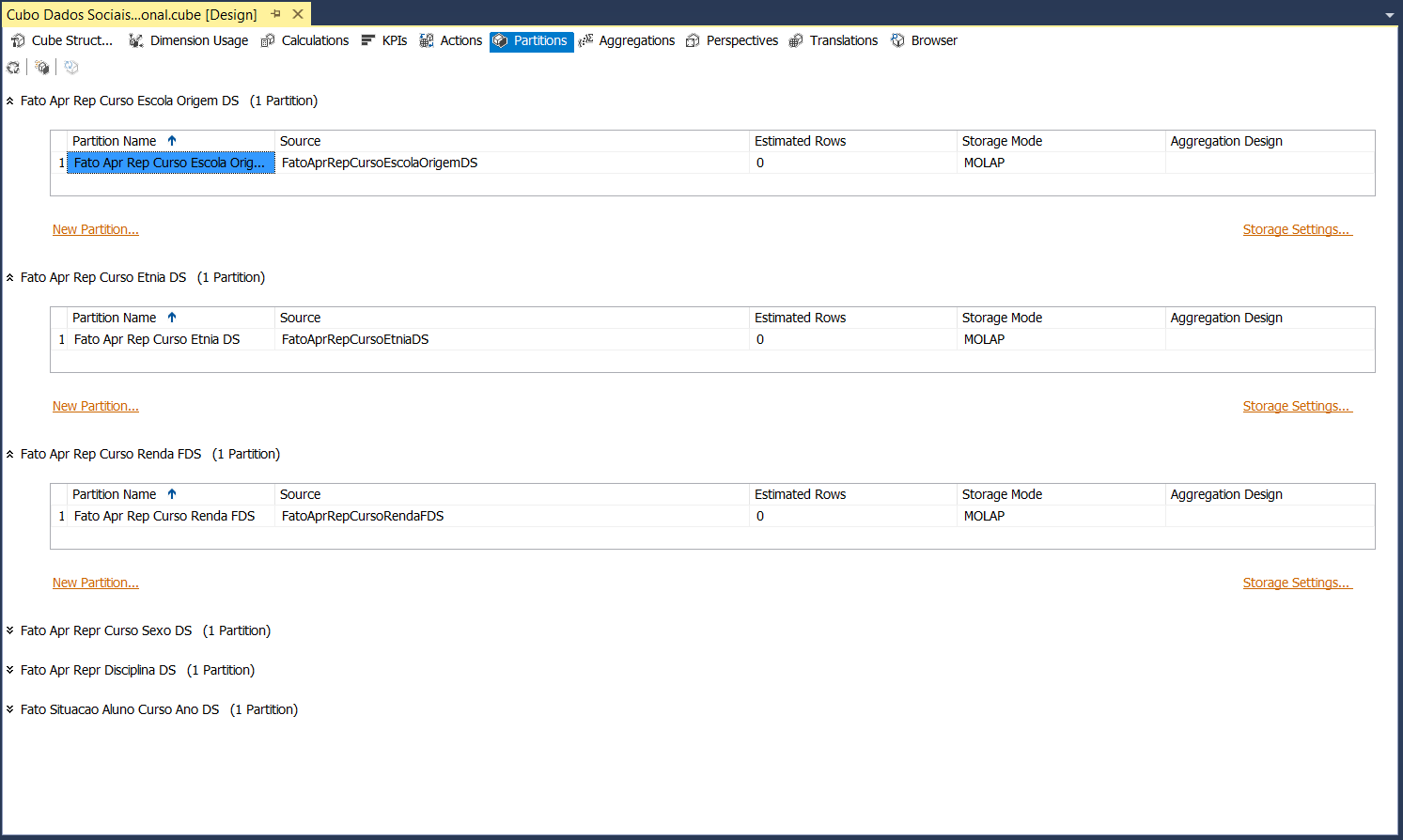
**Figura 3.22** O atributo “Situacao\_Matricula” como uma agregação.

**Fonte**: Autor.

A figura 3.22 mostra o calcula de agregações para o cubo relacionado aos dados sociais dos alunos, o cubo da figura 3.13. Este cubo é composto por seis fatos e quatro dimensões. Então, como já foi explicado anteriormente, é gerado automaticamente para cada fato uma partição em função de suas medidas. E para cada nova medida criada ou hierarquias criadas são geradas outras agregações. Como já foi dito, uma agregação é uma totalização pré-calculada de dados detalhados que habilitam o servidor a responder consultas raoidamente.

Uma outra coisa que preocupa os administradores de dados é a explosão de dados. No caso específico do cubo dados sociais, isso deve ser levado em conta pois, a dimensão dimDadosSoicia, só ela tem mais de milhão (precisamente 1074226).

A figura 3.23 mostra as partições padrão criadas para cada fato do cubo em função de suas medidas, como ainda não foi criado nenhuma outra medida e também ainda não foram definidas hierarquias para as dimensões, a informação de “Estmated Rows” é zero e, “Aggregation Desgn” está vazio.

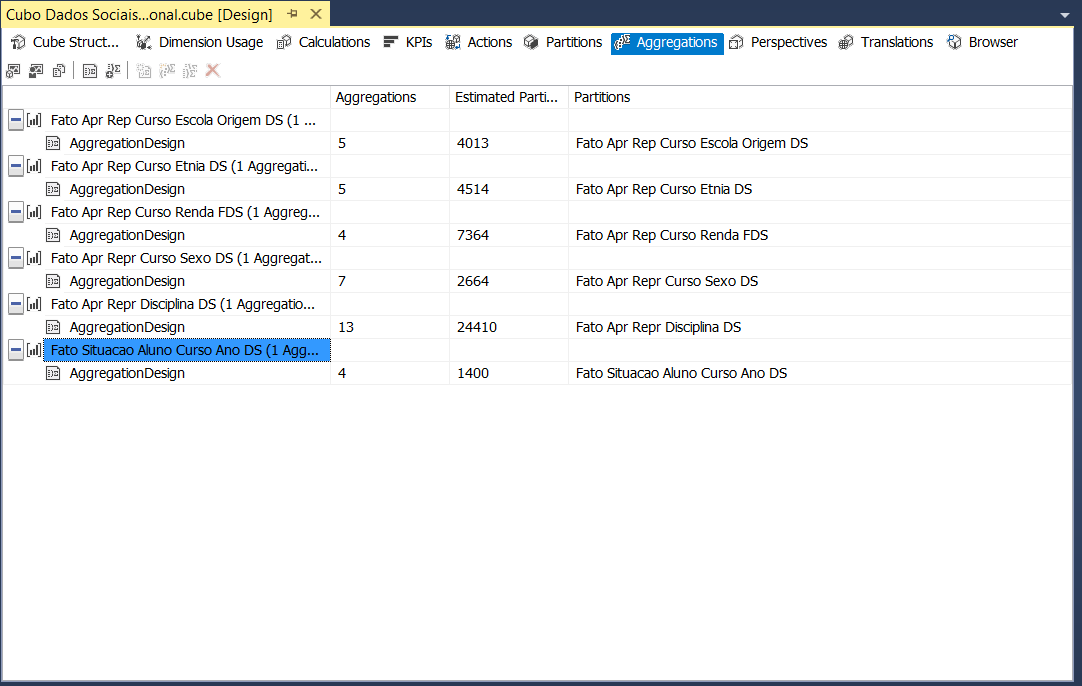


**Figura 3.23** Partições padrão do cubo Dados Sociais.

**Fonte**: Autor.

A Tabela 3.21 detalha as hierarquias criadas para os cubos do sistema. E após ter sido criadas as hierarquias da tabela 3.21, foi executado o assistente de agregação para o cálculo das agregações. Para esse estudo de caso, foi levado em conta o desempenho de 30% em função de um baixo uso de disco e, também foi assumido o padrão de estimativa de linha, ou seja, não foi alterado o valor padrão da contagem de linhas de cada tabela. A figura 3.24 mostra a quantidade de agregações geradas para cada tabela de fato.

Como já foi dito, o processo de otimização do desempenho das consultas, não é um processo estande, muito pelo contrário, é um processo que deve ser realizado em toda vida do sistema em produção. A todo instante você pode está adicionando novas medidas e criando novas hierarquias e, portanto, reprocessando o cálculo de agregações. Além do mais, existe, como foi mostrado na seção anterior, o recurso de se fazer os ajustes das agregações, com base no log de consultas. Ou seja, o administrador do sistema, ativa a gravação em log, de todas as consultas realizadas pelos usuários e, então como base nas informações armazenadas no log, fazer a otimização das consultas.



**Figura 3.24** Agregações criadas para cada tabela de fato do cubo dados sociais.

**Fonte**: Autor.

**3.5.3. Projetando Hierarquias**

Quando o usuário está navegando por um cubo, ele pode usar as hierarquias de atributos para facilitar esta tarefa. O usuário ao invés de arrastar para as colunas e linhas do navegador do cubo, ou até mesmo, para as colunas ou linhas de uma tabela dinâmica do Excel, os atributos individualmente, ele pode arrastar uma hierarquia de atributos, que tenham sido “empacotados previamente” essa coleção de atributos como uma hierarquia. Portanto, serão criadas algumas hierarquias de atributos para esse cubo em estudo no momento, para facilitar a navegação pelo cubo por parte dos usuários.

Para exemplificar o usuo das hierarquias, tome como exemplo a seguinte situação. Se o usuário desejar consultar o curso por código, sigla e descrição do curso. Então, pode-se empacotar esses atributos em uma hierarquia, como mostra a tabela 3.21 a hierarquia da tabela Curso “ “CODIGO – SIGLA - CURSO.

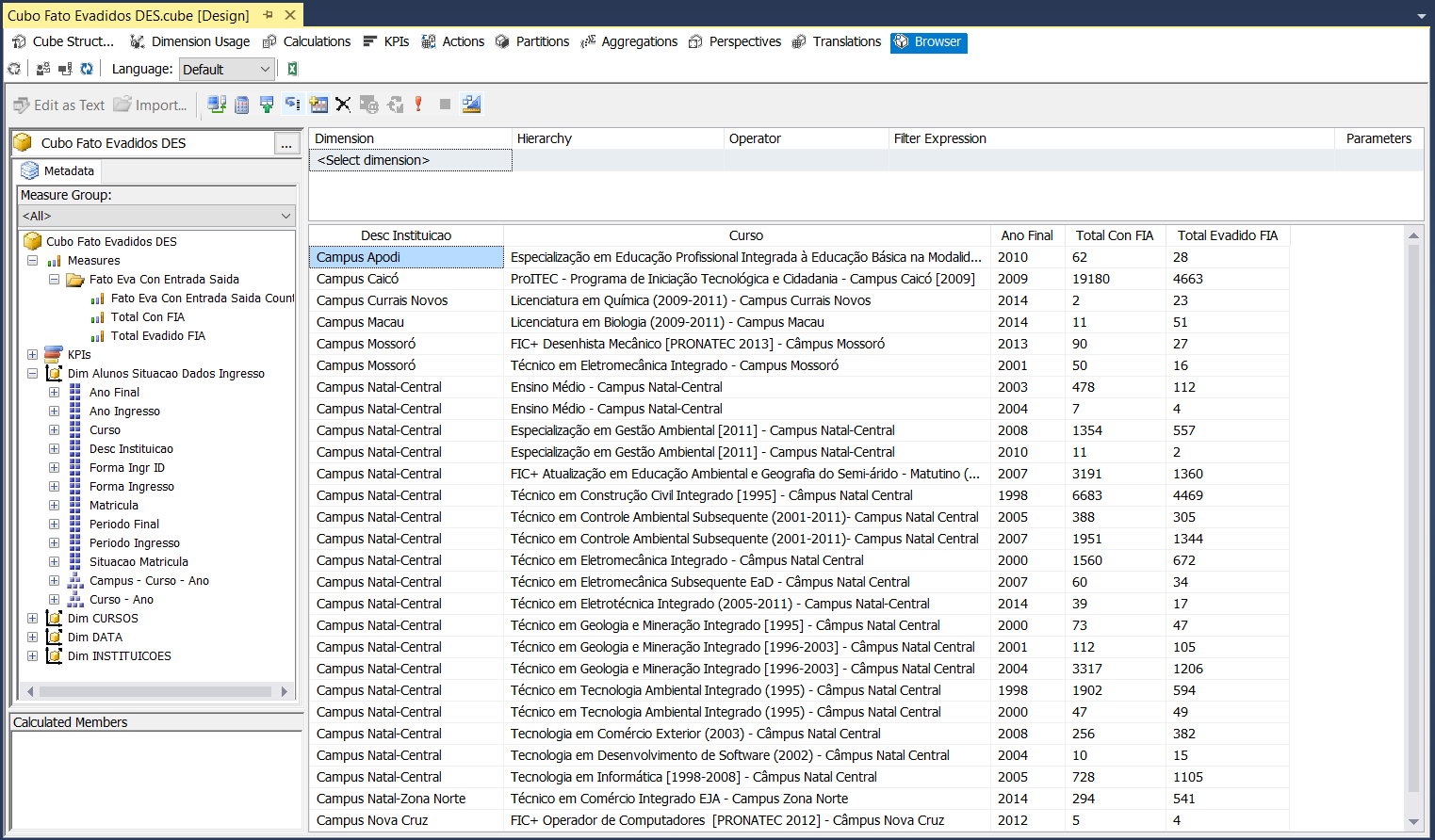
**Tabela 3.21** Hierarquias criadas para o cubo dados sociais

|  |  |
| --- | --- |
| Hierarquia | Estrutura de atributos |
| Tabela Dados Sociais | |
| Curso – Silga | Desc\_Curso 🡪 Sigla\_Curso |
| Renda – Escola – Etnia | Renda\_Familiar 🡪 Tipo\_Escola\_Origem 🡪 Etnia |
| Curso – Forma Ingresso | Desc\_Curso 🡪 Forma\_De\_Ingresso |
|  |  |
| Tabela Data | |
| Ano – Semestre | Ano 🡪 Semestre |
|  |  |
| Tabela Cursos | |
| Curso – Sigla | Desc\_Curso 🡪 Silga\_Curso |
| Codigo – sigla- curso | Cod\_Curso 🡪 Sigla\_Curso 🡪 Desc\_Curso |
| Campus – Curso – Data | Codigo\_Instituicao 🡪 Desc\_Curso 🡪 Data\_Inicio |
|  |  |
| Tabela Disciplinas |  |
| Disciplina – Sigla | Desc\_Disciplina 🡪 Sigla |
| Tabela dados Ingresso |  |
| campus – curso - ano | Desc\_Instituicao 🡪 Desc\_Curso 🡪 Ano Final |
| curso – ano | Desc\_Curso 🡪 Ano Final |
|  |  |
| tabela Professores |  |
| Professor – Regime trabalho | Nome\_Usual 🡪 Tipo\_Regime\_Trabalho |

Para exemplicar a utilização das hierarquias, será montado um relatório Ad Hoc, que mostre os totais das situações dos alunos, por instituição, o curso e o ano. Para isto será utilizado o próprio navegador de cubo do Microsoft Analysis Services. A Figura 3.25 mostra o resultado obtidos.

Observe na Figura 3.25 que foi expandida a dimensão dim Alunos Situacao Dados Ingresso, onde são exidos todos os seus atributos e também as hierarquias criadas previamente. Para este exemplo, foi arrastado para a coluna do navegador a hierarquia “Campus – Curso – Ano” e para a área de dados as medidas

**Figura 3.25** Navegador do cubo mostrando o relatório montado com a hierarquia e as medias Total Concluído (TotalCon FIA) e Total Evadidos (TotalEvadido FIA)



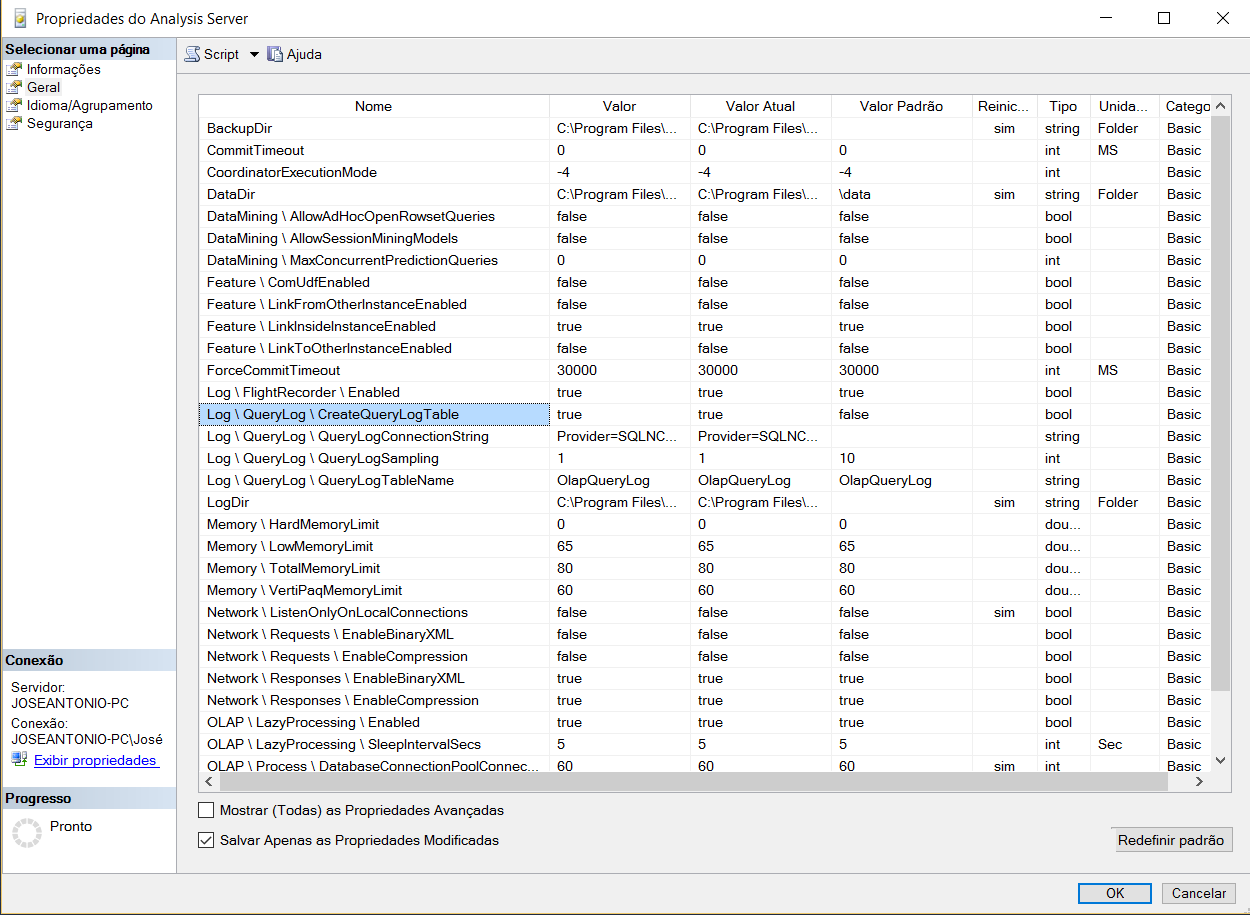
**Fonte**: Autor.

**3.5.3.1. Otimizando as agregações**

Como foi explanado na seção anterior, o assistente de agregações seleciona as agregações de um pool padrão de possíveis agregações. O pool de possíveis agregações inlcui todos os atributos de granularidade de cada dimensão. Foi mostrado também que podemos alterar esse pool alterando as propriedades dos atributos e adicionando hierarquias de usuário de atributos relacionados. No entanto, essa estratégia de projeto é adequada para cubos novos que esteja sendo colocado em produção, mas pode não ser otimizada para os usuários. Então, para um sistema em produção, a melhor estratégia para projetar agregações otimizadas é considerar os dados reais das consultas solicitadas pelos usuários do cubo[Jacobson, 2005][Harinath, 2012].

Os padrões de uso dos cubos são provenientes do log de consultas. O log de consulta pode ser armarzenado em um banco de dados relacional ou em um arquivo em disco. O servidor do Analysis Service tem como padrão registrar 1 de 10 consultas.

Para esse exemplo será criado um log de consultas que será preenchido com as consultas executadas no cubo. Para isto, será criado um banco de dados extra denominado “ASQueryLog”, no servidor SQL Server. Criado o banco de dados, o próximo passo é configurar o Analysis Services para iniciar o processo de registro em log. Isto é feito mudando a propriedade Log \ QueryLog \ CreateQueryLogTable do Analysis Services para true. A Figura 3.26 mostra essa configuração.



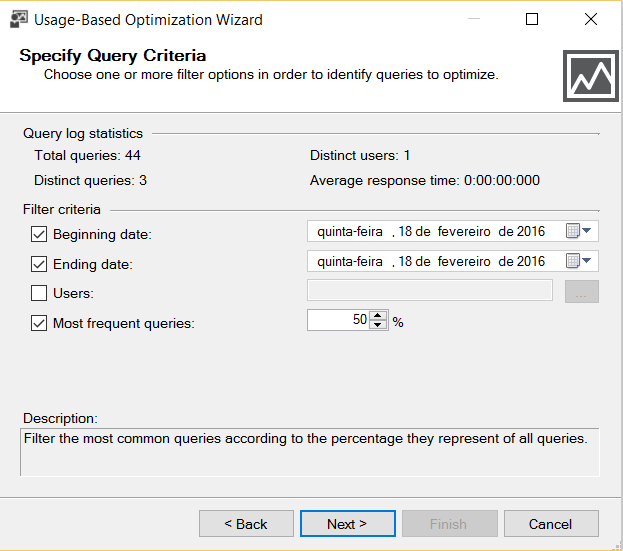
**Figura 3.26** Configurando o Analysis Services para registra as consultas nos cubos em uma tabela em um banco de dados relacional.

**Fonte**: Autor.

Observe que a propriedade Log \ QueryLog \ QueryLogTableName está padronizada para OlapQueryLog. Então, quando é finalizada a operação de configura, a tabela é adicionada ao banco de dados ASQueryLog que foi criado. É claro, que se pode editar esse nome se preferir.

A medida que os usuários forem solicitando consultas aos cubos, vão sendo registras informações na tabela de log e você pode a qualquer instante consultar esses dados.

Com as consultas gravadas no log, pode-se fazer a otimização das agregações com base nesse log. O Analysis Service usa o assistente “Usage-Based Optimization Wizard” como mostra a Figura 3.27.



**Figura 3.27** Otimizando agregações com base no registro de log de consultas.

**Fonte**: Autor.

O assistente de otimização de agregação, permite que se possa filtrar as consultas por vários critérios, como data de início e fim das consultas e também pela frequência com que essas consultas foram executadas, para se obtenha uma melhor otimização possível das agregações.

**3.5.4. Adicionando medidas calculdas ao cubo**

Se necessário, pode-se criar medidas calculadas em um cubo para estendem a capacidade analítica do mesmo. Isto significa que além das medidas definidas no cubo, você terá medidas extras que auxiliarão na análise dos dados. Por exemplo, para o nosso caso, onde se estar trabalhando com total de evadidos e conclusão, pode-se criar duas medidas calculadas para representar o percentual de evadidos e de concluídos.

O modelo multidimensional usa a linguagem **MDX** (um pseudo-acrônimo para Expressões Multidimensionais) como linguagem de consulta, para recuperar relatórios de um cubo e, como uma linguagem de expressões, usada para calcular valores. Portanto, será utilizada a **MDX** para criar as medidas calculadas, do percentual de evadidos e do percentual de concluídos. A listagem 3.1 mostra essas medidas.

**Listagem 3.1** Novas medidas calculadas para algumas tabelas de fato dos cubos usando a linguagem **MDX**.

**FatoEvaConEntradaSaida:**

**PercentualConclusão**: Cálcula o percentual de alunos concluídos

**MDX**: ([Measures].[Total Concluido]\*100)/([Measures].[Total Concluido]+[Measures].[Total Evadido])

**PercentualEvadidos**: Cálcula o percentual de alunos evadidos

**MDX**: (([Measures].[Total Evadido]\*100)/([Measures].[Total Concluido]+[Measures].[Total Evadido]))

**PercentualReprovadoEscolaOrigem**: Calcula o percentual de reprovados por escola de origem

**MDX**: ([Measures].[Total Reprovado C Esc Orig]\*100)/([Measures].[Total Aprovado C Esc Orig]+[Measures].[Total Reprovado C Esc Orig])

**FatoAprRepCursoEscolaOrigem FDS:**

**PercentualAprovadoEscolaOrigem**: Calcula o percentual de aprovados por escola de origem

**MDX**: ([Measures].[Total Aprovado C Esc Orig]\*100)/([Measures].[Total Aprovado C Esc Orig]+[Measures].[Total Reprovado C Esc Orig])

**FatoAprRepCursoEtnia FDS:**

**PercentualAprovadoPorCursoEtnia**: Calcula o percentual de aprovados por curso e etnia

**MDX**: ([Measures].[Total Aprovado CE]\*100)/([Measures].[Total Aprovado CE]+[Measures].[Total Reprovado CE])

**PercentualReprovadoPorCursoEtnia**: Calcula o percentual de reprovados por curso e etnia

**MDX**: ([Measures].[Total Reprovado CE]\*100)/([Measures].[Total Aprovado CE]+[Measures].[Total Reprovado CE])

**FatoAprRepCursoRenda FDS:**

**PercentualAprovadoPorCursoRenda**: Calcula o percentual de aprovados por curso e renda

**MDX**: ([Measures].[Total Aprovado C Renda F]\*100)/([Measures].[Total Aprovado C Renda F]+[Measures].[Total Reprovado C Renda F])

**PercentualReprovadoPorCursoRenda**: Calcula o percentual de reprovados por curso e renda

**MDX**: ([Measures].[Total Reprovado C Renda F]\*100)/([Measures].[Total Aprovado C Renda F]+[Measures].[Total Reprovado C Renda F])

**FatoAprRepCursoSexo**:

**PercentualAprovadoPorCursoSexo**: Calcula o percentual de aprovados por curso e sexo

**MDX**: ([Measures].[Total Aprovados C Sexo]\*100)/([Measures].[Total Aprovados C Sexo]+[Measures].[Total Reprovados C Sexo])

**PercentualReprovadoPorCursoSexo**: Calcula o percentual de aprovados por curso e sexo

**MDX**: ([Measures].[Total Reprovados C Sexo]\*100)/([Measures].[Total Aprovados C Sexo]+[Measures].[Total Reprovados C Sexo])

**FatoAprRepDisciplina:**

**PercentualAprovadoPorDisciplina:** Calcula o percentual de aprovados por disciplina

**MDX:** ([Measures].[Total Aprovados Disc DS]\*100)/([Measures].[Total Aprovados Disc DS]+[Measures].[Total Reprovados Disc DS])

**PercentualReprovadoPorDisciplina:** Calcula o percentual de reprovados por disciplina

**MDX:** ([Measures].[Total Reprovados Disc DS]\*100)/([Measures].[Total Aprovados Disc DS]+[Measures].[Total Reprovados Disc DS])

**FatoAprRepCursoAno:**

**PercentualAprovadoPorCursoAno:** Calcula o percentual de aprovados por curso e ano

**MDX:** ([Measures].[Total Aprovados]\*100)/([Measures].[Total Aprovados]+[Measures].[Total Reprovados])

**PercentualReprovadoPorCursoAno:** Calcula o percentual de Reprovados por curso e ano

**MDX:** ([Measures].[Total Reprovados]\*100)/([Measures].[Total Aprovados]+[Measures].[Total Reprovados])

**FatoAprRepDisciplinaAnoMD:**

**PerAprovadoDisciplina:** Calcula o percentual de aprovados por disciplina e ano

**MDX:** ([Measures].[Total Aprovado Disciplina Ano MD]\*100)/([Measures].[Total Aprovado Disciplina Ano MD]+[Measures].[Total Reprovado Disciplina Ano MD])

**PerReprovadoDisciplina:** Calcula o percentual de reprovados por disciplina e ano

**MDX:** ([Measures].[Total Reprovado Disciplina Ano MD]\*100)/([Measures].[Total Aprovado Disciplina Ano MD]+[Measures].[Total Reprovado Disciplina Ano MD])

**FatoAprRepCursoDiscAnoMD**:

**PerAprovadoCursoDisc**: Calcula o percentual de aprovados por curso e disciplina e ano

**MDX:** ([Measures].[Total Aprovado Curso Disc Ano MD]\*100)/([Measures].[Total Aprovado Curso Disc Ano MD]+[Measures].[Total Reprovado Curso Disc Ano MD])

**PerReprovadoCursoDisc**: Calcula o percentual de reprovados por curso e disciplina e ano

**MDX:** ([Measures].[Total Reprovado Curso Disc Ano MD]\*100)/([Measures].[Total Aprovado Curso Disc Ano MD]+[Measures].[Total Reprovado Curso Disc Ano MD])

**FatoAprRepCursoAnoMD:**

**PerAprovadosCurso**: Calcula o percentual de aprovados por curso e ano

**MDX**: ([Measures].[Total Aprovado Curso Ano MD]\*100)/([Measures].[Total Aprovado Curso Ano MD]+[Measures].[Total Reprovado Curso Ano MD])

**PerReprovadosCurso**: Calcula o percentual de Reprovados por curso e ano

**MDX**: ([Measures].[Total Reprovado Curso Ano MD]\*100)/([Measures].[Total Aprovado Curso Ano MD]+[Measures].[Total Reprovado Curso Ano MD])

**3.5.5. Indicadores-chave de desempenho** (**KPI**)

Um **KPI** (Key performance indicator) mede o progresso que uma empresa está tendo rumo ao cumprimento de suas metas. Nesse sentido, neste trabalho serão criados alguns KPIs. Para o cubo do fato “**FatoAprRepEntradaSaida**”, será criado um **KPI** para acompanhar a evolução da evasão escolar, segundo uma meta definida a ser atingida, que será de uma taxa de evasão escolar menor do que 10%.

Para criara um **KPI** precisa-se desenvolver expressões que calculam o valor, meta, status atual e tendência do **KPI**. A expressão Valor representa o valor atual e às vezes é mencionado como real quando comparado a uma meta. A expressão Meta define a meta a ser alcançada. A expressão Status mede como a expressão Valor se compara a meta. A expressão Tendência é usada para comparar o valor atual de Status com o valor dele em um ponto anterior no tempo. A listagem 3.2 MDX mostra as definições do KPI “KPI Evasão Escolar”.

**Listagem 3.2 Código** **MDX** para o **KPI** Evasão Escolar

**VALOR**:

([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido]))

**META**:

([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido]))<10.0

**STATUS**:

Case

When ([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido]))<10.0 Then 1

When (([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido]))>=10.0)

(([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido]))<=15.0) Then 0

When (([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido])) > 15.0) Then -1

End

**TENDÊNCIA**:

Case

When IsEmpty([Data].[Calendar].PrevMember) then 1

When

([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido])) >

([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido],[Data].[Calendar].PrevMember) ) then 1

When

([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido])) <

([Measures].[Total Evadido]\*100/([Measures].[Total Concluido]+[Measures].[Total Evadido],[Data].[Calendar].PrevMember) ) then -1

End

A declaração de tendência, compara a porção do período de tempo atual com o período de tempo anterior na hierarquia calendar. Por exemplo, o período de tempo anterior a janeiro de 2010 é dezembro de 2009. Como também, se o período de tempo for o primeiro semestre de 2010, o período anterior será o segundo semestre de 2009.

**3.5.6. Gerenciando Partições em Cubos OLAP**

Como foi mostrado na seção anterior, quando é criado um cubo OLAP, o sistema criar para cada tabela de medidas, uma partição como padrão. No entanto, para banco de dados muito grande, em muitas situações, é desejado o particionamento do banco de dados.

Como já se sabe, cada cubo **OLAP** consiste de pelo menos uma partição. Deve-se saber que, o processo de planejamento de armazenagem é o mesmo, quer o cubo tenha uma ou várias partições [Harinath, 2012][Jacobson, 2005].

Um dos benefícios de criar várias partições é que, pode-se projetar armazenagens defirentes para partes diferentes do cubo. Por exemplo, você pode projetar uma partição com informações mais recentes. Se essas informações são acessadas com frequência, então, pode ser especificado o armazenamento **OLAP** multidimensional (**MOLAP**) com agregações para proporcionar um aumento de desempenho de 50%. Uma outra partição poderia ser criada para armazenar as informações de dois, três ou vários anos anteriores, por exemplo. As informações desses anos, provavelmente, são acessadas a nível de totalizações e ocasionalmente, assim pode-se especificar a armazenagem (**HOLAP**), com agregações para o nível de desempenho de 30%.

Um benefício muito importante das partições, é você poder processar uma partição independentemente do restante do cubo. Imagine você um cubo muito grande que tenha que ser processado a cada 10 minutos. Em raríssima exceção, teríamos tempo de processar um grande cubo a cada dez minutos. Então, uma saída seria particionar o cubo, de forma que as informações que precisam ser processadas a cada dez minutos, ficassem em uma partição independete, e então, aí sim, essa partição poderia ser processada isoladamente, a cada dez minutos, sem comprometer o desempenho do sistema.

Para este projeto, o cubo dados sociais da figura 3.13, tem vários fatos, formando uma constelação de fatos. No entanto, cada fato não ultrapassa um total de 3000 linhas de dados. Dessa forma, não justifica ser fazer um particionamento desse cubo.

3.6. **Módulo Mineração de Dados**

Devido à complexidade do processo de descoberta de conhecimento em base de dados (**KDD**), será adotada uma extensão ou adaptação da metodologia **CRISP-DM** (Cross-Industry Stantard Process for Data Mining), pois esse modelo foi dos primeiros e bem aceitos para esse processo [GoldSchmit, 2015].

A metodologia de **KDD** escolhida recomenda que a execução do processo ocorra de forma iterativa e interativa. Nessa execução, dependendo dos resultados alcançados, os especialistas em **KDD,** podem retornar a qualquer etapa realizada anteriormente para fazer refinamentos em busca de melhores resultados. Dessa forma, a metodologia requer uma documentação detalhada das ações realizadas e dos resultados produzidos.

O modelo **CRISP-DM**, utiliza dois conceitos de controles em sua aplicação: **sessão de KDD** e **ciclo de KDD** [GoldSchmit, 2015].

Cada sessão de **KDD** representa uma linha de raciocínio e de condução do processo. Toda sessão de **KDD** possui um, e somente um, objetivo a ser alcançado e, pode ser realizada em um ou mais ciclos de **KDD**. Cada ciclo envolve uma ou mais etapas de execução de um plano de ação e pertence a exatamente a uma sessão de **KDD**. Cada ciclo de **KDD** corresponde a uma iteração do processo de **KDD**.

A Tabela 3.17 mostra o formulário para o nosso primeiro estudo de caso que, é sobre a repetência escolar.

**Tabela 3.17** Formulário que documenta a execução do processo de KDD

|  |  |  |  |
| --- | --- | --- | --- |
| **Aplicação**: Repetência Escolar | | | |
| **Sessão:** 1  **Objetivo**:  Previsão de comportamento de valores discretos e contínuos  **Tarefas de KDD**:  Árvore de Decisão | **Resum**o: Esta sessão tem como objetivo extrair modelos baseados em dados do sistema acadêmico do **IFRN**, que consiga, através do relacionamento de seus atributos, traçar um perfil da repetência escolar | | |
| Expectativa quanto ao modelo de conhecimento:   * Representação do modelo de forma transparente e fácil de interpretar * Representação em forma de árvore * Representar os atributos que influenciam na repetência escolar | | | |
| **Plano de ação**:   * Utilizar o algoritmo Árvore de decisão * Aplicar os vários métodos de análise para comparar os resultados * Selecionar o melhor método de análise para ser usado pelo algoritmo árvore de decisão * Analisar o resultado obtido pelo método escolhido | | | |
| **Ciclo nº 1** |  |  |  |
| Métodos |  |  |  |
| Partição do BD em treino e teste | 70% treino  30% teste |  |  |
| **Algoritmo**: Microsoft Árvore de decisão | | | |
| Codificação continuo – categórica (Discretização) |  |  |  |
| **Requisitos**:   * **Uma única coluna chave (key)**: Cada modelo deve conter uma coluna de texto ou numérica que identifique exclusivamente cada registro. Não são permitidas chaves compostas. * **Uma coluna previsível**: Requer, pelo menos, uma coluna previsível. Você pode incluir vários atributos previsíveis em um modelo, e o atributo previsível pode ser de diferentes tipos, tanto continuo como discreto. * **Colunas de entrada**: Requer colunas de entrada que podem ser discretas ou contínuas. | | | |
| **Métodos de análise:**   * 1 Entropia de Shannon * 3 Bayesian com K2 a priori * 4 Bayesian Dirichlet com uniforme a priori (padrão) | | | |

O algoritmo Árvores de Decisão da Microsoft é um algoritmo de classificação e regressão fornecido pelo Microsoft SQL Server Analysis Services para uso em modelagens de previsão de atributos discretos e contínuos, da seguinte forma: [Bassan, 2014].

* **Atributo Diescreto**: Para atributos discretos, o algoritmo faz previsões fundadas nas relações entre colunas de entrada em um conjunto de dados. Ele usa os valores, conhecidos como estados dessas colunas, para prever os estados de uma coluna que você define como previsível. Especificamente, o algoritmo identifica as colunas de entrada que são correlacionadas com a coluna previsível.
* **Atributo Contínuo**: No caso de atributos contínuos, o algoritmo usa a regressão linear para determinar onde uma árvore de decisão se divide.

Quando mais de uma coluna é definida como prevísil, ou se acoaso tiver uma tabela aninhada configurada como previsível, o algoritmo criará uma árvore de decisão separada para cada coluna previsível.

O modelo de mineração de dados gerado pelo algoritmo Árvores de Decisão da Microsoft é uma série de divisões/nós na árvore. Quando um atributo de entrada é considerado significativamente correlacionado a uma coluna prevísil, então é adicioando um nó ao modelo. A forma que o algoritmo determina uma divisão depende do fato de ele estar prevendo uma coluna contínua ou discreta.

O algoritmo Árvores de Decisão da Microsoft usa a seleção de recurso para guiar a seleção dos atributos mais úteis. A seleção de recurso é usada por todos os algoritmos de mineração de dados do Analysis Services para melhorar o desempenho e a qualidade da análise. A seleção de recurso é importante para impedir que atributos sem-importância usem tempo do processador.

A seleção de recursos ajuda a resolver o problema, de ter dados demais de pouco valor, ou de ter pouquíssimos dados de alto valor.

A seleção de recurso, calcula uma pontuação para cada atributo e, em seguida, seleciona apenas os atributos com a pontuação mais alta. E isto sempre é feita antes do treinamento do modelo, para que os atributos de um conjunto de dados com maior probabilidade de uso no modelo sejam escolhidos de forma automática.

Segundo (Bassan, 2014), o algoritmo Árvore de Decisão da Microsoft, cria o conjunto de valores de entrada possíveis, ele executa feature selection para identificar os atributos e os valores que fornecem a maioria das informações e remove os valores considerados muito raros e, para otimizar o desempenho, ele agrupa esses valores em compartimentos, que podem ser processados como uma unidade.

Para criar uma árvores com as correlações das entradas e o resultado pretendido, o algoritmo, depois de ter correlacionado todos os atributos, identifica o único atributo que separa mais claramente os resultados. O ponto da melhor seperação é medido com o uso de uma equação que calcula o ganho de informações. Portanto, o atributo escolhido é usado para dividir os casos em subconjuntos, que são analisados recursivamente pelo mesmo processo, até que não seja mais possível dividir a árvore.

A determinação da equação exata para avaliar o ganho de informações depende dos parâmetros definidos na criação do algoritmo, do tipo de coluna previsível e do tipo de dados de entrada.

O algoritmo Árvores de Decisão da Microsoft, oferece três fórmulas para pontuar o ganho de informação: **entropia de Shannon**, **rede bayesiana com antecedente K2** e **rede bayesiana** com uma **distribuição Dirichlet uniforme de antecedentes**. Neste trabalho, serão realizadas experiências com parâmetros e métodos de pontuação diferentes para determinar aqueles que fornecem os melhores resultados

3.6.1. **Pontuação de interesse**

Por padrão, a pontuação de interesse é usada sempre que, a coluna contiver dados numéricos contínuos não binários [Bassan, 2014].

A medida de interesse usada no SQL Server Analysis Services baseia-se em entropia, ou seja, os atributos com distribuições aleatórias têm maior entropia e menor ganho de informações; sendo assim, esses atributos são menos interessantes. A entropia de qualquer atributo em particular é comparada à entropia de todos os outros atributos, como segue:

**Interestingness(Attribute) = - (m - Entropy(Attribute)) \* (m - entropy(Attribute)) (1)**

Entropia central ou **m**, significa a entropia de todo o conjunto de recursos. Ao subtrair a entropia do atributo de destino da entropia central, é possível avaliar a quantidade de informações fornecida pelo atributo.

#### 3.6.2. **Entropia de Shannon**

A entropia de Shannon mede a incerteza de uma variável aleatória para um resultado em particular. Dessa forma, a entropia pode ser representada como uma função da probabilidade de um evento ocorrer [Bassan, 2014].

O Analysis Services usa a seguinte fórmula para calcular a entropia de Shannon (Para atributos discretos):

**H(X) = -∑ P(xi) log(P(xi))**

(2)

Esse método de pontuação está disponível para atributos discretos.

**3.6.3. Bayesiano com K2 a priori**

Segundo Bassan (2014), o Analysis Services fornece duas pontuações para seleção de recursos que se baseiam em redes Bayesianas. Uma rede Bayesiana é um gráfico direcionado ou acíclico de estados e transições entre estados, ou seja, alguns estados vêm sempre antes do estado atual, alguns ocorrem depois, e o gráfico não se repete nem gera um loop. Por definição, as redes Bayesianas permitem o uso do conhecimento prévio.

O algoritmo **K2** usado com a rede Bayesiana foi desenvolvido por Cooper e Herskovits. É escalável e analisa diversas variáveis, mas requer ordenação das variáveis usadas como entrada [Bassan, 2014]. Esse método de pontuação está disponível para atributos discretos.

3.6.4. **Bayesiano Dirichlet Equivalente com Uniforme a priori**

O método de pontuação do **BDE** (Bayesiano Dirichlet Equivalente), foi desenvolvido por Heckerman e se baseia na métrica de BD desenvolvida por Cooper e Herskovits. A distribuição Dirichlet é uma distribuição multinomial que descreve a probabilidade condicional de cada variável da rede e possui muitas propriedades que devemos conhecer [Bassan, 2014].

O método **BDEU** (*Bayesiano Dirichlet Equivalente* com *Uniforme a priori*) assume um caso especial da distribuição *Dirichlet*, na qual uma constante matemática é usada para criar uma distribuição fixa ou uniforme de estados anteriores. A pontuação do **BDE** também assume equivalência de probabilidade, o que significa que não se pode esperar que os dados separem estruturas equivalentes. Em outras palavras, se a pontuação de **If A Then B** for igual à pontuação de **If B Then A**, não será possível distinguir as estruturas com base nos dados nem deduzir a causa [Bassan, 2014].

**3.6.5. Personalizando o algoritmo Microsoft árvore de decisão**

O algoritmo Árvores de Decisão da Microsoft tem parâmetros que afetam o desempenho e a precisão do modelo de mineração resultante. A Tabela 3.18 descreve os parâmetros que você pode usar com o algoritmo Árvores de Decisão da Microsoft.

**Tabela 3.18** Parãmetros do algoritmo Microsoft Árvore de Decisão

|  |  |
| --- | --- |
| Parâmetro | Descrição |
| COMPLEXITY\_PENALTY | Controla o crescimento da árvore de decisão. Um valor baixo aumenta o número de divisões e um valor alto diminui o número de divisões.   * Para os atributos 1 a 9, o padrão é 0,5. * Para 10 a 99 atributos, o padrão é 0,9. * Para 100 ou mais atributos, o padrão é 0,99. |
| FORCE\_REGRESSOR | Força o algoritmo a usar as colunas especificadas como regressores, independentemente da sua importância quando calculadas pelo algoritmo. Esse parâmetro é usado apenas para árvores de decisão que preveem um atributo contínuo. |
| MAXIMUM\_INPUT\_ATTRIBUTES | Define o número de atributos de entrada que o algoritmo pode manipular antes de invocar a seleção de recurso. |
| MAXIMUM\_OUTPUT\_ATTRIBUTES | Define o número de atributos de saída que o algoritmo pode manipular antes de invocar a seleção de recurso. |
| MINIMUM\_SUPPORT | Determina o número mínimo de casos folha necessário para gerar uma divisão na árvore de decisão. |
| SCORE\_METHOD | Determina o método o usado para calcular a pontuação da divisão. As seguintes opções estão disponíveis:   |  |  | | --- | --- | | ID | Nome | | 1 | Entropia | | 3 | Bayesian com K2 a priori | | 4 | Bayesiano Dirichlet Equivalente (BDE) com uniforme a priori |   O padrão é 4 ou BDE. |
| SPLIT\_METHOD | Determina o método usado para dividir o nó. As seguintes opções estão disponíveis:   |  |  | | --- | --- | | ID | Nome | | 1 | **Binary:** Indica que, independentemente do número real de valores do atributo, a árvore deverá ser dividida em duas ramificações. | | 2 | **Complete:** Indica que a árvore pode criar tantas divisões quanto há valores de atributo. | | 3 | **Both:** Especifica que o Analysis Services pode determinar se uma divisão binária ou completa deve ser usada para produzir os melhores resultados. |   O padrão é 3. |

**3.6.6. Aplicar o Algoritmo Árvore de Decisão**

Seguindo o plano de ação da Tabela 3.17, vamos aplicar o algoritmo Árvore de Decisão para prever a situação da repetência escolar. Para esse pimeiro caso, vamos utilizar os dados de uma tabela com os atributos descriminados na Tabela 3.19.

**Tabela 3.19** relação de atributos da tabela onde será aplicado o algoritmo Árvore de Decisão para análise.

|  |  |  |
| --- | --- | --- |
| Atributo | Descrição | Tipo |
| Area procedencia escola origem | Urbana, Rural | Discreta |
| Aulas dadas | Quantidade de aulas ministradas | Continuo |
| Coeficiente de rendimento | Desempenho do aluno no curso | Continuo |
| Etnia | Branca, Preta, Parda e amarela | Discreto |
| Faltas | Total de fatlas no semestre | Continuo |
| Forma\_Ingresso | Exame seleção, ENEM, SISU a assim por dante. | Discreto |
| Idade | Idade do aluno | Continuo |
| Media final | Média final do aluno | Continuo |
| Renda | Valor em real | Continuo |
| Renda\_Familiar | Valor textual: até 1 salário, de 1 até 2 salarios e assim por diante. | Discreto |
| Reside | Com quem o aluno reside: com os pais ou não. | Discreto |
| Sexo | Masculino ou feminino | Discreto |
| Situacao | Aprovado, reprovado, jubilado, cancelado, evasão, etc. | Discreto |
| Tipo escolA origem | Pública estadual, federal, municipal ou priada. | discreto |

Para este estudo de caso, será escolhido o atributo “Situacao” para ser o atributo previsível e os demais serão informados como atributos de entrada. O método usado para calcular a pontuação de divisão, foi escolhido a Entropia e o método especificado para dividir os nós foi o binário. A Figura 3.15 mostra a árvore gerada para o atributo previsível “Situacao=Reprovado” e a Listagem 3.3 mostra o código **DMX** (Data Mining Extensions) usado para criar a estrutura de mineração de dados e definie o modelo a ser usado pela estrutura.

**Listagem 3.3** Código **DMX** para criar a estrutura de mineração de dados e os modelos (algoritmo) usados pela estrutura.

--Criando uma estrutura de mineração de dados

CREATE MINING STRUCTURE [Mine TB Juntando Dados Alunos]

(

[Codigo Curso] LONG KEY,

[Area Procedencia Escola Origem] TEXT DISCRETE,

[Aulas Dadas] LONG CONTINUOUS,

[Coeficiente Rendimento] DOUBLE DISCRETE,

[Disciplina] TEXT DISCRETE,

[Etnia] TEXT DISCRETE,

[Faltas] LONG CONTINUOUS,

[Forma Ingresso] TEXT DISCRETE,

[Idade] LONG CONTINUOUS,

[Media Final] DOUBLE CONTINUOUS,

[Renda Familiar] TEXT DISCRETE,

[Reside] TEXT DISCRETE,

[Sexo] TEXT DISCRETE,

[Situacao] TEXT DISCRETE,

[Situacao Ing] TEXT DISCRETE

)

WITH HOLDOUT (30 PERCENT or 20000 CASES)

--Após criar a estrutura de mineração de dados, adiciona-se o algoritmo (modelo) desejado.

--Adicionando o modelo de árvore de decisão

ALTER MINING STRUCUTURE [Mine TB Juntando Dados Alunos]

ADD MINING MODEL [Decision Tree]

(

[Codigo Curso],

[Area Procedencia Escola Origem],

[Aulas Dadas],

[Coeficiente Rendimento],

[Disciplina],

[Etnia],

[Faltas],

[Forma Ingresso],

[Idade],

[Media Final],

[Renda Familiar],

[Reside],

[Sexo],

[Situacao] PREDICT,

[Situacao Ing] PREDICT

) USING Microsoft\_Decision\_Trees (algoritmo parâmetros)

WITH DRILLTHROUGH

--Adicionando um modelo cluster

ALTER MINING STRUCUTURE [Mine TB Juntando Dados Alunos]

ADD MINING MODEL [Clustering]

USING Microsoft\_Clustering

WITH DRILLTHROUGH

Para treinar o modelo da estrutura foi usado o código **DMX** da listagem 3.4.

**Listagem 3.4** Treinando o modelo da estrutura criada na listagem 3.4.

--Treinando o modelo

INSERT INTO STRUCTURE [Mine TB Juntando Dados Alunos]

(

[Codigo Curso],

[Area Procedencia Escola Origem],

[Aulas Dadas],

[Coeficiente Rendimento],

[Disciplina],

[Etnia],

[Faltas],

[Forma Ingresso],

[Idade],

[Media Final],

[Renda Familiar],

[Reside],

[Sexo],

[Situacao],

[Situacao Ing]

)

OPENQUERY (DDSEducacional),

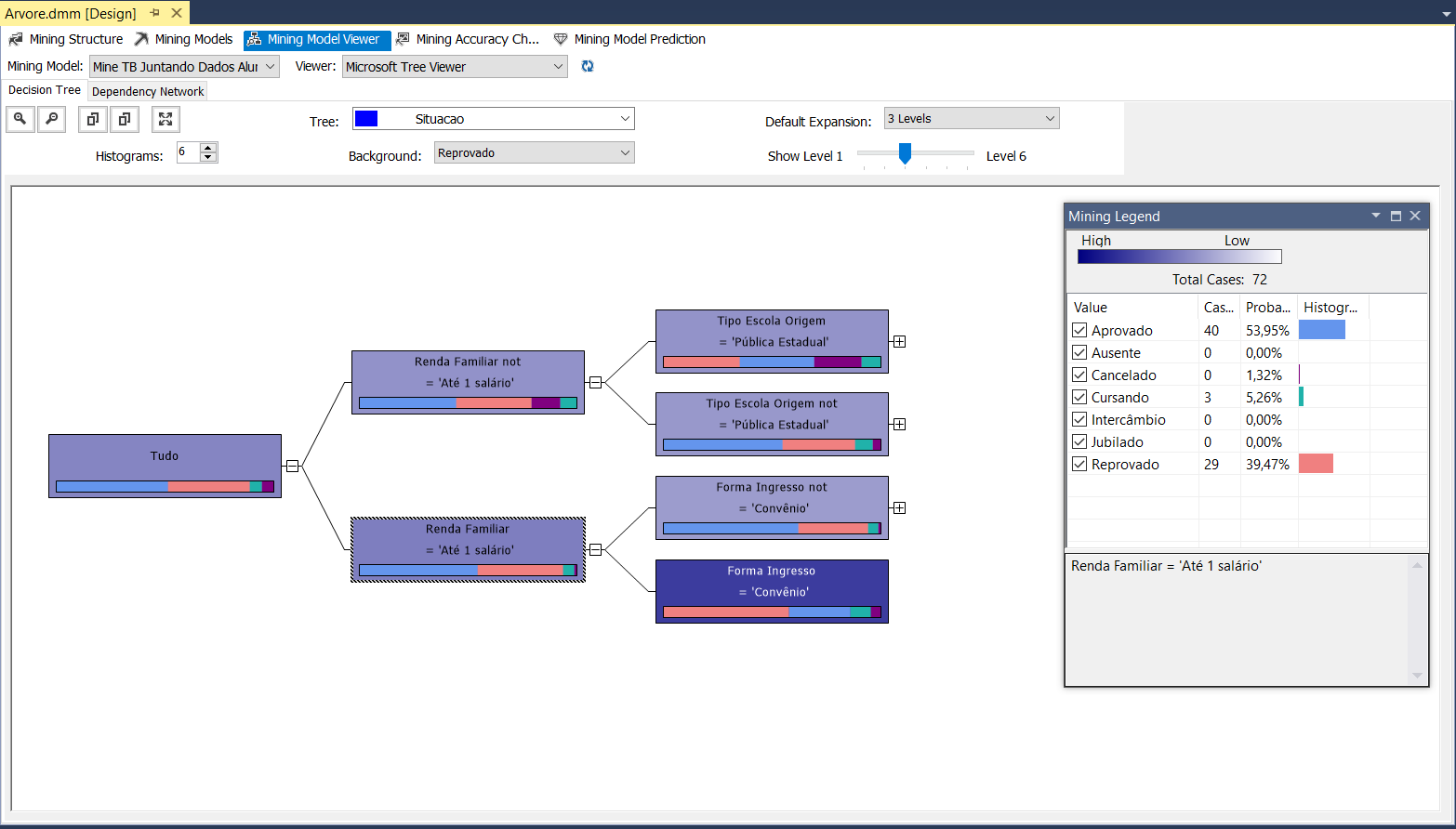
'SELECT [Codigo Curso], [Area Procedencia Escola Origem],

[Aulas Dadas], [Coeficiente Rendimento],

[Disciplina], [Etnia], [Forma Ingresso], [Idade], [Media Final], [Renda Familiar], [Reside], [Sexo], [Situacao], [Situacao Ing]

FROM TB\_JuntandoDadosAlunos')

A figura 3.15 mostra a árvore de decisão criada para esse modelo. Como se pode ver ela tem duas ramificações. Clicando no nó “Renda familiar = Até 1 salário”, temos, como mostra a tabela “Mining Legend” que a taxa de Repetência e de 39,47%.



**Figura 3.15** Árvore de Decisão para o atributo previsível “Situacao=Reprovado”.

**Fonte**: Autor.

O modelo que se pode extrair dessa árvore é mostrado no código **SQ**L da listagem 3.5:

**Listagem 3.5** Código do modelo extraído da àrvore de decisão da figura 3.15

IF ([Renda Familiar] = 'Até 1 salário') Then

Probabilidade de reprovação = 39,47%

IF ([Forma Ingresso]='Convênio') Then

Probabilidade de reprovação = 57,14%

ELSE ([Forma Ingresso]!='Convênio') Then

Probabilidade de reprovação = 25,53%

IF ([Forma Ingresso] ='Seleção Geral Curso FIC') then

Probabilidade de reprovação = 50%

ELSE ([Forma Ingresso]!='Seleção Geral Curso FIC') Then

Probabilidade de reprovação = 32%

IF (Sexo = 'M') Then

Probabilidade de reprovação = 23,8%

ELSE (Sexo = 'F') Then

Probabilidade de reprovação = 28%

IF (Etnia = 'Branca') Tjen

Probabilidade de reprovação = 33,33%

ElSE

Probabilidade de reprovação = 21,43%

ELSE

Probabilidade de reprovação = 34,62%

IF ([Tipo Escola Origem] = 'Pública Estadual') Then

Probabilidade de reprovação = 34,38%

IF ([Renda Familiar] = 'De 1 até 2 salários') Then

Probabilidade de reprovação = 29,41%

ElSE

Probabilidade de reprovação = 36,84%

ELSE ([Tipo Escola Origem] != 'Pública Estadual') Then

Probabilidade de reprovação = 33%

IF (Reside = 'Com os pais') Then

Probabilidade de reprovação = 28,57%

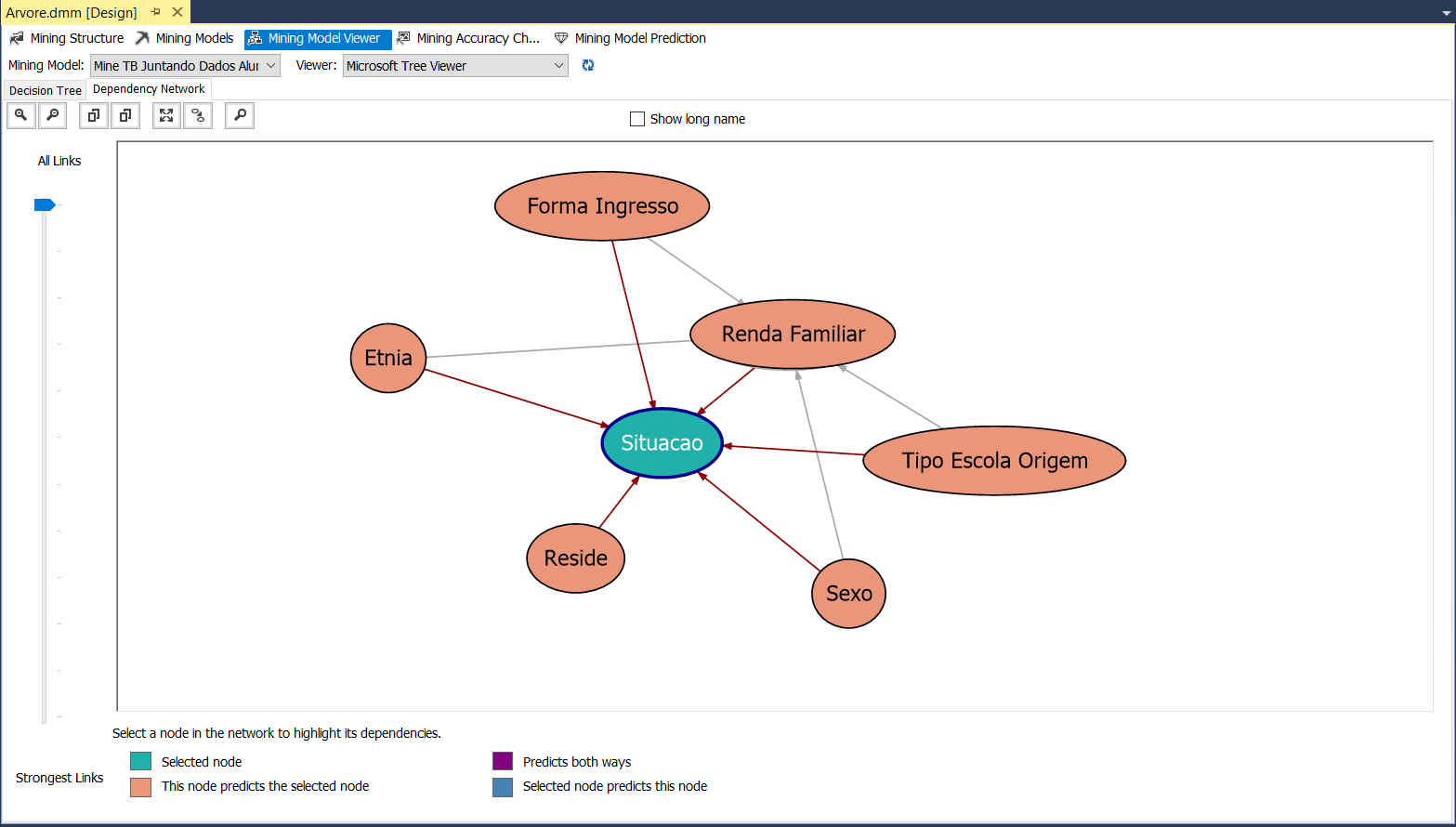
ELSE

Probabilidade de reprovação = 35,71%

Observe que a árvore da figura 3.15 tem duas ramificações, que estão representadas no modelo pelo **IF**...**ELSE**.

Como foi explicado anteriormente, a árvore gera para cada atributo de entrada, que tenha correlação significativa com o atributo previsível um nó, como mostrado na figura 3.15. Portanto, resta agora verificar quais atributos apresentam correlação significativa com o atributo previsível, ou seja, quais atributos mais influenciam a formação de nós na árvore.

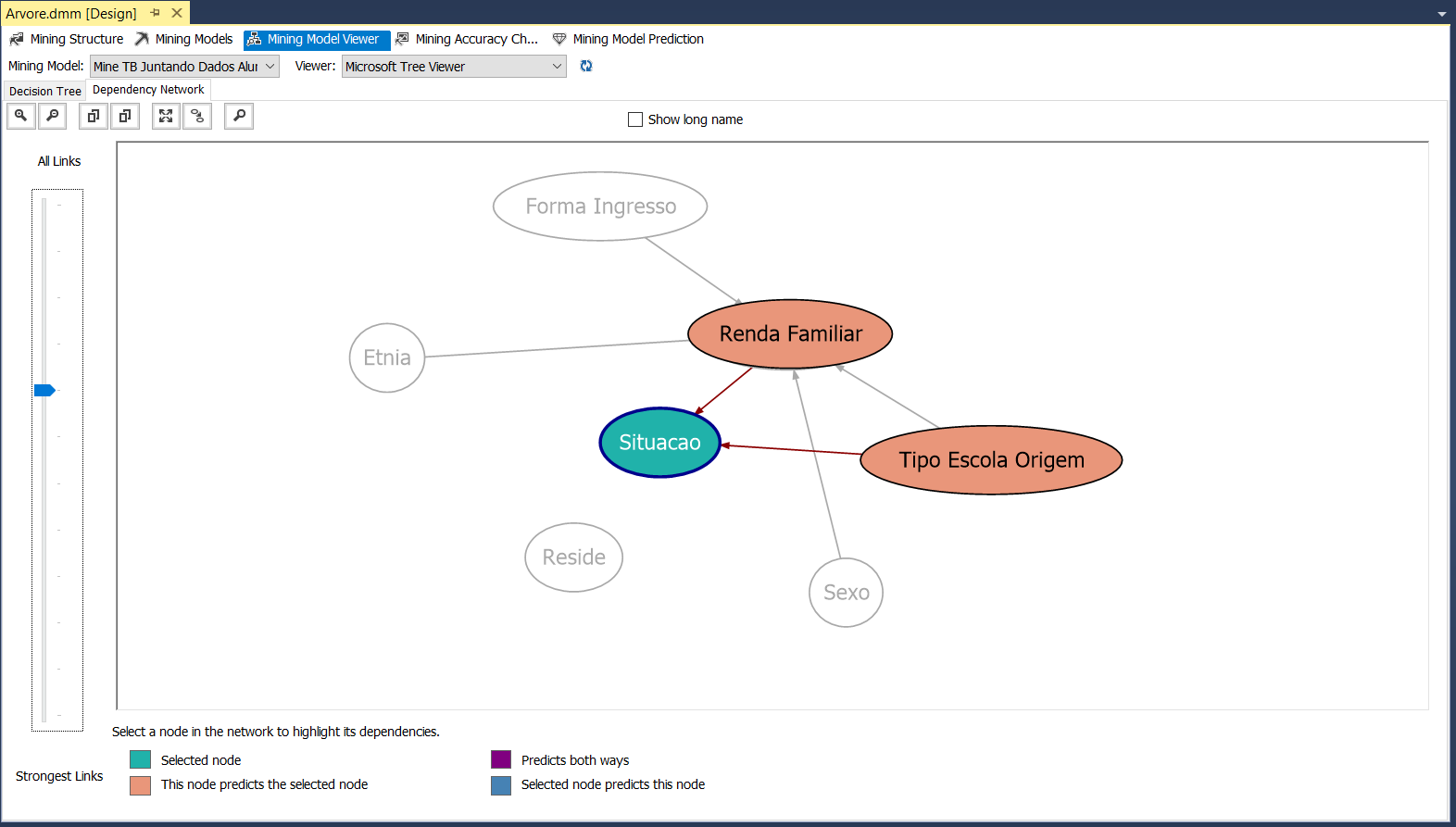
A Figura 3.16 mostra, através de um gráfico de correlação, os atributos que mais influenciaram na situação reprovado. Observe que do lado esquerdo tem uma barra de rolagem e que a mesma esta posicionada no topo. Isto significa que o gráfico está exibindo todos os atributos de entrada que têm correlação significativa com o atributo previsívele, portanto, foram usados para gerar nós na árvore. Se deslocarmos a barra para baixo, ela vai desconectando os atributos que influenciam menos e deixando aqueles tem têm mais influencia na formação da árvore. Então posicinando a barra de rolagem além do centro, temos o resultado mostrado na figura 3.17.



**Figura 3.16** Rede de dependência dos atributos em relação ao atributo situação (Reprovado).

**Fonte**: Autor.

Na figura 3.17 pode-se observar que os atributos de entrada que mais influenciaram na situação (atributo previsível) foram os atributos forma de ingresso, faltas e media final.



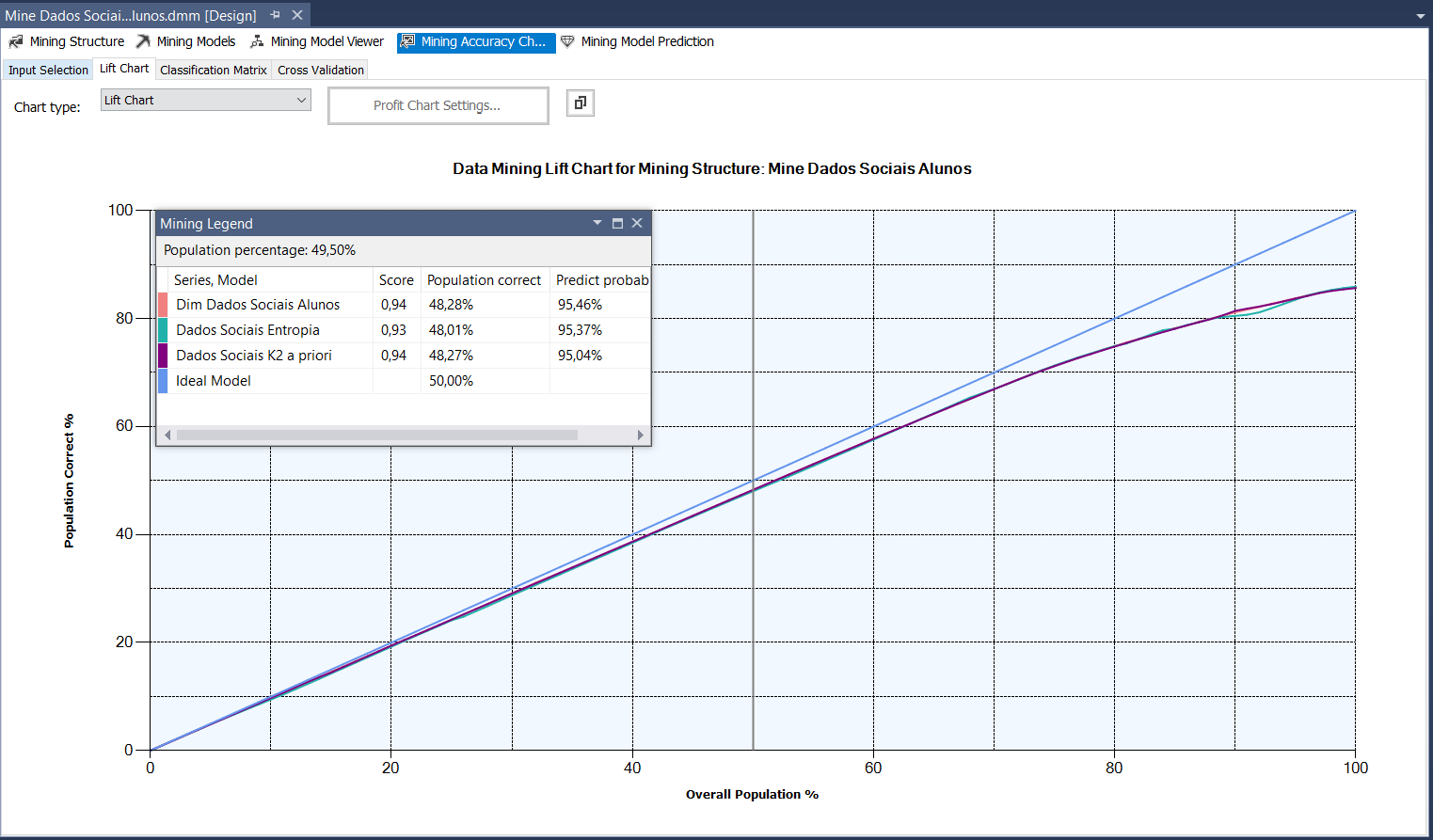
**Figura** 3.17 Rede de dependência dos atributos em relação ao atributo situação (Reprovado).

**Fonte**: Autor.

Usando o mesmo estudo de caso, foram criadas outras árvores de decisão, apenas modificando o parâmetro “*SCORE\_METHOD*”, que determina como os nós da árvore são divididos e, observou-se que, para esse mesmo conjunto de dados, os resultados foram muito parecidos. Então, para que podesse ser comprovada esta semelhança entre os resutados obtidos, foi verificada a acuracia entre o métodos selecionados. A Figura 3.18 mostra um gráfico, onde mostra que os métodos obteram resultados muito próximos um do outro.

Os métodos para cálculo dos nós selecionados foram os seguintes: o padrão (4) *Bayesiano Dirichlet Equivalente* (**BDE**) com uniforme a priori, o entropia (1) e Bayesian com K2 a priori (3).

Veja na Figura 3.18 que as linhas laranja, verde e roxa estão praticamente uma em cima da outra. Se você observar a tabela gerada pela ferramenta, você vai vê que a probabilidade é de 95,46% para o método padrão, 95,37% para o método de entropia e de 95,04% para o método K2 a priori. Portanto, não faz diferença em usar qualquer um dos métodos de pontuação, para este estudo de caso, é claro.

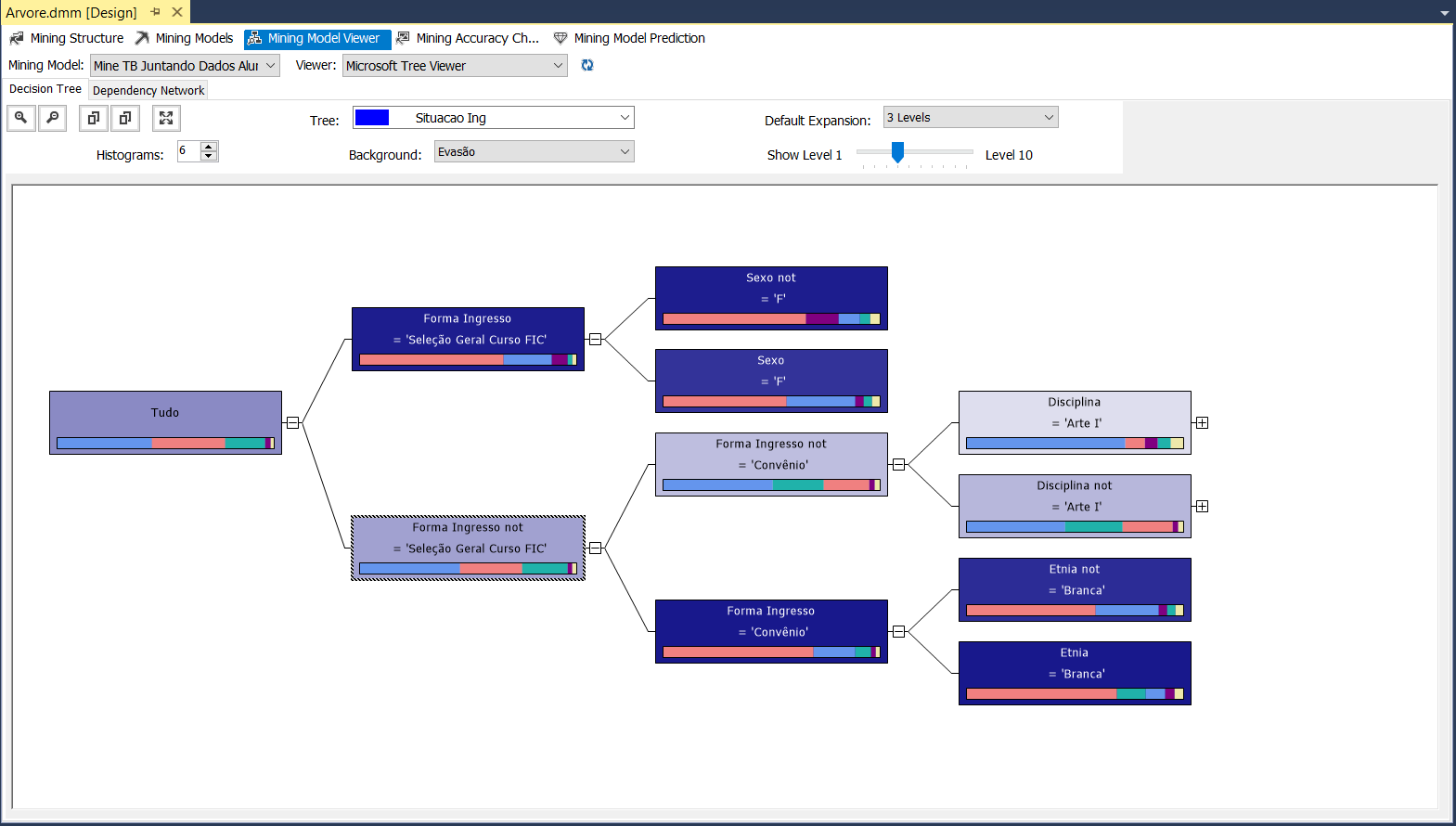


**Figura 3.18** Gráfico de acurácia entre os métodos de divisão dos nós da árvore de decisão.

**Fonte**: Autor.

Quando da configuração do algoritmo Árvore de Decisão, foram definidos dois atributos como previsíveis, o atributo “Situacao” e o atributo “SituacaoIng”. Onde o primeiro está relacionado os dados sociais dos alunos e correlacionados a repetência escolar, enquanto o segundo atributo previsíveis, está relacionado os dados de ingresso dos alunos nos respectivos cursos. Portanto, o algoritmo cria duas árvores, uma para cada atributo previsível. No caso, anterior, foi analisada uma das árvores criada, a que estava relacionada a repetência escolar. A segunda árvore, que será analisada a partir desse momento, os atributos de entrada estam correlacionados a evasão escolar. A figura 3.19 mostra a árvore de decisão criada para a situação evasão escolar.

O atributo previsível “SituacaoIng” possui vários estados, dentre eles pode-se citar Cancelado, Jubilado, Concluído, evasão, intercâmbio e auesente. No entanto, será analisada a situação evasão escolar. O método usado para calcular a pontuação de divisão, foi escolhido a Entropia e o método especificado para dividir os nós foi o binário.



**Figura 3.19** Árvore de Decisão usando o método de pontuação a Entropia

**Fonte**: Autor.

O modelo que se pode extrair dessa árvore é mostrado no código SQL da listagem 3.6:

**Listagem 3.6** Código do modelo extraído da àrvore de decisão da figura 3.19

IF ([Forma Ingresso] = 'Seleção Geral Curso FIC') Then

IF (Sexo = 'F') Then

evasão = 13%

ELSE

evasão = 12%

ELSE

IF ([Forma Ingresso] = 'Convênio') Then

IF (Etnia = 'Branca') Then

evasão = 14%

ELSE

evasão = 13%

ELSE

IF (Disciplina = 'Arte 1') Then

evasão = 2%

Else

evasão = 42%

IF ([Forma Ingresso] = 'Seleção Dif. Téc. Subsequente') Then

evasão = 8%

IF (Sexo = 'F') Then

evasão = 1%

ELSE

evasão = 7%

ELSE

evasão = 34%

IF (Reside = 'Conjuge') Then

evasão = 8%

IF (Etnia = 'Branca') Then

evasão = 1%

ELSE

evasão = 7%

ELSE

evasão = 26%

IF ([Forma Ingresso] = 'Seleção Geral Téc. Sub.') Then

evasão = 2%

ELSE

evasão = 24%

IF ([Forma Ingresso] = 'Seleção Dif. Grad. Vest.') Then

evasão = 4%

ELSE

evasão = 20%

IF (Etnia = 'Parda') Then

evasão = 9%

IF ([Tipo escola origem] = 'Pública Estadual') Then

evasão = 5%

ELSE

evasão = 4%

ELSE

evasão = 11%

IF ([Forma Ingresso]='Seleção Dif. Téc. Integ./PROITEC') Then

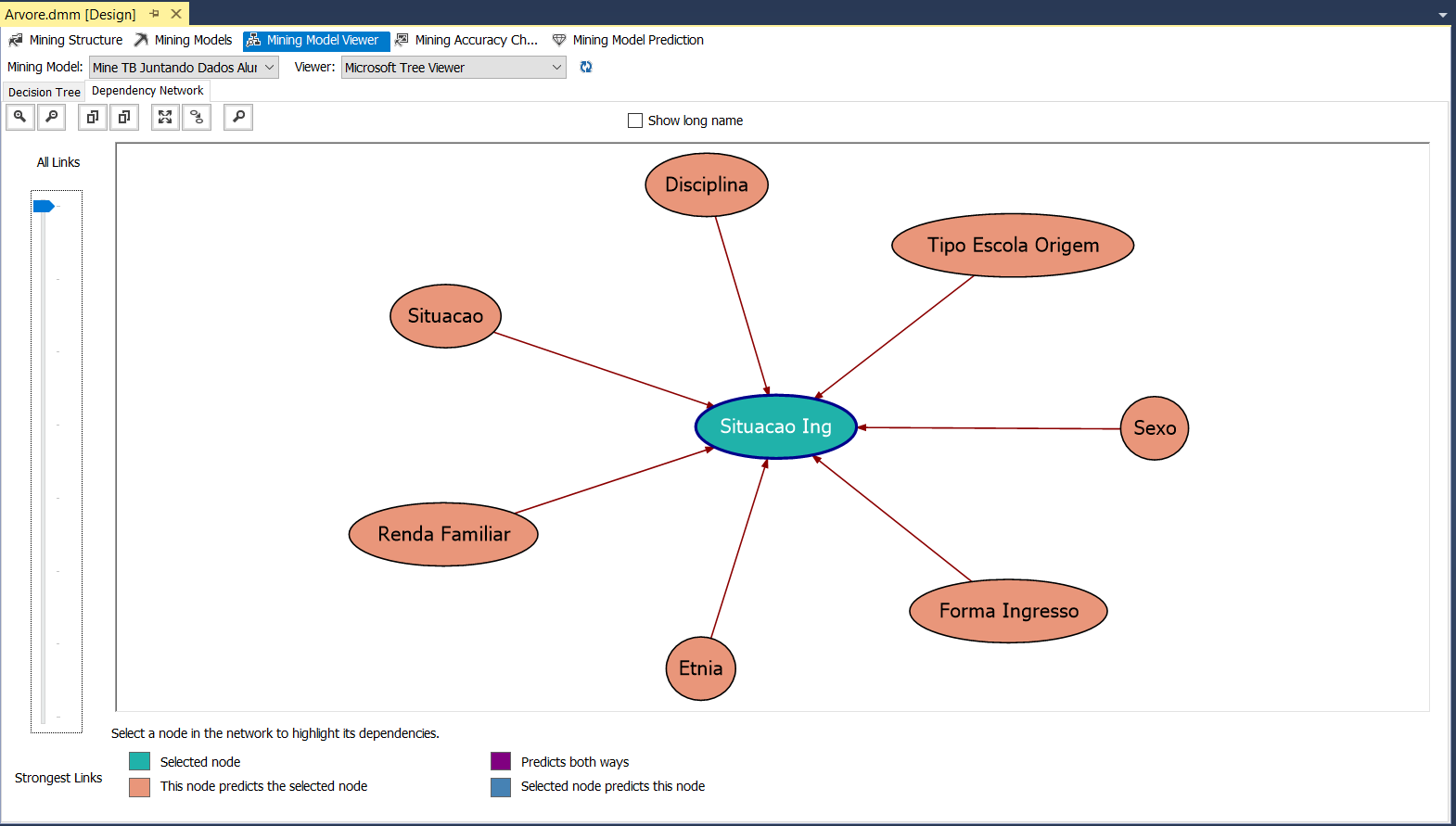
evasão = 1%

ELSE

evasão = 10%

O modelo da listagem 3.4 representa os dois ramos da árvore de decisão da figura 3.19. Como se pode ver, a árvore gera para cada atributo de entrada, que tenha correlação significativa com o atributo previsível um nó, como mostrado na figura 3.20.

A Figura 3.20 mostra, através de um gráfico de correlação, os atributos que mais influenciaram na situação evasão.



**Figura 3.20** gráfico de correlação dos atributos de entrada com o atributo previsível “Situacao Ing” destacado na cor verde.

**Fonte**: Autor.

Na figura 3.20 os atributos que estão destacados na cor laranja são justamente os atributos de entrada que tem correlação com o atributo previsível.

O próximo passo é, utilizar o algoritmo de clustering para verificar o percentual de influencia de cada atributo de entrada, na formação do atributo previsível “Situacao Ing”.

3.6.7. **Aplicando o Algoritmo Cluster**

Vamos aplicar um algoritmo de clusterização sobre o mesmo conjunto de dados usado anteriormente, para tentar mapear o quanto cada atributo da base de dados selcionada, influencia na sitação “Reprovado”.

O algoritmo Microsoft Clustering é um algoritmo de segmentação. Ele usa técnicas iterativas para agrupar casos em um conjunto de dados em clusters que contenham características semelhantes. Esses agrupamentos podem ser usados para explorar dados, identificar anomalias nos dados e criar previsões[Bassan, 2014].

Diferentemente do algoritmo Árvore de Decisão, não é necessário designar um atributo privisível para poder criar o modelo de clustering. O mesmo treina o modelo a aprtir das relações existentes nos dados e a partir dos clusters que o algoritmo identifica.

O algoritmo do Microsoft Clustering, fornece dois métodos para criar clusters e atribuir pontos de dados aos clusters. O primeiro, o algoritmo ***K-means***, um método de clustering rígido. Isso significa que um ponto de dados pode pertencer somente a um cluster e que, uma única probabilidade é calculada para a associação de cada ponto de dados nesse cluster. O segundo método, de *Maximização de Expectativa* (**EM**), é um método de *clustering flexível*. Isso significa que, um ponto de dados sempre pertence a vários clusters e que uma probabilidade é calculada para cada combinação de ponto de dados e cluster [Bassan, 2014].

Pode-se escolher o algoritmo que será usado definindo o parâmetro ***CLUSTERING\_METHOD***. O método padrão de cluster é o **EM** evolutivo.

3.6.7.2. **Cluster EM**

No cluster **EM**, o algoritmo refina de modo iterativo, um modelo de clustering inicial para ajustar os dados e determina a probabilidade de um ponto de dados existir no cluster. O algoritmo termina o processo quando, o modelo de probabilidade ajusta os dados. A função usada para determinar o ajuste é a probabilidade de log dos dados de acordo com o modelo [Bassan, 2014].

Se clusters vazios forem gerados durante o processo ou se a associação de um ou mais clusters estiver abaixo de um determinado limite, os clusters com baixas populações serão propagados novamente em novos pontos e o algoritmo **EM** será executado mais uma vez [Bassan, 2015].

Os resultados do método de cluster **EM** são probabilidades. Isso significa que, cada ponto de dados pertence a todos os clusters, mas cada atribuição de um ponto de dados a um cluster, tem uma probabilidade diferente. Como o método permite a sobreposição dos clusters, a soma dos itens de todos os clusters pode ultrapassar o total de itens do conjunto de treinamento [Bassan, 2014].

3.6.7.2. **Cluster K-means**

O algoritmo **K-means**, atribui cada ponto de dados exatamente a um cluster e não permite incertezas na associação. A associação em um cluster é expressa como uma distância do centroide [Bassan, 2014].

De acorodo com Bassan (2014), o algoritmo **K-means** é usado para criar clusters de atributos contínuos, onde calcular a distância até o centro é simples. No entanto, a implementação do Microsoft, adapta o método **K-means** ,para atributos de distinção de cluster usando probabilidades. Para os atributos de distinção, a distância de um ponto de dados a partir de um cluster específico é calculada da seguinte maneira ***P(data point, cluster)***.

3.6.7.3. **Personalizando o Algoritmo Cluster**

O algoritmo Microsoft Clustering dá suporte a vários parâmetros que afetam o comportamento, o desempenho e a exatidão do modelo de mineração resultante.

A Tabela 3.18 descreve os parâmetros que podem ser usados com o algoritmo do Microsoft Clustering. Esses parâmetros afetam o desempenho e a exatidão do modelo de mineração resultante.

**Tabela 3.18** Especifica o método de cluster para o algoritmo a ser usado.

|  |  |  |
| --- | --- | --- |
| Parâmetro | Descrição | Padrão |
| CLUSTERING\_METHOD | Especifica o método de cluster para o algoritmo a ser usado.  1 - EM evolutivo  2 - EM não evolutivo  3 - K-Means evolutivo  4 - K-means não evolutivo | O padrão é 1. |
| CLUSTER\_COUNT | Especifica o número aproximado de clusters a serem criados pelo algoritmo. | O padrão é 10. |
| CLUSTER\_SEED | Especifica o número de propagação, usado apenas para gerar clusters aleatoriamente para o estágio inicial de criação de modelo. | O padrão é 0. |
| MINIMUM\_SUPPORT | Especifica o número mínimo de casos necessários para criar um cluster. Se o número de casos do cluster for menor que esse número, o cluster será tratado como vazio e descartado. | O padrão é 1 |
| MODELLING\_CARDINALITY | Especifica o número de modelos de exemplo construídos durante o processo de cluster. | O padrão é 10. |
| STOPPING\_TOLERANCE | Especifica o valor usado para determinar quando a convergência é alcançada e, o algoritmo terminou de criar o modelo. A convergência é alcançada quando, a alteração geral nas probabilidades do cluster é menor do que a taxa do parâmetro STOPPING\_TOLERANCE dividida pelo tamanho do modelo. | O padrão é 10. |
| SAMPLE\_SIZE | Especifica o número de casos, que o algoritmo usará em cada passagem, se o parâmetro CLUSTERING\_METHOD for definido como um dos métodos de cluster evolutivo. | O padrão é 50.000. |
| MAXIMUM\_INPUT\_ATTRIBUTES | Especifica o número máximo de atributos de entrada, que o algoritmo pode manipular, antes de invocar a seleção de recurso. | O padrão é 255. |
| MAXIMUM\_STATES | Especifica o número máximo de estados de atributo aos quais, o algoritmo dá suporte. Se um atributo tiver mais estados que o valor máximo, o algoritmo usará os estados mais populares e ignorará os demais estados. | O padrão é 100. |

A Tabela 3.19 mostra o formulário para o nosso primeiro estudo de caso que, é sobre a repetência escolar, para o segundo ciclo.

**Tabela 3.19** Formulário que documenta a execução do processo de KDD

|  |  |  |  |
| --- | --- | --- | --- |
| **Aplicação**: Repetência Escolar | | | |
| **Sessão:** 1  **Objetivo**:  identificam as relações em um conjunto de dados  **Tarefas de KDD**:  Cluster | **Resumo**: Identificar as relações entre os dados do conjunto de dados e, gera uma série de clusters com base nelas. Fazer a representação dos clusters através de gráfico. Verificar observando os clusters, quais atributos mais influenciam a repetência escolar. | | |
| Expectativa quanto ao modelo de conhecimento:   * Representação do modelo de forma transparente e fácil de interpretar * Representação em forma de gráfico de dispersão * Representar os atributos que mais influenciam na repetência escolar | | | |
| Plano de ação:   * Utilizar o algoritmo cluster * Aplicar os algoritmos EM e K-means e analisar os resultados obtidos * Selecionar o melhor resultado ou melhor algoritmo | | | |
| **Ciclo nº 2** |  |  |  |
| **Métodos** |  |  |  |
| Partição do BD em treino e teste | 70% treino  30% teste |  |  |
| **Algoritmo**: Microsoft Cluster | | | |
| Codificação numérica – categórica (Discretização) |  |  |  |
| **Parâmetros**:   * CLUSTERING\_METHOD = 1, 2, 3, 4 * CLUSTER\_COUNT = 10 * MAXIMUM\_STATES = 10   Nota: Os demais parâmetros foram definidos com o valor padrão. | | | |
| **Métodos de clusterização:**   * 1 EM evolutivo (padrão) * 2 EM não evolutivo * 3 K-Means evolutivo * 4 K-Means não evolutivo | | | |

A figura 3.21 mostra o gráfico de dispersão, gerado pelo algoritmo cluster **EM Evolutivo**, para o conjunto de dados especificado, com a configuração dos parâmetros definidos na Tabela 19.

Observe na figura que os clusters com maior número de casos de reprovados ão o cluster 3 e 6. A cor azul escura, indica exatamente o cluster onde ocorreram o maior número de casos para situação selecionada na opção “state”.

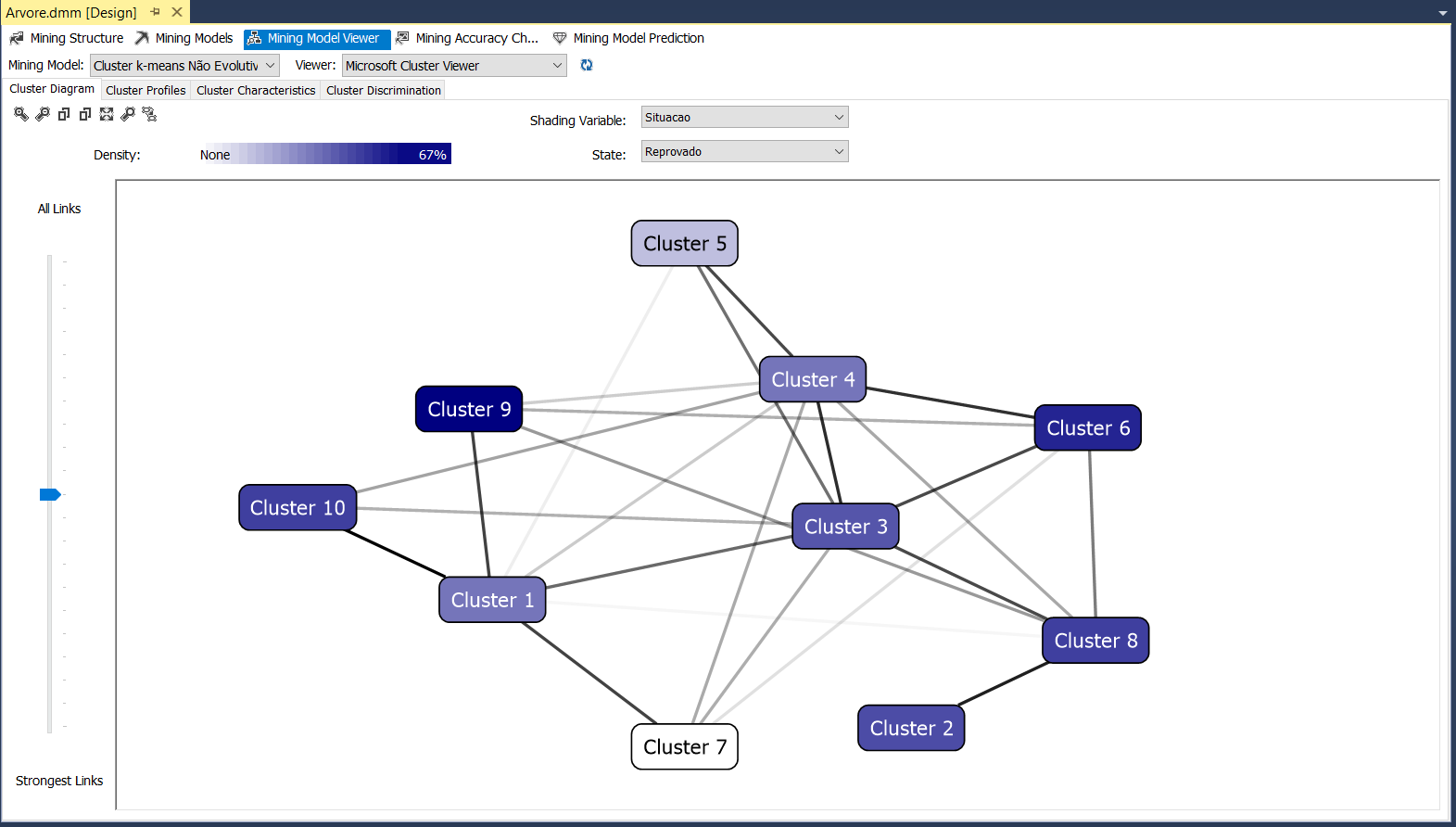
A Figura 3.22 mostra o gráfico gerado usando o método de clusterização o EM não Evolutivo. Para este modelo, os clusters com maior concentração de casos de repetência escolar, foram os clusters 6 e 9. O cluster 9 com 67% de reprovados e o cluster 3 com 57% de reprovados e o cluster 7 0% de reprovados.

Comparando com a situação anterior, onde o cluster 3 tem 58% de reprovados e o cluster 6 tem 57% de reprovados, pode-se preceber que os resultados obtidos pelos métodos **EM Evolutivo** e o **EM Não Evolutivo** são bastante próximos.



**Figura 3.21** Gráfico dos cluster gerados para a situação “Reprovado” usando o método de clusterização o EM Evolutivo.

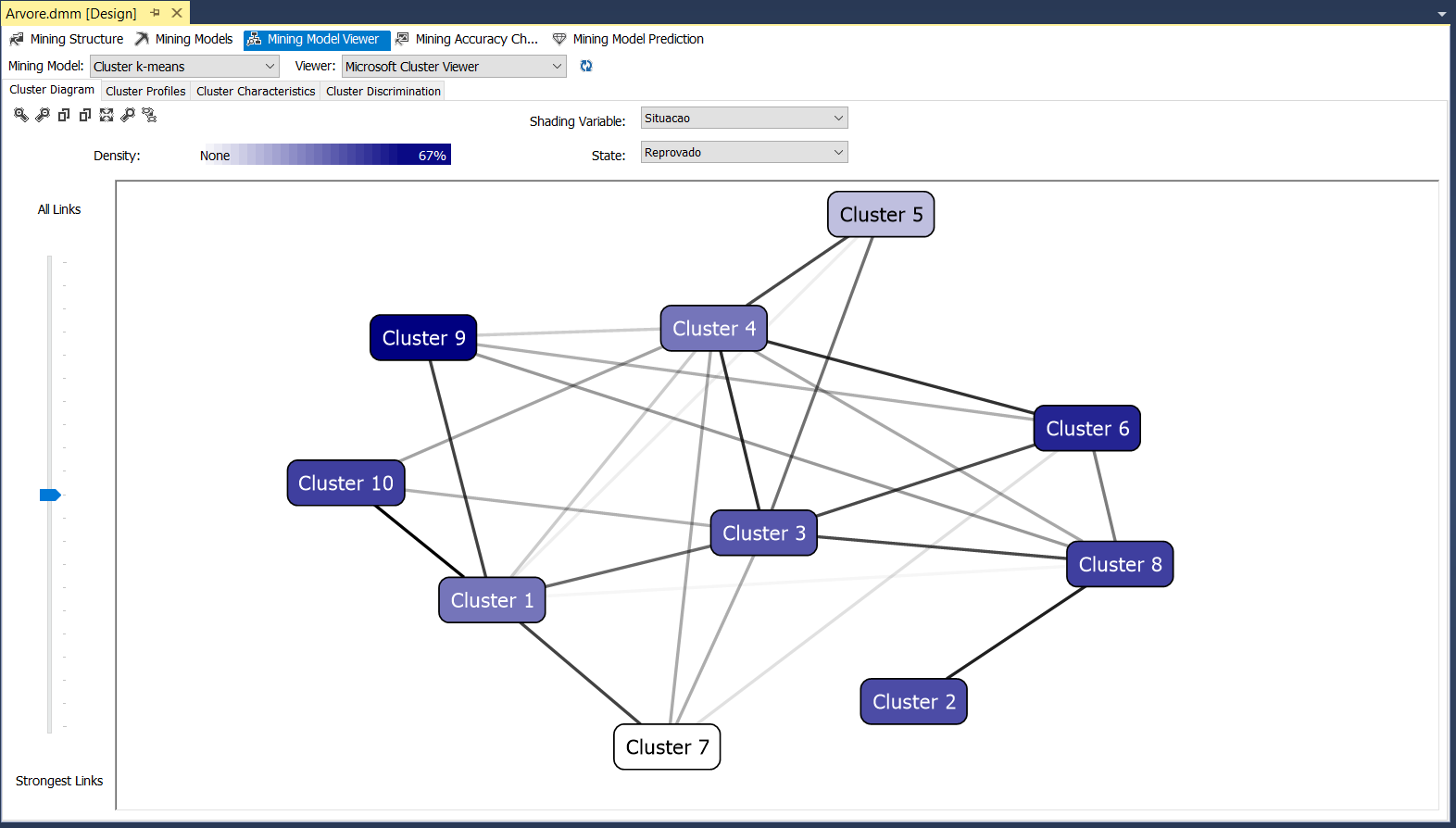
**Fonte**: Autor.



**Figura 3.22** Cluster gerado com o método **EM** não Evolutivo.

Fonte: Autor.

A figura 3.23 mostra o gráfico gerado pelo método K-means Evolutivo para o mesmo DataSet e as mesmas configurações do métodos anteriores.

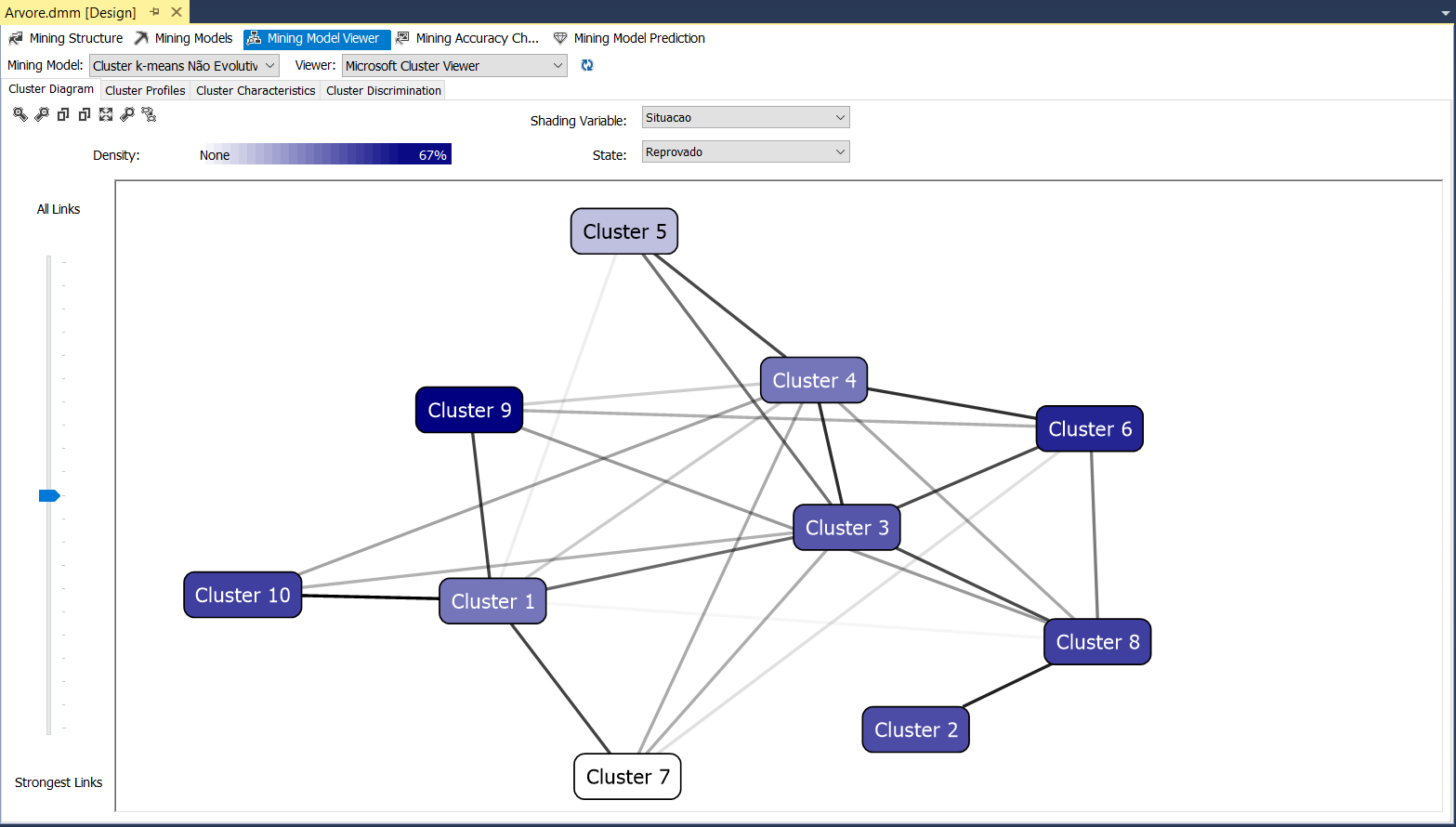


**Figura 3.23** Gráfico de clusters gerado pelo método K-means Evolutivo.

**Fonte**: Autor.

Mais uma vez os clusters 6 e 9 aparecem com a maior concentração de casos de repetência escolar. Ness caso o clusters 9 tem 67% de reprovados e o cluster 6 57% de reprovados, enquanto o cluster 7 tem 0% de reprovados. Resultado muito semelhante ao obtido pelo método EM Evolutivo.

A figura 3.24 mostra o gráfico gerado pelo método K-means Não Evolutivo para o mesmo DataSet e as mesmas configurações do métodos anteriores.



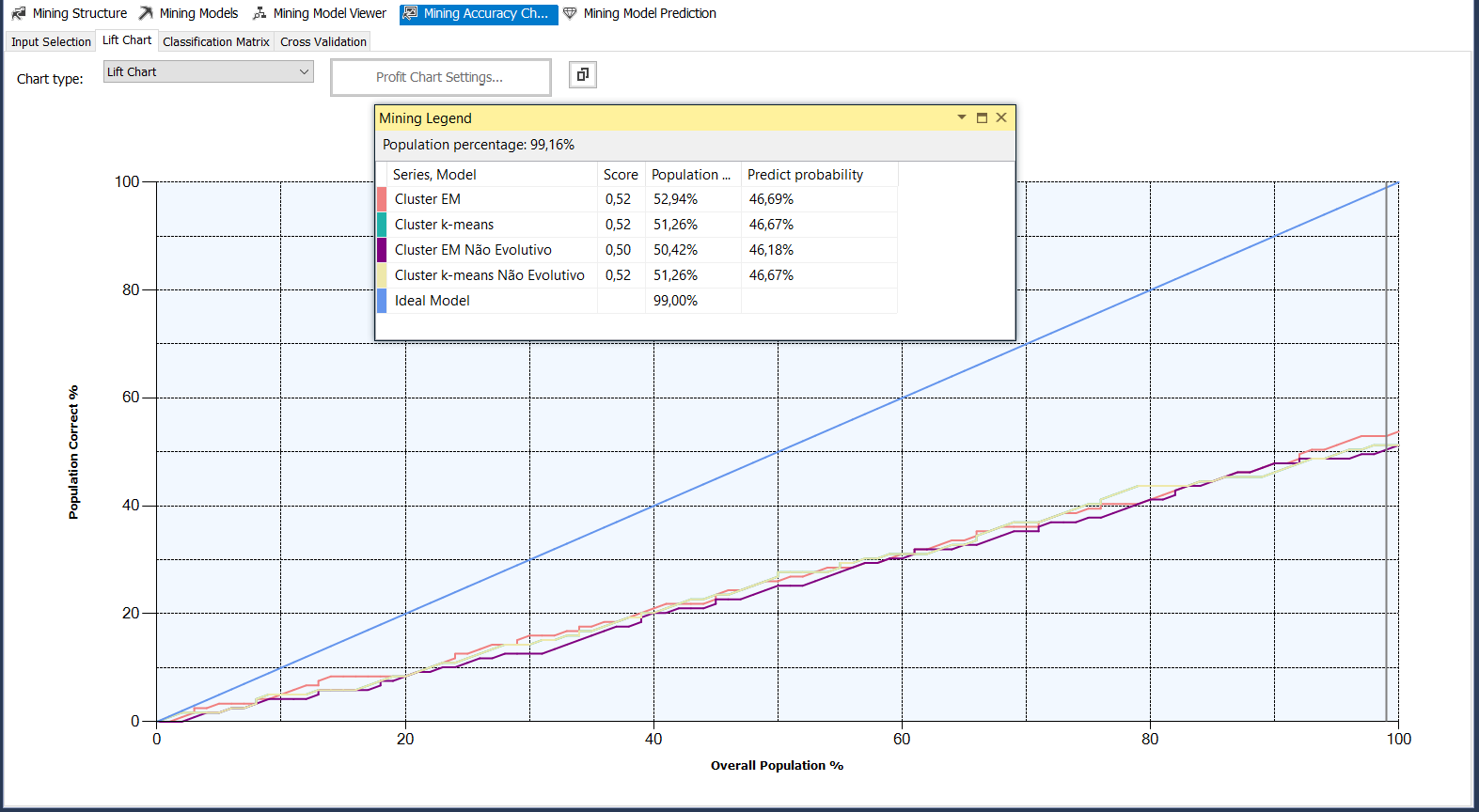
**Figura 3.24** Gráfico de clusters gerado pelo método K-means Não Evolutivo.

**Fonte**: Autor.

Como se pode perceber, o resultado é praticamente o mesmo do gerardo pelo método K-means Evolutivo. O cluster 9 tem 67% e o cluster 6 tem 57% dos casos de repetência escolar.

Para uma melhor análise comparativa desses métodos, apresento a seguir, um gráfico da acurácia desse método de clusterização. A figura 3.23 mostra o gráfico de acurácia.

Obervando-se a figura 3.24, perceb-se que as linhas com as cores amarela, laranja, roxa e verde estão muito proíxmas uma das outras. A linha azul representa a situação ideal. Então, os métodos geraram resultado longe do ideal. Além do mais, a medida que a população de casos aumenta, as linhas vão se distanciando ainda mais. Porém, a seguir será feita uma análise dos clusters 6 e 9, para se ter uma ideia o quanto cada atributo de entrada influencia na formação desses clusters.



**Figura 3.24** Gráfico comparativo entre os métodos de cluster.

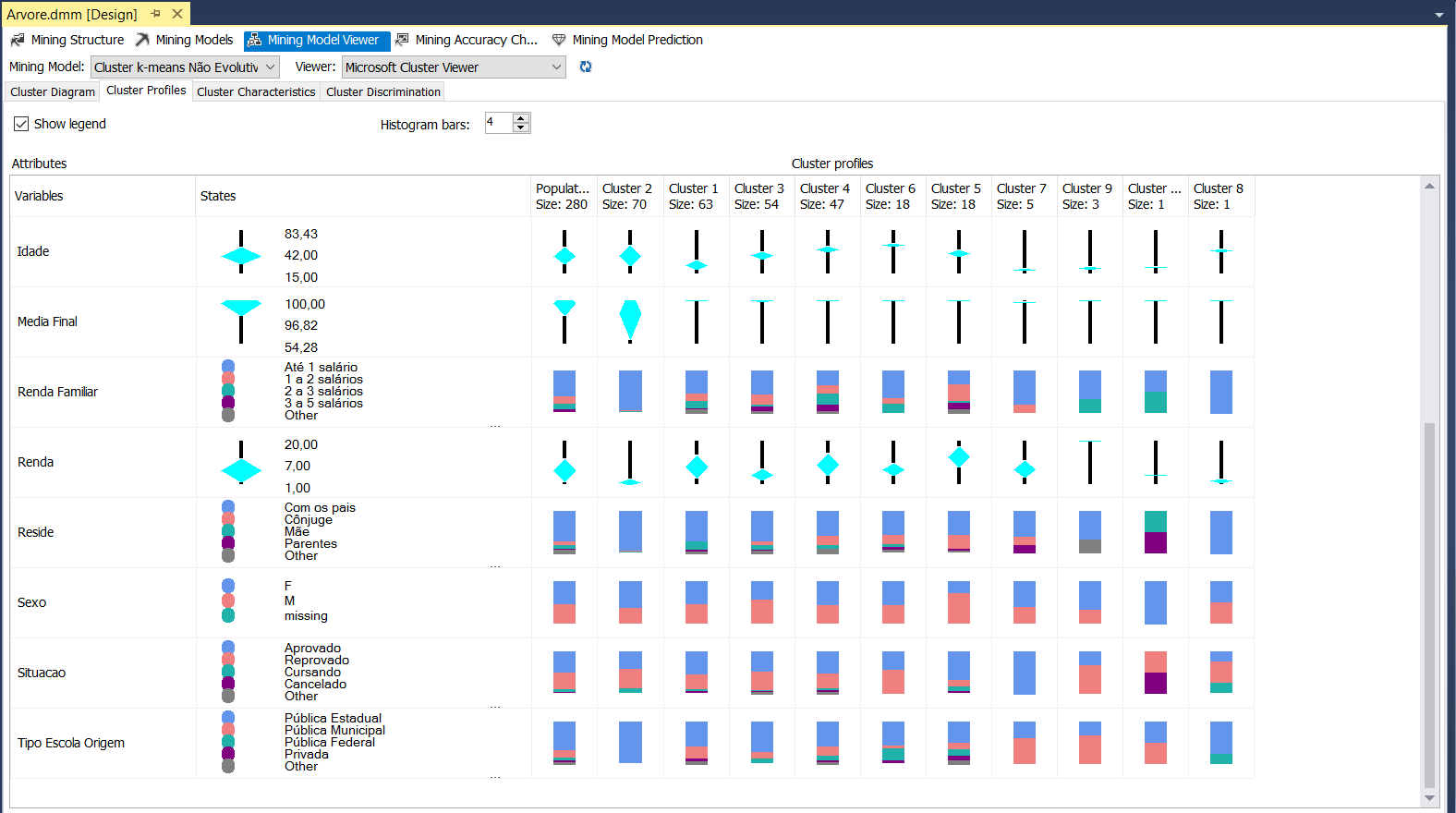
**Fonte**: Autor.

Então, o próximo passo, é analisar os clusters 6 e 9, afim de identificar quais são os atributos que mais ifluenciaram na formação desses cluster.

A figura 3.25 mostra o perfil de cada cluster, o seja, mostra exatamente o quanto cada atributo está influenciando a formação do cluster. A tabela 3.20 detalha esses valores.

**Tabela 3.20** Tabela comparativa entre os clusters 6 e 9 da figura 3.24.

|  |  |  |
| --- | --- | --- |
| Cluster | Atributo | Valor |
| 6 | Tipo Escola Origem Pública Estadual  Pública Municipal | 57% |
| 9 | 66,7% |
| 6 | Situação Reprovado | 57% |
| 9 | 66,7% |
| 6 | Sexo ‘F’ | 57% |
| 9 | 66,7% |
| 6 | Reside com os pais | 57% |
| 9 | 67% |
| 6 | Renda familiar até 1 salário | 64,3% |
| 9 | 66,7% |



**Figura 3.25** Perfis de Cluster algoritmo EM evolutivo

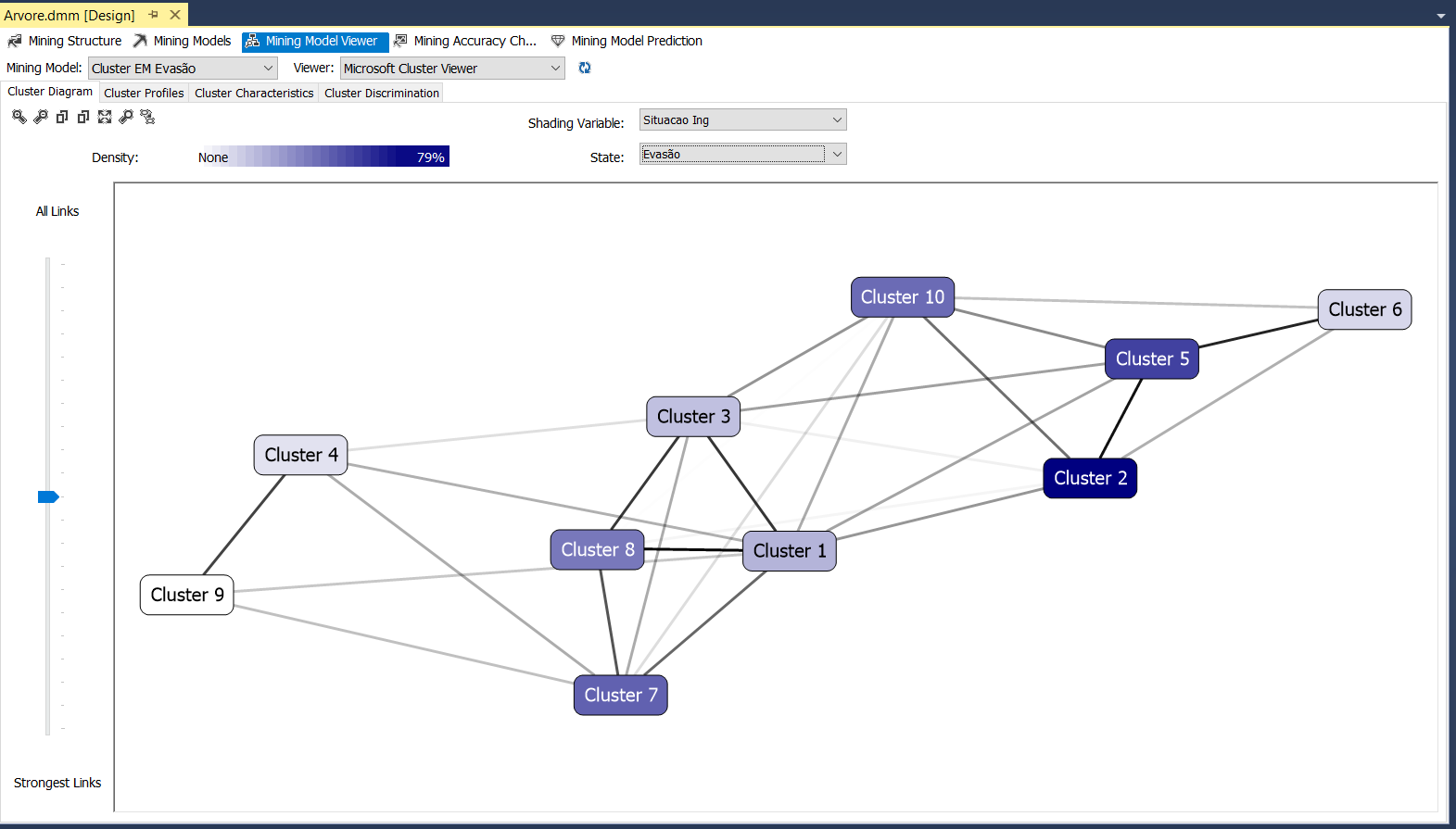
**Fonte**: Autor.

Portanto, o perfil dos alunos com maior probabilidade de repetência escolar são aqueles oriundos das escolas públicas estaduais ou municipais, cuja familia tem renda familiar de até 1 salário mínimo e residem com os pais.

A figura 3.26 mostra o gráfico de dispersão, gerado pelo algoritmo cluster **EM Evolutivo**, em função do atributo previsível “Situacao Ing = evasão”, cujo gráfico de correlação de atributos corresponde a figura 3.20.

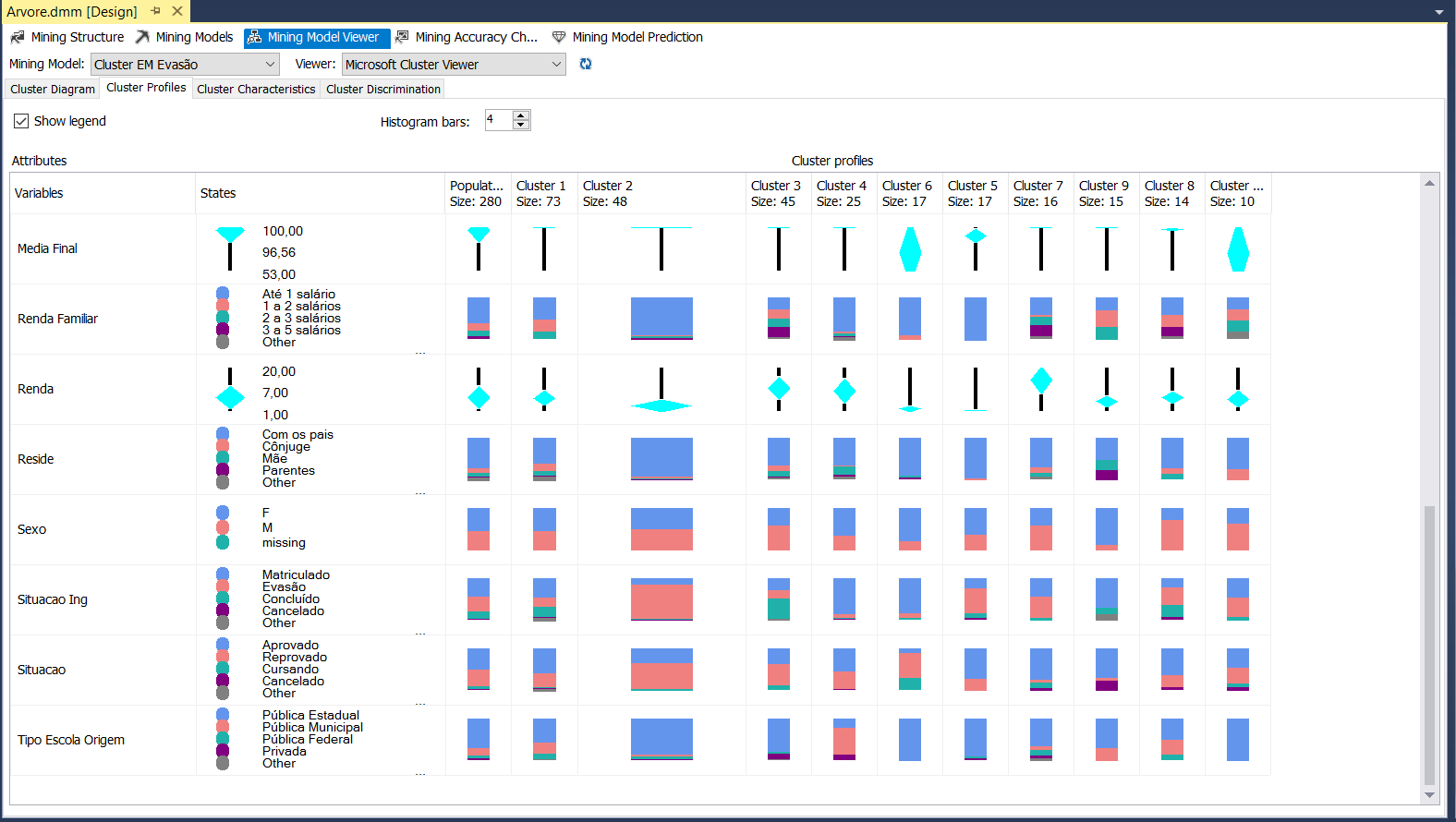
O cluster 2 na cor azul mais escura, contêm a maior concentração de casos de evasão escolar, com exatamente 79% dos casos. O cluster 5 com 59% e o cluster 7 com 49% dos casos de evasão escolar. O cluster 9, como se pode ver com 0% dos casos de evasão escolar.

Analisando o cluster 2, se pode comprovar que, os atributos que mais influenciam na formação do cluster 2 são: Renda Familiar de até 1 salário com 89%, Reside com os pais com percentual de 91%, situacao Reprovado com percentual de 62% e Tipo de escola de origem igual a Pública estadual com percentual de 85% dos casos de evasão escolar. A figura 3.27 mostra de forma gráfica esses dados.



**Figura 3.26** Cluster **EM Evolutivo** para o atributo previsível “Situacao Ing=Evasão”.

**Fonte**: Autor.



**Figura** 3.27 Gráfico tipo histograma mostrando o quanto cada atributo influencia percentualmente na formação de cada cluster.

**Fonte**: Autor.

**3.6.8. Criando Consultas de Previsão**

Com os modelos de mineração de dados treinados e comparados e, escolhido o que melhor apresentou os resultados. Então, pode-se usar os modelos para fazer previsões, como por exemplo, fazer provisões para identificar os possíveis perfis de alunos com maior probabilidade de evasão escolar.

3.6.8.1. Prevendo a Evasão Escolar

Nesta consulta pretende-se criar uma consulta para prever qual é o perfil dos alunos com a maior probabilidade de evasão escolar, tomando como base os dados as informações do modelo de mineração de Árvore de Decisões, explorado nesse trabalho e os dados da tabela base TB\_JuntandoDadosAlunos.

A listagem 3.8 mostra a consulta **DMX** criada para a previsão de evasão escolar em função dos atributos ‘Tipo Escola Origem = Pública Estadual’, ‘Renda Familiar = Até 1 salário’ e ‘Etnia = Branca’.

**Lisategm 3.8** Código de previsão escrito na linguagem **DMX**

SELECT

(PredictProbability([Mine TB Juntando Dados Alunos].[Situacao Ing])) as [Probabilidade%],

[Mine TB Juntando Dados Alunos].[Situacao Ing],

t.[Tipo\_Escola\_Origem],

t.[Renda\_Familiar],

t.[Etnia]

From

[Mine TB Juntando Dados Alunos]

PREDICTION JOIN

OPENQUERY([DDS Educacional],

'SELECT

[Tipo\_Escola\_Origem],

[Renda\_Familiar],

[Etnia],

[Data\_de\_Nascimento],

[Idade],

[Forma\_Ingresso],

[SituacaoIng],

[Coeficiente\_Rendimento],

[Disciplina],

[Area\_Procedencia\_Escola\_Origem],

[Media\_Final],

[Aulas\_Dadas],

[Faltas],

[Reside],

[Sexo],

[Situacao],

[Renda]

FROM

[dbo].[TB\_JuntandoDadosAlunos]

') AS t

ON

[Mine TB Juntando Dados Alunos].[Data De Nascimento] = t.[Data\_de\_Nascimento] AND

[Mine TB Juntando Dados Alunos].[Idade] = t.[Idade] AND

[Mine TB Juntando Dados Alunos].[Forma Ingresso] = t.[Forma\_Ingresso] AND

[Mine TB Juntando Dados Alunos].[Situacao Ing] = t.[SituacaoIng] AND

[Mine TB Juntando Dados Alunos].[Coeficiente Rendimento] = t.[Coeficiente\_Rendimento] AND

[Mine TB Juntando Dados Alunos].[Disciplina] = t.[Disciplina] AND

[Mine TB Juntando Dados Alunos].[Area Procedencia Escola Origem] = t.[Area\_Procedencia\_Escola\_Origem] AND

[Mine TB Juntando Dados Alunos].[Etnia] = t.[Etnia] AND

[Mine TB Juntando Dados Alunos].[Media Final] = t.[Media\_Final] AND

[Mine TB Juntando Dados Alunos].[Aulas Dadas] = t.[Aulas\_Dadas] AND

[Mine TB Juntando Dados Alunos].[Faltas] = t.[Faltas] AND

[Mine TB Juntando Dados Alunos].[Renda Familiar] = t.[Renda\_Familiar] AND

[Mine TB Juntando Dados Alunos].[Reside] = t.[Reside] AND

[Mine TB Juntando Dados Alunos].[Sexo] = t.[Sexo] AND

[Mine TB Juntando Dados Alunos].[Tipo Escola Origem] = t.[Tipo\_Escola\_Origem] AND

[Mine TB Juntando Dados Alunos].[Situacao] = t.[Situacao] AND

[Mine TB Juntando Dados Alunos].[Renda] = t.[Renda]

WHERE

[Mine TB Juntando Dados Alunos].[Situacao Ing] ='Evasão' AND

t.[Tipo\_Escola\_Origem] ='Pública Estadual' AND

t.[Renda\_Familiar] ='Até 1 salário' AND

t.[Etnia] ='Branca'

Para obter todas as probabilidades, basta modificar os parâmetros dos filtros para os atributos Tipo\_Escola\_Origem, Renda\_Familiar e Etnia. A Tabela 3.21 mostra os possíveis atributos que esses parâmetros podem assumir.

**Tabela 3.21** Valores dos atributos

|  |  |
| --- | --- |
| Atributo | Valores |
| Tipo\_Escola\_Origem | Privada  Pública Municipal  Pública Estadual  Pública Federal |
| Renda\_Familiar | Até 1 salário  1 a 2 salários  2 a 3 salários  3 a 5 salários  5 a 10 salários  10 a 20 salários  Mais de 20 salários |
| Etnia | Branca  Parda  Preta  amarela |

Ao ser executada a consulta da listagem 3.9, será obtido o valor de 57% de probabilidade de evasão escolar para o perfil especificado. E se mudarmos o perfil para ‘Tipo Escolar Origem = Privada’, será obtido o valor de 51% de probabilidade de evasão escolar, como a tabela 3.22 mostra os resultado da consulta com vários perfis.

**Tabela 3.22** Previsão de evasão de acordo com o perfil especificado na tabela.

|  |  |  |  |
| --- | --- | --- | --- |
| PERFIL  (EVASÃO ESCOLAR) | | | |
| TIPO ESCOLA  ORIGEM | RENDA  FAMILIAR | ETNIA | Probabilidade |
|  |  |  |  |
| Pública Estadual | Até 1 salário | Branca | 57% |
| Privada | Até 1 salário | Branca | 51% |
|  |  |  |  |

Na cláusula where do código **MDX** da listagem 3.8 tem-se a seguinte instrução [Mine TB Juntando Dados Alunos].[Situacao Ing] ='Evasão', se mudarmos esta instrução para [Mine TB Juntando Dados Alunos].[Situacao] ='Reprovado'. Então, pode-se fazer previsão em relação a repetência escolar. A tabela 3.23 mostra um resumo das previsões calculadas.

**Tabela 3.22** Prevendo repetência de acordo com o perfil especificado na tabela.

|  |  |  |  |
| --- | --- | --- | --- |
| PERFIL  (REPETÊNCIA ESCOLAR) | | | |
| TIPO ESCOLA  ORIGEM | RENDA  FAMILIAR | ETNIA | Probabilidade |
|  |  |  |  |
| Pública Estadual | Até 1 salário | Branca | 90% |
| Privada | Até 1 salário | Branca | 75% |
| PÚBLICA FEDERAL | Até 1 salário | Branca | 88% |
| PÚBLICA MUNICIPAL | Até 1 salário | Branca | 86% |

Continua...

Fazer mais previsões e aplicar outros algoritmos de mineração de dados e, também aplicar outros algoritmos de mineração de dados sobre os resultado obtidos dos algoritmos anteriores. Por exemplo, pegar o resultado da Árvore de Decisão e aplicar uma Rede Neural ou um algoritmo de regressão linear.

3.7. **Módulo Caso Baseado em Raciocínio**

3.7.1. **Introdução**

Segundo Blay Whitby (2003), Inteligência Artificial(**IA**) é o estudo do comportamento inteligente (em homens, animais e máquinas) e a tentativa de encontrar formas pelas quais esse comportamento possa ser transformado em qualquer tipo de artefato por meio da engenharia. Entre os diversos processos de **IA** está o aprendizado de máquinas, que por sua vez pode ser seccionado em aprendizado supervisionado e aprendizado não-supervisionado, como já foi explanado no referencial teórico desse trabalho.

Nesse módulo, serão utilizadas Raciocínio Baseado em Casos (**RBC**), para implementar um sistema especialista de aconcelhmaneto pedagócigo para o IFRN, aliando as técnicas do algoritmo de clustering, denominado de n-Vizinhos Mais Próximos (**kNN**) e à plataforma Web e suas tecnologias.

**RBC** é um conjunto de técnicas para o funcionamento de um sistema inteligente, tendo como centro de operações, uma base de conhecimento prévio, composta de experiências contextualizadas, que descrevem os problemas e suas respectivas soluções, chamada Base de Casos. Estas soluções podem ser sugeridas (reutilização) para serem aplicadas a quaisquer novos casos, ou problemas, que sejam apresentados, a partir da relação de similaridade (recuperação) do problema apresentado com os casos existentes, podendo inclusive, se servir destes novos casos (retenção) para ampliar a base de conhecimento (Watson 1997).

Portanto, O principal objetivo desse módulo é aplicar o Raciocínio Baseado em Casos (**RBC**), como técnica de Inteligência Artificial, no desenvolvimento de um sistema especialista que, auxilie uma equipe de acompanhamento pedagógico, na tarefa de aconselhamento de alunos, através da proposição de soluções aos problemas atuais, baseando-se no conhecimento adquirido com os casos anteriores que, por sua vez, foram inseridos na Base de Casos.

Então, para que tudo isso aconteça, objetivas-se desenvolver as seguintes tarefas:

* Desenvolver uma arquitetura ideal utilizando linguagem de programação e ambiente de desenvolvimento e produção amplamente utilizado.
* Utilizar a plataforma web, a linguagem de programação Javascript e as notações JSON (Javascript Object Notation) e BSON (Binary Javascript Object Notation) no desenvolvimento do sistema proposto.
* Testar a aplicação e analisar os resultados do Estudo de Caso utilizando Raciocínio Baseado em Casos com dados obtidos com auxílio da equipe de Orientação Pedagógica do Instituto Federal de Educação, Ciência e Tecnologia do RN – IFRN.

3.7.2. **Contextualização**

Em entrevista realizada com a Equipe Técnica Pedagógica – **ETEP** da Diretoria Acadêmica de Gestão e Tecnologia da Informação do IFRN, onde foi exposto como os procedimentos de atendimento pedagógico são realizados, e de que maneira são registrados esses casos de atendimentos, onde ficou evidente que, a dinâmica aplicada, não atingir a problemática central do atendimento que é a qualificação das informações e a transformação do conhecimento tácito de cada membro da equipe em conhecimento explícito de toda **ETEP**. Além dos mais, existe uma evolução do número de casos nos últimos anos. Portanto, ficou evidente a necessidade do desenvolvimento de uma ferramenta de apoio neste contexto de aconselhamento pedagógico,

Em anos anteriores a 2012, o atendimento sequer contava com um documento padronizado para registro dos casos, sendo posteriormente criada uma ficha de atendimento, ou seja, todo registro dos casos de atendimento pedagógico são registrados, até o memento em fichas de papel. Mais recentemente, foi desenvolvido um formulário utilizando a plataforma Google Docs, que gera uma planilha compartilhada entre os membros da **ETEP**, com as principais informações cadastradas.

A equipe pedagógica relata na entrevista que:

“O acompanhamento dos alunos acontecia por verificação das notas nos boletins no final dos semestres, escrita de situações em livro ATA (registro das situações problemas) ou agenda da equipe (Até 2011). A partir de 2012, adotou-se um modelo de ficha individual que facilitava o acompanhamento da evolução dos encaminhamentos e das soluções e permitia identificar recorrências. Esse modelo foi importante, também, pela possibilidade que trouxe de partilhar informações entre as pedagogas da Diretoria sobre os atendimentos. [...] Verificou-se já um grande avanço com as fichas, mas começou-se a perceber que algumas situações de atendimento eram bem comuns (problemas que requeriam acompanhamento psicológico, por exemplo). E como verificar isso? Olhando de ficha em ficha a natureza do problema? Sem falar que requer um tempo para descrever a situação na ficha, descrever o problema, descrever o encaminhamento dado e outras informações importantes.”

A equipe pedagógica, afirma ainda, a necessidade de uma ferramenta que, tornasse os dados mais “palpáveis” e mais “úteis”, de forma que, fossem permitidos produzir relatórios e gráficos que exibissem as problemáticas mais comuns, a evolução dos casos e ainda a evolução dos encaminhamentos abordados e as melhores soluções ou ainda, as mais úteis. Dessa forma, foi apresentada a equipe pedagógica a proposta do RBC, onde pode-se, através dessa técnica, reutilizar soluções anteriores, em novas situações problema.

3.7.3. **Metodologia**

Como primeiro passo para a estruturação do sistema especialista, que utilize RBC, no auxílio de aconselhamento pedagógico aos alunos, foi realizar entrevista com a Equipe Técnico Pedagógica – ETEP, da diretoria Acadêmica de gestão e Informática, no sentido que fosse feito levantamento das principais situações que demandam atenção, bem como a classificação destas situações que se baseasse numa maior ou menor necessidade de atenção por parte dos técnicos da ETEP. Como resultado deste levantamento, foram classificadas um total de 26 demandas.

Com as demandas identificadas, as memas foram distribuídas em cinco grupos, ou classes, de forma que possibilitou a determinação de um peso em ordem de relevância, de acordo com a avaliação dos especialistas da equipe pedagógica, onde o maior peso determina maior importância e necessidade de atenção.

A tabela 3.7.1 mostra a classificação feita e seus respectivos pesos.

**Tabela 3.7.1**. Descriminação de classes e seus respectivos pesos, em ordem de relevância.

|  |  |
| --- | --- |
| Classe | Peso |
| Dificuldade cognitiva/psicopedagógica | 5 |
| Conflitos psicológicos | 4 |
| Problemas de relacionamento/comportamento no âmbito escolar | 3 |
| Problemas disciplinares | 2 |
| Dificuldade de relacionamento afetivo/familiar | 1 |

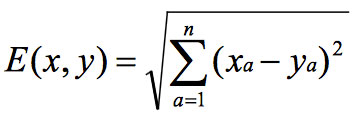
Determinou-se os pesos das classes de 1 a 5, de forma que, as demandas que necessitam de atenção maior, por parte do profissional pedagogo esteja contida nas classes de peso maior, enquanto que demandas consideradas menos relevantes pertencem a classes que possuem peso menor. A Tabela 3.7.1 relaciona as classes e os respectivos pesos.

Na Tabela 3.7.2, estão descritas todas as demandas que foram elencadas, no total de 26, que serão os atributos calculados na busca por similaridade entre os casos, relacionando estes atributos com os pesos que receberam de acordo com a análise pedagógica, pela classificação anteriormente determinada.

**Tabela 3.7.2**. Distribuição de demandas nas respectivas classes e pesos, conforme análise da especialista psicopedagógica por ordem alfabética.

|  |  |  |
| --- | --- | --- |
| Demandas | Classe | Peso |
| Conflito familiar | E | 1 |
| Pais em separação | E | 1 |
| Problema de relacionamento com a mãe | E | 1 |
| Problema de relacionamento com o pai | E | 1 |
| Problema de relacionamento em casa | E | 1 |
| Problema de ordem socioeconômica do aluno ou da família | E | 1 |
| Problema de ordem disciplinar grave | D | 2 |
| Problema de ordem disciplinar leve | D | 2 |
| Problema de ordem disciplinar médio | D | 2 |
| Problema de comportamento | C | 3 |
| Problema de relacionamento aluno x aluno | C | 3 |
| Problema de relacionamento professor x aluno | C | 3 |
| Bulling | B | 4 |
| Conflito com relação a opção sexual | B | 4 |
| Conflito devido a situação relacional ou afetiva | B | 4 |
| Desmotivação pela opção de curso | B | 4 |
| Desmotivação por baixo rendimento | B | 4 |
| Necessidade de orientação secular  (pessoal/não acadêmica) | B | 4 |
| Situação de abuso (moral, sexual etc.) | B | 4 |
| Situação de exclusão em sala | B | 4 |
| Situação relacionada à timidez | B | 4 |
| Atrasos constantes | A | 5 |
| Desequilíbrio ou problema de ordem psicológica | A | 5 |
| Dificuldade de aprendizagem em disciplina | A | 5 |
| Muitas faltas | A | 5 |
| Necessidade de orientação pedagógica | A | 5 |

Seguiu-se no processo de desenvolvimento do sistema **RBC**, o passo seguinte é definir um método de cálculo de similaridade, ou dissimilaridade, entre os casos, de forma a tornar os processos de indexação e recuperação efetivamente funcionais. Foi decidido a utilização da Distância Euclidiana para calcular as distâncias entre os casos, onde é calculada a distância para cada atributo que representa uma demanda entre os casos. A equação utilizada é a seguinte:

 Equação x

A Equação 1 descreve o cálculo euclidiano para medidas de dissimilaridade, onde ***E(x, y)*** é a distância e *x* e *y* são dois vetores com *n* atributos numéricos.

A Distância Euclidiana é uma medida de dissimilaridade, que representa um segmento de reta com a menor distância entre dois pontos, sendo escolhida para o desenvolvimento deste módulo por ser aplicável em espaços multidimensionais, com valores discretos ou contínuos. Os vetores a serem utilizados nos cálculos deste projeto, são o conjunto de demandas de cada caso registrado na base de casos, os atributos correspondem às demandas discriminadas no levantamento e classificação descrito anteriormente.

A Base de Casos de referência, foi populada com vários casos, todos bem definidos, completos e originados de experiências do mundo real. Para o levantamento dos mesmos, contamos com o apoio de profissionais especialistas em orientação pedagógica, para realizar o desenvolvimento da lista de demandas e principais soluções aplicadas aos casos, os quais tiveram conhecimento durante o primeiro período letivo do ano 2014, no Instituto Federal de Educação, Ciência e Tecnologia do RN – IFRN.

Na definição da Base de Casos, cada demanda está representada como um atributo do caso, sendo que, entre os casos elencados, alguns possuem mais de um atributo, porém, dados os limites da nossa fonte de informações, nem todos os atributos classificados estão contemplados na Base de Casos.

Estes atributos selecionados pelos especialistas, podem ser avaliados de forma condicional a estarem relacionados com o caso, e foram valorados numericamente com 0 (ausente) e 1 (presente). Ou seja, caso o atributo exista ele recebe o valor 1 (um), se não existir, o valor 0 (zero). Os atributos que não forem respondidos ou que não são de ciência do orientador pedagógico recebem igualmente o valor 0.

Outros atributos também foram selecionados, de forma a dar uma maior especificidade aos casos e que servirão para outros propósitos, como a geração de relatórios e fichas de acompanhamento pedagógico e assim por diante. São atributos multivalorados que receberam código para seus valores, de acordo com a forma de classificação. Por exemplo, o atributo SEXO, recebe valor 0 para masculino e valor 1 para feminino. Dados com valores escalares absolutos, ou com valores textuais, como por exemplo, o percentual de rendimento acadêmico, ou a quantidade de salários da renda familiar, recebem seu valor nominal.

Essa distribuição de atributos, é importante, caso deseje-se utilizar, o processo de mapeamento, onde serão aplicadas funções para calcular a distancia (dissimilaridade) entre os casos. Não deve se confundir esta função de mapeamento com o modelo de programação MapReduce, proposto por Dean e Ghemawat (2004), que está intrinsecamente ligado ao processamento e geração de dados massivos, sendo sua principal utilização em grandes massas de dados e processamento em clusters de computadores.

MapReduce foi inspirado por estruturas similares no mundo da linguagem funcional, onde *map* é uma função que é aplicada em cada membro de uma coleção e *reduce* é uma função aplicada através da coleção (Tiwari 2011).

É possível aplicar MapReduce na Base de Casos, e o Sistema Gerenciador de Banco de Dados – SGBD – MongoDB tem suporte nativo a esta técnica, porém, já que nossa base de dados é significativamente limitada, esta é uma recomendação para trabalhos futuros que queiram estender a aplicação deste trabalho ou aplica-lo a uma base de casos de amplitude massiva.

3.7.4**. Implementando o Sistema RBC**

De acordo com a definição de Raciocínio Baseado em Casos (RBC), é uma técnica de Inteligência Artificial, como um conjunto de atividades que auxilia na resolução de problemas, propondo soluções incontestavelmente utilizadas e documentadas, ao recuperar e adaptar experiências passadas – chamadas casos – armazenadas em uma Base de Casos. Um novo caso é resolvido com base na adaptação de solução de casos similares já conhecidos (Wangenheim *et al* 2013).

Segundo Watson (1997), RBC pode ser resumido como um processo cíclico que compreende os quatro R’s:

* Recuperação dos casos mais similares;
* Reutilização para solucionar o caso;
* Revisão da solução proposta;
* Retenção (aprendizado) da nova solução como parte de um novo caso.

Portanto, é necessário que, exista uma base de casos e um mecanismo de inferência que, através de similaridade apresente soluções para o caso apresentado fazendo uso, quando necessário, de adaptação dos casos existentes ao novo que foi apresentado. A Figura 2.8 do referencial teórico, descreve o processo cíclico que compõe as fases do RBC.

3.7.4.1. **Casos**

Como primeiro passo na construção da base de casos, está a aquisição e representação dos casos. No quesito aquisição, podemos encontrar diversos cenários (Abel e Castilho 1996) que podem incluir:

* Os casos não estão disponíveis em uma fonte externa;
* Os casos estão disponíveis parcialmente em uma fonte externa;
* Os casos estão disponíveis e contém erros;
* Os casos estão disponíveis e estão corretos.

Depois de coletados os casos, deve-se definir a forma como serão representados. Não existe um consenso a respeito de que tipo de dados ou quais informações devem estar contidas em um caso, mas devemos ter em mente dois aspectos significativos destas informações: a facilidade de aquisição da informação e a funcionalidade desta.

Um caso ideal pode ser estruturado de forma a conter os seguintes dados, de forma estruturada: uma descrição dos aspectos relevantes do caso – para este projeto, as demandas que são objeto de atenção por parte da ETEP; o contexto no qual o caso está inserido – pode ser, um texto descritivo ou palavras-chave que, associam o caso com um contexto específico; a descrição da solução associada ao caso – neste projeto, foi descrito uma solução, como o conjunto de encaminhamentos dados a cada atendimento pedagógico, inserido na base de casos; e a avaliação da solução empregada. Na Figura 3.7.4.1, descreve-se a composição de um caso ideal com demandas e encaminhamentos.

|  |
| --- |
| {  nome: {type: String, required: true},  sexo: Number,  telefone: {type: String},  email: {type: String},  matricula: {type: String, required: true},  curso: {type: String, required: true},  periodo: {type: Number, required: true},  data: {type: String, required: true},  observacoes: {type: String},  demandas: {  desequilibrioPsicologico: Number,  orientacaoSecular: Number,  orientacaoPedagogica: Number,  muitasFaltas: Number,  problemaSocioEconomico: Number,  atrasosConstantes: Number,  bulling: Number,  desmotivacaoRendimento: Number,  desmotivacaoCurso: Number,  dificuldadeAprendizagem: Number,  problemaDisciplinarLeve: Number,  problemaDisciplinarMedio: Number,  problemaDisciplinarGrave: Number,  problemaComportamento: Number,  conflitoFamiliar: Number,  problemaRelacionamentoPai: Number,  problemaRelacionamentoCasa: Number,  problemaRelacionamentoAluno: Number,  problemaRelacionamentoProfessor: Number,  situacaoAbuso: Number,  situacaoExclusao: Number,  situacaoTimidez: Number,  separacaoPais: Number  },  encaminhamentos: {  psicologiaEscolar: Number,  servicoSocial: Number,  setorMedico: Number,  chamarPais: Number,  chamarProfessores: Number,  acionarCoordenacao: Number,  advertencia: Number,  repreensao: Number,  suspesaoEscolar: Number,  trancarPeriodoCompulsorio: Number,  trancarPeriodoVoluntario: Number,  visitaDomiciliar: Number,  atendimentoDomiciliar: Number,  rotinaEstudo: Number,  centroAprendizagem: Number,  mudarTurno: Number,  conversar: Number  },  contexto: {type: String, required: true},  atendidoPor: {type: String, required: true},  classe: {type: String, required: true}  } |

**Figura 3.7.4.1** Descrição de um caso ideal utilizando a notação **JSON**

O que foi feito, foi o seguinte, pegarmos o domínio do sistema proposto, que é o aconselhamento pedagógico de estudantes, um caso ideal poderia ser descrito como, um documento onde existem registros com informações sobre cada estudante: nome, matrícula, turma, período e assim por diante. Estes seriam dados de contextualização.

Tem-se ainda, os aspectos relevantes do caso, geralmente nomeados de feição ou atributos, seriam as demandas de atenção, ou problemas que este aluno possa estar passando ou gerando para outrem. Ou seja, se o aluno está tendo problemas de relacionamento com outros alunos e/ou professores, se existe algum conflito familiar ou afetivo, se algum problema de ordem psicológica, econômica ou social está interferindo na aprendizagem e no convívio do aluno e assim por diante.

A descrição das soluções empregadas, incluiriam todas as ações tomadas pela orientação pedagógica, incluindo encaminhamentos e acompanhamento, por exemplo. Onde, Encaminhamentos podem ser: encaminhar efetivamente o aluno ao atendimento médico ou psicológico da escola, fazer visita à residência do aluno, elaborar rotinas de estudo, chamar os pais ou responsáveis para conversar e assim por diante.

A avaliação da solução empregada, poderia ser a descrição, ou consideração por parte da orientação pedagógica, sobre os resultados obtidos com a solução descrita e como estas contribuíram para o caso anterior, de forma a embasar a decisão de retomar esta como solução para o novo caso-problema.

3.7.4.2. **Base de Casos**

A estrutura de dados composta de casos, é chamada Base de Casos. Pode ser uma base de dados em forma de documentos, pares de chave-valor, objetos, predicados, redes semânticas, e assim por diante. O mais utilizado é o caso representado como um objeto, ou um frame, embora seja comum combinar um ou mais formalismos entre os citados (Wangenheim *et al* 2013).

Na estrutura de dados orientado a documentos, uma notação usada amplamente e que será usada na solução proposta é a **JSON** (Javascript Object Notation – Notação de Objetos Javascript), que permite descrição de objetos como documentos, agrupados em conjuntos de chave-valor, onde o valor pode ser um documento ou objeto agregado e subsequentemente. O Formato de Comunicação de Dados **JSON** é descrito pela **RFC 4627** (Crockford 2006), como sendo independente de linguagem, leve, baseado em texto e sua origem deriva dos padrões da linguagem de programação **ECMAScript**.

Este trabalho utiliza a notação JSON, e sistema de banco de dados orientado a documentos para representação de casos. Retomando o exemplo do item anterior, poderíamos escrever o documento que representa um exemplo de caso ideal, utilizando a notação JSON, da maneira descrita na Figura 3.7.4.1.

3.7.4.3. **Indexação**

A indexação é um processo importante e decisivo dentro do **RBC**, assim como numa base de dados convencional, porém não necessariamente, todas as informações relevantes deverão ser indexadas. As informações indexadas servirão para acelerar o processo de recuperação enquanto que informações não-indexadas podem servir de contextualização para a decisão de reutilização do caso, ou em outros aspectos do sistema de informação e não somente na recuperação.

Devemos ter em mente que os casos serão indexados de forma automatizada. Existem vários tipos de indexação. Indexação por medidas ou especificações, baseadas em diferenças, por similaridade e métodos indutivos de aprendizado são alguns exemplos que podemos utilizar na indexação. Para este projeto será utilizado, o mesmo algoritmo do k-Vizinho mais Próximo (neste trabalho abreviado para kNN, do inglês *k-nearest neighbor*) na indexação e na recuperação de casos.

Deve-se ficar atento para o fato de que, não tem uma maneira simples de como identificar, quais são as informações que deverão indexar os casos? Porém devemos nos ater a algumas características que índices devem possuir: ser preditivo; estar direcionado ao caso ao qual será aplicado; ser abstrato o suficiente para que através de extrapolação possa ser usado; ser concreto o suficiente para ser reconhecido no futuro (Wangenheim *et al* 2013).

3.7.4.4. **Recuperação**

A recuperação deve utilizar um algoritmo que verifique os clusters que tem maior similaridade com o caso-problema apresentado e, permita otimizar a reutilização e revisão do caso. Essa recuperação pode se dar como: uma busca por proximidade, em processos como o proposto por este trabalho; busca hierárquica, utilizando processos semelhantes ao algoritmo **ID3**, ou derivados deste, conforme descrito por Nagy et al (2013); existem ainda busca por simulação paralela, busca serial etc.

Ao contrário de um banco de dados relacional, uma base de casos, requer que a recuperação dos casos se dê de forma avaliada, envolvendo heurísticas e julgamentos sobre a relevância do dado recuperado. Entre os algoritmos mais relevantes e divulgados na recuperação de casos estão: Recuperação de Padrões, Algoritmo de Indução e Algoritmo de Vizinhança ou k-Vizinhos mais Próximos.

3.7.4.5. **Reutilização**

Raciocínio Baseado em Casos aplica-se na solução de problemas pela possibilidade de que, caso haja conhecimento do domínio aplicado ou das constantes existentes neste, pode-se aplicar uma técnica interpretativa, através da qual, podemos reutilizar uma solução conhecida de um caso semelhante, ou adaptar uma ou mais soluções conhecidas, para propor a solução para o caso apresentado. Existem duas etapas no processo de reutilização: Adaptação e Justificação e Crítica (Abel e Castilho 1996).

Adaptação ocorre quando, a solução ou conjunto de soluções propostas, não correspondem especificamente às ações que serão tomadas, porém, são feitos ajustes na solução para que, a mesma sirva ao problema dado. Já na justificação e crítica é necessária uma busca no domínio por uma combinação dos atributos que porventura possam causar problemas na solução, e corrigir estas distorções.

Retomando-se o exemplo do domínio educacional, e supondo-se que uma das soluções propostas seja o encaminhamento para orientação psicológica, com o psicólogo da escola. Agora imagina-se que a escola em questão não possua um profissional disponível, deverá então, ser feita adaptação na solução para que, o aluno, seja encaminhado a um psicólogo que fará o acompanhamento externo ao ambiente escolar.

Qualquer que seja, este trabalho conta com a interação do usuário especialista do domínio que, irá determinar e irá realizar os ajustes, críticas ou adaptações nas situações que julgar necessárias.

3.7.4.6. **Revisão e Retenção**

Revisão é o processo de julgar quão boa uma solução é. Neste passo, a solução deverá ter sido experimentada no mundo real e, recebe um feedback sobre o que ocorreu e, se foi correspondente ao esperado, ou se gerou alguma situação nova como resultado. Neste caso, será de suma importância, receber o feedback sobre a anomalia com descrição detalhada dos fatores, e como prevenir o surgimento de novas anomalias.

Essas informações podem ser: explicação das diferenças, justificativas diferentes, comparação de possibilidades alternativas e avaliação. Se o conjunto formado pelo problema proposto mais a solução avaliada resultar em um caso com características distintas, este novo caso é retido e indexado à Base de Casos.

3.7.4.7. **Similaridade e Problema de Agregação**

Uma questão generalizada enfrentada por pesquisadores das mais diversas áreas é, como organizar os dados observados em estruturas manipuláveis, isto é, como desenvolver taxonomia. Em vistas de resolver este problema, diversas técnicas e algoritmos vem sendo propostos. Ao conjunto destas técnicas, dá-se o nome de Análise Cluster, ou *clustering* (agrupamento), que faz menção a característica comum de agrupamento das características ou variáveis.

Agrupamentos são encontrados nas mais diversas situações no nosso dia-a-dia. Por exemplo, no ambiente escolar, existem os diversos grupos de estudantes separados por sala, ou em um restaurante, pode-se separar as pessoas em grupos, pelas mesas que compartilham.

Análise *cluster* é uma ferramenta de análise exploratória de dados que pega diferentes objetos e os separa em grupos, de forma que o grau de associação (similaridade) entre dois objetos é máxima, se eles pertencem ao mesmo grupo e mínima caso contrário. Dessa forma, análise *cluster*, pode ser usada para descobrir estruturas de dados, sem recorrer a uma explicação ou interpretação.

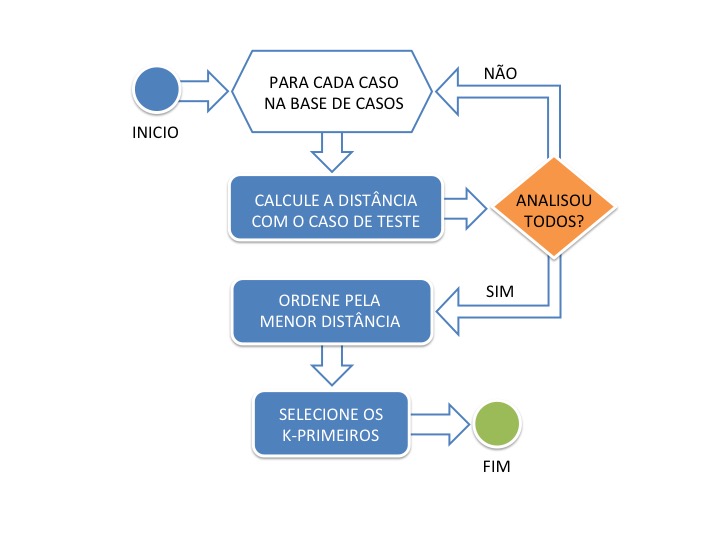
É importante observar que, a similaridade entre os membros das classes será cada vez menor quanto maior o nível de agregação. Um exemplo importante de *clustering*, é a classificação que os biólogos fazem dos homens separando-os em grupos (clusters) diversos, onde homem faz parte dos primatas, mamíferos, e animais, por exemplo. Homem tem similaridade maior com chipanzés (primatas) do que com jacarés (animais).

A taxonomia aplicada neste projeto, baseou-se nas demandas que compõe os casos, e nos grupos (classes) que podem ser construídos a partir destas informações. O resultado foi a separação em 5 classes de atributos, separados por aspectos do mundo real e a influência na vida acadêmica dos alunos. As tabelas 3.7.1 e 3.7.2 descrevem as classes os pesos e as demandas definidas para este projeto.

3.7.4.8. **Desenvolvimento do projeto**

O módulo RBC, faz parte de um sistema maior (Portal Acadêmico do IFRN), de onde os usuários do módulo RBC terão acesso. Está sendo proposto para o Portal, uma aplica aplicação web, desenvolvida com a linguagem de marcação HTML5, juntamente com a linguagem de programação Javascript para a construção da interface com o usuário e a lógica de negócio.

O sistema proposto por este módulo (RBC), faz uso do algoritmo intitulado Algoritmo **KNN**. O fluxograma simplificado do algoritmo é descrito na Figura 3.7.4.2.



**Figura 3.7.4.2**. Fluxograma simplificado do algoritmo kNN

Em princípio, o usuário fornece um conjunto de descrições do problema atual. Então, aplica-se uma função de cálculo para a medida de similaridade escolhida, à todas as instâncias de conhecimento contidas na base de casos. Em seguida, ordena-se os casos por ordem de similaridade, retornando um ou mais casos, até o limite estipulado por k, entre os mais próximos (Wangenheim *et al* 2013).

Uma desvantagem clara dessa abordagem, é o alto valor computacional do algoritmo, pois, é necessário comparar os dados de testes de cada novo caso apresentado, com todos os outros presentes na base de casos, recalculando as distâncias cada vez que é utilizado.

As principais características que, levam a utilizar-se este algoritmo, se encontra na sua facilidade de implementação e, na possibilidade de adaptação para diferentes entradas com múltiplas dimensões, inclusive.

Uma etapa natural deste algoritmo, é a classificação da entidade testada, inferindo a que grupo o conjunto de dados de teste pertence, com base na classe dominante, à qual pertencem os vizinhos mais próximos. Neste trabalho, esta etapa do algoritmo é descartada, em vista de que, optou-se por agrupar os casos, de acordo com a classe a qual pertence a demanda de maior peso presente no caso.

A Figura 3.7.4.3 exibe uma implementação de parte deste algoritmo, com ênfase no cálculo euclidiano de distâncias.

|  |
| --- |
| // Realiza o cálculo euclidiano para distâncias  var calcularDistancias = function (caso, vizinho) {  var pesos = {  atrasosConstantes: 5,  desequilibrioPsicologico: 5,  dificuldadeAprendizagem: 5,  muitasFaltas: 5,  orientacaoPedagogica: 5,  bulling: 4,  conflitoOpcaoSexual: 4,  conflitoRelacionalAfetivo: 4,  desmotivacaoCurso: 4,  desmotivacaoRendimento: 4,  orientacaoSecular: 4,  situacaoAbuso: 4,  situacaoExclusao: 4,  situacaoTimidez: 4,  problemaComportamento: 3,  problemaRelacionamentoAluno: 3,  problemaRelacionamentoProfessor: 3,  problemaDisciplinarGrave: 2,  problemaDisciplinarLeve: 2,  problemaDisciplinarMedio: 2,  conflitoFamiliar: 1,  separacaoPais: 1,  problemaRelacionamentoMae: 1,  problemaRelacionamentoPai: 1,  problemaRelacionamentoCasa: 1,  problemaSocioEconomico: 1  };  // insere o peso nas demandas  var c = [];  var v = [];  for (demanda in caso) {  c[demanda] = caso[demanda] \* pesos[demanda];  v[demanda] = (vizinho[demanda] != undefined) ? (vizinho[demanda] \* pesos[demanda]) : 0;  }  // calcula a soma dos quadrados das diferenças  var soma = 0;  for (demanda in c) {  soma += Math.pow(c[demanda] - v[demanda], 2);  }  return Math.sqrt(soma / 26);// retorna a distância normalizada  } |

Figura 3.7.4.3. Implementação do Cálculo de distância dos Casos na linguagem Javascript

O primeiro passo, é definir o peso de cada entrada, conforme classificação da Tabela 3.7.1. Em seguida, percorre-se os vetores de entrada, ou seja, o caso-problema (caso) e um caso presente na Base de Casos (vizinho), multiplicando cada atributo pelo peso correspondente. Segue-se então a soma dos quadrados das diferenças de todos os atributos e por fim divide-se pelo intervalo de demandas e faz-se o cálculo da raiz quadrada, retornando seu valor que, conforme o cálculo euclidiano normalizado, é a distância entre os casos. Repete-se o cálculo para cada entrada registrada na Base de Casos.

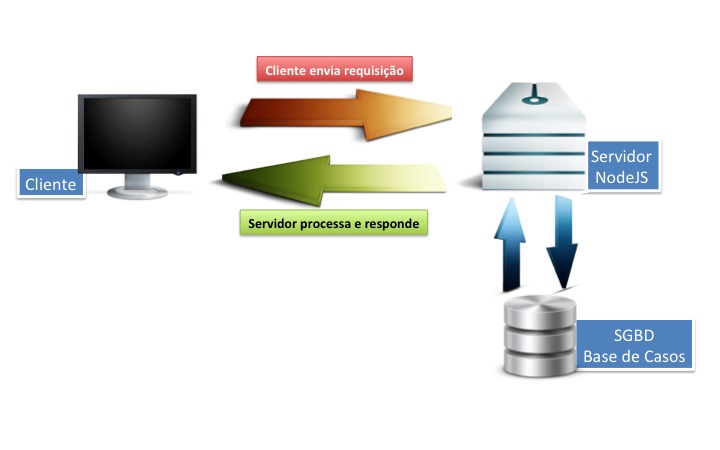
Uma outra particularidade deste trabalho, é o fato de que, todos os casos deverão ser armazenados, já que deverão ser registradas, todas as atividades de atendimento pedagógico atendidas pela ETEP. Para tanto, será utilizada uma segunda coleção de registro de casos, esta apenas para fins estatísticos e administrativos, sendo que, estes aspectos da aplicação, não serão exploradas no presente trabalho.

3.7.4.8.1. **Arquitetura do sistema RBC**

A arquitetura de um sistema pode ser definida como a descrição da organização dos diversos componentes do sistema e como estes, e o sistema, interagem entre si e com o ambiente no qual estão inseridos. A arquitetura selecionada para o desenvolvimento deste trabalho é o de cliente-servidor, comum em ambiente web. A Figura 3.7.4.4 demonstra essa arquitetura.

Na arquitetura cliente-servidor, conforme descrito na Figura 3.74.4, o usuário através da aplicação cliente realiza uma requisição e envia os dados ao servidor, que processa e responde ao cliente com as informações (Mendes 2002). Os dados são persistidos na base de casos e podem ser processados pelo servidor.

Esse modelo arquitetural é especialmente entendido como um modelo em camadas, onde permite-se compreender o papel de cada camada do sistema de forma isolada.



**Figura 3.7.4.4**. Modelo arquitetural de aplicação cliente-servidor.

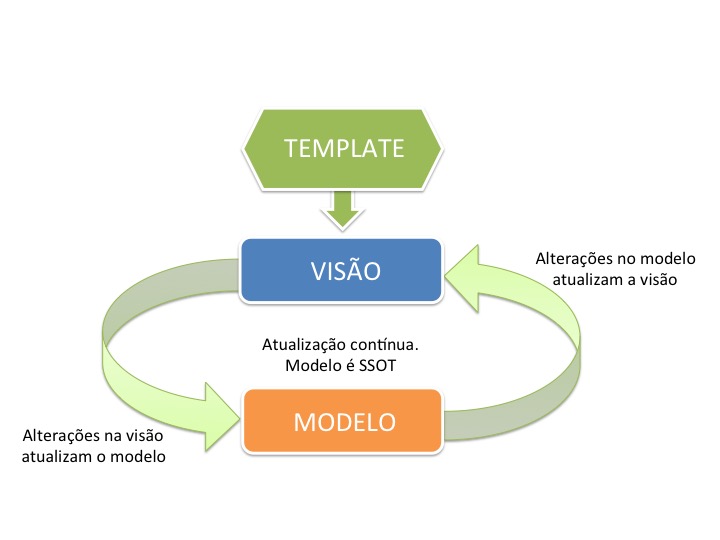
O sistema proposto é compreendido em três camadas: 1) interface com usuários – é a camada que os usuários utilizam para interagir com o sistema; 2) camada de negócio – presente no lado servidor, corresponde a lógica de domínio da aplicação; 3) camada de persistência – nesta camada os dados da aplicação são guardados em um repositório ou banco de dados.

3.7.4.8.1. **Interface com o Usuário**

A camada mais superficial da aplicação é a interface com o usuário. A interface deste módulo foi desenvolvida usando um conjunto de documentos e formulários escritos utilizando HTML5 e algumas bibliotecas de script Javascript que podem ser interpretadas pelo navegador de internet.

Entre as bibliotecas utilizadas, destaca-se **AngularJS**, por ser uma biblioteca escrita em Javascript que tem seu foco na manipulação de interface através de uma tecnologia nomeada *two-way data biding*. Esta tecnologia permite que a interface seja alterada conforme a lógica do sistema mude, sem a necessidade de recarregar a aplicação para visualizar as novas informações, assim como, também reflete as ações do usuário na lógica do sistema em tempo de execução (Google Inc. 2010). Ou seja, se relacionarmos os campos de um formulário a um objeto que pode ser manipulado com AngularJS, a cada interação que o usuário tenha no preenchimento deste, o objeto é modificado simultaneamente, permitindo, por exemplo, que os dados preenchidos possam ser verificados mesmo antes do usuário decidir encerrar a interação.

Com o uso desta tecnologia, o modelo passa a ser o recurso único da verdade (**SSOT** – *Single Source of Truth*, em inglês). **SSOT** refere-se a prática de estruturar a informação, de maneira que exista apenas uma fonte de informação daquele dado, quaisquer outras formas de acesso aos dados, se dá por referência apenas (Google Inc. 2010). A Figura 3.7.4.5 ilustra o funcionamento do *two-way data biding*.



**Figura 3.7.4.5**. Esquema de *Two-way data biding* na biblioteca AngularJS

**Fonte**: Adaptado de Google Inc, 2010.

3.7.4.8.2. **Servidor de Aplicações**

A segunda camada da aplicação fica alocada no servidor, responsável por receber as solicitações da interface com o usuário e, responder de acordo com o resultado do processamento da requisição. Esta camada intermediária é executada, em um ambiente de servidor desenvolvido para este fim, através da tecnologia usada para processar **Javascript** no lado do servidor chamada **NodeJS**.

**NodeJS** é uma plataforma de processamento, que utiliza a tecnologia direcionada a eventos, com o modelo de **I/O** não-bloqueante, leve e eficiente. Geralmente utilizada em aplicações de tempo real, com intensa transmissão de dados e, que funcionam como sistema distribuído. **I/O** é a sigla, em inglês, para o processo de entrada e saída de informações no servidor (Joyent Inc 2014).

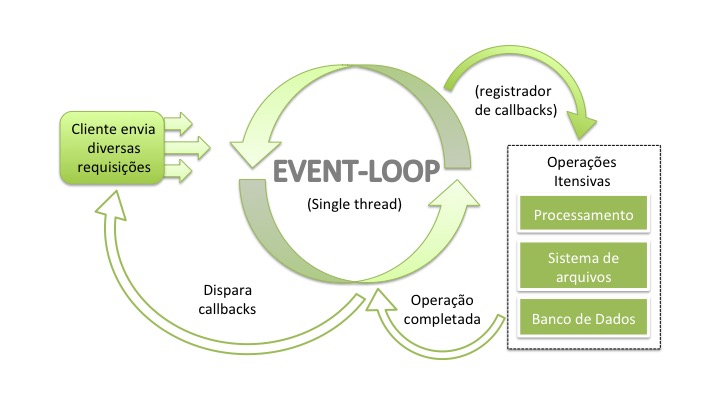
Dizer que, o modelo de entrada e saída é não-bloqueante, significa que o servidor, quando receber uma requisição, irá processá-la e em algum momento ele poderá responder aquela requisição, porém, para tanto, não deixará de receber e responder outras requisições que porventura cheguem até o mesmo.

Uma das características que permite o **NodeJS** ser não-bloqueante é que este é *single-thread*, ou seja, a aplicação terá somente uma instância de cada processo, sendo, no entanto, possível a criação de clusters, como no caso da biblioteca **Mongoose**, uma biblioteca escrita em Javascript para facilitar o trabalho do desenvolvedor que deseja conectar a aplicação com o **SGBD MongoDB** (LearnBoost 2010).

Outra particularidade, é a forma como se dá, a orientação a eventos do **NodeJS**. Enquanto que, nos navegadores de internet, os eventos aos quais o Javascript, estão relacionados serem, por exemplo, o clique do mouse, o pressionar de uma tecla etc., o servidor de aplicação **NodeJS** não responderá a estas ações. Os eventos a que estão relacionadas as atividades dele são, entre outros, *connect* (conexão com banco de dados), *open*, *read*, *close* (operações em arquivos em geral), *data* (como em transmissão de dados) e assim por diante.

A tecnologia de *callback* do Javascript tem um papel importante em todas as etapas de processamento do **NodeJS**, já que este é um item crucial para o gerenciador de eventos do servidor. É através dela que, podemos mesmo com a limitação de ser *single-thread*, realizar operações paralelas assíncronas de forma não-bloqueante (Joyent Inc 2014).

O mecanismo utilizado para o gerenciamento destes eventos é o *Event-Loop*. O *event-loop*, na verdade, é um ciclo infinito que percorre, a cada iteração, toda a sua lista de eventos e verifica se algum evento foi emitido. Quando identificado um evento, ele o executa e o encaminha para a fila de executados. O poder deste mecanismo de processo está na possibilidade de, enquanto um evento é processado, podermos definir funcionalidades que serão disparadas, e descrever a lógica destes. A Figura 3.7.4.6 exemplifica a execução do mecanismo de *Event-Loop* do NodeJS.



**Figura 3.7.4.6**. Mecanismo Event-Loop do NodeJS  
  
**Fonte**: Adaptado de Joyent Inc., 2014

3.7.4.8.3. **Persistência de Dados**

A terceira e última camada, é a camada de persistência dos dados. Foi utilizado um SGBD (Sistema Gerenciador de Bancos de Dados) que nos permitisse armazenar os casos de forma otimizada e mantivesse a mesma estrutura de documentos utilizada nas camadas superiores da aplicação, fazendo assim, com que os dados sofressem o mínimo de alterações possíveis e que somente as transformações necessárias fossem realizadas.

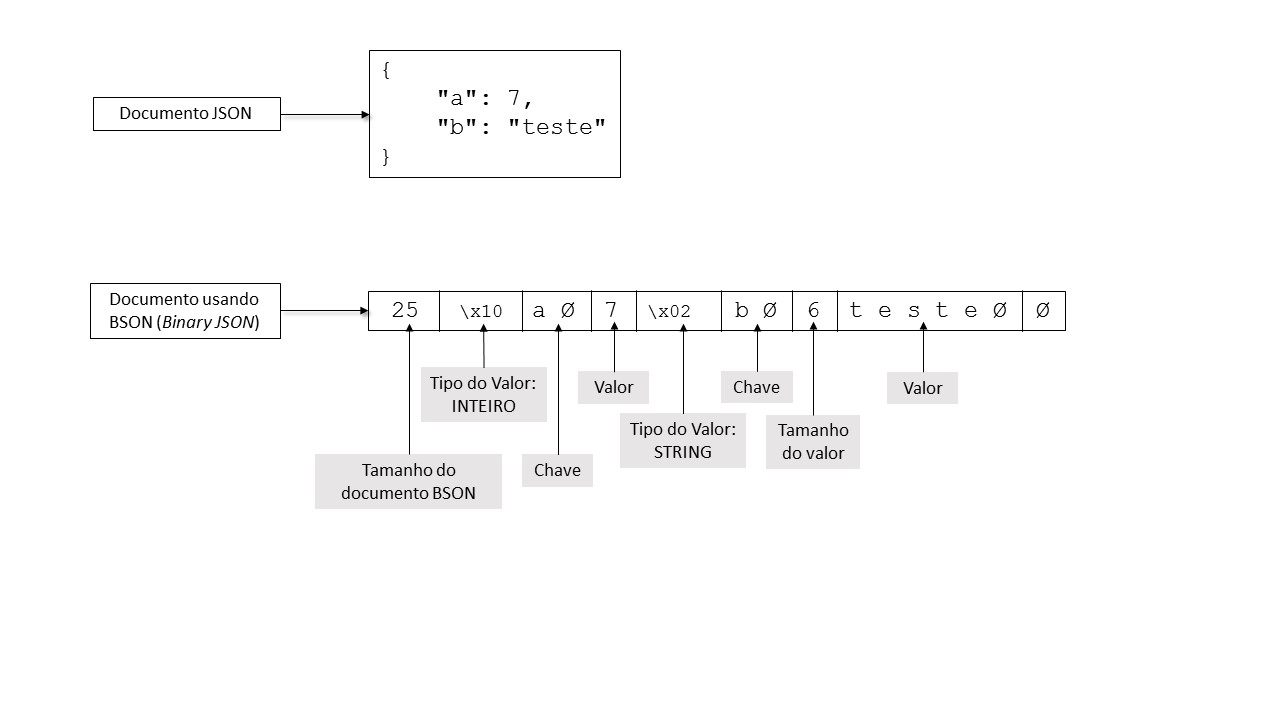
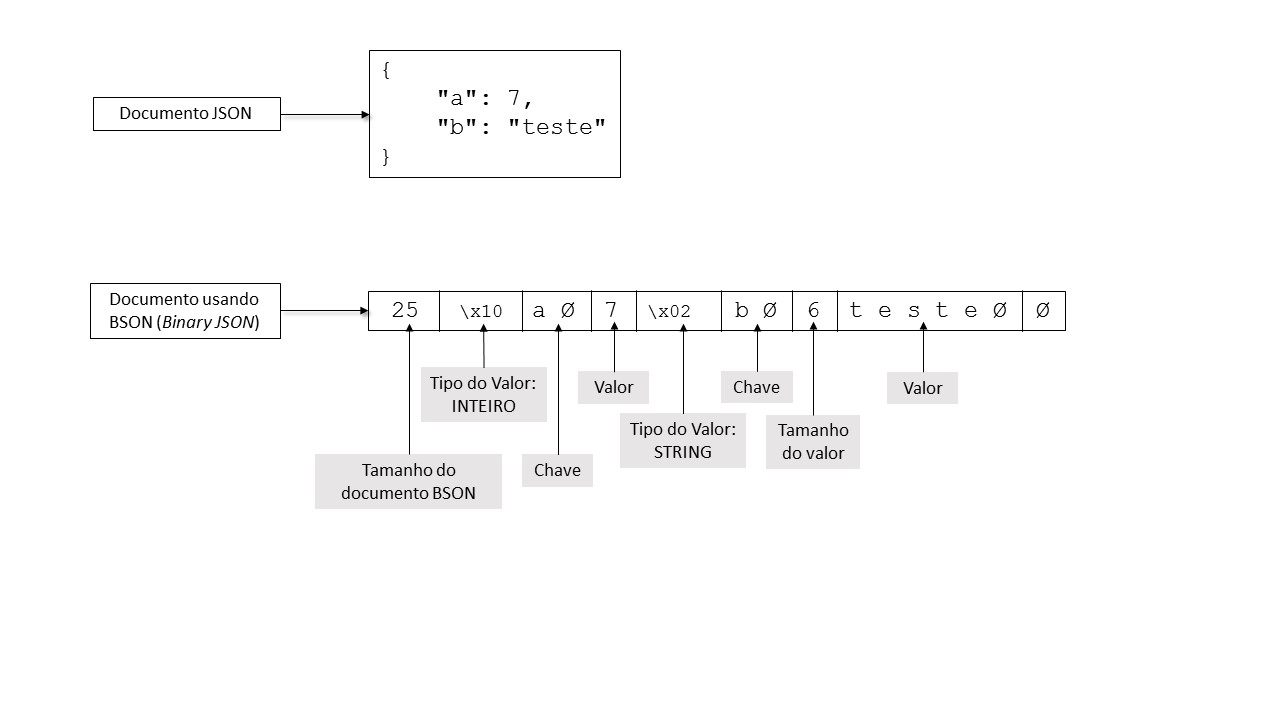
Como um fator importante ao **RBC**, a base de casos ser persistida em um banco de dados, que mantém sua estrutura na forma de coleção de documentos, e ainda permite a utilização nativa de algoritmos de classificação, nos auxiliando a obter resultados de forma nativa, tende a ser altamente eficiente.

Foi utilizado, para desenvolvimento do projeto objeto deste trabalho, um **SGBD NoSQL**, orientado a documentos, de código aberto e uso livre, de nome **MongoDB** (MongoDB Inc. 2014).

O termo **noSQL** é usado amplamente para descrever estruturas de bancos de dados que vão além da estrutura relacional, e que podem incluir conceitos de tuplas, grafos, pares de chave-valor, árvore, documento e assim por diante. Sadalage e Fowler (2013) afirmam que as características comuns dos bancos de dados NoSQL são:

* Não utilizam modelo relacional;
* Tem uma boa execução em clusters;
* Seu código é aberto;
* São criados para propriedades na web do século XXI;
* Não tem esquema (*schemeless*).

O **MongoDB**, utiliza também Javascript como linguagem nativa para suas rotinas e aplicações, além de armazenar os dados, em um formato semelhante à notação de objetos Javascript (**JSON**), só que em binário ao invés de texto pleno. A Figura 3.7.4.7 ilustra a comparação entre um documento representado em texto pleno e sua representação em binário utilizando a notação Javascript.



**Figura 3.7.4.7**. Comparação da representação de um mesmo documento usando **JSON** em texto pleno e na sua representação em binário, **BSON**.

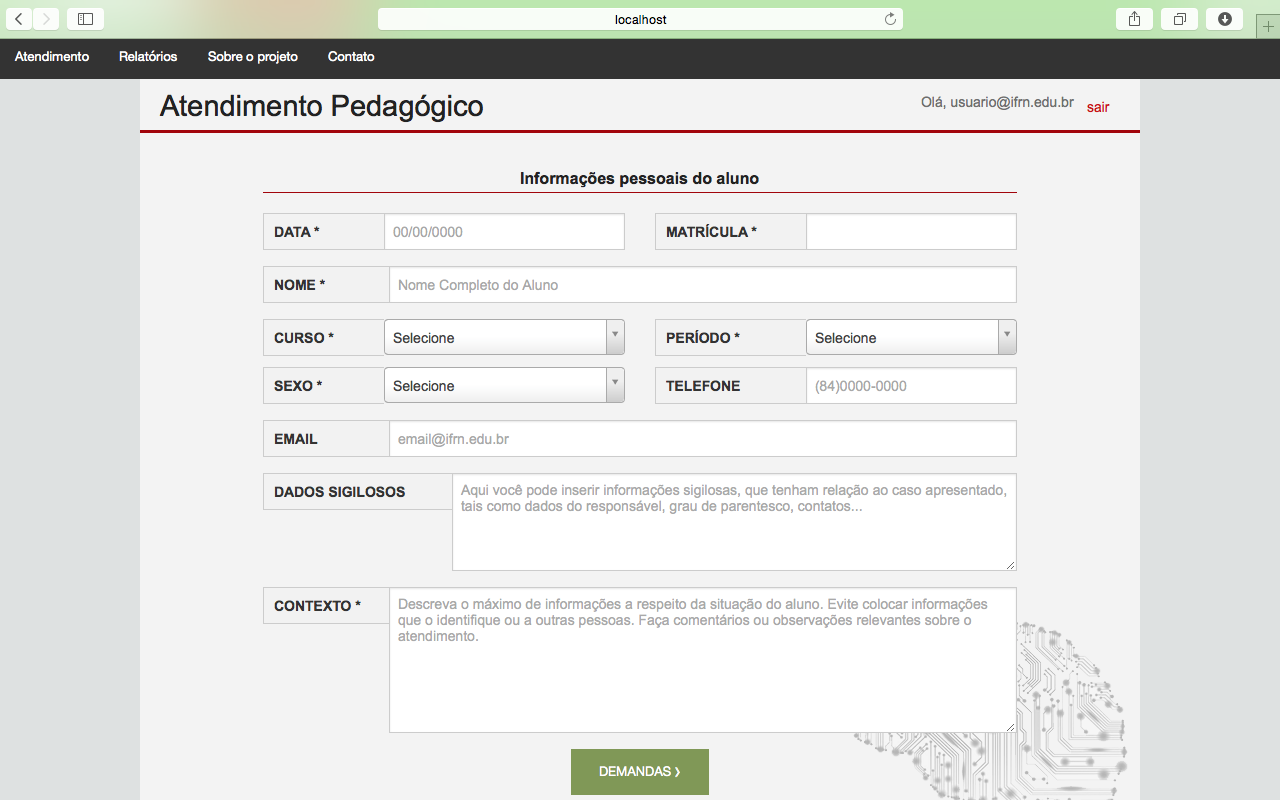
**Fonte**: Adaptado de Especificação BSON

3.7.4.9. **Resultados do Módulo RBC**

O principal resultado, deste módulo do projeto, foi o desenvolvimento de um sistema inteligente de apoio à decisão, que tem como público alvo de usuários, aconselhadores pedagógicos e técnicos de **ETEP**’s. O sistema foi desenvolvido com o conjunto de tecnologias descritas neste documento, e está disponibilizado como software de código aberto, sob a licença **MIT**. O **Anexo II** – **LICENSE**, é o texto integral da licença sob a qual está disponibilizado o software.

Como meio de disponibilização, e visando a possibilidade de futuras contribuições da comunidade, escolhemos criar um repositório público de código usando a plataforma web, na comunidade de desenvolvedores GitHub.com. O endereço **URL** de acesso público é **https://github.com/embits/educase**. Esta plataforma, além de permitir a publicação, possui uma ferramenta de versionamento de código, que se mostrou bastante útil durante o período de desenvolvimento.

A Figura 3.7.4.8 representa uma das telas do sistema, mais especificamente, o cadastro de informações de contextualização e discriminação do caso, com informações sobre o aluno, curso, informações de contato.



**Figura 3.7.4.8**. Interface com o usuário do sistema. Passo 1 – Informações pessoais do aluno

**Fonte**: Autor

Através de um formulário interativo, é realizado o cadastro do atendimento pedagógico em 3 passos, sendo que estes passos incluem as etapas de recuperação, reutilização, revisão e retenção do **RBC**, a partir das informações fornecidas e da consulta na base de dados, conforme descrito a seguir.

No primeiro passo, são fornecidas as informações pessoais do aluno, como nome, matrícula, curso, sexo, data do atendimento, entre outras. Nesta tela também deve ser fornecido uma descrição do contexto no qual o caso está inserido, descrevendo locais, pessoas e situações, que poderão servir inclusive em um momento posterior, quando este novo caso já fizer parte da base de casos e vier a ser sugerido como uma das prováveis soluções.

No segundo passo, o usuário deverá elencar as demandas, dentre a lista de 26 demandas elencadas pela equipe da **ETEP** e descrita na Tabela 3.7.2 supracitada. Após seleciona as demandas que estão relacionadas a este novo caso, o usuário deverá clicar no botão “Encaminhamentos”, de cor verde, ilustrado na Figura 3.7.4.9, que realizará uma requisição assíncrona ao servidor informando as demandas que foram selecionas. O servidor por sua vez calculará, dentre todos os casos registrados na Base de Casos, os vizinhos mais próximos com base no cálculo euclidiano de distância e irá responder a solicitação sugerindo as soluções dos casos mais similares. Este é o primeiro passo do processo de Raciocínio Baseado em Casos, a recuperação.



**Figura 3.7.4.9** Passo 2 – Seleção de demandas do novo caso.

**Fonte**: Autor

Quando o servidor responde à solicitação, o usuário poderá optar por, aplicar a solução que melhor julgar conveniente à situação do novo caso, sendo esta a etapa de reutilização, ou ainda verificar uma ou mais soluções sugeridas e adaptar ao cenário do novo caso. A tela de sugestões de solução também permite que seja visualizado o contexto no qual aquelas sugestões estão relacionadas. Esta é a etapa de revisão do **RBC**.

Por último, é feito o cadastro do caso novo, sendo enviada, nova requisição ao servidor, de forma que este deverá calcular se, o conjunto formado das demandas e dos encaminhamentos (soluções) apresentados, forma uma nova unidade de conhecimento, a qual deverá ser inserida à base de casos, encerrando o ciclo **RBC** com a fase de retenção, ou aprendizado.

O processo de cadastro de um novo caso, no último passo, possui algumas particularidades que merecem uma maior atenção, inclusive, porque aqui são realizadas algumas etapas importantes que, irão influenciar não somente nossa base de dados, mas também os registros gerais dos atendimentos pedagógicos e os problemas enfrentados pela **ETEP** do **IFRN**, conforme apontado pela pedagoga entrevistada.

Quando o servidor recebe a requisição, para persistir o registro de um caso, a primeira ação que este deve tomar é a de classificar o novo caso. Como exposto nessa seção, onde falamos sobre o algoritmo **kNN**, este é comumente utilizado para classificação, é através desse algoritmo que feita a proposição de classes com base na classe dominante entre os vizinhos mais próximo ao caso dado.

O **kNN** foi utilizado neste trabalho, tanto para fins de indexação, quanto para recuperação de casos na Base de Casos, através do uso do cálculo de distância e da seleção dos vizinhos mais próximos, porém na fase de classificação, utilizou-se o critério de classificar conforme a classe do atributo mais significativo presente no caso-problema, visto que, de acordo com a metodologia de enfrentamento de problemáticas que a equipe técnico-pedagógica utiliza, classificar os casos com base não em seus vizinhos mais próximos, mas sim com base na classe mais significativa entre as demandas que o compõe, facilitará a produção de dados estatísticos de forma a elencar mais evidentemente os problemas que necessitam de maior atenção.

Isto se dá pelo fato de que, muitas vezes, apesar de a classificação sugerida pelo **kNN** indicar determinado agrupamento, como a classificação foi inicialmente realizada em cima de cada uma das demandas especificamente, ocorre de serem sugeridas, às vezes, classificar o novo caso dentro de um grupo, ou classe, na qual este não possui nenhuma demanda relacionada.

Identificou-se que, este fenômeno ocorre devido a classe dominante estar presente em casos, que possui uma variedade de demandas, que influenciam aqueles casos que, de outra forma, estariam agrupados de maneira distinta à sugerida pelo algoritmo do vizinho mais próximo.

Outro ponto importante de abordar, na persistência da informação, tem a ver com a produção de conhecimento *versus* a necessidade de acumular informações dos atendimentos pedagógicos. A Base de Casos deve ser constituída de informações únicas. Cada caso ali presente, deve representar um conjunto isolado de demandas e soluções, de maneira que, ao ser solicitada a persistência de um caso que não produza um exemplar único e realmente novo de conhecimento a ser agregado, este deverá ser descartado, de forma a manter a integridade e a performance da técnica de **RBC**. Por outro lado, a atividade da **ETEP** necessita que todos os atendimentos sejam registrados, de forma a criar um banco de informações que reflitam as situações do mundo real. A solução encontrada para que, ambas condições sejam satisfeitas, foi a utilização de duas coleções distintas no **SGBD**. Uma será nossa base de casos, onde as solicitações de recuperação, reutilização, revisão e retenção serão cumpridas e calculadas com base nos princípios da Inteligência Artificial, com unidades distintas de conhecimento, da forma descrita anteriormente.

A outra coleção, irá persistir todas as solicitações de retenção de casos, mesmos aqueles que não gerem uma nova unidade de conhecimento para o **RBC**, mais que, registra e representa o conjunto global de atendimentos realizados. Esta segunda coleção irá auxiliar, entre outras coisas, na produção de relatórios, como demonstrado na Figura 3.7.4.10, que ilustra o relatório de demandas mais atendidas. Este tipo de relatório será um ferramental importante para a ETEP realizar atividades de prevenção e cuidados básico, tornando as informações colhidas durante os atendimentos “mais palpáveis e mais úteis”.



**Figura 3.7.4.10**. Relatório de Demandas mais atendidas.

**Fonte**: Autor

Por não ser, todavia, o cerne deste trabalho, a geração de relatórios e gráficos, mas sim, a construção de um sistema inteligente aplicado ao aconselhamento pedagógico, estas ferramentas deverão ser construídas em atividades posteriores, inclusive podendo ser explorada por outro trabalho, que tenham um enfoque mais estatístico ou voltado ao desenvolvimento deste tipo.

3.7.4.10 **Considerações Finais do Módulo RBC**

O desenvolvimento de Sistemas Inteligentes usando RBC tem demonstrado que, apesar do esforço na construção de uma Base de Casos suficientemente grande, e com a definição de algoritmos e técnicas de indexação e recuperação cada dia mais eficazes, um problema recorrente que ainda deve ser solucionado é o da aplicabilidade de soluções que se baseiam em quão similar são os casos entre si.

Isso se deve ao fator similaridade, referindo-se ao conhecimento, de forma a trazer um conjunto de experiências passadas, que possam ser utilizadas na solução do problema atual. Dado que, na maioria das vezes, a quantidade de casos é insuficiente ou inadequada para refletir a solução dos problemas, uma abordagem que tem uma grande chance de ser mais eficiente, ou que traga soluções mais adequadas, seria a utilização de algoritmos que proponham as soluções mais úteis, ao invés de sugerir soluções com base nos casos mais similares.

3.7.4.11. **Referências do módulo RBC**

ABEL, Mara; CASTILHO, José Mauro Volkmer. **Um Estudo Sobre Raciocínio Baseado em Casos** – 1996. UFRS – Porto Alegre, RS.

Crockford, D. **RFC 4627** – 2006. The Internet Society – Acessado em: 23/9/2014. Disponível em: http://www.ietf.org/rfc/rfc4627.txt.

FERREIRA, Silvio. **Guia Prático de HTML5** – 2013. Editora Universo dos Livros – São Paulo, SP.

FLANAGAN, David. **Javascript: O guia definitivo** – Tradução de Edson Furmankiewicz – 2004. Editora Bookman – Porto Alegre, PR.

JEFFREY, Dean; GUEMAWAT, Sanjay. **MapReduce: Simplified Data Processing on Large Clusters** – 2004 – Google Inc. Acessado em 14/9/2014. Disponível em: http://static.googleusercontent.com/media/research.google.com/pt-BR//archive/mapreduce-osdi04.pdf.

MENDES, Antônio. **Arquitetura de Software: desenvolvimento orientado para arquitetura** 2002 – Editora Campus. Rio de Janeiro – RJ.

NAGY, Heba Mohammed; ALY, Walid Mohamed; HEGAZY, Osama Fathy. **An Educational Data Mining System for Advising Higher Education Students** – 2013. International Journal of Computer, Information Science and Engineering, Vol:7, N:10, 2013. World Academy of Science, Engineering and Technology.

SADALAGE, Pramod J; FOWLER, Martin. **NoSQL Essencial, Um guia conciso para o mundo emergente da persistência poliglota** – 2013. Novatec Editora – São Paulo, SP.

TIWARY, Shashank. **Professional NoSQL, a hands-on guide to leveraging NoSQL databases** – 2011. John Wiley & Sons.

WANGENHEIM, Christiane Gresse von; WANGENHEIM, Aldo von e RATEKE, Thiago. **Raciocínio Baseado em Casos com Softwares Livres e Aplicativos Móveis** – 2ª edição - 2013. Bookess Editora – Florianópolis, SC.

WATSON, Ian Duncan. **Applying Case-Based Reasoning** – 1997. Morgan Kauffman Publishers – San Francisco, CA.

WHITBY, Blay. **Inteligência Artificial – Um guia para iniciantes** - 2004. Editora Madras – São Paulo, SP.

\_\_\_\_. **AngularJS** – 2010. Google Inc. – Acessado em: 14/8/2014. Disponível em: http://docs.angularjs.org

\_\_\_\_. **BSON Specs** – 2014. Creative Commons (sem direitos autorais). – Acessado em: 6/8/2014. Disponível em: http://bsonspec.org/spec.html

\_\_\_\_. **MongoDB** – 2014. MongoDB Inc. – Acessado em: 14/8/2014. Disponível em: http://www.mongodb.com/what-is-mongodb

\_\_\_\_. **Mongoose** – 2010. LearnBoost. – Acessado em: 14/8/2014. Disponível em: http://mongoosejs.com/docs/guide.html

\_\_\_\_. **NodeJS** – 2014. Joyent Inc. – Acessado em: 10/6/2014. Disponível em: http://nodejs.org/documentation

\_\_\_\_. **Standard ECMA-262** – 2011. ECMA International – Acessado em: 23/9/2014. Disponível em: http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf

3.8. **Módulo Game Educativo**

3.9. **Módulo Portal**

Este módulo do projeto, tem como objetivo o desenvolvimento de um sistema de informação, que está sendo denominado de Portal. É através dele que, todos os utilizadores do sistema irão interagir com cada módulo, desenvolvido, o armazém de dados (*Data warehouse*), a mineração de dados (*Data Mining*), o sistema de aconselhamento pedagógico (**RBC**) e o jogo.

Para implementar uma solução informatizada, deve-se em primeira mão, fazer a modelagem do sistema. Portanto, a seguir é descrita toda a modelagem do sistema, ou seja, do Porta.

3.9.1. **Atores do sistema**

O primeiro passo é definir que são os atores do sistema. Então, a Tabela 3.9.1 mostra os possíveis atores do Portal.

**Tabela 3.9.1.** Atores do Portal IFRN

**Fonte**: Autor

|  |  |  |
| --- | --- | --- |
| **Nome** | **Descrição** | **Responsabilidades** |
| Aluno | Todos os estudantes matriculados no IFRN | Acompanhar seu desempenho nas disciplinas e acessar o jogo para desenvolver alguma habilidade em assuntos que o mesmo tenha dificuldades de aprendizagem. |
| Professor | Todos que fazem parte do corpo docente do IFRN | Acomanhar o rendimento da turma, disciplina a diciplina. E com isso, poder identificar com antecedência, os alunos problema e, com essa medida preventiva, tentar dimunuir a taxa de repetência e ou evasão escolar. |
| Gestor | Corresponde aos Pedagogos, cordenadores, etc. | Consultar dados e relatórios dos mais variados tipos, fornecidos pelo data warehouse e pelo data minig (consultas Ad hoc). |
| Usuários eventuais | Pessoas externas | Algumas informações serão disponibilizadas ao público externo, como por exemplo, aos pais dos alunos. |
| Administrador | José Antônio da Cunha | Acesso a todo o sistema. |

3.9.2. **Visão do Produto**

3.9.2.1. **Requisitos Funcionais**

Segundo Pressman (2001), os requisitos funcionais são aqueles que descrevem os serviços que o sistema deve oferecer, assim como qual é seu comportamento, como deve reagir a certas entradas e como deve comportar-se em determinadas situações.

De acordo com Sommerville (2007), os requisitos funcionais descrevem a funcionalidade ou o serviço que se espera que o sistema forneça.

Então, partindo-se das definições de requisitos dos autores e, após o entendimento de todas as operações do sistema, foram definidos os requisitos funcionais do sistema. Estas funcionalidades foram definidas conjuntamente com o gerente do projeto e todos os envolvidos direta ou indiretamente com o sistema. Para isto, foram realizadas reuniões e entrevistas com os gestores e a toda a equipe pedagógica do instituto.

A Tabela 3.9.2 mostra os requisitos funcionais do sistema identificados através de consultas e entrevistas realizadas junto a comunidade acadêmica do **IFRN** e também, entrevistas realizadas com os administradores do sistema acadêmico do instituto (Equipe do centro de processamento de dados do instituto). Lembrando que, o sistema Acadêmico é a principal fonte de dados, de onde são coletados, a grande massa de dados para o armazém de dados e, para a análise de dados. A Tabela 3.9.2 mostra os requisitos funcionais identificados.

**Tabela 3.9.2**. Requisitos funcionais

**Fonte**: Autor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cód | **Requisito** | **Descrição** | **Ator(es)** | **Prioridade** |
| F01 | Importar dados das planilhas para o modelo multidimensional (Storage área) | Este requisito permite usar os dados das planilhas no modelo multidimensional. | Administrador | Alta |
| F02 | Realizar todas as agregações para as fact tables, onde estão armazenadas as medidas do modelo multidimensional | Este requisito permite armazenar as informações na tabela central. | Administrador | Alta |
| F03 | Gerar Relatórios AD HOC usando Excel | Este requisito é possível gerar relatórios usando excel | Gestor e Professores | Média |
| F04 | Gerar Relatórios AD HOC usando MDX | Este requisito é possível gerar relatórios usando MDX | Administrador | Média |
| F05 | Selecionar Cubo que o usuário deseja trabalhar | Neste requisito pode-se escolher o cubo que será usado. | Gestor | Alta |
| F06 | Selecionar técnica de Mineração de Dados | Neste requisito é capaz de selecionar o tipo de técnica de mineração que mostrará os dados de acordo com sua classificação, como árvore e clusters. | Administrador | Alta |
| F07 | Definir parâmetro a ser analisado pela técnica selecionada | Os parâmetro podem ser reprovação escolar, evasão escolar, aprovação escolar, conclusão de cursos. | Administrador | Média |
| F08 | Processar o modelo | Processando o modelo, será obtido os resultados | Gestor | Média |
| F09 | Exibir o resultado de forma: gráfica, árvore, tabela, etc | Permite ver os dados em forma de gráficos, tabelas, para um melhor entendimento | Gestor | Alta |
| F10 | Interpretar o resultado | Permite analisar os índices indicadores de desempenho escolar | Gestor | Alta |
| F11 | Armazenar o resultado para uso posterior | Os dados ficarão guardados para uma consulta futura. | Administrador | Alta |
| F12 | Montar a tela com os campos a serem preenchidos com as informações relacionadas ao problema a ser solucionado | Na tela deve conter os resultados com os dados coletados para testes dos modelos, do campus Natal Central, de todos os cursos, referente aos anos de 2010 até 2013 | Gestor | Média |
| F13 | Buscar solução similar na base de casos | Comparar os resultados com a base de casos | Gestor | Média |
| F14 | Analisar a solução encontrada | Com a solução em mãos, deverá empregá-la da melhor maneira possível. | Gestor | Média |
| F15 | Adaptar a solução para o problema em questão | Adaptando a solução poderá ter um melhor aproveitamento do resultado. | Gestor | Alta |
| F16 | Gravar nova solução na base de casos | A nova solução deverá ficar na base de dados para uma consulta futura. | Gestor | Alta |
| F17 | Verificar qual é a sua dificuldade em fazer certas disciplinas | Neste requisito é possível analisar a situação do aluno. | Aluno | Média |
| F18 | Sugerir ao aluno alguma tarefa a ser desenvolvida para melhorar o seu desempenho escolar | A tarefa sugerida será através de Game Compettitions, jogos educativos. | Professor | Média |
| F18 | Cadastrar usuário |  |  |  |

3.9.2.2 **Requisitos Não-Funcionais**

De acordo com Pressman (2001), os requisitos não funcionais descrevem as restrições de serviços ou de funções do sistema, assim como de tempo, restrição do processo de desenvolvimento e padrões. Eles expressam como esse processo deve ser feito, definindo as propriedades do sistema e as suas restrições. De um modo geral, estão relacionados a padrões de qualidade como confiabilidade, tempo de resposta, robustez e restrições técnicas.

Os requisitos funcionais normalmente surgem conforme a necessidade dos usuários, contemplando restrições de orçamento e de políticas da organização, dentre outras restrições. Para este sistema, foram realizadas reuniões com a equipe de TI e discursões com os usuários do domínio da aplicação, para se identificar quais os requisitos não funcionais do sistema em questão. A Tabela 3.9.3 descreve os requisitos não funcionais onde, alguns são padrões da TI, alguns fazem parte da arquitetura do sistema, alguns são restrições do sistema operativo (OS), alguns estão relacionados a forma como o projeto irá rodar, tais como os relacionados a tempo, recursos, localização e orçamento e assim por diante.

**Tabela 3.9.3**. **Requisitos Não-funcionais**

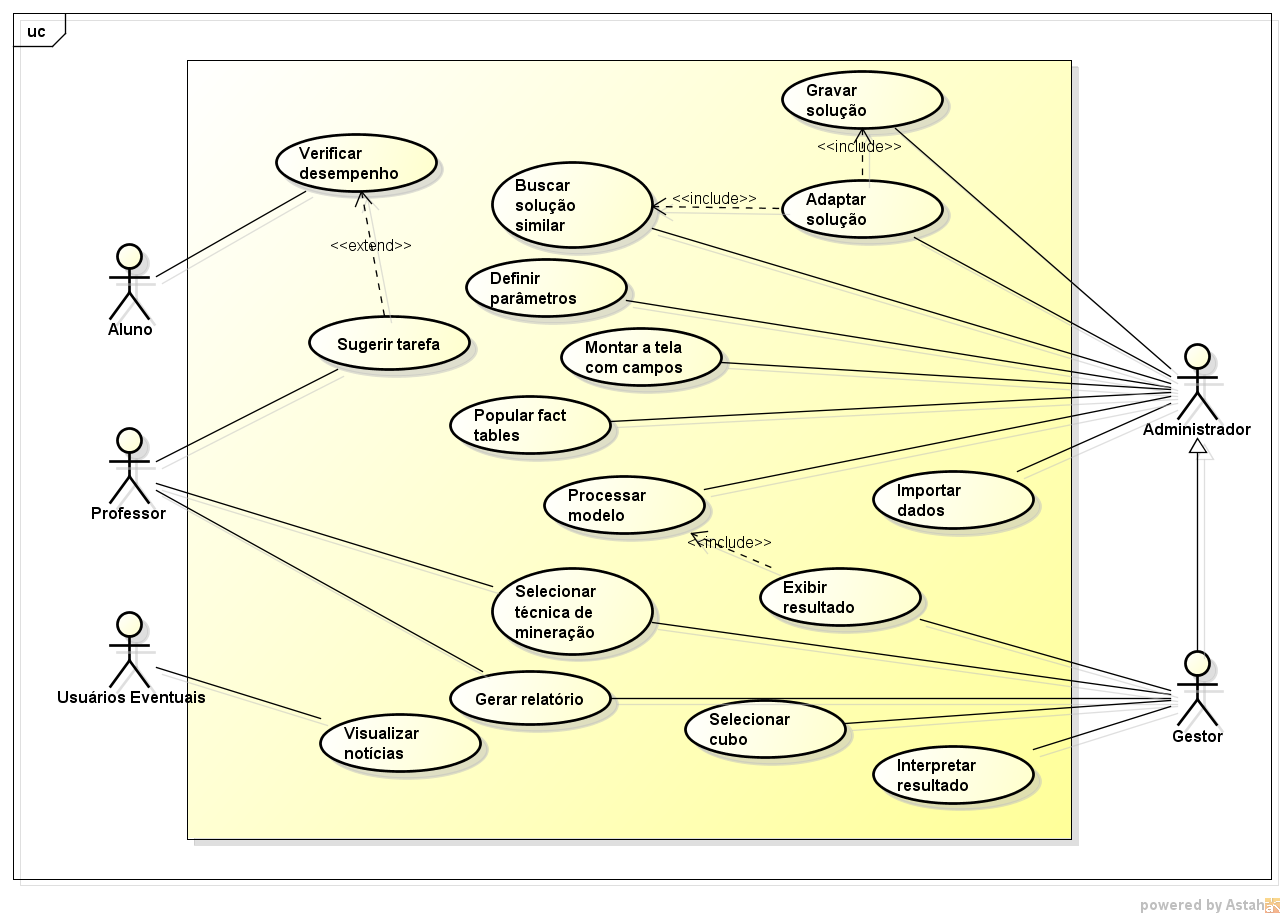
Fonte: Autor

|  |  |  |
| --- | --- | --- |
| **Código** | **Requisitos** | **Prioridade** |
| NF01 | SQL Server | Alta |
| NF02 | MySql | Alta |
| NF03 | Planilhas eletrônicas | Alta |
| NF04 | Mecanismos de acessibilidade | Alta |
| NF05 | Mecanismos de navegação | Alta |
| NF06 | Interface | Alta |
| NF07 | Controle de acesso | Alta |
| NF08 | Estabilidade | Alta |
| NF09 | Confidencialidade | Alta |
| NF10 | Tolerância a falhas | Alta |
| NF11 | Flexibilidade de navegadores | Alta |

3.9.2.3. **Casos de Uso**

A descrição de um Caso de Uso define o que o sistema faz quando o caso de uso é executado. A funcionalidade de um sistema está definida por um conjunto de casos de usos em que cada um representa um fluxo especifico de eventos [Kruchten, 2003][Bezerra, 2002].

O caso de uso flui os eventos expressos do comportamento do sistema, mostrando como o caso de uso é executado de fato, em termos de objetos interagindo e classes. Uma maneira de representar os casos de usos de um sistema é através de um diagrama de casos de uso. A Figura 3.9.1 mostra o diagrama de casos de uso para o sistema proposto[Kruchten, 2003][Bezerra, 2002].



**Figura 3.9.1**. Diagrama de Casos de Uso para o Portal

**Fonte**: Autor

Resumindo, casos de uso são meios de expressar exigências na funcionalidade do sistema. Além disso, os casos de uso ajudam a sincronizar o conteúdo de vários modelos.

A Tabela 3.9.4 faz um detalhamento de cada caso de uso do diagrama da Figura 3.9.1.

**Tabela 3.9.4** Detalhamento de cada caso de uso do sistema Portal

**Fonte**: Autor

|  |  |  |
| --- | --- | --- |
| **Nome** | **Atores** | **Descrição** |
| 1 UC – Verificar desempenho | Aluno, Professor | A turma ou o aluno terá seu desempenho avaliado, sendo avaliado constantemente. |
| 2 UC - Sugerir tarefa | Professor | A tarefa é sugerida aos alunos com desempenho inferior a média. |
| 3 UC - Gerar relatório | Gestor e administrador | Com os dados corretos será possível gerar o relatório com as informações necessárias. |
| 4UC - Importar dados | Administrador | Os dados serão importados de diversos sistema de informação (acadêmico, formulários de pesquisas do aluno, do professor e da instituição). |
| 5UC - Definir parâmetro | Administrador | Definir os seguintes parâmetros (dados demográficos como idade, sexo, situação financeira dos pais, nível de escolaridade dos pais, região de procedência, etc; e dados comportamentais, como o número de reprovações, a situação no curso, e os dados sobre avaliações. |
| 6UC - Selecionar cubo | Gestor e administrador | Cada cubo será um tipo de dados que o gestor poderá selecionar. |
| 7UC - Popular fact tables | Administrador | Preencher as tabelas com valores. |
| 8UC - Selecionar técnica de mineração | Administrador, gestor e professor | O gestor poderá selecionar entre as técnicas: Árvores, Clusters, etc. |
| 9UC - Processar modelo | Administrador | O modelo será processado pelo sistema mostrando o resultado. |
| 10UC - Interpretar resultado | Gestor, Administrador | O resultado deverá ser interpretado pelo gestor. |
| 11UC - Montar tela com campos | Administrador | Com o resultado interpretado será montada a tela com os campos reprovação escolar, evasão escolar, aprovação escolar, conclusão de cursos |
| 12UC - Buscar solução similar | Administrador | A solução encontrada será comparada com outras. |
| 13UC - Adaptar solução | Administrador | A solução deve ser adaptada de acordo com a necessidade. |
| 14UC - Gravar nova solução | Administrador | Todas as soluções devem ser arquivadas para consulta. |
| 15UC - Exibir resultado | Administrador e Gestor | O resultado das soluções de ser exibidos. |

3.9.2.3.1. **Descrição detalhada dos casos de uso**

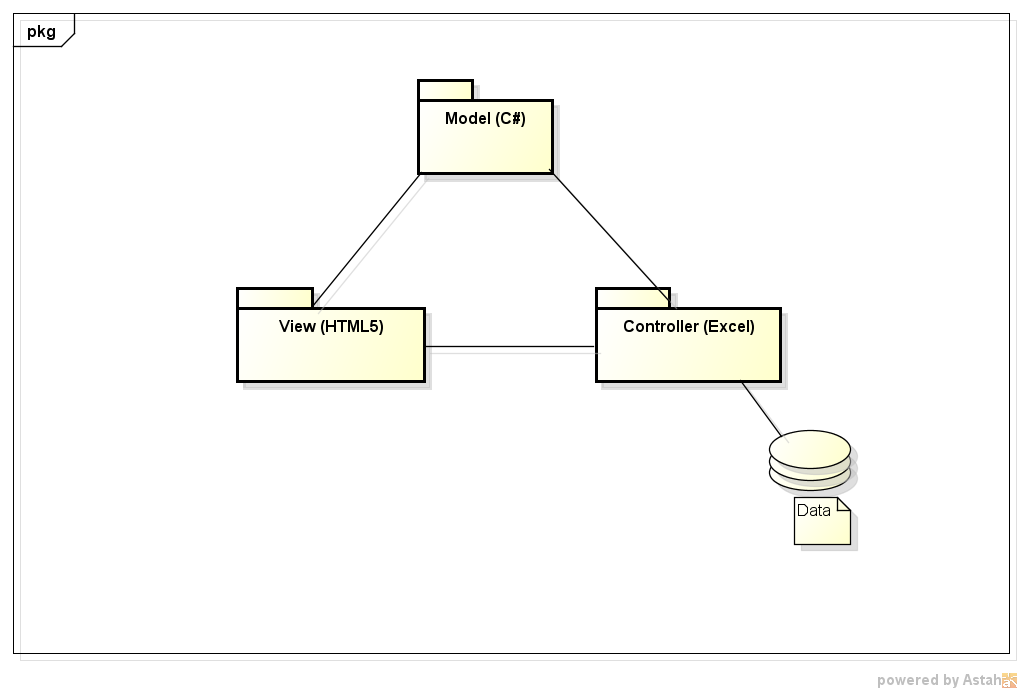
A Figura 3.9.2 descreve de forma detalhada cada caso de uso.

|  |
| --- |
| **Caso de uso - Verificar desempenho**  **RESUMO:** Com os resultados obtidos será possível visualizar o desempenho da turma ou aluno.  **ATOR PRINCIPAL:** Aluno  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**  1. [IN] O usuário na página inicial terá no menu a opção ”Verificar Desempenho”.  2. [OUT] O sistema irá retorna todas as informações referente ao aluno ou turma.    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Sugerir Tarefa**  **RESUMO:** A tarefa será sugerida aos alunos com nota inferior a média  **ATOR PRINCIPAL:** Aluno, professor  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema, e desempenho abaixo da média.  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**  1. [IN] O usuário na página inicial terá no menu a opção ”Verificar Desempenho”.  2. [OUT] O sistema irá retorna todas as informações referente ao aluno ou turma.  3. [OUT] Caso o sistema esteja baixo, será sugerido tarefas em forma de games educativos.    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Gerar relatórios**  **RESUMO:** Todos os dados analisados deverão gerar relatórios.  **ATOR PRINCIPAL:** Gestor, professor.  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Importar dados**  **RESUMO:** Os dados serão importados de diversas fontes, através de planilha eletronica.  **ATOR PRINCIPAL:** Gestor  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Definir parâmetro**  **RESUMO:** Os parâmetros serão definidos depois de selecionar a técnica de mineração  **ATOR PRINCIPAL:** Gestor  **PRÉ-CONDIÇÃO:** Selecionar a técnica de mineração  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Selecionar cubo**  **RESUMO:** O modelo multidimensional será usado pela ferramenta OLAP, para criação dos cubos de dados.  **ATOR PRINCIPAL:** Aluno  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**  1. [IN] Haverá uma lista de cubos do lado direito da página, no qual o usuário poderá selecionar.    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Popular fact tables**  **RESUMO:**  **ATOR PRINCIPAL:** Administrador  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Selecionar técnica de mineração**  **RESUMO:** Deverá ser escolhida uma técnica de mineração, para prosseguir com a análise dos dados  **ATOR PRINCIPAL:** Gestor, professor  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**      **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Processar modelo**  **RESUMO:** Depois de escolhida a técnica de mineração os dados serão processados  **ATOR PRINCIPAL:** Gestor, administrador  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Exibir Resultado.**  **RESUMO:** Com os resultados obtidos será possível editar dados de acordo com a necessidade.  **ATOR PRINCIPAL:** Gestor  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Interpretar resultado**  **RESUMO:** Com os resultados obtidos será possível interpretá-los, ajudando na tomada de decisões educacionais, mais eficientes.  **ATOR PRINCIPAL:** Gestor  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Montar tela com campos**  **RESUMO:** Os campos deverão conter as seguintes informações: (**dados demográficos** como idade, sexo, situação financeira dos pais, nível de escolaridade dos pais, região de procedência, etc), e **dados comportamentais**, como o número de reprovações, a situação no curso, informações de evasão e os **dados sobre avaliações**.  **ATOR PRINCIPAL:** Gestor  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Buscar solução similar**  **RESUMO:** A solução obtida, deverá ser comparada com outros que estarão no sistema.  **ATOR PRINCIPAL:** Gestor  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Adaptar solução**  **RESUMO:** Tentar adaptar a solução da melhor forma possível.  **ATOR PRINCIPAL:** Gestor  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |
| **Caso de uso - Gravar solução**  **RESUMO:** A solução deverá ser gravada para uma consulta futura.  **ATOR PRINCIPAL:** Administrador  **PRÉ-CONDIÇÃO:** O usuário deverá estar cadastrado no sistema  **PÓS-CONDIÇÕES:**  **FLUXO PRINCIPAL:**    **FLUXO DE EXCEÇÃO:**  **FLUXO ALTERNATIVO:** |

Figura 3.9.2. Detalhamento dos casos de uso

3.9.3. **Arquitetura do Portal**

É através da arquitetura de sistema que é realizada a o mapeamento de funcionalidade para componentes de hardware e software, e também um mapeamento da arquitetura de software de hardware para a arquitetura de hardware e uma interação humana com esses componentes. A Figura 3.9.3 mostra a arquitetura do Portal.



**Figura 3.9.3**. Modelo arquitetural do Portal

**Fonte**: Autor

Como pode-se perceber, o modelo utilizado será o modelo arquitetural **MVC**, onde **M** – *Model*, **V** – *View* e **C** – *Control*.

* **Model**: Ele é **responsável** pela **leitura** e **escrita** de **dados,** e também de suas **validações**.
* **View**: A camada de interação com o usuário. Responsável por mostrar informações ao usuário e interpretar os comandos do usuário [Evans, 2010]. Ela apenas faz a **exibição** dos **dados**, sendo ela por meio de um **html** ou **xml**.
* **Control**: É responsável por **receber** todas as **requisições** do **usuário**. Seus métodos chamados *actions* são responsáveis por uma página, controlando qual model usar e qual view será mostrado ao usuário.

O **MVC** é um padrão de arquitetura de software, separando a aplicação em três camadas. A camada de interação com o usuário (*View*), a camada de manipulação dos dados (*Model*) e a camada de controle (*Control*).

**Referência do módulo Portal**

BEZERRA, Eduardo. Princípios de análise e projeto de sistemas com UML/Eduardo Bezerra. – Rio de Janeiro: Campus, 2002

CHADWICK, Jess; Synider Todd; Panda, Hrusikesh. Programming ASP.NET MVC 4. First Edition – O’Reilly Media, Inc., 2012

CILIBERTI, John. ASP.NET MVC 4 Recipes. A problem-Solution Approach. A Pratical Guide for Developers Creating Modern Web Applications on the Microsoft Plataform. Apress, 2013

EVANS, Eric. Domain-Driven Design; Atacando as Complexidades no Coração do Software; 2ª Edição Revisada; Editora ALTA BOOKS; Rio de Janeiro, 2010

GALLOWAY, Jon; Haack, Phil; Wilson, Brad; Allen, K. Scott. Professional ASP.NET MVC 4. Published by Wiley & Sons, Inc. Indianapolis - Indiana, 2012

KRUCHTEN, Philippe. Intrudção ao **RUP** – Rational Unified Process. Rio de Janeiro: Editora Ciência Moderna Ltda., 2003

PALERMO, Jeffrey; Bogard J.; Hexter Eric; Hinze M; Skinner J. ASP.NET MVC4 in Action. A revised edition of ASP.NET MVC 4 in Action – MANNING Shelter Island, 2012

PRESSMAN, R. S. Software engineering: a practitioner’s approach. 5 ed. New York: McGraw-Hill, 2001

SOMMERVILLE, Ian. Engebharia de Software / Ian Sommerville; tradução André Maurício de Andrade Ribeiro; revisão técnica Kechi Hirama. – São Paulo: Addison Wesley, 2007.

WAZLAWICK, Raul Sidnei. Análise e projeto de sistemas de informações orientados a objetos / Rual Sidnei Wazlawick. – Rio de Janeiro: Elsevier, 2004

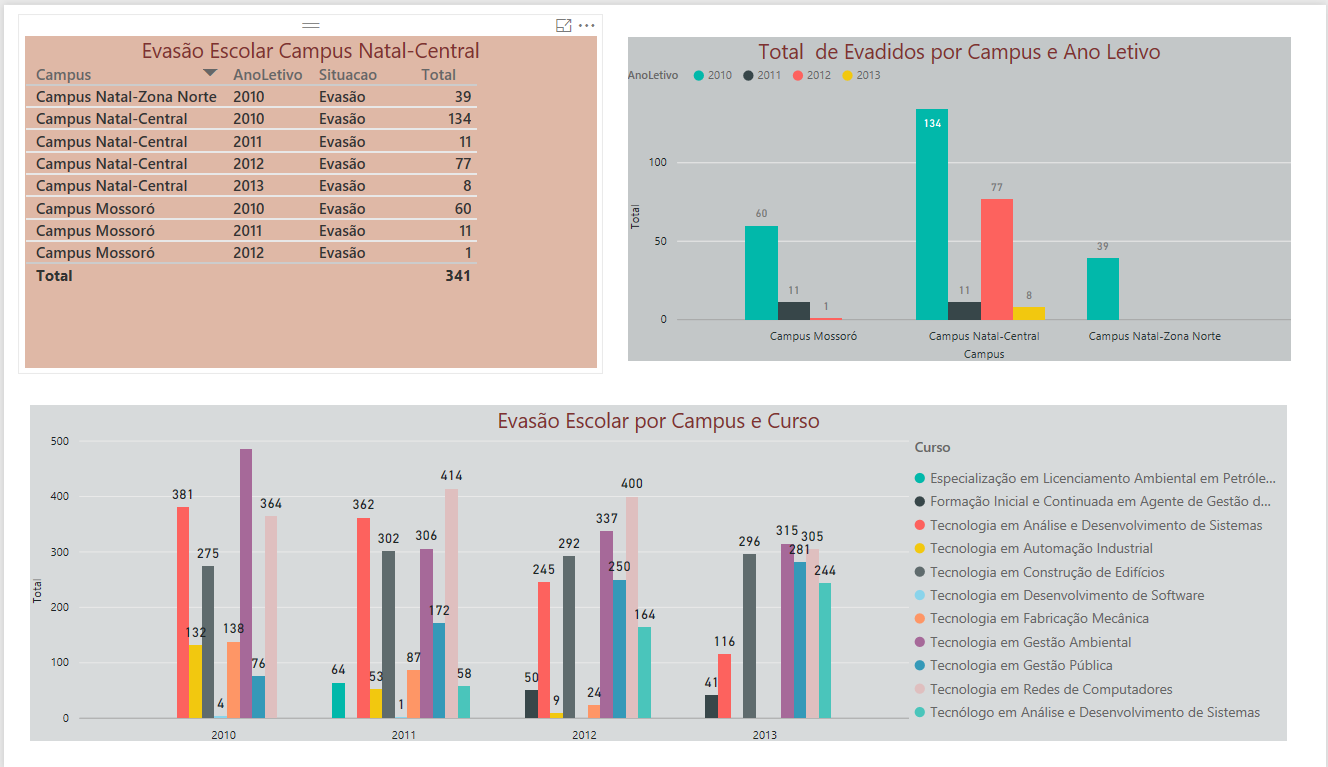
**Capítulo 4 – Resultados obtidos**

4.1. **Comparando os resultados obtidos com outros dados**

A Figura 4.1 mostra um deshdoard que exibe dados sobre a evasão escolar no campus Natal-Central, campus Mossotó e campus Natal-Zona Norte. A tabela, no lado esquerdo, mostra a evasão escolar no campus para os anos de 2010 à 2013. O gráfico do lado direito da tabela, mostra as situações aprovado, Cancelado, Evasão e Reprovado, em cada campus e o gráfico na parte inferior do dashboard, mostra a evasão por curso, para os anos de 2010 à 2013. Observe que alguns cursos aparecem mais de uma vez para o mesmo campus. Isto acontece porque, um curso pode mudar a sua grade curricular e, o aluno tem a opção de continuar na mesma grade um passar para a grade nova, por questões legais. Dessa forma, tem aluno que prefere permanecer na grande que iniciou e, então, fica o mesmo curso funcionando com grades curriculares diferentes e, portanto, com indetificação diferente, mas com o mesmo nome. Nesse dashdoard os dados podem ser filtrados por qualquer campo que se deseje. No estudo de caso apresentado, os dados estão filtrados por campus Natal Central, cursos de tecnologias, situação e para os anos de 2010 até 2013.

Observando a Figura 4.1, percebe-se que em 2010 o curso de Tecnologia em Construção de Edifícios, apresenta um alto índice de evasão escolar. E o curso de Tecnologia em Redes de Computadores, praticamente mantém um índice de evasão elevado para todos os anos apresentados no dashboard. O curso de Tecnologia em Análise e Desenvolvimento de Software, por outro lado, teve o índice de evasão alto em 2010, mais diminui em 2011 em diante. Observa-se também que alguns cursos apresentam baixo índice de evasão, como é o caso dos cursos: Especialização em Lincenciatura Ambiental e Petróleo e Gás, Formação Inicial e Continuada em Agente de Gestão de Residuos Sólidos, Tecnologia em Automação Industrial e Tecnologia em Fabricação Mecânica.

Os dados apresentados na Figura 4.1 foram filtrados para exibir apenas os campus onde se estar concentrado a maior quantidade de alunos, pois esses campis são os mais antigos, campus Natal Central, a primeira sede construída, o campus Mossoró, o segundo campus a entrar em funcionamento e o campus Natal-Zona Norte, o terceiro campus a entrar em funcionamento de toda rede federal do Rio Grande do Norte.



**Figura 4.1** Bashboard para Evasão Escolar com base nos dados do Boletim Escolar.

**Fonte**: Autor.

Segundo o **PNUD** (Programa das Nações Unidas para o Desenvolvimento), o Brasil, com uma taxa de 24,3%, tem a 3ª maior taxa de evasão escolar entre 100 países com maior **IDH** (Índice de Desenvolvimento Humano), só atrás da Bosnia Herzegovina (26,8%) e das ilhas de São Cristovam e Névis, no Caribe (26,5%).

Na América Latina, só Gautemala (35,2%) e Nicarágua (51,6%) tem taxa de evasão superiores.

Anida segndo o **PNUD**, algumas das principais causas da “evasão escolar” no Brasil são a pobreza, a dificuldade de acesso a escola, a necessidade de trabalho e, principalmente, o desinteresse pelos estudos.

A Tabela 4.1 mostra dados estatísticos do relatório **PNUD**, com o ranking de um alguns países, em relação a taxa de evasão escolar e outros dados. Onde se pode observar que o Brasil apresenta uma taxa de evasão escolar muito superior a outros países, inclusive, países da América do Sul e Latina. A taxa de evasão escolar para o Brasil, explanada na Tabela 4.1, é de 24,3%, enquanto Chile, Uruguai e México, por exemplo, apresentavam em 2013, taxas de evasão escolar, bem menores do que o Brasil.

**Tabela 4.1** Dados relativos à Educação no Relatório do **PNUD**.

**Fonte**: PNUD, 2013

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| País | Ranking | IDH | População Alfabetizada | Ensino médio completo | Taxa de evasão |
| Noruega | 1 | 0,955 | 1 | 0,952 | 0,5% |
| Austrália | 2 | 0,938 | 1 | 0,922 | - |
| Estados Unidos | 3 | 0,937 | 1 | 0,945 | 6,9% |
| Holanda | 4 | 0,921 | 1 | 0,889 | - |
| Alemanha | 5 | 0,92 | 1 | 0,965 | 4,4% |
| Chile | 40 | 0,819 | 0,986 | 0,74 | 2,6% |
| Argentina | 45 | 0,811 | 0,978 | 0,56 | 6,2% |
| Uruguai | 51 | 0,792 | 0,981 | 0,498 | 4,8% |
| México | 61 | 0,775 | 0,931 | 0,539 | 6,0% |
| Brasil | 85 | 0,73 | 0,903 | 0,495 | 24,3% |

Na Europa foi desenvolvido, no período de dezembro de 2011 à novembro de 2013, um estudo sobre o abandono escolar precose (Em inglês *Early School Leaving* – **ESL**). Esta pesquisa foi desenvolvida pelo “Grupo de Trabalho Temático” e, teve como resultado um relatório sobre "abandono escolar precoce". Este grupo de trabalho incluía peritos nomeados pelos 31 países europeus e organizações de partes interessadas e, foi assistido pelos consultores da Comissão, Anne-Marie Hall e Ms Ilona Murphy[ <http://ec.europa.eu/education/schooleducation/leaving_en.htm>, 2013].

Segundo o relatório do Grupo de Trabalho Temático, o abandono escolar precoce (**ESL**) é um problema complexo e multifacetado, causada por um processo cumulativo de contingências. É um processo pessoal, social, econômico, educacional, e razões relacionadas a família. A escola tem um papel importante na abordagem do abandono escolar, no entanto, ela não deve trabalhar isoladamente, na função de diminuir esse índice. Ainda, segundo o relatório gerado pelo Grupo de Trabalhos Temático, existe uma política entre estes países Europeus para diminuir o índice de Abandono Escolar Precoce e, com metas a serem atingidas até 2020. A Tabela 4.2 mostra um quadro do abandono escolar precoce, gerado pelo Grupo de Trabalho Temático, para países da União Europeia (**EU**).

**Tabela 4.2 Taxa de Abandono Escolar Precoce na Europa**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 2009 | 2012 | | | 2020 |
| Total | Masculino | Feminino | Média | Meta |
| EU | 14.3 | 14.5 | 11.0 | 12.7 | < 10.0 |
| Bélgica | 11.1 | 14.4 | 9.5 | 12.0 | 9.5 |
| Bulgária | 14.7 | 12.1 | 13.0 | 12.5 | 11.0 |
| República Checa | 5.4 | 6.1 | 4.9 | 5.5 | 5.5 |
| Dinamarca | 11.3 | 10.8 | 7.4 | 9.1 | <10.0 |
| Alemanha | 11.1 | 11.1p | 9.8p | 10.5p | <10.0 |
| Estônia | 13.9 | 14.0 | 7.1 | 10.5 | 9.5 |
| Irlanda | 11.6 | 11.2 | 8.2 | 9.7 | 8.0 |
| Grécia | 14.5 | 13.7 | 9.1 | 11.4 | 9.7 |
| Espanha | 31.2 | 28.8 | 20.8 | 24.9 | 15.0 |
| França | 12.2 | 13.4 | 9.8 | 11.6 | 9.5 |
| Croácia | 3.9 | (4.6) | (3.6) | 4.2 | 4.0 |
| Itália | 19.2 | 20.5 | 14.5 | 17.6 | 15.0 |
| Chipre | 11.7 | 16.5 | 7.0 | 11.4 | 10.0 |
| Letônia | 13.9 | 14.5 | 6.2 | 10.5 | 13.4 |
| Lituânia | 8.7 | 8.2 | (4.6) | 6.5 | <9.0 |
| Luxemburgo | 7.7b | 10.7p | 5.5p | 8.1p | <10.0 |
| Hungria | 11.2 | 12.2 | 10.7 | 11.5 | 10.0 |
| Malta | 27.1n | 27.5 | 17.6 | 22.6 | - |
| Países Baixos | 10.9 | 10.2p | 7.3p | 8.8p | <8.0 |
| Áustria | 8.7 | 7.9 | 7.3 | 7.6 | 9.5 |
| Polônia | 5.3 | 7.8p | 3.5p | 5.7p | 4.5 |
| Portugal | 31.2 | 27.1 | 14.3 | 20.8 | 10.0 |
| Romênia | 16.6 | 18.0 | 16.7 | 17.4 | 11.3 |
| Eslovênia | 5.3 | 5.4 | (3.2) | 4.4 | 5.0 |
| Eslováquia | 4.9 | 6.0 | 4.6 | 5.3 | 6.0 |
| Finlândia | 9.9 | 9.8 | 8.1 | 8.9 | 8.0 |
| Suécia | 7.0 | 8.5 | 6.3 | 7.5 | <10.0 |
| Reino Unido | 15.7 | 14.6 | 12.4 | 13.5 | - |
| Montenegro | : | : | : | : | - |
| Islândia | 21.3 | 23.6 | 16.5 | 20.1 | - |
| MK\* | 16.2 | 11.1 | 12.3 | 11.7 | - |
| Sérvia | : | : | : | : | - |
| Turquia | 44.3 | 36.1 | 43.0 | 39.6 | - |
| Noruega | 17.6 | 17.6 | 11.9 | 14.8 | - |
| Suíça | 9.1d | 5.7 | 5.3 | 5.5 | - |

**Fonte**: Eurostat (LFS). Intermediate breaks in time series for NL (2010) and LV (2011). Notes: "b" = break in time series; "p" = provisional; "()" = Data lack reliability due to small sample size; ":" = data either not available or not reliable due to very small sample size; “n” = national data. \*MK: The former Yugoslav Republic of Macedonia.

Os dados mostrados sobre o abandono escolar na União Europeia nesse trabalho, tem como finalidade, fazer um comparativo entre o abandono escolar na EU e a evasão escolar no Brasil e na Rede Federal de Ensino.

Como já foi mostrado através do relatório do PNUD de 2013, o Brasil tem uma taxa de evasão escolar, em torno de 24,3% que, é muito superior as apresentadas por Chile, Uruguai e México, por exemplo. Este percentual de 24,3% inclui a evasão escolar em todos os níveis da educação no Brasil. No entanto, nessa pesquisa será feito um estudo sobre a evasão escolar no IFRN, com base nos dados do sistema acadêmico. Portanto, a Tabela 4.3 mostra um resumo da evasão escolar no IFRN por curso, no período de 2010 à 2015. Esses dados foram extraídos do sistema acadêmico, apenas para o campus Natal-Central (CNAT). Cumpus CNAT concentra o maior número de estudantes, em torno de 5 mil alunos.

**Tabela** 4**.3** mostra um resumo da evasão escolar no IFRN por curso, no período de 2010 à 2015

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Curso | 2010 | 2011 | 2012 | 2013 | 2014 | 2015 | Média |
| 01025 | 0 | 0 | 34,21 | 15,58 | 22,10 | 13,05 | 36,99 |
| 01104 | 0 | 0 | 43,53 | 34,93 | 35,00 | 19,83 | 33,32 |
| 01404 | 6,25 | 40,43 | 43,51 | 33,86 | 37,05 | 21,29 | 30,40 |
| 01022 | 0 | 0 | 44,59 | 30,46 | 25,96 | 19,54 | 30,14 |
| 01436 | 0 | 55,56 | 31,25 | 28,21 | 22,27 | 10,75 | 29,61 |
| 01045 | 0 | 0 | 39,51 | 39,51 | 23,86 | 14,83 | 29,43 |
| 01102 | 0 | 33,33 | 33,93 | 27,43 | 31,28 | 20,75 | 29,34 |
| 01201 | 40,0 | 40,0 | 27,27 | 24,44 | 27,48 | 0 | 26,53 |
| 01435X | 23,68 | 39,73 | 14,29 | 0 | 0 | 0 | 25,90 |
| 405X | 30,00 | 45,23 | 29,84 | 31,43 | 17,54 | 0 | 25,67 |
| 104 | 0 | 47,37 | 20,69 | 4,35 | 0 | 0 | 24,14 |
| 502 | 24,81 | 30,60 | 39,45 | 23,33 | 9,52 | 13,33 | 23,51 |
| 01112 | 0 | 62,50 | 30,61 | 15,91 | 27,42 | 0 | 22,74 |
| 01435 | 0 | 0 | 15,79 | 28,99 | 28,57 | 16,17 | 22,38 |
| 01502 | 0 | 0 | 18,00 | 23,66 | 24,51 | 22,02 | 22,05 |
| 01405 | 10,00 | 5,71 | 26,02 | 30,71 | 39,61 | 18,80 | 21,81 |
| 21 | 29,39 | 31,34 | 17,72 | 7,89 | 0 | 0 | 21,59 |
| 01057 | 0 | 33,33 | 23,08 | 19,31 | 17,63 | 12,19 | 21,11 |
| 01032 | 0 | 0 | 39,19 | 14,81 | 23,64 | 1,64 | 19,82 |
| 404X | 29,46 | 31,97 | 28,73 | 21,24 | 2,90 | 0 | 19,05 |
| 01051 | 0 | 7,69 | 24,72 | 23,36 | 26,94 | 10,75 | 18,69 |
| 204 | 18,69 | 31,67 | 2,94 | 0 | 0 | 0 | 17,77 |
| 01434 | 0 | 0 | 7,89 | 28,21 | 22,07 | 10,90 | 17,27 |
| 01433 | 0 | 33,33 | 8,70 | 24,36 | 19,83 | 0 | 17,24 |
| 01436X | 29,55 | 24,35 | 38,46 | 8,62 | 2,27 | 0 | 17,21 |
| 201X | 15,42 | 24,12 | 21,50 | 15,11 | 6,82 | 0 | 16,59 |
| 22 | 20,18 | 25,32 | 19,30 | 8,70 | 4,08 | 0 | 15,52 |
| 01304 | 0 | 0 | 27,47 | 20,00 | 22,61 | 7,44 | 15,50 |
| 25 | 21,09 | 28,88 | 12,11 | 6,90 | 7,14 | 0 | 15,22 |
| 57 | 19,54 | 24,71 | 9,80 | 2,08 | 0 | 0 | 14,03 |
| 01301 | 0 | 25,00 | 10,98 | 14,38 | 13,35 | 0 | 12,74 |
| 01434X | 18,92 | 17,91 | 14,89 | 6,67 | 4,0 | 0 | 12,48 |
| 202 | 17,91 | 18,52 | 8,98 | 9,35 | 7,14 | 0 | 12,38 |
| 103 | 17,16 | 22,11 | 13,03 | 8,09 | 0,90 | 0 | 12,26 |
| 01202 | 0 | 0 | 11,11 | 23,08 | 14,78 | 0 | 12,24 |
| 32 | 13,36 | 18,92 | 2,96 | 0 | 0 | 0 | 11,75 |
| 01432 | 0 | 0 | 15,91 | 9,33 | 18,42 | 0 | 10,92 |
| 01433X | 6,67 | 11,39 | 12,66 | 11,43 | 0 | 0 | 10,54 |
| 401X | 8,92 | 16,76 | 11,62 | 8,97 | 5,98 | 0 | 10,45 |
| 102X | 13,74 | 21,05 | 4,38 | 2,56 | 2,22 | 0 | 10,43 |
| 01101 | 0 | 0 | 9,28 | 10,50 | 10,37 | 10,0 | 9,93 |
| 304X | 16,80 | 12,35 | 8,72 | 8,18 | 4,59 | 0 | 8,44 |
| 101X | 5,87 | 9,94 | 6,10 | 4,04 | 11,68 | 0 | 7,53 |
| 01432X | 2,44 | 10,53 | 10,81 | 4,48 | 0 | 0 | 7,07 |
| 301X | 9,86 | 8,71 | 7,24 | 2,42 | 3,57 | 0 | 7,06 |
| 01111 | 0 | 0 | 5,36 | 8,33 | 12,50 | 6,67 | 5,48 |
| 302 | 8,73 | 3,95 | 2,31 | 2,90 | 0 | 0 | 4,47 |
| 01110 | 0 | 0 | 2,33 | 2,47 | 4,17 | 5,16 | 2,36 |

Calculando a média geral (Coluna média) obtem-se o resultado de 17,69% (devo lembrar que é apenas para os anos de 2010 à 2015) para a média geral de todos os cursos da Tabela 3.22 do campus Natal-Central (CNAT). A taxa de evasão escolar do campus Natal-Central do IFRN de 17,69% é menor do que a de alguns países da União Europeia (**EU**), e inferior da taxa da evasão escolar do Brasil, segundo o relatório **PNUD**. No entanto, ainda é alta e precisa de ser trabalhada no intuito de sua diminuição. Observa-se também na Tabela 4.3 que, a maioria dos cursos têm taxa de evasão escolar acima de 10%e, enquanto, apenas quatro tem taxa de evasão escolar abaixo de 10%. Alguns valores atribuídos a evasão escolar, na Tabela 3.22 tem valor zero (0) para alguns anos. Isto acontece porque aquele curso não existe naquele ano. Por exemplo, o curso 01025 não existia em 2010 e 2011, ou seja, ele só foi implantado a partir de 2012.

A Tabela 4.4 faz uma descriminação de cada curso listado na Tabela 4.3. Esses cursos são todos do campus Natal-Central.

**Tabela 4.4** Relação de Cursos e suas siglas do campus Natal-Central (**CNAT**).

|  |  |
| --- | --- |
| Sigla do curso | Descrição |
| 01025 | Técnico de Nível Médio em Petróleo e Gás, na forma Subsequente. |
| 01104 | Técnico de Nível Médio em Estradas, na forma Subsequente. |
| 01404 | Tecnologia em Análise e Desenvolvimento de Sistemas. |
| 01022 | Técnico de Nível Médio em Mecânica, na forma Subsequente. |
| 01436 | Tecnologia em Gestão Pública. |
| 01045 | Técnico de Nível Médio em Eletrotécnica, na forma Subsequente. |
| 01102 | Tecnologia em Construção de Edifícios. |
| 01201 | Técnico de Nível Médio em Eletrotécnica, na Forma Integrado. |
| 01435X | Técnico em Geologia Subsequente. |
| 405X | Tecnologia em Redes de Computadores. |
| 104 | Técnico em Estradas Subsequente. |
| 502 | Tecnologia em Comércio Exterior. |
| 01112 | Técnico de Nível Médio em Manutenção e Suporte em Informática, na Forma Integrado. |
| 01435 | Técnico de Nível Médio em Geologia, na forma Subsequente. |
| 01502 | Tecnologia em Comércio Exterior. |
| 01405 | Tecnologia em Redes de Computadores. |
| 21 | Técnico em Eletrotécnica Subsequente. |
| 01057 | Técnico de Nível Médio em Segurança do Trabalho, na forma Subsequente. |
| 01032 | Técnico de Nível Médio em Controle Ambiental, na Forma Subsequente. |
| 404X | Tecnologia em Análise e Desenvolvimento de Sistemas. |
| 01051 | Técnico de Nível Médio em Edificações, na Forma Subsequente. |
| 204 | Tecnologia em Fabricação Mecânica. |
| 01434 | Técnico de Nível Médio em Mineração, na forma Subsequente. |
| 01433 | Técnico de Nível Médio em Geologia, na Forma Integrado. |
| 01436X | Tecnologia em Gestão Pública. |
| 201X | Técnico em Eletrotécnica Integrado. |
| 22 | Técnico em Mecânica Subsequente. |
| 01304 | Tecnologia em Gestão Ambiental. |
| 25 | Técnico em Petróleo e Gás Subsequente. |
| 57 | Técnico em Segurança do Trabalho Subsequente. |
| 01301 | Técnico de Nível Médio em Controle Ambiental, na Forma Integrado. |
| 01434X | Técnico em Mineração Subsequente. |
| 202 | Técnico em Mecânica Integrado. |
| 103 | Técnico em Edificações Subsequente. |
| 01202 | Técnico de Nível Médio em Mecânica, na Forma Integrado. |
| 32 | Técnico em Controle Ambiental Subsequente. |
| 01432 | Técnico de Nível Médio em Mineração, na Forma Integrado. |
| 01433X | Técnico em Geologia Integrado. |
| 401X | Técnico em Informática Integrado. |
| 102X | Tecnologia em Construção de Edifícios. |
| 01101 | Técnico de Nível Médio em Edificações, na Forma Integr. |
| 304X | Tecnologia em Gestão Ambiental. |
| 101X | Técnico em Edificações Integrado. |
| 01432X | Técnico em Mineração Integrado. |
| 301X | Técnico em Controle Ambiental Integrado. |
| 01111 | Técnico de Nível Médio em Informática para Internet, na Forma Integrado. |
| 302 | Técnico em Geologia e Mineração Integrado. |
| 01110 | Técnico de Nível Médio em Administração, na Forma Integrado. |

A Tabela 4.5 mostra alguns indicadores de desempenho de todos os campi, no período de 2005 a 2015. Estes dados foram extraídos do sistema consolidado, ou seja, na base de dados multidimensional, montada para esse trabalho.

**Tabela 4.5**. Indicadores de Ensino de todos os campi de 2005 a 2015.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Taxas (%) | | | |
| Anos | Retenção | Conclusão | Evasão | Reprovação |
| 2005 | 25,02 | 9,76 | 13,29 | 16,45 |
| 2006 | 18,48 | 5,15 | 11,89 | 17,64 |
| 2007 | 29,4 | 10,98 | 16,38 | 16,98 |
| 2008 | 23,99 | 9,15 | 14,01 | 15,81 |
| 2009 | 22,17 | 13,42 | 12,7 | 15,81 |
| 2010 | 18,12 | 10,21 | 15,63 | 19,77 |
| 2011 | 18,56 | 3,5 | 16,93 | 19,48 |
| 2012 | 22,31 | 6,06 | 18,55 | 18,35 |
| 2013 | 20,97 | 5,31 | 15,83 | 18,38 |
| 2014 | 25,33 | 7,72 | 15,64 | 18,76 |
| 2015 | 26,19 | 8,14 | 14,90 | 16,83 |

De acordo com os números da Tabela 4.5, a taxa média para retenção, conclusão, evasão e reprovação são de 22,78%, 8,14%, 15,07% e de 16,83% respectivamente. Observe que, a taxa de evasão obtida com os dados da tabela 4.3 foi de 15,07% e que, a taxa obtida com os dados da tabela 4.5 foi de 14,90%. Esta diferença deve-se pelo fato de que, no caso anterior foi considerado apenas os anos de 2010 a 2015, enquanto que, no último caso foi considerado um período maior, de 2005 a 2015.

**Nota**: Retenção (alunos reprovados em três ou mais disciplinas, ficam retidos no módulo que estão cursando).

**Bibliografia Geral**

**ANEXO A -** Script SQL para popular as dimensões e tabelas de fatos Dados Sociais dos Alunos

/\* Script para criar e popular a tabela FatoSituacaoAlunoDadosSociais

A função deste fato é armazenar os totais de alunos:

- Aprovados

- Reprovados

-

- Os dados serão extraídos da base de dados do sistema acadêmico

-

Autor: José Antônio da Cunha

Data: 26/05/2015

Versão: 01

/\* Cria a exibição de dados, que calcula o total de alunos aprovados por curso e ano letivo\*/

CREATE View vAprovadosCursoAnoDS

AS

select distinct Cod\_Curso as Curso, Ano\_Let AS Ano,

Count(Situacao) AS TotalAprovado

from dimDadosSociaisAlunos

where Situacao='Aprovado' AND Ano\_Let between 2010 and 2014

group by Cod\_Curso, Ano\_Let

/\* Cria a exibição de dados, que calcula o total de alunos reprovados por curso e ano letivo\*/

CREATE View vReprovadosCursoAnoDS

AS

select distinct Cod\_Curso as Curso, Ano\_Let as Ano,

Count(Situacao) AS TotalReprovado

from dimDadosSociaisAlunos

where Situacao='Reprovado' AND Ano\_Let between 2010 and 2014

group by Cod\_Curso, Ano\_Let

--Neste ponto, faz-se a população da tabela fato

INSERT INTO FatoSituacaoAlunoCursoAnoDS

(Curso,

Ano,

TotalAprovados,

TotalReprovados,

Data\_Insercao,

Data\_Atualizacao)

select A.Curso,A.Ano,

A.TotalAprovado,

R.TotalReprovado,

GetDate(),

GetDate()

from vAprovadosCursoAnoDS A LEFT JOIN vReprovadosCursoAnoDS R

ON (A.Curso = R.Curso AND A.Ano = R.Ano)

/\* Sexo: Script para popular a tabela FatoAprovadosCursoSexoDS

DS - Significa: Dados Sociais dos alunos (obtidos do sistema acadêmico)

Autor: José Antônio da Cunha

Data: 27/05/2015

Versão: 01

\*/

Create View vAprovadosCursoSexoDS

AS

select Cod\_Curso as Curso,

Sexo,

Ano\_Let AS Ano,

Count(Situacao) AS TotalAprovadoCSexo

from dimDadosSociaisAlunos

where Situacao='Aprovado' AND Ano\_Let between 2010 and 2014

group by Sexo, Cod\_Curso,Ano\_Let

--Reprovados

Create View vReprovadosCursoSexoDS

AS

select Cod\_Curso as Curso,

Sexo,

Ano\_Let AS Ano,

Count(Situacao) AS TotalReprovadoCSexo

from dimDadosSociaisAlunos

where Situacao='Reprovado' OR Situacao='Dispensado'

AND Ano\_Let between 2010 and 2014

group by Sexo, Cod\_Curso,Ano\_Let

INSERT INTO FatoAprReprCursoSexoDS

(Curso,

Sexo,

Ano,

TotalAprovadosCSexo,

TotalReprovadosCSexo,

Data\_Insercao,

Data\_Atualizacao)

select Ap.Curso,

Ap.Sexo,

Ap.Ano,

ap.TotalAprovadoCSexo,

Rep.TotalReprovadoCSexo,

GetDate(),

GetDate()

from vAprovadosCursoSexoDS Ap LEFT JOIN vReprovadosCursoSexoDS Rep

ON (Ap.Sexo = Rep.Sexo AND Ap.Curso = Rep.Curso AND Ap.Ano = Rep.Ano)

Select \* from FatoAprReprCursoSexoDS

/\* Disciplina: Fatos aprovados e reprovados por disciplina\*/

--View Cursando

CREATE View vAprovadosDisciplinaCursoANODS

AS

select Sigla\_Disciplina AS Sigla,Cod\_Curso AS Curso,Ano\_Let AS Ano,

Count(Situacao) AS TotalCursandoDisc

from dimDadosSociaisAlunos

where Situacao='Aprovado'

group by Sigla\_Disciplina, Cod\_Curso,Ano\_Let

/\* Script para popular a tabela FatoAprovadosDisciplinaDS

Agrupado pela sigla da disciplina

Autor: José Antônio da Cunha

Data: 27/05/2015

Versão: 01

\*/

INSERT INTO FatoAprReprDisciplinaDS

(Curso,

Sigla\_Disciplina,

Ano,

TotalAprovadosDiscDS,

TotalReprovadosDiscDS,

Data\_Insercao,

Data\_Atualizacao)

select ap.Curso, ap.Sigla, ap.Ano,

ap.TotalCursandoDisc,

repdis.TotalRepDispDisc,

GetDate(),

GetDate()

from vAprovadosDisciplinaCursoANODS Ap Left JOIN

vDispensadosDisciplinaCursoAnoDS RepDis

ON (Ap.Sigla = RepDis.Silga AND Ap.Curso = RepDis.Curso AND

Ap.Ano = RepDis.Ano)

/\* Etnia: Script para totalizar situação de alunos, quanto aprovação,

Reprovação, cursando e dispensados, em relação dos dados sociais

dos mesmos por curso e Etnia \*/

--Views Aprovados

CREATE View vAprovadosCursoEtniaDS

AS

select Cod\_Curso as Curso, Etnia,Ano\_Let AS Ano,

Count(Situacao) AS TotalAprovadoCE

from dimDadosSociaisAlunos

where Situacao='Aprovado' AND Ano\_Let between 2010 and 2014

group by Etnia, Cod\_Curso,Ano\_Let

--View Reprovados

CREATE View vReprovadosCursoEtniaDS

AS

select Cod\_Curso as Curso, Etnia,Ano\_Let As Ano,

Count(Situacao) AS TotalReprovadoCE

from dimDadosSociaisAlunos

where Situacao='Reprovado' AND Ano\_Let between 2010 and 2014

group by Etnia, Cod\_Curso, Ano\_Let

--View Cursando

CREATE View vCursandoCursoEtniaDS

AS

select Cod\_Curso as Curso, Etnia,Ano\_Let AS Ano,

Count(Situacao) AS TotalCursandoCE

from dimDadosSociaisAlunos

where Situacao='Cursando' AND Ano\_Let between 2010 and 2014

group by Cod\_Curso,Etnia, Ano\_Let

/\* Script para popular a tabela FatoSituacaoAlunoCursoEtniaDS

Autor: José Antônio da Cunha

Data: 27/05/2015

Versão: 01

\*/

INSERT INTO FatoAprRepCursoEtniaDS

(Curso,

Ano,

Etnia,

TotalAprovadoCE,

TotalReprovadoCE,

Data\_Insercao,

Data\_Atualizacao)

select ApCE.Curso, ApCE.Ano, ApCE.Etnia,

TotalAprovadoCE,

TotalReprovadoCE,

getdate(),

getdate()

from vAprovadosCursoEtniaDS ApCE LEFT JOIN vReprovadosCursoEtniaDS RepCE

ON (ApCE.Curso = RepCE.Curso And ApCE.Etnia = RepCE.Etnia

AND ApCE.Ano = RepCE.Ano)

/\* Exibições para totalizar situação de alunos, quanto aprovação e reprovação, em relação dos dados sociais dos mesmos por curso e Escola de Origem \*/

--Views Aprovados

CREATE View vAprovadosCursoEscolaOrigemDS

AS

select Cod\_Curso as Curso, Tipo\_Escola\_Origem,Ano\_Let AS ANO,

Count(Situacao) AS TotalAprovadoCEscOrig

from dimDadosSociaisAlunos

where Situacao='Aprovado'

group by Tipo\_Escola\_Origem, Cod\_Curso, Ano\_Let

--View Reprovados

CREATE View vReprovadosCursoEscolaOrigemDS

AS

select Cod\_Curso as Curso, Tipo\_Escola\_Origem,Ano\_Let AS Ano,

Count(Situacao) AS TotalReprovadoCEscOrig

from dimDadosSociaisAlunos

where Situacao='Reprovado' OR Situacao='Dispensado'

group by Tipo\_Escola\_Origem, Cod\_Curso, Ano\_Let

--View Cursando

CREATE View vCursandoCursoEscolaOrigemDS

AS

select Cod\_Curso as Curso, Tipo\_Escola\_Origem,Ano\_Let AS Ano,

Count(Situacao) AS TotalCursandoCEscOrig

from dimDadosSociaisAlunos

where Situacao='Cursando'

group by Tipo\_Escola\_Origem, Cod\_Curso,Ano\_Let

/\* Script para popular a tabela FatoAprRepCursoEscolaOrigemDS

Autor: José Antônio da Cunha

Data: 27/05/2015

Versão: 01

\*/

INSERT INTO FatoAprRepCursoEscolaOrigemDS

(Curso,

DadosSociais\_ID,

Tipo\_Escola\_Origem,

Data\_ID,

Ano,

TotalAprovadoCEscOrig,

TotalReprovadoCEscOrig,

Data\_Insercao,

Data\_Atualizacao)

select ApCEO.Curso,null AS DadosSociais\_ID ,ApCEO.Tipo\_Escola\_Origem,

null as Data\_ID,

ApCEO.Ano,

TotalAprovadoCEscOrig,

TotalReprovadoCEscOrig,

getdate(),

getdate()

from vAprovadosCursoEscolaOrigemDS ApCEO LEFT JOIN vReprovadosCursoEscolaOrigemDS RepCEO

ON (ApCEO.Curso = RepCEO.Curso And

ApCEO.Tipo\_Escola\_Origem = RepCEO.Tipo\_Escola\_Origem AND

ApCEO.Ano = RepCEO.Ano)

/\* Fato sobre a Renda familiar

Calcular os totais de aprovados e reprovados em função da renda familiar,

com base nos dados sociais dos alunos, obtidos do acadêmico\*/

--Views Aprovados

CREATE View vAprovadosCursoRendaFDS

AS

select Renda\_Familiar AS Renda,cod\_curso AS Curso,Ano\_Let AS Ano,

Count(Situacao) AS TotalAprovadoCRendaF

from dimDadosSociaisAlunos

where Situacao='Aprovado'

group by Renda\_Familiar,Cod\_Curso,Ano\_Let

--View Reprovados

CREATE View vReprovadosCursoRendaFDS

AS

select Renda\_Familiar AS Renda,Cod\_Curso AS Curso,Ano\_Let AS Ano,

Count(Situacao) AS TotalReprovadoCRendaF

from dimDadosSociaisAlunos

where Situacao='Reprovado' OR Situacao='Dispensado'

group by Renda\_Familiar,Cod\_Curso,Ano\_Let

--View Cursando

CREATE View vCursandoCursoRendaFDS

AS

select Renda\_Familiar AS Renda,Cod\_Curso AS Curso,Ano\_Let AS Ano,

Count(Situacao) AS TotalCursandoCRendaF

from dimDadosSociaisAlunos

where Situacao='Cursando'

group by Renda\_Familiar,Cod\_Curso,Ano\_Let

/\* Script para popular a tabela FatoSituacaoAlunoCursoRendaFDS

Autor: José Antônio da Cunha

Data: 27/05/2015

Versão: 01

\*/

INSERT INTO FatoAprRepCursoRendaFDS

(Curso,

Renda\_Familiar,

Ano,

TotalAprovadoCRendaF,

TotalReprovadoCRendaF,

Data\_Insercao,

Data\_Atualizacao)

select ApCRF.Curso, ApCRF.Renda,

ApCRF.Ano,

ApCRF.TotalAprovadoCRendaF,

RepCRF.TotalReprovadoCRendaF,

getdate(),

getdate()

from vAprovadosCursoRendaFDS ApCRF LEFT JOIN vReprovadosCursoRendaFDS RepCRF

ON (ApCRF.Curso = RepCRF.Curso And ApCRF.Renda = RepCRF.Renda

AND ApCRF.Ano = RepCRF.Ano)

--Fim dos scripts Dados Sociais dos alunos - Autor: José Antônio da Cunha

/\* Script para criar e popular a tabela FatoEvaConEntradaSaida

Serão criadas as exibições de dados para realizar as agregações

Será então definida a tabela de Fato e populada com as totalizações

Temos uma granularidade - Curso, Campus e Ano.

Autor: José Antônio da Cunha

Data: 28/05/2015

Versão: 01

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

USE Sistema\_Academico

GO

--Criar exibição de dados para calcular Evadidos

CREATE VIEW vEvadidosFormIngresso

AS

Select Desc\_Instituicao,curso, Forma\_Ingresso, Ano\_Ingresso,

count(situacao\_matricula) AS TotalEvaFormaIngresso

From dimAlunosSituacaoDadosIngresso

Where Situacao\_Matricula='Evasão' OR Situacao\_Matricula='Cancelado'

Group By Forma\_Ingresso, Desc\_Instituicao,curso, Ano\_Ingresso

--Criar exibição de dados para calcular Concluídos

CREATE VIEW vConcluidosFormIngresso

AS

Select Desc\_Instituicao,curso, Forma\_Ingresso, Ano\_Ingresso,

count(situacao\_matricula) AS TotalConFormaIngresso

From dimAlunosSituacaoDadosIngresso

Where Situacao\_Matricula='Concluído' OR Situacao\_Matricula='Concludente'

Group By Forma\_Ingresso,Desc\_Instituicao,curso, Ano\_Ingresso

--Criar exibição de dados para calcular Matriculados

CREATE VIEW vMatriculadosFormIngresso

AS

Select Desc\_Instituicao,curso, Forma\_Ingresso, Ano\_Ingresso,

count(situacao\_matricula) AS TotalMatFormaIngresso

From dimAlunosSituacaoDadosIngresso

Where Situacao\_Matricula='Matriculado'

Group By Forma\_Ingresso, Desc\_Instituicao,curso, Ano\_Ingresso

--Criar e popular a tabela FatoSituacaoPorFormaIngresso

INSERT INTO FatoEvaConPorFormaIngresso (

Campus,

Curso,

Forma\_Ingresso,

Ano\_Ingresso,

TotalEvadidoFIA,

TotalConFIA,

Data\_Insercao,

Data\_Atualizacao

)

Select Eva.Desc\_Instituicao,

Eva.Curso,

eva.Forma\_Ingresso,

eva.Ano\_Ingresso,

Eva.TotalEvaFormaIngresso,

Con.TotalConFormaIngresso,

GetDate(),

GetDate()

From vEvadidosFormIngresso Eva JOIN vConcluidosFormIngresso Con

ON (Eva.Forma\_Ingresso = con.Forma\_Ingresso AND

Eva.Ano\_Ingresso = Con.Ano\_Ingresso) AND

Eva.curso = Con.curso