

UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE – UERN  
FACULDADE DE CIÊNCIAS EXATAS E NATURAIS – FANAT  
DEPARTAMENTO DE INFORMÁTICA – DI

Samuel da Silva Praxedes

**Aplicação do Algoritmo *Association Rules* para Recomendação de Produtos**

MOSSORÓ - RN

2017

Samuel da Silva Praxedes

## **Aplicação do Algoritmo *Association Rules* para Recomendação de Produtos**

Monografia apresentada à Universidade do Estado do Rio Grande do Norte como um dos pré-requisitos para obtenção do grau de bacharel em Ciência da Computação, sob orientação do Prof. D. Sc. Marcelino Pereira dos Santos Silva.

MOSSORÓ - RN

2017

Samuel da Silva Praxedes

**Ficha catalográfica gerada pelo Sistema Integrado de Bibliotecas e Diretoria de Informatização (DINF) - UERN, com os dados fornecidos pelo(a) autor(a)**

P919a PRAXEDES, SAMUEL DA SILVA.

Aplicação do Algoritmo Association Rules para Recomendação de Produtos / SAMUEL DA SILVA PRAXEDES - 2017.

70 p.

Orientador: Marcelino Pereira dos Santos Silva.

Monografia (Graduação) - Universidade do Estado do Rio Grande do Norte, Ciência da Computação, 2017.

1. Mineração de Dados. 2. Regras de Associação. 3. SQL Server. 4. Banco de Dados. 5. Sugestão de Produtos. I. Silva, Marcelino Pereira dos Santos, orient. II. Título.

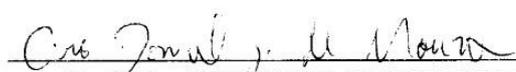
APLICAÇÃO DO ALGORITMO ASSOCIATION RULES PARA RECOMENDAÇÃO DE PRODUTOS

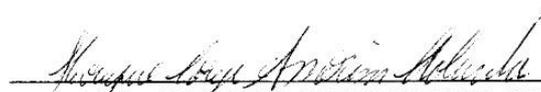
Monografia apresentada como pré-requisito para a obtenção do título de Bacharel em Ciência da Computação da Universidade do Estado do Rio Grande do Norte – UERN, submetida à aprovação da banca examinadora composta pelos seguintes membros:

Aprovada em: 15/05/2017

Banca Examinadora

  
\_\_\_\_\_  
**Prof. Dr. MARCELINO PEREIRA DOS SANTOS SILVA**  
Universidade do Estado do Rio Grande do Norte - UERN

  
\_\_\_\_\_  
**Prof. Me. CIRO DANIEL GURGEL DE MOURA**  
Instituto Federal do Rio Grande do Norte - IFRN

  
\_\_\_\_\_  
**Prof. Dr. HENRIQUE JORGE AMORIM HOLANDA**  
Universidade do Estado do Rio Grande do Norte - UERN

*Dedicatória*

*Aos que fazem parte da minha vida e  
que sempre estiveram ao meu lado nos  
momentos de dificuldades: Meus pais  
Jailson e Josineide, e a minha futura  
esposa, Isadora.*

## AGRADECIMENTOS

Toda honra e toda glória seja dada a aquele por intermédio Dele todas as coisas acontecem, e nada acontece sem sua permissão. Minha gratidão é primeiramente para Deus que até o momento me sustentou e me manteve de pé mesmo quando tudo parecia não favorável. Diante dos problemas de saúde que enfrentei para concluir este trabalho, apenas as misericórdias do meu Senhor puderam me dar a capacidade de saúde física e mental para concluí-lo.

É fato que durante todas as etapas da vida conhecemos diversas pessoas, muitas delas marcam para sempre pelo que fizeram ou significaram. Porém, há pessoas que não surgem por um mero acaso do destino, elas foram pré-determinadas a nos acompanhar até depois de partirem fisicamente. Estas podem ser definidas como insubstituíveis como companheiras inseparáveis, como tutores, como eternos amigos, ou simplesmente como pais. Estes, acima de todas as pessoas, merecem a maior homenagem por quaisquer feitos que porventura eu venha a conquistar.

A estes, por nunca desistirem de me ensinar para vida, por nunca medirem esforços para me oferecer as melhores oportunidades, além do enorme cuidado e carinho que me deram e me dão, ofereço o presente trabalho que representa o resultado destes últimos anos de dedicação, aos meus pais, Jailson Praxedes e Josineide Praxedes.

De maneira única e particular, deixo meus agradecimentos também à pessoa que me proporciona amor e um cuidado especial, minha namorada Isadora. Esta esteve presente ao meu lado todo o tempo, me dando suporte e encorajamentos necessários para concluir este desafio. Para ela, dedico meu amor e companheirismo como gratidão.

Por sua incontestável contribuição, pela disposição de seu tempo e ensino, agradeço ao orientador Marcelino Pereira, que desde o início depositou confiança na minha capacidade de realizar este trabalho com a seriedade e empenho necessários.

Seria difícil citar todos os nomes que de alguma forma contribuíram para que eu chegasse até este momento, mas agradeço principalmente aos professores do Departamento de Informática da UERN, que garantiram os conhecimentos necessários para a minha formação profissional. Somado a estes, incluo em minha gratidão os colegas de sala de aula e amigos para a vida que conheci e os guardo sempre no coração: João Neto, Hitalo Vinícius, André Noberto e Daniel Alves, além de todos os alunos ingressantes na turma do ano de 2013.

*“A noite é mais sombria um pouco  
antes de amanhecer”.*

*Batman - O Cavaleiro das Trevas.*

## RESUMO

No mercado das empresas que atuam no varejo, a concorrência é um fator altamente importante para determinar a expansão ou o encolhimento de uma companhia. Para manter-se sempre em nível competitivo, estas adotam estratégias e principalmente sistemas de *software* que garantam a organização dos processos do negócio, além do próprio atendimento ao cliente. Estes *softwares* são chamados de *Enterprise Resource Planning* (ERP), que utilizam um sistema de gerenciamento de banco de dados (SGBD) para armazenar seus dados. Sabendo da grande quantidade de informações que uma empresa varejista precisa armazenar, principalmente a respeito de seus produtos e clientes, é fundamental aproveitar o potencial destas bases de dados. Uma possibilidade é a utilização de técnicas de mineração de dados para prover melhorias em um ERP, ideia esta que será aplicada no presente trabalho. Por meio deste tipo de solução é possível melhorar rotinas de vendas, permitindo que através de um algoritmo de aprendizagem de máquina sejam encontradas regras de associação extraídas diretamente do banco de dados para recomendação de produtos. Nesta linha, o aumento e dinamização das vendas será natural e eficiente, economizando tempo e ampliando o faturamento, além de aumentar a satisfação do cliente.

**Palavras-chave:** Sistemas ERP, SGBD, Mineração de Dados, Aprendizagem de Máquina.

## ABSTRACT

In the retail business market, competition is a highly important factor in determining the expansion or shrinkage of a company. In order to be always competitive, they adopt strategies and mainly software systems that guarantee the organization of the business processes, as well as the customer service itself. These softwares are called Enterprise Resource Planning (ERP) and rely on a database management system (DBMS) to store your information. Knowing the great amount of information a retailer needs to keep, especially regarding their products and customers, the need to employ other utilities for this data. One of these possibilities is the use of data mining techniques to help improve ERP, an idea that will be applied in the present work. By means of this type of solution, it is possible to improve sales service routines, allowing an association algorithm to be found directly from the database for suggestion of products. With this application, making sales from the sellers becomes simpler and saves time by avoiding the nominal search for each product.

**Keywords:** ERP Systems, DBMS, Data Mining, Machine Learning.

## LISTA DE SIGLAS

TI	Tecnologia da Informação
ERP	<i>Enterprise Resource Planning</i>
CIM	<i>Computer Integrated Manufacturing</i>
CAD	<i>Computer Aided Design</i>
CAM	<i>Computer Aided Manufacturing</i>
MRP	<i>Material Requirement Planning</i>
KDD	<i>Knowledge Discovery in Database</i>
MBA	<i>Market Basket Analysis</i>
SGBD	Sistema de Gerenciamento de Banco de Dados
ANSI	<i>American National Standards Institute</i>
SQL	<i>Structured Query Language</i>
BI	<i>Business Intelligence</i>
DMX	<i>Data Mining Extensions</i>
MAR	<i>Microsoft Association Rules</i>
IDE	<i>Integrated Development Environment</i>
PHP	<i>Personal Home Page</i>
HTML	<i>HiperText Markup Language</i>

## LISTA DE FIGURAS

Figura 1: Estrutura comum de um ERP.....	19
Figura 2: Uma visão geral das etapas que compõem o processo KDD. ....	21
Figura 3: Conhecimentos presentes na etapa de mineração de dados. ....	23
Figura 4: Exemplo prático de associação de produtos.....	26
Figura 5: Regras de associação obtidas de uma aplicação do MAR. ....	33
Figura 6: Exemplo de modelo transação aceita pelo Weka.....	37
Figura 7: Script de criação da base de dados. ....	40
Figura 8: Templates de projetos do Visual Studio.....	41
Figura 9: Conexões disponíveis para criar a fonte de dados. ....	42
Figura 10: Lista de tabelas da fonte de dados.....	43
Figura 11: Diagrama das views usadas para a mineração de dados.....	44
Figura 12: Seleção de tabelas views para estrutura de mineração. ....	45
Figura 13: Especificação dos papéis das colunas.....	46
Figura 14: Visualização das regras no menu rules.....	49
Figura 15: Visualização de itens frequentes.....	50
Figura 16: Produtos associados visualizados na rede de dependências. ....	51
Figura 17: Consulta DMX para visualização de regras. ....	53
Figura 18: Query para previsão de produtos. ....	54
Figura 19: Query de recomendação de produtos. ....	55
Figura 20: Tabela de regras de associação. ....	56
Figura 21: Formulário de busca de produtos.....	56
Figura 22: Pesquisa de produtos pela descrição.....	57
Figura 23: Carrinho de compras com sugestão de produtos.....	58

## LISTA DE TABELAS

Tabela 1: Atributos selecionados e suas descrições.....	44
Tabela 2: Parâmetros usados nos modelos de mineração .....	47
Tabela 3: Tipos de variação de nós. ....	54

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>15</b>
<b>2. REFERENCIAL TEÓRICO.....</b>	<b>17</b>
2.1 Histórico e Evolução do <i>Enterprise Resource Planning</i> (ERP).....	17
2.2 Componentes de um ERP.....	18
2.3 Descoberta de conhecimento em bancos de dados.....	19
2.3.1 Especialistas envolvidos no processo de KDD.....	20
2.3.2 Etapas do processo de KDD .....	21
2.3.3 A influência do KDD para tomada de decisões corporativas.....	22
2.4 Mineração de Dados .....	23
2.4.1 Mineração de dados com Aprendizagem supervisionada .....	24
2.4.1.1 Classificação e Regressão .....	24
2.4.2 Mineração de dados com Aprendizagem não supervisionada.....	24
2.4.2.1 Associação e <i>Clustering</i> .....	24
2.5 Market Basket Analysis .....	25
2.6 A Ferramenta SGBD Microsoft SQL Server.....	28
2.7 Considerações acerca das aplicações e técnicas descritas.....	33
<b>3. TRABALHO PROPOSTO.....</b>	<b>34</b>
3.1 Visão Geral.....	34
3.2 A problemática .....	35
3.3 A base de dados utilizada .....	36
3.4 A escolha das ferramentas.....	36
3.4.1 A ferramenta Weka e suas particularidades.....	37
3.4.2 O mecanismo Analysis Services e o pacote Microsoft Visual Studio .....	38
<b>4. DESENVOLVIMENTO DA SOLUÇÃO .....</b>	<b>39</b>
4.1 Extração dos dados nativos .....	39
4.2 Criando um projeto de mineração de dados no <i>Visual Studio</i> .....	40
4.2.1 Criando uma fonte de dados e selecionando as tabelas.....	41
4.2.2 Criando uma estrutura e um modelo de mineração .....	44
4.3 Visualizando as regras geradas .....	48
4.3.1 Visualizando regras com o Visual Studio .....	48
4.3.2 Visualizando regras por meio de consultas DMX .....	52

4.4 A implementação de uma rotina de venda com recomendação de produtos .....	55
<b>5. CONSIDERAÇÕES FINAIS .....</b>	<b>Erro! Indicador não definido.</b>
REFERÊNCIAS.....	60

## 1. INTRODUÇÃO

No ambiente das organizações faz-se necessário coordenar e manter processos organizados, um desafio para as companhias, sejam elas grandes ou pequenas. Diante dessa necessidade e do frequente avanço tecnológico, ferramentas de Tecnologia da Informação (TI) se firmaram como o meio mais indicado para atender tais demandas dentro e fora das corporações.

No contexto de *software*, existem aqueles que são desenvolvidos especificamente para a gestão de empresas, conhecidos como *Enterprise Resource Planning* (ERP). Porém, como se sabe, estas ferramentas não são desenvolvidas para um cliente específico, mas para vários. Como todo sistema genérico, algumas necessidades específicas de determinadas empresas acabam não sendo atendidas ou não correspondendo satisfatoriamente, gerando demandas por soluções próprias.

Além disso, é natural que em companhias que utilizam sistemas de informação há algum tempo, estas mantenham sob seu domínio um repositório de dados contendo um grande volume de informações. Neste contexto, os sistemas gerenciadores de banco de dados (SGBD), responsáveis por gerenciar tais registros, não exercem outra funcionalidade além de armazenar e recuperar dados acerca das operações diárias da empresa. Essa subutilização motivou a concepção, estudo e aplicação de uma área relativamente nova da Ciência da Computação, conhecida como *Knowledge Discovery in Database* (KDD), que tem como etapa principal a mineração de dados.

Para Han e Kamber (2001), mineração de dados é um processo que necessita de interação entre homens e máquinas, visando a exploração de grandes bancos de dados, com a finalidade de extrair conhecimento através da busca por padrões e relacionamentos entre variáveis. Essa definição contraria o conceito bastante difundido de que sistemas de mineração de dados podem de forma automática encontrar informações valiosas sem um acompanhamento humano.

Uma aplicação muito relevante para o comércio, usando técnicas de mineração de dados, é a recomendação de produtos, ou seja, no momento que o cliente está no

estabelecimento comercial, o sistema sugere outros itens de compra a partir do histórico de vendas dos produtos inicialmente escolhidos.

O presente trabalho tem como objetivo buscar através de mineração de dados, mais especificamente usando regras de associação, padrões em históricos de vendas para a recomendação de produtos. O estudo utilizará o repositório de dados de uma empresa do comércio varejista que atua no mercado de materiais de construção. A proposta visa encontrar resultados relevantes e suficientes para dar suporte à implementação de melhorias do ERP da empresa.

Este trabalho está organizado como apresentado a seguir. No capítulo 2 serão apresentados os principais conceitos e definições teóricas dos sistemas de informação empresariais, seu histórico, as etapas do estudo de KDD, os tipos de técnicas de mineração de dados, os formalismos de regras de associação em análise de cesta de compras, além do *software* e o algoritmo que serão usados. No capítulo 3, será exposta a visão geral do estudo proposto e da problemática encontrada, realizando um aprofundamento do mecanismo de análise usado e a apresentação da plataforma de desenvolvimento da solução. O capítulo 4 irá expor as principais atividades práticas para realização da etapa de mineração dos dados, os métodos de interpretação de resultados, assim como a implementação de um módulo genérico de sugestão de produtos utilizando as regras extraídas no experimento. Este módulo foi desenvolvido para ambiente Web e busca mostrar o potencial das regras de associações mineradas para a proposta de sugestão de produtos. Por último, o capítulo 5 mostra as considerações finais e perspectivas futuras do presente trabalho.

## 2. REFERENCIAL TEÓRICO

### 2.1 Histórico e Evolução do ERP

O'Leary (2000) define um ERP como sendo um pacote de *software* para ambientes de arquitetura cliente-servidor que interliga a maioria dos processos de negócios, processa as transações dentro da organização, usa uma base de dados empresarial e permite o acesso aos dados em tempo real, tudo isso para manter o mais alto nível de competição no mercado e organização setorial e unificada da corporação.

Para se chegar ao conceito de ERP que hoje é conhecido no mercado, foram necessários diversos avanços nos modelos de *software* que eram usados como soluções corporativas para gerenciar as operações dentro de uma empresa. Segundo Alsène (1999) o uso de computadores em empresas data desde o início da década de 60, onde surgiu daí também a ideia de se construir sistemas integrados onde houvesse comunicações entre os diversos setores.

Conforme Bancroft *et al.* (1998) diz, nos primórdios os sistemas de informação eram desenvolvidos de maneira customizada de acordo com a solicitação de um departamento da empresa, porém, devido à visão muitas vezes limitada à responsabilidade operacional, o produto entregue acabava deixando a desejar em vários aspectos, pois cada departamento tinha definido suas prioridades e objetivos conforme suas necessidades particulares, dificultando cada vez mais a interligação coordenada de todos os setores. Além disso, tornava-se cada vez mais complexa a manutenção do sistema e conseqüentemente a permanência dele.

Com a adesão dos modelos integrados de software pelos profissionais de TI da época, surgiram diversos conceitos novos de sistemas voltados para as corporações. Vários autores defendem o conceito de Produção Integrada por Computação (*Computer Integrated Manufacturing - CIM*) como sendo o modelo primário de sistemas integrados, que pode ser definido como um conjunto de vários sistemas computacionais interligados. Podem-se citar como exemplo desses sistemas, que são utilizados hoje, os *softwares* da área da automação industrial CAD (*Computer Aided Design*), CAM (*Computer Aided Manufacturing*) e o MRP (*Material Requirement Planning*) que mais tarde viria a ser o MRP II.

## 2.2 Componentes de um ERP

O *software* ERP, por ter a característica de ser um pacote comercial, não é desenvolvido para um cliente específico. O fornecedor do sistema, durante a sua concepção do projeto tenta corresponder ao maior número de possíveis clientes. Isso é realizado por meio do uso de requisitos genéricos, que são obtidos através do trabalho de experimentação, além da contínua coleta de parâmetros e informações pertinentes aos diversos modelos de negócios presentes no mercado alvo, onde a ferramenta deseja atuar.

Um ERP é composto por uma base de dados centralizada, por diversos módulos e processos:

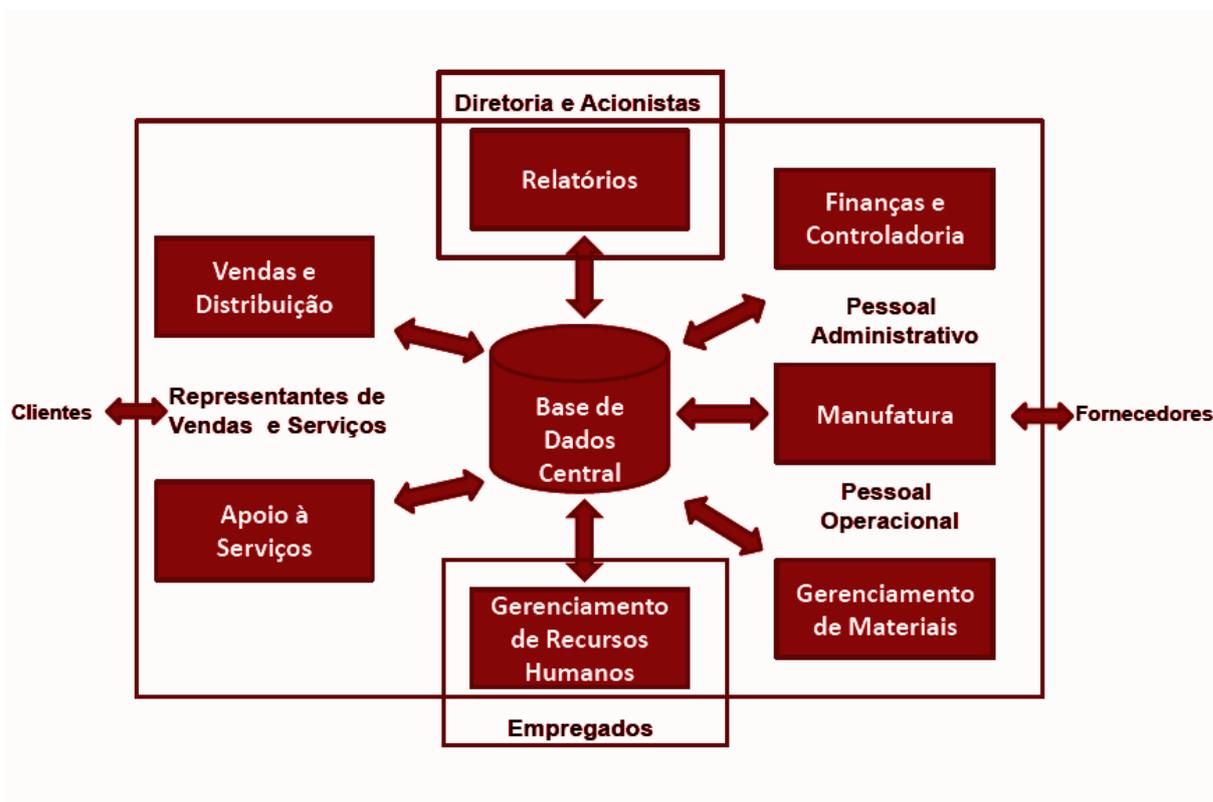
- **Módulos:** Componentes do sistema que armazenam informações específicas de determinado setor da empresa, além de oferecer ferramentas de processamento e recuperação de dados, bem como integração com os outros módulos.

- **Processos:** Durante o desenvolvimento do sistema ERP, os fornecedores das ferramentas são os responsáveis por definir os modelos de processos que serão implementados. Para isso, Davenport (1998) defende que o fornecedor seja o responsável por definir o que de melhor deve ser desenvolvido, defende ainda que *“em alguns casos as suposições do sistema podem ir realmente de encontro aos interesses da empresa”*.

- **Base de Dados:** Onde estão armazenados os dados, podendo admitir diversas modelagens, sendo a mais usada o modelo de banco de dados relacionais. Um banco de dados bem modelado e estruturado é o primeiro e mais importante passo para se ter um *software* ERP de qualidade, tendo em vista que toda a informação é armazenada nele.

A Figura 1 exibe a estrutura comum de um ERP contendo alguns módulos principais, bem como os respectivos setores e pessoas envolvidas nos processos por ele gerido.

Figura 1: Estrutura comum de um ERP.



Fonte: Adaptado de Davenport (1998).

### 2.3 Descoberta de conhecimento em bancos de dados

Na era digital, tornou-se trivial o uso da tecnologia em diversos setores da sociedade, que vai desde o acesso pessoal de um usuário em sua rede social, até o uso de sistemas de monitoramento em áreas médicas. Diante desse cenário, o acúmulo de dados cresce proporcionalmente ao uso constante dos sistemas de informação, superando o poder de interpretação e exame dessas informações coletadas.

Devido à ineficiência dos métodos tradicionais de análise para encontrar informações relevantes nos dados, os repositórios tornam-se subutilizados, tendo apenas a função de armazenamento, haja vista que as informações estratégicas para seus detentores não são acessíveis explicitamente. Essa problemática gera a necessidade de novas ferramentas e técnicas para a análise inteligente e eficiente de banco de dados (FAYYAD et al., 1996).

A técnica que possui a capacidade de obter conhecimento extraído de bancos de dados denomina-se *Knowledge Discovery in Database* (KDD), que segundo Fayyad *et al.* (1996) foi proposta em 1989 para referir-se ao processo de descoberta de conhecimento útil a partir de dados. Neste processo estão presentes várias etapas, das quais a mais conhecida delas é a etapa de mineração de dados, onde é realizada a aplicação de algoritmos específicos para se extrair padrões dos dados.

Outra definição, segundo Berry e Linoff (1997), remete ao termo mineração de dados como sendo a exploração e análise de maneira automática ou semiautomática de um grande contingente de dados, com a finalidade de descobrir padrões e regras.

### **2.3.1 Especialistas envolvidos no processo de KDD**

Antes do aprofundamento no processo em si, faz-se necessário entender os papéis e as aptidões necessárias de cada pessoa envolvida, pois o processo de KDD envolve várias etapas com muitas decisões a serem feitas pelo responsável (ou responsáveis) pela sua condução.

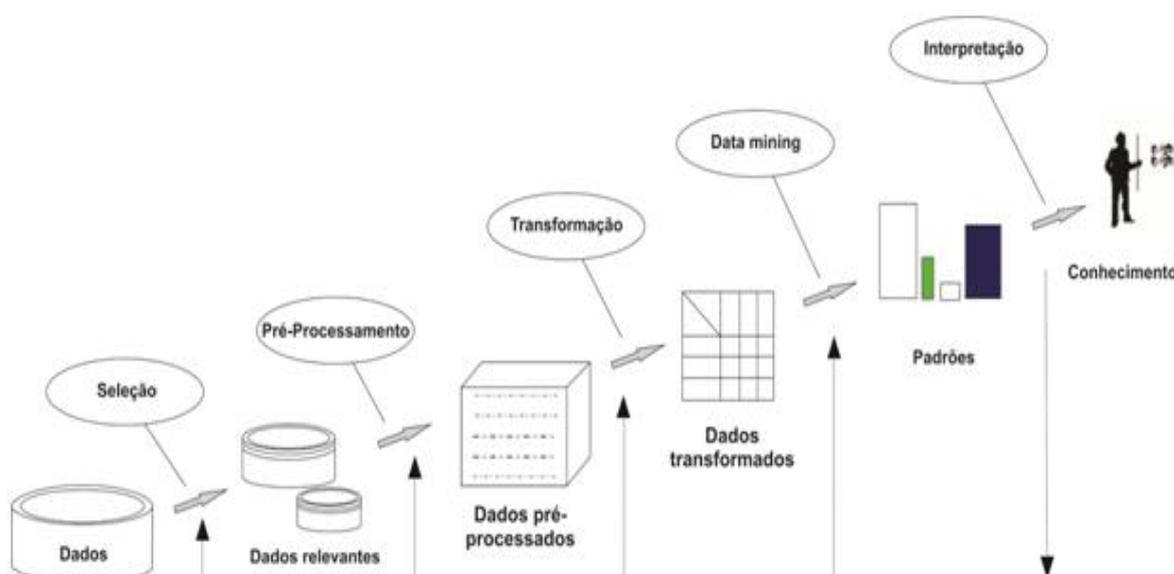
- Analista de Dados: Responsável pelo domínio das ferramentas e dos algoritmos que serão utilizados no processo.
- Especialista de Domínio: Conhece o domínio aos quais os dados pertencem e como são aplicados, a origem deles e sua importância no contexto geral do sistema.
- Usuário: A pessoa que irá utilizar o resultado de todo o processo de KDD para tomada de decisões.

Os papéis e aptidões das pessoas envolvidas em KDD não precisam necessariamente estar direcionados a uma única pessoa, ou ainda que um indivíduo envolvido no projeto deva ter conhecimento sobre outros domínios necessários para a execução do processo.

### 2.3.2 Etapas do processo de KDD

O processo de KDD é o composto por várias etapas que devem ser seguidas para que se obtenha uma resposta satisfatória da análise dos dados, segundo Fayyad *et al.* 1996, conforme apresentado na Figura 2.

Figura 2: Uma visão geral das etapas que compõem o processo KDD.



Fonte 2: Adaptado de Fayyad et al. 1996

- A primeira etapa diz respeito às tarefas de definir e selecionar os dados alvos que possam conter a informação que se deseja descobrir. Para isso é necessário um conhecimento do domínio e as melhorias que podem ser obtidas através das informações extraídas deles.
- A segunda etapa é onde acontece o pré-processamento, que consiste da limpeza dos dados e seleção dos atributos que serão usados. É nessa fase que são aplicadas algumas técnicas para remoção de ruídos e estratégias para preencher campos de dados que porventura estejam ausentes. Este tratamento dos dados é sumariamente importante, pois o objetivo dessa etapa é adequá-los para o uso na ferramenta de mineração (SILVA, 2004).
- A terceira etapa diz respeito à etapa de transformação e formatação dos dados, onde se busca reduzir o número de variáveis a serem consideradas para o

conjunto de dados, com a finalidade de reorganizar estes de maneira específica selecionando características e atributos relevantes para o que se deseja representar e interpretar nas etapas posteriores.

- A quarta etapa é a mineração de dados, onde serão aplicados um ou mais algoritmos, selecionados de acordo com o padrão de conhecimento que se deseja localizar. Essa é a fase mais importante do processo, e para se descobrir os padrões torna-se necessário ajustar os parâmetros nestes algoritmos, de forma que eles se adequem ao objetivo do processo de descoberta.
- Na última etapa ocorre a interpretação de resultados. Segundo Figueira (1998), é o momento em que os padrões encontrados pelo processo de mineração de dados serão interpretados e avaliados para verificar se são válidos em uma possível solução para o problema proposto.

### **2.3.3 A influência do KDD para tomada de decisões corporativas**

Assim como uma pessoa comum precisa tomar decisões diariamente, com as empresas não é diferente. Diante desse comparativo, recaem sobre os gestores das empresas as responsabilidades de saber escolher as decisões corretas e planejá-las antes mesmo de tomá-las, tendo em vista que uma má decisão pode acarretar prejuízos diversos à sua companhia. Aliado à necessidade em questão, está o uso das ferramentas de TI para dar suporte gerencial necessário, mantendo a empresa no patamar da concorrência e apresentando sempre um diferencial no mercado.

Diante desse quadro, Torres (1995) afirma que já algum tempo as corporações têm usado mal as tecnologias da informação, onde um grande contingente de empresas se restringe a usar recursos voltados simplesmente para suas operações diárias dando suporte apenas ao trabalho operacional. Esse cenário, bastante comum nos dias de hoje, torna essas tecnologias subutilizadas, deixando de lado seu potencial de apoio à tomada de decisões.

Nesse contexto, o processo de KDD encontra justificativas práticas para a sua existência e importância, pois como mostram Berry e Linoff, 1997, uma empresa deve tomar suas decisões de negócios com base na aprendizagem adquirida através do relacionamento com seus clientes. Eles definem ainda que para garantir a formação desse aprendizado uma empresa deve ser capaz observar o que seus clientes estão

fazendo atualmente, lembrar o que ela e seus clientes tem feito ao longo do tempo, aprender com o que foi lembrado e agir de acordo com tudo isso para tornar o seu negócio cada vez mais rentável.

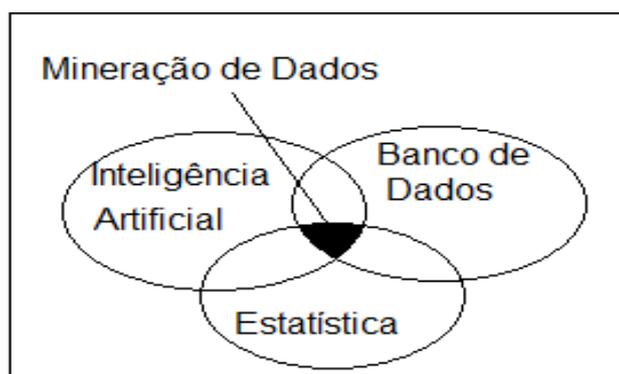
Por estes motivos, as técnicas de KDD se firmam cada vez mais como uma solução para o desafio da análise de grandes volumes de dados. Além disso, é capaz de dar uma utilidade maior aos repositórios de dados, que agora não só armazenam as informações do cotidiano, mas são diretamente responsáveis por deter o conhecimento que pode alavancar a maneira com que as empresas alcançam seus objetivos.

## 2.4 Mineração de Dados

Sendo a etapa mais importante do processo de KDD, a mineração de dados abrange um conjunto de diversas áreas de conhecimento, como inteligência artificial no que tange às técnicas da aprendizagem de máquina, recursos de banco de dados para manipular e consultar bases de dados, além de aplicar conhecimentos de domínio da área da estatística para validar e avaliar resultados (SILVA, 2004).

A Figura 3 ilustra a relação das três áreas de conhecimento que estão presentes na fase de Mineração de Dados.

**Figura 3: Conhecimentos presentes na etapa de Mineração de Dados.**



**Fonte: Adaptado de Silva (2004).**

A etapa de Mineração de dados se caracteriza conforme as tarefas que se desejam resolver. As técnicas dessa etapa podem ser divididas em dois grupos: os algoritmos de aprendizagem supervisionada e de aprendizagem não supervisionada.

### **2.4.1 Mineração de dados com Aprendizagem supervisionada**

Esta nomenclatura se dá pelo fato de que seus algoritmos atribuem uma classe para cada amostra de dados presentes no treinamento. Esse tipo de aprendizagem é dita preditiva, pois a análise tenta fornecer previsões e tendências com base na classificação de cada situação presente nas informações disponíveis nos dados.

#### **2.4.1.1 Classificação e Regressão**

A tarefa de classificação consiste em encontrar uma função para mapear um subconjunto de dados em um conjunto de classes, onde uma vez descoberta, a função encontrada poderá ser aplicada em novos registros com a capacidade de prever a qual classe esses dados pertencem. Esta técnica pode ser usada para prever comportamentos de clientes e tendências do mercado.

A técnica de regressão consiste em buscar uma função que mapeie os registros selecionados de um banco de dados em um intervalo de valores numéricos reais. É similar à técnica de Classificação, diferenciando-se o tipo de dado do atributo alvo que se deseja prever e do método empregado (estatístico). As ferramentas que aplicam esse tipo de algoritmo podem usar diversos meios de representação de conhecimento, podendo-se destacar o uso de árvores, além de modelos matemáticos e redes neurais.

### **2.4.2 Mineração de dados com Aprendizagem Não Supervisionada**

Nesta técnica, não se conhece a rotulação de classes de cada amostra usada no processo, nem se sabe o número de classes a serem processadas pelo algoritmo. Esse tipo de aprendizagem é chamado de não supervisionado justamente pelo fato de que os algoritmos devem descobrir as relações, padrões, regularidades ou categorias dos dados que estão sendo analisados como entrada.

#### **2.4.2.1 Associação e *Clustering***

Associação consiste em identificar combinações de itens que ocorrem com frequência em um banco de dados. O registro do conjunto de dados pode ser chamado de transação. O problema principal da mineração de regras de associação está em encontrar todas as regras que possuam um determinado índice de confiança (ou probabilidade), pois quanto maior esse índice maior a chance de ocorrência de

determinada associação de registros. Mais adiante será aprofundado o estudo sobre a aplicabilidade desta técnica em MBA (*Market Basket Analysis*), as características das regras, bem como quais métodos podem ser usados para validar resultados obtidos de algoritmos de regras de associação.

A técnica conhecida por *Clustering* (ou Agrupamento) pode ser definida como a técnica que consiste em identificar similaridades entre conjuntos de dados, segundo um determinado grau de semelhança entre estes. O objetivo dessa tarefa é maximizar a similaridade *intracluster* e minimizar similaridade *intercluster*. Possui uma grande aplicação no mercado para *Marketing*, podendo agrupar clientes de acordo com seu grupo de interesse e partir disso trabalhar na recomendação ou propaganda de outros produtos.

## **2.5 Market Basket Analysis**

Está técnica, é também conhecida como análise de cesta de compras, e pode ser definida como uma técnica de mineração de dados que utiliza regras de associação para encontrar padrões em hábitos de compras dos clientes de um determinado estabelecimento. O conhecimento extraído desse tipo de análise pode fornecer uma visão aos gestores sobre como melhorar as vendas aumentando o potencial de recomendação de produtos associados.

Como todo o conceito que embasa o MBA está centrado na técnica de mineração de dados que usa regras de associações, faz-se necessário um aprofundamento conceitual dessas regras.

Segundo Gonçalves (2013), a problemática da mineração de dados usando associação foi apresentada pela primeira vez em 1993, onde as regras eram mineradas em bases de dados de transações. As transações podem ser representadas em uma tabela de um banco de dados relacional, onde as linhas podem conter informações acerca de produtos e vendas, cada qual com atributos identificadores. A cesta de compras corresponde aos itens adquiridos por um cliente no processo de compra, ignorando as quantidades e o preço dos produtos.

### 2.5.1 Formalismos de Regras de Associação aplicados no MBA

Conforme Fonseca (2009) e Gonçalves (2013), existe um conjunto de produtos distintos  $P = \{p_1, p_2, p_3, \dots, p_n\}$  e  $DB$  um banco de dados formados por transações ou registros relacionais de compras, onde uma transação  $T$  é composta por um conjunto de produtos tal que  $T \subseteq P$ . Uma regra de associação pode ser expressa como  $A \rightarrow B$  onde  $P \subseteq A$  e  $P \subseteq B$ . Partindo dessas premissas podemos determinar que  $A \neq \emptyset$  e  $B \neq \emptyset$  e portanto, conseqüentemente  $A \cap B \neq \emptyset$ . Diante disso, podemos dizer que em uma base dados a probabilidade de ocorrência do conseqüente ( $B$ ) é maior nos registros que contém ( $A$ ), ou seja, a ocorrência do antecedente ( $A$ ) aumenta a chance de ( $B$ ) aparecer e vice versa. Essa regra permite que tanto o antecedente quanto o conseqüente sejam formados por conjuntos que possam conter um ou mais itens. Um exemplo de regras de associação entre produtos pode ser visto na Figura 4.

**Figura 4: Exemplo prático de Associação de produtos.**



**Fonte: Fracalanza (2009).**

A relevância de uma regra de associação pode ser medida por duas características, a medida de suporte e a medida de confiança. Dada a regra  $A \rightarrow B$  demonstrada anteriormente, a sua medida de suporte ( $Sup$ ) diz respeito à porcentagem de transações ou registros da base de dados que possuem os itens  $A$  e  $B$ , mostrando a relevância dessa associação. Enquanto isso, a medida que tange à confiança ( $Conf$ ) representa, nas transações que contêm os itens de  $A$ , a porcentagem de transações que igualmente possuem os itens de  $B$ , indicando a validação da regra associada. A seguir seguem exemplos dessas duas medidas de interesse conforme visto em Silva (2004).

- **Confiança:**

Temos a regra mostrada na definição acima, a partir dela podemos determinar a seguinte confiança:

$$Conf A \rightarrow B = \frac{T \subseteq A + T \subseteq B}{T \subseteq A}$$

Exemplo: Minerando-se uma base dados de uma empresa do varejo na área de materiais de construção, podemos encontrar uma confiança de 90% (0,90) na regra de associação *compra (cliente, cerâmica porcelanato) → compra (cliente, rejunte)*. Significa dizer que 90% dos clientes que compram o produto cerâmica porcelanato, também compram algum tipo de rejunte.

- **Suporte:**

O suporte de uma regra de associação é dado pela seguinte expressão:

$$Sup A \rightarrow B = \frac{T \subseteq A + T \subseteq B}{Total\ de\ transações}$$

Isso implica dizer que, em um possível suporte de 10% todos os registros ou transações comerciais realizadas em uma base dados, 10% são feitas por clientes que compram produto *A* e também compram produto *B*.

Há ainda outra medida de interesse muito usada para validar resultados de regras de associações, denomina-se *lift*. Em resumo, o *lift* da regra de associação do exemplo anterior  $A \rightarrow B$  indicaria o quanto mais frequente se torna *B*, quando ocorre a aparição de *A*.

- **Lift:**

$$Lift A \rightarrow B = \frac{Conf A \rightarrow B}{Sup A \rightarrow B}$$

Esta medida de interesse aplicando-se hipoteticamente no exemplo da empresa de materiais de construção (*cerâmica porcelanato*) → (*rejunte*), seria esboçado como:

$$Lift A \rightarrow B = \frac{Conf(cerâmica\ porcelanato) \rightarrow (rejunte)}{Sup(rejunte)}$$

Se por exemplo, o resultado dessa razão fosse igual à 3 (três), isto implicaria dizer que os clientes que compram *cerâmica porcelanato* possuem uma probabilidade três vezes maior de comprar *rejunte*.

## 2.6 A Ferramenta SGBD Microsoft SQL Server

Também conhecido apenas como *SQL Server*, a ferramenta é um Sistema Gerenciador de Banco de Dados relacional desenvolvido pela Microsoft. Sua principal função é armazenar e recuperar dados solicitados por outros *softwares*, oferecendo suporte de comunicação com redes locais e a Internet.

Segundo Aguiar (2008), desde sua criação em 1988, onde foi desenvolvida inicialmente em parceria com a *Sybase*, a ferramenta ganhou várias versões trazendo diversas novas atualizações em cada nova edição. A parceria durou apenas até o lançamento da versão para o Windows NT em 1994, onde a partir daí apenas a Microsoft passou a manter a manutenção da ferramenta.

A ferramenta possui sua linguagem de consulta primária no padrão definido pelo *American National Standards Institute* (ANSI) SQL, além de oferecer uma extensão própria deste padrão, conhecida como *Transact-SQL*.

### 2.6.1 Analysis Services e seus recursos de Mineração de Dados

O *Analysis Services* é um mecanismo de dados analíticos da Microsoft usado em soluções de apoio à decisão de negócios e *Business Intelligence* (BI). Além disso, é capaz de organizar os dados utilizando o conceito multidimensional para permitir a geração de relatórios sofisticados e consultas complexas para dar suporte à análise de grandes volumes de informações, bem como em diversas dimensões.

Conforme visto em *Analysis Services* (2016), a metodologia de trabalho deste mecanismo inclui também criar um modelo de dados da forma tabular, além da já

citada forma multidimensional. Os modelos criados podem ser alimentados com dados e metadados externos, geralmente de *data warehouses* ou banco de dados relacionais, hospedados em instâncias de um SGBD SQL Server. Quando configurado por completo o modelo, é possível ainda atribuir permissões de acesso aos usuários finais por meio de qualquer aplicativo cliente que possua suporte ao *Analysis Services* como fonte de dados.

Dentro do contexto de modelo de soluções, o mecanismo oferece uma plataforma integrada capaz de executar tarefas de mineração de dados. Nesta plataforma, é possível criar soluções de BI usando dados relacionais ou multidimensionais para a obtenção de análises preditivas de negócio.

De acordo com o portal para desenvolvedores da Microsoft em Mineração de Dados (2016), são vários os recursos disponíveis para as soluções de mineração de dados, alguns deles são:

- Diversas fontes de dados. Não é necessário criar um *data warehouse*, para fazer mineração de dados, pois o mecanismo permite usar dados de tabelas externas, de planilhas, ou ainda de arquivos de texto.
- Vários algoritmos personalizáveis. Estão disponíveis algoritmos de *clustering*, rede neurais, árvores de decisão, regras de associação e regressão. É possível também desenvolver algoritmos personalizados através de plug-ins.
- Infraestrutura de testes de modelo. A plataforma oferece a possibilidade de testar modelos e conjuntos de dados usando ferramentas estatísticas como validação cruzada, matrizes de classificação, gráficos de comparação de precisão e dispersões.
- Podem ser criadas consultas de previsões, recuperação de padrões modelos e estatísticas.
- Gerar *scripts* através do suporte de linguagens capazes de criar objetos programáveis, como a linguagem *Data Mining Extensions (DMX)*.

Além de todas estas possibilidades, a ferramenta *Analysis Services* da Microsoft conta com um acesso fácil de sua documentação, com a disponibilização de

artigos e fóruns específicos para cada tarefa, além de versões gratuitas de seus mecanismos para estudantes e profissionais que pretendem se aprofundar neles.

### **2.6.2 O Algoritmo *Microsoft Association Rules (MAR)***

Como descrito na seção anterior, o mecanismo *Analysis Services* disponibiliza um conjunto de algoritmos prontos para soluções de mineração de dados, dentre estes está o algoritmo conhecido por *Microsoft Association Rules*, que tem a capacidade de encontrar regras de associações em itens frequentes em banco de dados. Este é geralmente aplicado em bases de dados que guardam informações sobre vendas e produtos.

Segundo *Data Mining Algorithms* (2016), o algoritmo é uma implementação inspirada no já conhecido algoritmo Apriori, que foi o pioneiro na geração e análise de itens em conjuntos.

### **2.6.3 Preparação de dados para o modelo de associação**

Antes de executar qualquer algoritmo de mineração de dados, é necessário preparar os dados alvos de maneira que atendam aos requisitos do modelo a ser usado, o que inclui a quantidade de dados necessária e como estes devem ser usados para se obter os resultados desejados.

Conforme pode ser visto em *Data Mining Algorithms* (2016), os modelos de regras de associação possuem os seguintes requisitos:

- Uma coluna chave única. Este requisito elimina automaticamente a possibilidade de existir chaves compostas, pois o modelo de dados deverá conter uma coluna que identifique de maneira única cada registro, podendo esta possuir um valor do tipo texto ou numérico.
- Uma única coluna preditiva. Um modelo de associação requer preferencialmente que exista apenas uma única coluna previsível, geralmente dos produtos que foram comprados, pois é onde mais se desejam encontrar regras de associação.
- Colunas de Entrada. As colunas de entrada devem possuir um conteúdo definido como discreto, isto é, quantidades finitas de valores sem

continuidade entre eles. Estas colunas de entradas podem estar presentes em mais de uma tabela, pois o modelo descrito permite o uso de tabelas aninhadas.

#### 2.6.4 Principais parâmetros e medidas de importância do algoritmo MAR

*“O algoritmo Regras de Associação da Microsoft não executa qualquer tipo de seleção de recursos automática. Em vez disso, ele fornece parâmetros que controlam os dados usados pelo algoritmo. Podem estar incluídos limites para o tamanho de cada conjunto de itens ou a configuração do suporte máximo e mínimo necessário para adicionar um conjunto de itens ao modelo (DATA MINING ALGORITHMS, 2016).”*

No algoritmo MAR é possível fazer ajustes nos valores dos diversos em seus parâmetros, sendo necessário saber a função e o nome que cada um possui. As medidas de importância e seus respectivos parâmetros dentro do algoritmo são:

- Suporte. Também é conhecido como frequência, que consiste no número de casos que contém o item destino ou a combinação de itens.
  - *MAXIMUM\_SUPPORT*: Designa o número máximo de casos que um conjunto de itens pode ter.
  - *MINIMUM\_SUPPORT*: Designa o número mínimo de casos que um conjunto de itens deve ter antes de gerar uma regra. Apenas os itens que têm a quantia mínima especificada de suporte podem ser incluídos no modelo.
- Probabilidade. Também chamado de confiança, determina a probabilidade de uma regra ser verdadeira através do seu percentual.
  - *MINIMUM\_PROBABILITY*: Fornece uma restrição do número de regras que um modelo produz de acordo com o valor definido neste parâmetro.
- Importância. É a pontuação gerada para cada regra criada. A importância de uma regra é calculada pela probabilidade do log do lado direito da regra, dado o lado esquerdo da regra. Após isso, é feita uma normalização da

proporção obtida, através de uma escala logarítmica;

- *MINIMUM\_IMPORTANCE*: Define o limite mínimo de importância para as regras extraídas do modelo de associação. Caso uma regra tenha um valor menor que o definido por este parâmetro, ela será descartada.

O mecanismo de *Analysis Services* oferece ainda outros parâmetros que podem ser facilmente configurados no modelo do algoritmo MAR, além de permitir também a criação de novos parâmetros.

### **2.6.5 Interpretação de regras em um conjunto de itens**

A implementação do algoritmo MAR parte do modelo mais comum de regras de associação, isto é, variáveis Booleanas que representam sempre valores binários como sim/não ou Existente/Ausente são atribuídos às colunas. Conforme afirma *Data Mining Algorithms (2016)*, uma análise de cesta de compras é um exemplo de modelo dessas regras, pois usa essas variáveis Booleanas para determinar se um item da cesta está ausente ou presente na compra realizada por um cliente.

Durante o treinamento dos dados o algoritmo é capaz de a partir de um conjunto de itens, criar pontuações com base nos valores de confiança e suporte, onde estas pontuações podem ser usadas para determinar se as regras encontradas são interessantes para a solução.

Particularmente, o MAR oferece suporte a modelos de associação com banco de dados que possuem atributos numéricos ou de texto. Isso se deve ao fato de que o algoritmo implementa em suas funcionalidades a capacidade de tratar esses valores diferenciados como booleanos ou como pares atributo-valor. Esta característica torna-se uma grande vantagem em relação a implementação básica do Apriori. Isso se deve ao fato de que, a responsabilidade de tratar os atributos e transformá-los em valores Booleanos para compreensão do algoritmo é uma tarefa exclusiva do analista responsável pela solução.

A Figura 5 mostra um conjunto de regras obtido a partir de um determinado banco de dados comercial onde o algoritmo MAR foi aplicado sobre um conjunto de dados, cujas informações dizem respeito às compras feitas diariamente por clientes e seus respectivos produtos nas cestas de compras.

**Figura 5: Regras de associação obtidas de uma aplicação do MAR.**

Rule
REJUNTE COLACERAMICA IMPER FLEX PLATINA = Existente -> ARGAMASSA COLACERAMICA AC2 15KG = Existente
REJUNTE COLACERAMICA IMPER FLEX CINZA = Existente -> ARGAMASSA COLACERAMICA AC2 15KG = Existente
ARGAMASSA COLACERAMICA AC2 15KG = Existente -> REJUNTE COLACERAMICA IMPER FLEX PLATINA = Existente
ARGAMASSA COLACERAMICA AC2 15KG = Existente -> REJUNTE COLACERAMICA IMPER FLEX CINZA = Existente

A imagem acima ilustra um modelo de regras de associação que é expresso por um antecedente seguido de um conseqüente. Para exemplificar este modelo basta analisar a primeira linha da imagem, onde o lado esquerdo que corresponde ao antecedente da regra foi atribuído o produto “REJUNTE COLACERAMICA IMPER FLEX PLATINA” acrescido de um valor “Existente”, e no lado direito da regra onde se localiza o conseqüente, o produto “ARGAMASSA COLACERAMICA AC2 15KG” recebe também o valor “Existente”. Diante disso, implica dizer que ambos os produtos tendem a serem comprados juntos de acordo com a confiança dessa regra, que é determinada pelo parâmetro *MINIMUM\_PROBABILITY* definido no tópico anterior.

## 2.7 Considerações acerca das aplicações e técnicas descritas

As técnicas e algoritmos definidos pelos autores e pesquisadores da área de Descoberta de Conhecimento de Banco de Dados, mais especificamente de Regras de Associação, são sempre vinculadas a situações de vendas no comércio, com foco sempre no conceito de bases de dados de transações, entretanto, é possível extrair regras de associação de qualquer base de dados onde existam relacionamentos e atributos implícitos, principalmente se tratando de modelos de bancos de dados relacionais.

Portanto, diante dos principais conceitos e definições discorridas, torna-se possível realizar uma aplicação de projeto de mineração de dados que atenda a proposta definida para este trabalho.

### 3. TRABALHO PROPOSTO

#### 3.1 Visão Geral

Uma empresa que trabalha no setor de varejo, mais especificamente na área de materiais para construção, utiliza em suas operações um *software* ERP de código fechado para gerir de forma coordenada e estruturada suas operações. Este *software* chama-se Protheus, implementado e comercializado pela empresa Totvs S.A.

Segundo Meirelles (2016), em um estudo feito pela Fundação Getúlio Vargas entre 2015 e 2016, foi percebido que a Totvs S.A detém a maior parte do mercado de *software* ERP, o que corresponde a aproximadamente 35% das empresas que utilizam esta categoria de solução de TI. Além dela, empresas como a SAP que detém 31% do mercado e a Oracle com 15%, também aparecem como líderes na pesquisa realizada.

No Protheus é possível desenvolver novas funcionalidades ou melhorar as que já existem através de customizações. Essa possibilidade deve-se ao fato de que a Totvs adota o conceito de ponto de entrada em seus códigos-fontes, possibilitando assim a personalização de seu sistema sem alterar suas funcionalidades principais e nem liberar o acesso a seu código original.

Como está definido na arquitetura de um sistema ERP, este possui um banco de dados centralizado onde são armazenadas todas as informações pertinentes à empresa e às funcionalidades do próprio *software*. Sabendo disso, é natural que um banco de dados empresarial se torne, com o passar do tempo, um grande repositório de dados.

Trazendo essa realidade para o ambiente da empresa, fica evidente a necessidade de potencializar a base de dados que esta possui, e a partir disso encontrar recursos que apontem solução para desafios ou informações estratégicas para entendimento de problemas.

O acesso a estas informações se deu pela oportunidade de ser colaborador da companhia, mais especificamente trabalhando com o setor de TI. Diante disso, a ideia de desenvolver um projeto de melhoria do ERP foi proposta ao diretor do setor de TI,

onde seriam utilizadas as informações do histórico de vendas, obtidas por meio de mineração de dados.

Durante a formulação e discussão do projeto, foi estimado o período de vendas necessário, correspondendo a aproximadamente seis meses, o escopo temporal para mineração no presente trabalho. Para assegurar o sigilo da empresa varejista, será atribuído o pseudônimo “Construmat” para referir-se a ela e ocultar sua razão social.

### **3.2 A problemática**

Ao iniciar o projeto foi necessário, antes das etapas práticas, encontrar limitações do sistema Protheus em tomada de decisões ou ainda problemas apresentados por ele em sua operacionalidade. Essa busca se deu através de visitas presenciais aos respectivos setores da empresa Construmat, onde foram feitas diversas entrevistas com os funcionários de cada partição buscando detectar exatamente os problemas ali presentes.

Durante esse período, foi percebido que o setor que mais demandava melhorias era o setor de vendas e atendimento ao cliente. Neste setor, os usuários (vendedores) relataram diversas limitações do sistema, dentre estas estava a busca por produtos no momento de realizar a venda. O usuário nesse momento precisava selecionar produto por produto do orçamento ao efetuar a venda para o cliente, o que demandava uma quantidade considerável de tempo caso houvesse um número grande de itens a serem selecionados.

A partir dos principais conceitos e técnicas presentes em soluções de KDD, foi feita a opção de se realizar uma aplicação de regras de associação para MBA. A aplicação dessas técnicas permitiu, além da melhoria de *software*, um suporte maior para tomada de decisões e criação de estratégias de negócio para facilitar também ações de *marketing* e venda. Além disso, a escolha mostrou-se potencialmente capaz de contribuir na busca de padrões de produtos frequentemente associados, sabendo que essa informação é diariamente armazenada no banco de dados da Construmat.

### **3.3 A base de dados utilizada**

Como colaborador atuante no setor de TI da empresa, conhecer as informações a respeito da base de dados da Construmat se tornou tarefa cotidiana para esta etapa do estudo.

A companhia utiliza um SGBD da *Microsoft Corporation*, mais especificamente a ferramenta *SQL Server 2008 versão Datacenter Edition*, que permite o gerenciamento de um banco de dados com grande capacidade de armazenamento e consulta de seus registros. O SGBD é executado como um serviço em uma máquina servidora, possibilitando o acesso aos dados para os usuários através da rede corporativa presente no ambiente da empresa.

O SGBD é integrado exclusivamente com ERP Protheus, que armazena dados desde a sua implantação no ano de 2007, possuindo atualmente um arquivo de dados bruto com um tamanho aproximado de 90 GB. Além disso, adota o modelo de banco de dados relacional, onde as informações são gravadas em tabelas que possuem relacionamentos entre si. Entre os dados armazenados estão os registros de todos os setores da empresa tais como, contas a pagar, contas a receber, impostos, folha salarial, informações dos clientes, registro de compras e vendas, além de todas as informações sobre documentos fiscais e burocráticos.

Como a ideia proposta se restringe à análise de histórico de compras e cesta de produtos, duas tabelas do banco de dados são relevantes para esta tarefa, a tabela de orçamento de vendas chamada SL1010 que contém as informações acerca do cabeçalho geral da venda, e a tabela de itens do orçamento de venda SL2010 que contém sobre os produtos de seus respectivos orçamentos. Ambas as tabelas possuem um atributo que permitem a consulta de seus dados, uma chave primária da venda.

### **3.4 A escolha das ferramentas**

Pensando em aplicar algoritmos de regras de associação no contexto de MBA, foram feitas várias pesquisas com o objetivo de encontrar as melhores ferramentas e os melhores algoritmos para realizar esta tarefa. Levando em consideração o mecanismo de SGBD utilizado pela Construmat, foram analisados alguns *softwares*.

### 3.4.1 A ferramenta Weka e suas particularidades

Buscando uma ferramenta gratuita e com facilidade de comunicação com o *SQL Server*, foi feita a análise do *Waikato Environment Knowledge Analysis (Weka)*, que possui algoritmos para mineração de dados voltados para o tipo de estudo desejado, como pode ser visto em Corrêa (2013). Sobre esta possibilidade, o Weka permite minerar regras de associação diretamente das tabelas do banco de dados, sem que seja necessária uma extração dos dados para um arquivo de texto ou de planilha.

A partir disso, foi feita a opção pela aplicação do algoritmo Apriori, já implementado na ferramenta Weka. Após a escolha, surgiram grandes obstáculos que posteriormente levaram a substituição do *software* de mineração, dentre estes, vale ressaltar a dificuldade de adaptação dos dados brutos para que eles pudessem ser lidos pelo algoritmo. Este problema se deve ao fato de que o algoritmo espera que os dados já estejam formatados no modelo de transação com exposto na Figura 6.

**Figura 6: Exemplo de modelo Transação aceita pelo Weka.**

TID	Produtos Comprados
1	biscoito, cerveja, chá, salaminho
2	cerveja, couve, linguiça, pão, queijo
3	café, brócolis, couve, pão
4	brócolis, café, cerveja, couve, pão, salaminho
5	brócolis, café, couve, pão, refrigerante
6	couve, linguiça

**Fonte: Corrêa (2013).**

No modelo esperado, conforme apresentado na Figura 6 o Weka receberia todos os produtos de suas respectivas vendas em um único registro de transação. Isto dificultaria o trabalho proposto, pois os dados estão dispostos em tabelas separadas, possuindo registros individuais de produtos, e uma enorme quantidade de itens cadastrados. Tentou-se encontrar uma solução através de consultas para esse problema, porém não houve êxito devido à sua complexidade e recursos

computacionais insuficientes. Diante disso a opção mais viável foi substituir a ferramenta por outra que viesse a facilitar a mineração dos dados.

### 3.4.2 O mecanismo Analysis Services e o pacote Microsoft Visual Studio

Após a dificuldade de trabalhar com a ferramenta Weka, foi então necessário encontrar outro *software* que pudesse satisfazer à necessidade de mineração dos dados. Durante a busca, foi descoberto que o próprio SQL Server oferece suporte à análise de dados por meio de mineração de dados. Esse suporte se dá através do mecanismo “*Analysis Services*”, que pode ser oferecido como um serviço de uma mesma instância do SQL Server.

Para então utilizar esse mecanismo, são necessários dois requisitos. O primeiro é fazer a instalação do serviço *Analysis Services* que pode ser obtido no arquivo de instalação do próprio SGBD SQL Server, sendo importante notificar que para seu correto funcionamento, todos os mecanismos precisam pertencer à mesma versão. Além disso, durante a instalação deve-se optar pela configuração no modo Multidimensional, pois as soluções de mineração de dados são viáveis apenas neste formato.

O segundo requisito é a instalação do pacote IDE (*Integrated Development Environment*) da Microsoft, o *Visual Studio*. Como a própria definição de IDE diz, o *Visual Studio* é um pacote de programas para desenvolvimento integrado de *software*, mais especificamente dedicado para soluções de ferramentas da própria Microsoft. Com isso, a escolha da IDE possibilitou uma fácil comunicação com o banco de dados SQL Server e com o mecanismo *Analysis Services*.

A Microsoft disponibiliza três tipos de distribuição de sua IDE, a versão *Community* com recursos gratuitos para estudantes e desenvolvedores individuais, a versão *Professional* apenas para assinantes, com ferramentas profissionais e serviços para pequenas equipes de desenvolvimento, e oferece ainda a versão *Enterprise* com soluções para atender projetos robustos voltados para a demanda de alta qualidade e em larga escala. Diante dessas três versões, foi adotada a distribuição *Community* por ser a mais acessível para o trabalho proposto, além de oferecer soluções suficientemente capazes de suprir a demanda do projeto.

## 4. DESENVOLVIMENTO DA SOLUÇÃO

Para o desenvolvimento da solução, um conjunto de passos foi seguido até se chegar aos resultados obtidos. As tarefas práticas para a aplicação de KDD seguiram um formato e uma sequência diferente da definida pelos autores e pesquisadores da área. Além destas, outras atividades realizadas serão descritas e mostradas neste capítulo.

### 4.1 Extração dos dados nativos

Como citado no capítulo 3, a base de dados da Construmat está instalada em um servidor da rede da empresa. Para manter a integridade dos dados reais da produção, foi necessário fazer a instalação de uma cópia desse banco dados para uma máquina local onde os testes pudessem ser executados sem prejuízos.

Todos os registros e tabelas do SQL Server são condensados em dois arquivos no seu diretório de instalação. O primeiro é um arquivo que possui o nome da base de dados e uma extensão chamada *Master Data File* (MDF), onde todos os dados brutos são salvos. O segundo arquivo é um arquivo que também possui o nome da base de dados, porém no seu nome é acrescentado a palavra “Log” e possui uma extensão *Log Data File* (LDF). Este segundo arquivo, é responsável por armazenar em formato de log todas as transações feitas pelo banco de dados. Como o estudo se limita à análise das informações guardadas nas tabelas, para a tarefa de cópia da base dados, apenas o arquivo de extensão MDF foi necessário.

Para a realização da cópia, foi preciso antes instalar e configurar uma instância do SQL Server. Como a versão do SGBD da Microsoft usado pela Construmat era ainda do ano de 2008, buscou-se então uma mais recente e estável. Por ser uma ferramenta paga, o SQL Server limita suas versões gratuitas permitindo o armazenamento de dados apenas até 10 GB. Dado o tamanho do arquivo de dados da Construmat, que ultrapassa esse limite, a solução encontrada foi aderir a uma versão de avaliação do SQL Server 2014 que limita seu funcionamento à apenas 180 dias.

Escolhido o SGBD, foi realizada então a sua instalação e configuração completa, incluindo o mecanismo *Analysis Services*. Em seguida, uma cópia do

arquivo “MDF” contendo os dados foi transferido para o diretório de instalação da instância recém-configurada do SQL Server 2014. O último passo foi criar através de um *script* SQL, uma nova base de dados na máquina local e anexar a ela o arquivo copiado do servidor conforme pode ser visto na Figura 7.

**Figura 7: Script de criação da base de dados.**

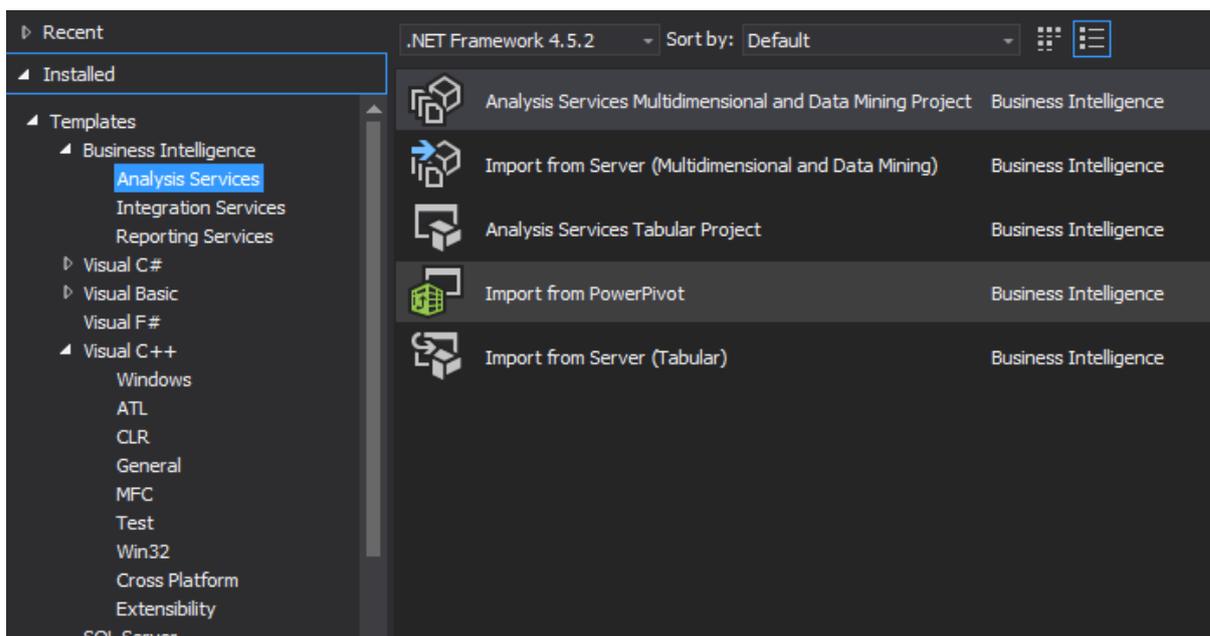
```
CREATE DATABASE Datamining
ON (FILENAME = 'C:\Program Files\Microsoft SQL
Server\MSSQL12\MULTIDSQL\MSSQL\DATA\Database_Protheus.mdf')
FOR ATTACH;
```

No *script* da Figura 7, a cláusula *FOR ATTACH* é responsável por anexar o arquivo de dados presente no diretório indicado na linha anterior. Depois de criado o banco de dados, o passo seguinte foi criar um projeto de mineração de dados no *Visual Studio*.

#### **4.2 Criando um projeto de mineração de dados no *Visual Studio***

O processo de criação de um projeto de mineração de dados no *Visual Studio* é simples e rápido, pois sua interface gráfica de usuário (do inglês *Graphical User Interface* - GUI) é um dos pontos positivos da ferramenta. Para criar um projeto é preciso selecionar a opção *File* na página principal, em seguida escolher a opção *Project*. Feito isso, uma tela com a opção de vários *templates* será apresentada como mostra a Figura 8.

**Figura 8: Templates de projetos do Visual Studio.**



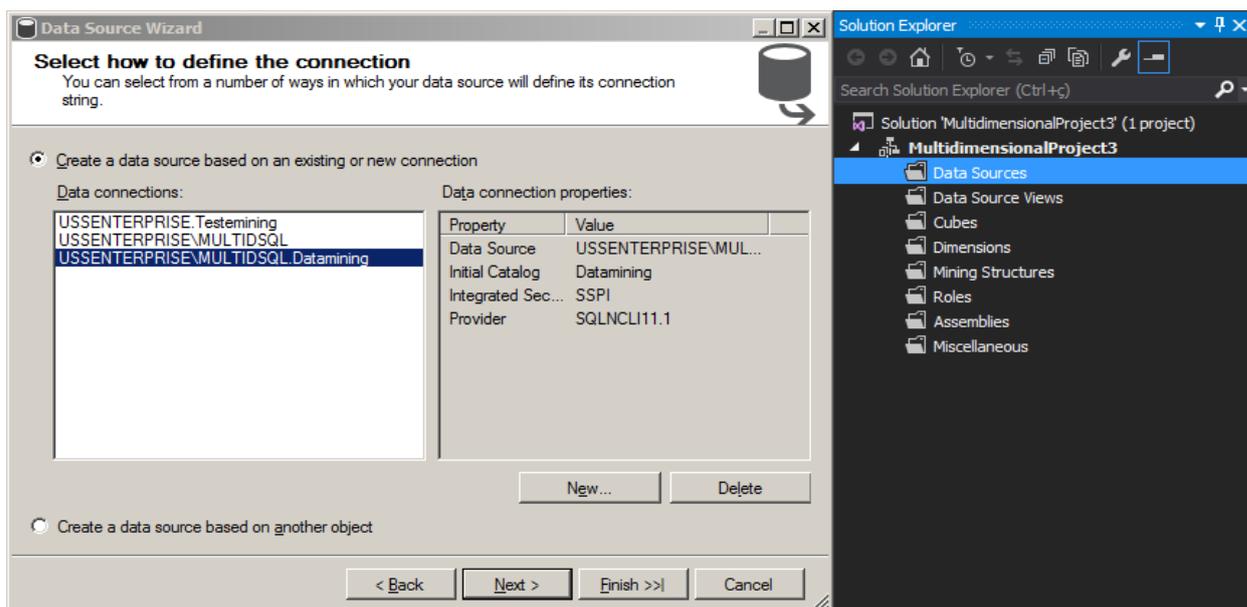
Como o mecanismo usado na aplicação de KDD foi o *Analysis Services*, o *template* que engloba esse tipo de solução é mostrado no Visual Studio como uma ferramenta de *Business Intelligence* (BI). Dentre as opções deste tipo de projeto, a mais adequada foi a primeira opção mostrada na Figura 8, identificada como “*Analysis Services Multidimensional and Data Mining Project*”.

As etapas fundamentais do processo de KDD conforme, Fayyad et al. 1996, serão mostradas implicitamente pelos passos seguidos no desenvolvimento do projeto. Algumas destas fases não foram necessárias, como limpeza dos dados e transformação. Isso ocorreu pela ausência de dados com ruídos, e pelo fato da técnica de mineração escolhida necessitar apenas de um grupo específico de atributos. Outro fator que facilitou a aplicação foi o uso de um único SGBD pela empresa Construmat.

#### **4.2.1 Criando uma fonte de dados e selecionando as tabelas**

Após a criação do projeto no Visual Studio, a tarefa seguinte foi criar uma fonte de dados que fizesse comunicação com o banco de dados *SQL Server* anexado. Para isso, na opção *Data Sources* da aba *Solution Explorer*, é possível criar uma nova fonte de dados através da propriedade *New Data Source*. A Figura 9 mostra a tela com as conexões de banco de dados disponíveis para criação da fonte de dados.

**Figura 9: Conexões disponíveis para criar a fonte de Dados.**



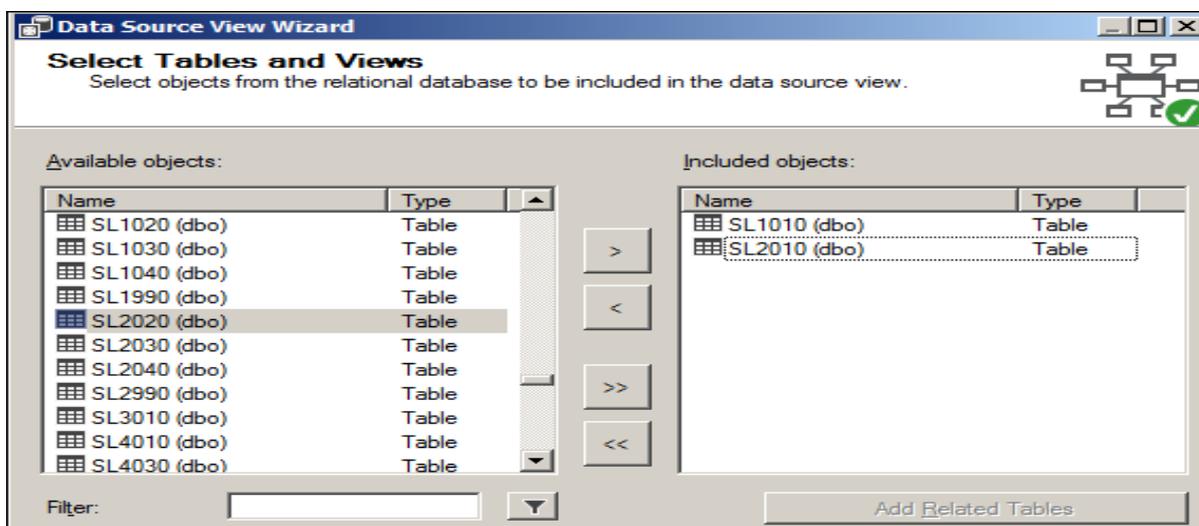
Ao selecionar a conexão com o banco de dados escolhido, os passos seguintes são as definições de usuário e o nome da fonte de dados criada. Até o momento nenhuma tabela foi escolhida para a mineração, portanto é possível realizar consultas em quaisquer tabelas usando o próprio Visual Studio.

Para selecionar os conjuntos de dados que vão ser treinados, é preciso especificar as tabelas que contêm os registros que serão usados. Isso é possível criando a *Data Source Views*, que são tabelas virtuais contendo os dados reais que se desejam mostrar. Como descrito anteriormente, para o trabalho proposto interessam apenas a tabela de orçamentos de vendas (SL1010) e a tabela de produtos de orçamento de vendas (SL2010). Dessa maneira, será necessário aninhar as duas tabelas para a aplicação do algoritmo. Este aninhamento foi possível na própria criação das *views*, onde será mostrado abaixo como fazê-la.

1. Deve-se clicar com o botão direito na opção “*Data Source Views*” na janela *Solution Explorer*;
2. Ao selecionar “*New Data Source Views*”, uma janela será mostrada onde deve ser selecionada a fonte de dados configurada na etapa anterior;
3. Em seguida, como mostra a Figura 10, uma lista contendo todas as tabelas do banco de dados será mostrada. Nela deverá ser selecionada apenas as tabelas que serão usadas no treinamento;

4. Após isso, o último passo é nomear a “Data Source View”.

Figura 10: Lista de tabelas da fonte de dados.



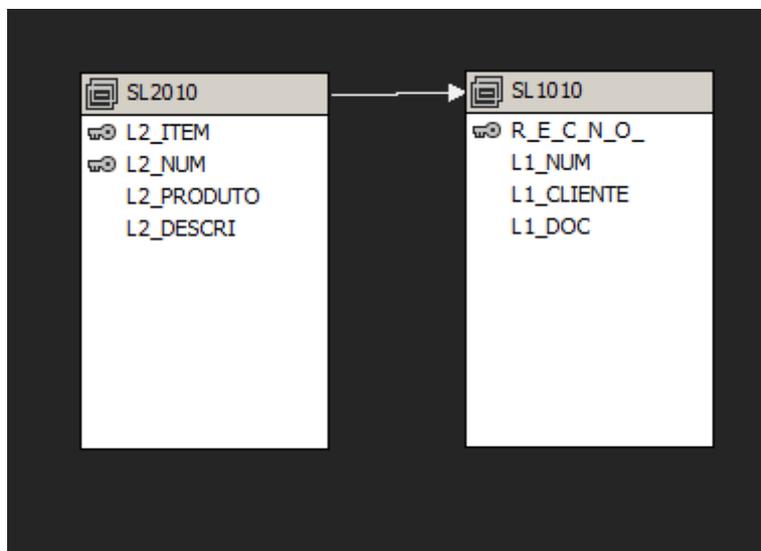
Por padrão, as *views* criadas são uma cópia completa das tabelas originais, contendo todos os atributos e registros delas. Porém, é possível alterá-las para mostrar apenas as colunas importantes para o treinamento dos dados, além permitir também a inclusão de filtros para a seleção dos registros apropriados. Essas alterações podem ser feitas facilmente através de consultas SQL inseridas diretamente dentro da *view*.

Com o objetivo de minerar dados correspondentes apenas às vendas do ano de 2015, as *views* utilizadas para a mineração de dados foram alteradas, resultando em mais de 300.000 registros de produtos de vendas. Além de filtrar os registros para o ano escolhido, também foram alteradas as visualizações das colunas a serem mostradas, retirando assim as que não tinham valor relevante para a análise. As consultas criadas para filtragem de colunas e registros estão nos Apêndices A e B.

Outra característica importante das tabelas virtuais, permitida pelo *Analysis Services*, é a definição de atributos chaves genéricas para identificar unicamente cada registro e possibilitar o aninhamento de tabelas criando relacionamento entre os atributos chaves. Como ambas as tabelas SL1010 e SL2010 são complementos uma da outra, foram utilizados, respectivamente, os atributos L1\_NUM e L2\_NUM, que correspondem ao número da venda, para criar um relacionamento entre as *views*. A

Figura 11 mostra o diagrama das tabelas virtuais criadas pelo Visual Studio e uma seta indicando o relacionamento entre elas.

**Figura 11: Diagrama das views usadas para a mineração de dados.**



A seleção das colunas destas *views* se deu pela importância destes atributos para a identificação de cada registro das vendas e dos produtos. A tabela 1 apresenta as descrições de cada atributo selecionado.

**Tabela 1: Atributos selecionados e suas descrições.**

Colunas	Descrição
L1_NUM, L2_NUM	Número da Venda
L1_CLIENTE	Código do Cliente
L1_DOC	Número do Documento Fiscal da Venda
R_E_C_N_O_	Número identificador Único do Registro
L2_ITEM	Número do Item da Cesta de Produtos
L2_PRODUTO	Código do Produto
L2_DESCRI	Descrição do Nome do Produto

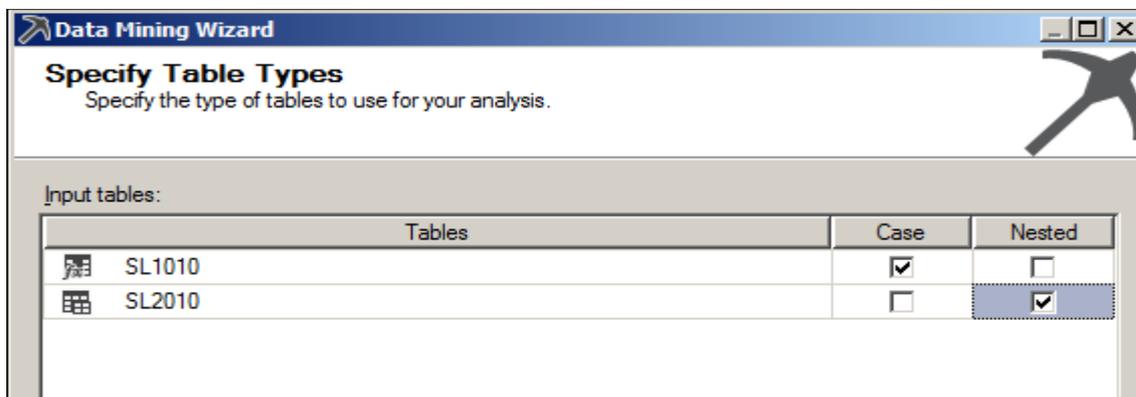
#### 4.2.2 Criando uma estrutura e um modelo de mineração

Tendo selecionado as tabelas e filtrados os dados, a etapa seguinte é o ponto principal da aplicação. Nela, é criada a estrutura do modelo de mineração, que pode ser configurado no menu *Mining Structure* pertencente também ao *Solution Explorer*.

Na rotina de criação da estrutura de mineração, o primeiro passo é escolher o algoritmo que deverá ser aplicado. Dado o objetivo da aplicação presente, foi utilizado

o algoritmo *Microsoft Association Rule*. Depois de selecionado, a rotina de criação da estrutura exibe as *views* criadas na fonte de dados, como mostra a Figura 12.

Figura 12: Seleção de tabelas *views* para estrutura de mineração.

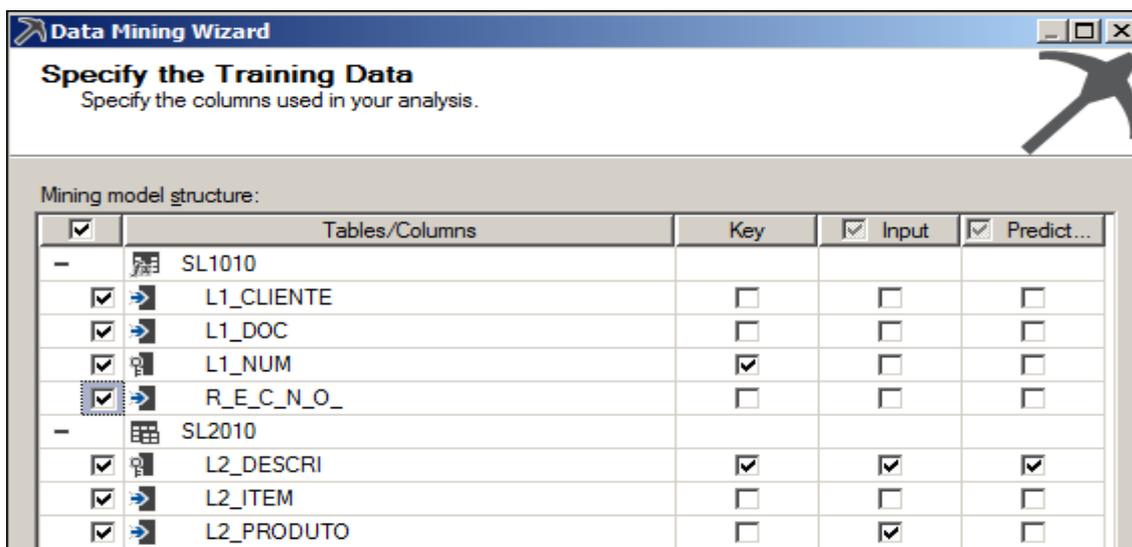


Na imagem da Figura 12, é possível perceber dois tipos de tabelas possíveis, a tabela do tipo *case* que corresponde à tabela principal, e a tabela tipo *nested* que diz respeito a uma tabela que pode ser aninhada com a principal. A tabela SL1010 foi escolhida como *case* por ser o cabeçalho geral das vendas, enquanto a tabela SL2010 é a tabela *nested* por conter o complemento das informações sobre as vendas, no caso os dados dos produtos.

Ao serem definidas as tabelas da estrutura, é preciso definir também os papéis de cada atributo das tabelas para a interpretação do algoritmo. A esse componente da estrutura dá-se o nome de Modelo de Mineração. Para uma mesma estrutura, podem ser criados diversos modelos diferentes, permitindo a mineração dos dados sobre outras óticas, oferecendo uma análise dos dados com parametrizações e funções de atributos distintos entre os vários modelos da estrutura de mineração criada.

As colunas podem assumir três funções diferentes, a função de atributo chave, a função de atributo de entrada e a função de atributo de previsão. A Figura 13 aponta os papéis escolhidos para os atributos da tabela de venda e de produtos.

Figura 13: Especificação dos papéis das colunas.



Para o modelo de mineração, são necessárias mais duas especificações de dados. Estas podem ser feitas na tela seguinte, e dizem respeito ao tipo de conteúdo armazenado na coluna e ao tipo de dado. Os conteúdos atribuídos às colunas foram do tipo discreto, ou seja, quantidades finitas de valores sem continuidade entre eles. Contudo, para os tipos de dados, foram definidos conforme as especificações originais do banco, isto é, todas as colunas possuindo um tipo textual, com exceção do atributo “R\_E\_C\_N\_O\_”, que contém valores numéricos.

#### 4.2.3 Definindo os parâmetros do modelo para o algoritmo

O *Analysis Services* fornece parâmetros com valores padrões pré-definidos, sendo possível alterar estes para adequá-los à necessidade das regras que se deseja encontrar. Para a análise de MBA da Construmat, dois modelos com parametrizações distintas foram criados. O primeiro modelo com valores padrão do algoritmo, enquanto o segundo sofreu algumas alterações para visualizar outro tipo de regra gerado. A Tabela 2 mostra os parâmetros dos dois modelos criados.

Tabela 2: Parâmetros usados nos modelos de mineração

PARÂMETROS	VALORES MODELO 1 (PADRÕES)	VALORES MODELO 2
<i>MAXIMUM_ITEMSET_COUNT</i>	200.000	200.000
<i>MAXIMUM_ITEMSET_SIZE</i>	3	30
<i>MAXIMUM_SUPPORT</i>	1.0	1.0
<i>MINIMUM_IMPORTANCE</i>	-	-
<i>MINIMUM_ITEMSET_SIZE</i>	1	1
<i>MINIMUM_PROBABILITY</i>	0.4	0.1
<i>MINIMUM_SUPPORT</i>	0.0	0.0

A Tabela 2 apresenta apenas duas distinções entre os valores padrões e os valores do segundo modelo de mineração. Essas distinções estão presentes nos parâmetros *MAXIMUM\_ITEMSET\_SIZE* e *MINIMUM\_PROBABILITY*, que foram alterados no modelo 2 para que o algoritmo pudesse gerar regras de formatos diferentes, isto é, as regras devem ser geradas com conjuntos que possuam no máximo 30 itens e uma probabilidade mínima de ocorrência de 10%. Através disso, este modelo de regra permite compreender os padrões de vendas com uma maior quantidade de produtos.

Para o restante dos parâmetros foram mantidos os valores pré-determinados. Abaixo estão descritos o que cada valor significa para o resultado da mineração:

- O valor de 200.000 atribuído ao parâmetro *MAXIMUM\_ITEMSET\_COUNT* implica dizer que este será o número máximo de conjuntos de itens a serem produzidos. Este valor foi mantido por ser suficientemente capaz de expressar uma quantidade satisfatória de regras para a análise proposta.
- O *MAXIMUM\_SUPPORT* recebe o valor real 1.0 para determinar que o número máximo de casos que um conjunto de itens pode conter está limitado ao total existente na base de dados, ou seja, 100% destes.
- O parâmetro *MINIMUM\_IMPORTANCE* não possui valor padrão atribuído, pois fica a critério do analista do projeto definir o limite mínimo de importância para as regras de associação. A depender do valor recebido o algoritmo descarta regras com importância menor que a definida neste parâmetro.

- Em *MINIMUM\_PROBABILITY*, o valor real 0.4 especifica a probabilidade mínima de 40% para uma regra ser verdadeira, ou seja, nenhuma regra com valor abaixo disso será gerada. Para iniciar a análise dos dados essa porcentagem foi mantida por ter sido considerada um valor relevante para o estudo, resultando em regras relevantes ao final da aplicação.
- O suporte mínimo definido como 0.0 em *MINIMUM\_SUPPORT* também foi mantido o valor padrão, com a finalidade de que houvesse regras com qualquer suporte.

Tendo definido todos os parâmetros, resta então compilar o projeto e executar o treinamento do algoritmo sobre os dados selecionados. Dada a quantidade de registros da Construmat referente às vendas do ano de 2015, o treinamento de cada modelo da estrutura durou aproximadamente 14 horas para ser finalizado.

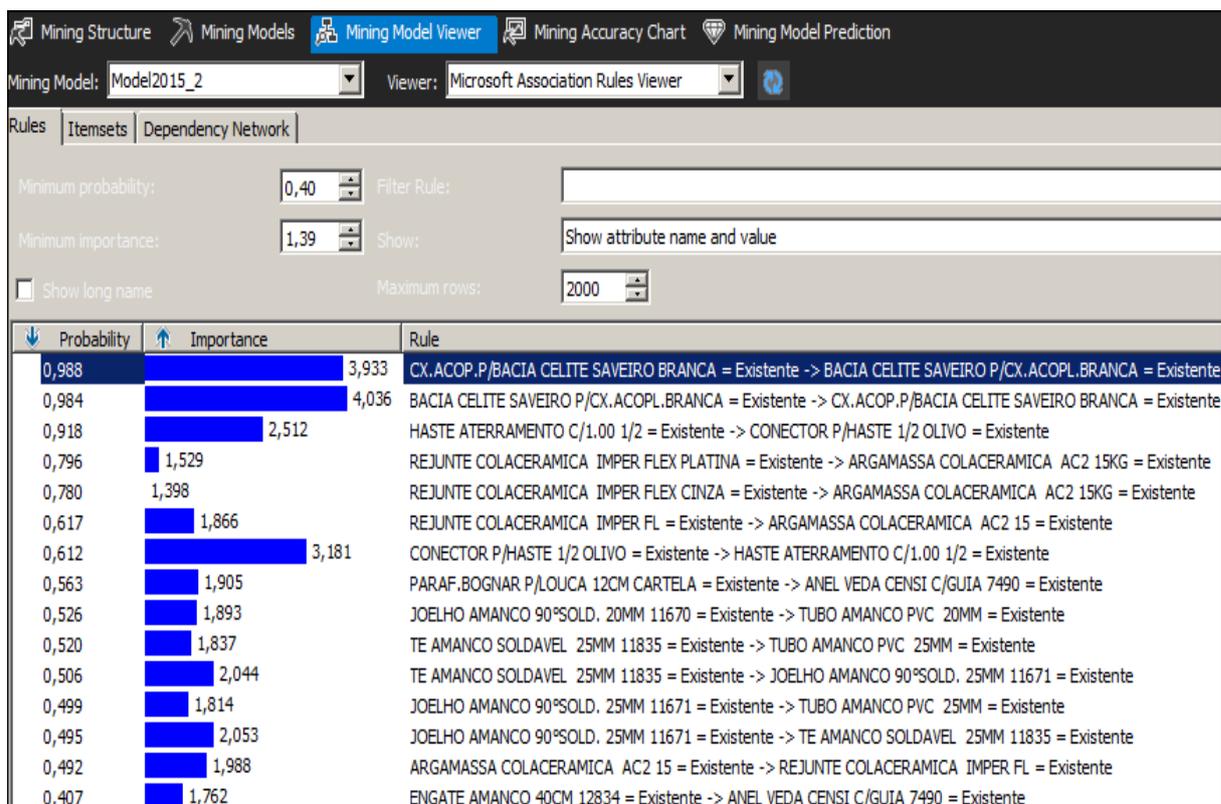
### **4.3 Visualizando as regras geradas**

A visualização das regras geradas se dá de duas formas. A primeira através da própria IDE *Visual Studio*, que contém ferramentas que facilitam o entendimento e a análise das regras via interface gráfica de usuário. O segundo método de visualização requer um conhecimento da linguagem DMX, que permite a seleção das regras que se deseja encontrar, usando cláusulas do padrão SQL e diversas funções próprias para a análise das regras.

#### **4.3.1 Visualizando regras com o Visual Studio**

Para visualizar as regras geradas, a IDE fornece três formas diferentes para compreender as associações dos itens presentes nos conjuntos. A primeira é a apresentação das regras no formato  $A \rightarrow B$ . Essa forma pode ser vista no próprio modelo de mineração criado, mais especificamente em “*Mining Model Viewer*” na guia *rules*. A Figura 14 mostra 15 regras geradas conforme a seleção de dados escolhida e os parâmetros padrões do algoritmo.

Figura 14: Visualização das regras no menu rules.



Este modelo de expressão de regras apresentado na Figura 14, facilita a compreensão do analista do projeto, bem como de seus gestores e responsáveis pela companhia, que na maioria das vezes não compreendem de forma clara o trabalho proposto. Para visualizar as regras encontradas no segundo modelo, consulte o Anexo A.

A segunda maneira de visualizar as regras, é por meio do detalhamento dos itens mais frequentes na base dados. Esse detalhamento toma como base a medida de suporte de cada produto e organiza-os conforme a força deste parâmetro, permitindo uma análise não só das regras de associação, mas também da movimentação desses itens durante o período de tempo escolhido. Este modelo de visualização é encontrado na guia *Itemsets* como mostra a Figura 15.

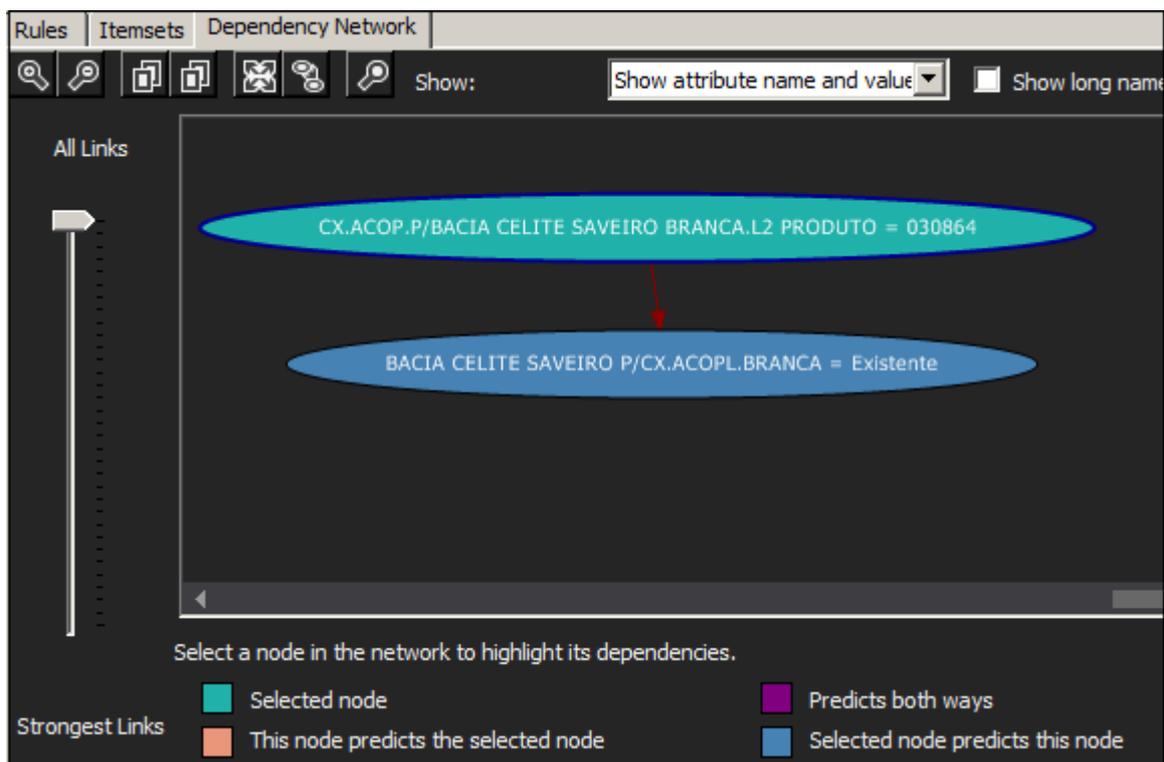
Figura 15: Visualização de itens frequentes.

Support	Size	Itemset
2324	1	REJUNTE COLACERAMICA IMPER FLEX PLATINA = Existente
2063	1	ARAME RECOZIDO 18 = Existente
1997	1	ARGAMASSA COLACERAMICA AC2 15 = Existente
1849	2	REJUNTE COLACERAMICA IMPER FLEX PLATINA = Existente, ARGAMASSA COLACERAMICA AC2 15KG = Existente
1641	1	COLA AMANCO PVC 75G 90060 = Existente
1591	1	REJUNTE COLACERAMICA IMPER FL = Existente
1478	1	TUBO AMANCO PVC 25MM = Existente
1406	1	ANEL VEDA CENSI C/GUIA 7490 = Existente
1365	1	ENGATE AMANCO 40CM 12834 = Existente
1337	1	BUCHA NYLON FIX 08MM 1.4G = Existente
1336	1	LAMINA STARRET 12X300MM BS1218 = Existente
1322	1	TUBO AMANCO PVC 20MM = Existente
1279	1	TUBO AMANCO ESGOTO 100MM = Existente

Dada quantidade de registros analisados, apenas alguns itens foram selecionados para apresentação na Figura 15. A coluna que indica o suporte está evidenciando numericamente a quantidade de casos em que o produto apresentado na linha ocorreu. Na coluna *size* está expressa a quantidade de itens do conjunto, seguida da declaração dos parâmetros do algoritmo, mais especificamente em *MINIMUM\_ITEMSET\_SIZE* e *MAXIMUM\_ITEMSET\_SIZE*. A visualização dos itens frequentes encontrados no segundo modelo de parâmetros, pode ser visto no Anexo B.

A IDE *Visual Studio* oferece ainda uma terceira e última forma de expressão de regras de associações. Esta se dá através da guia nomeada por *Dependency Network*, que significa rede de dependências. Nela as regras e os itens encontrados no treinamento são expressos graficamente através de um modelo de grafo orientado, onde os nós são os produtos e as arestas são as associações entre eles.

Figura 16: Produtos associados visualizados na rede de dependências.



Por meio desta representação, a compreensão das regras se torna mais dinâmica. Nela, o conjunto dos grafos iniciais contém todos os seus nós na cor branca indicando que não há visualização de dependências. Para visualizá-las, o analista deve selecionar um produto nó do grafo, onde a IDE deverá colorir os estes de acordo com o tipo de dependência de sua regra. O exemplo de rede encontrado no segundo modelo de mineração está apresentado no Anexo C.

Como indicado na Figura 16, um nó pode receber quatro colorações diferentes, sendo estas:

1. A coloração verde será atribuída ao nó selecionado pelo analista;
2. A cor rosa que indicará o nó ou os nós que predizem o nó selecionado;
3. A cor azul que indica que o nó selecionado (verde) prediz o nó que possui esta cor;
4. A cor roxa que indica uma dependência correspondida, onde um nó prediz o outro e vice-versa.

Sabendo do contingente de itens analisados pelo algoritmo, apenas um exemplo de regra foi selecionado para ilustrar a rede na Figura 16. Nela é possível perceber que há apenas dois itens no conjunto. O produto selecionado foi a “CX ACOP. P/BACIA CELITE SAVEIRO BRANCA”, onde este nó selecionado prever a existência do produto “BACIA CELITE SAVEIRO P/CX. ACOPL. BRANCA”, conforme as cores definidas pelo *Visual Studio*.

Portanto, diante dos três mecanismos de visualização mostrados e da expressão das regras, é possível afirmar que para a uma mineração de dados proposta e com os resultados obtidos, o *Visual Studio* atendeu satisfatoriamente o objetivo previsto.

#### **4.3.2 Visualizando regras por meio de consultas DMX**

Segundo *Data Mining Queries* (2016), consultas de mineração de dados podem ser usadas para diversos propósitos, como:

- Aplicar um novo modelo de dados para fazer diversas previsões, fornecendo valores de entrada como parâmetros ou em um lote.
- Obtenção de um resumo estatístico dos dados.
- Extração de padrões e regras ou criação de um perfil que representa um padrão no modelo usado.
- Obtenção de casos que se ajustem a um determinado padrão específico.
- Recuperação de detalhes sobre casos individuais usados no modelo, permitindo a inclusão de dados não usados no modelo.
- Treinar novamente um modelo adicionando novos dados para realizar uma previsão aninhada.

Além disso, as consultas DMX oferecem usabilidades diferentes para os resultados do treinamento, como a possibilidade de desenvolver-se uma solução de *software* com base nos resultados das *queries*. É dessa maneira que uma melhoria no ERP da Construmat seria possível, mais especificamente no módulo de vendas do Protheus.

Para a criação das consultas foi utilizado o *Management Studio* que é uma ferramenta do próprio SQL Server para manipulação e definição de dados. Nele, é possível selecionar os modelos de mineração de dados criados nas etapas anteriores com o *Visual Studio* e realizar consultas sobre as regras já mineradas. Para isso é necessária apenas a conexão com a instância do *Analysis Services* usada no projeto. A primeira consulta criada e suas respectivas saídas são apresentadas na Figura 17.

**Figura 17: Consulta DMX para visualização de regras.**

<pre> SELECT TOP 10 (NODE_SUPPORT) AS SUPORTE, ROUND (NODE_PROBABILITY, 2) AS CONFIANCA, NODE_CAPTION AS REGRA FROM Model2015.CONTENT WHERE NODE_TYPE = 8 ORDER BY NODE_SUPPORT DESC </pre>		
SUPORTE	CONFIANCA	REGRA
1552	0,42	ARGAMASSA COLACERAMICA AC2 15KG = Existente -> REJUNTE COLACERAMICA IMPER FLEX PLATINA = Existente
1552	0,8	REJUNTE COLACERAMICA IMPER FLEX PLATINA = Existente -> ARGAMASSA COLACERAMICA AC2 15KG = Existente
978	0,62	REJUNTE COLACERAMICA IMPER FL = Existente -> ARGAMASSA COLACERAMICA AC2 15 = Existente
978	0,49	ARGAMASSA COLACERAMICA AC2 15 = Existente -> REJUNTE COLACERAMICA IMPER FL = Existente
721	0,99	CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente -> BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente
721	0,99	BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente -> CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente
595	0,79	REJUNTE COLACERAMICA IMPER FLEX CINZA = Existente -> ARGAMASSA COLACERAMICA AC2 15KG = Existente
532	0,41	ANEL VEDA CENSI C/GUIA 7490 = Existente -> ENGATE AMANCO 40CM 12834 = Existente
532	0,46	ENGATE AMANCO 40CM 12834 = Existente -> ANEL VEDA CENSI C/GUIA 7490 = Existente
498	0,63	PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente -> ANEL VEDA CENSI C/GUIA 7490 = Existente

A *query* apresentada na Figura 17 retorna as dez regras que possuem maior índice de suporte ordenando-as de forma decrescente, acompanhada de suas respectivas probabilidades (confiança). No predicado *where* o valor “oito” filtrado em *NODE\_TYPE* indica o tipo de nó que deve ser mostrado pelo modelo de associação, havendo apenas três variações de modelos conforme apresentado na Tabela 3.

Tabela 3: Tipos de variação de nós.

ID do tipo de nó	Tipo de nó
1 (Modelo)	Nó raiz ou nó pai
7 (Conjunto de itens)	Conjunto de itens ou coleção de pares expressos na forma atributo-valor. Exemplo: Produto 1 = Existente, Produto 2 = Existente.
8(Regra)	Regras que definem como itens se relacionam entre si, por exemplo: Produto 1 = Existente -> Produto 2 = Existente.

Fonte: Adaptado de Data Mining Queries (2016).

Num outro tipo de consulta realizada, o valor de retorno é a previsão de ocorrência de determinados itens. Através desta, é possível determinar se um produto poderá compor uma regra de associação com outros produtos. Para esta determinação, o algoritmo verifica a probabilidade de ocorrência de cada produto mediante todos os registros em que estes apareceram. A *query* e as tuplas retornadas por ela estão presentes na Figura 18.

Figura 18: Query para previsão de produtos.

<pre>SELECT PredictAssociation([Model2015].[SL2010],INCLUDE_STATISTICS, 4) FROM [Model2015] NATURAL PREDICTION JOIN (SELECT (SELECT 'PRODUTOS' AS [L2_DESCRI]) AS [ITENS DE CESTA DA COMPRAS]) AS t</pre>			
L2 DESCRÍ	\$SUPPORT	\$PROBABILITY	\$ADJUSTEDPROBABILITY
ARGAMASSA COLACERAMICA AC2 15KG	3671	0,02996000979352	0,0295156573582409
ARGAMASSA COLACERAMICA AC2 15	1988	0,0162245980576185	0,0160936882093061
REJUNTE COLACERAMICA IMPER FLEX PLATINA	1941	0,0158410185260752	0,0157162094952211
REJUNTE COLACERAMICA IMPER FL	1587	0,0129519301395577	0,0128684148426456

Para a demonstração foram selecionados apenas quatro produtos dentre os diversos itens analisados. Nesta consulta foi utilizada a função *PredictAssociation*, que é nativa da extensão DMX, onde seus resultados correspondem às recomendações de produtos e tem vários argumentos que podem ser usados para personalizar o resultado da consulta.

Buscando uma análise mais profunda dos padrões presentes na base de dados da Construmat, outro modelo de consulta foi aplicado sobre as regras encontradas. Embora tais regras e a função *PredictAssociation* sejam úteis para gerar as recomendações como é mostrado na Figura 18, é possível examinar outros conjuntos de itens que contenham determinado produto para ter uma certeza maior se este

realmente é um item frequentemente comprado por clientes. A consulta usada para exemplificar como este tipo de recomendação pode ser feita é vista na Figura 19 com o produto “ANEL VEDA CENSI C/GUIA 7490”.

Figura 19: Query de recomendação de produtos.

<pre> SELECT * FROM   (SELECT FLATTENED NODE_CAPTION, NODE_SUPPORT,     (SELECT ATTRIBUTE_NAME FROM NODE_DISTRIBUTION      WHERE ATTRIBUTE_NAME = 'SL2010(ANEL VEDA CENSI C/GUIA 7490)')     AS PRODUTO    FROM MODEL2015.CONTENT    WHERE NODE_TYPE = 7) AS Itens  WHERE [PRODUTO.ATTRIBUTE_NAME] &lt;&gt; NULL  ORDER BY NODE_SUPPORT DESC </pre>	
NODE_CAPTION	NODE_SUPPORT
ANEL VEDA CENSI C/GUIA 7490 = Existente	1306
ENGATE AMANCO 40CM 12834 = Existente, ANEL VEDA CENSI C/GUIA 7490 = Existente	532
PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente, ANEL VEDA CENSI C/GUIA 7490 = Existente	498
BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente, ANEL VEDA CENSI C/GUIA 7490 = Existe...	373
CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente, ANEL VEDA CENSI C/GUIA 7490 = Existente	372
CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente, BACIA CELITE SAVEIRO P/CX.ACOPL.BR...	372

Diante do conjunto de informações de saída dessa *query*, torna-se fácil compreender que dado o produto escolhido, quaisquer produtos presentes nos nós mostrados no resultado da consulta podem ser recomendados como itens associados à uma mesma venda. Portanto, é com estas consultas que implementações de melhorias na rotina de vendas do ERP da Construmat seriam viáveis. Para simular isto, um módulo genérico de vendas foi desenvolvido e será apresentado a seguir.

#### 4.4 A implementação de uma rotina de venda com recomendação de produtos

Para testar a funcionalidade das regras em uma melhoria de *software*, foi criado um módulo genérico de vendas com sugestão de produtos. Para estas sugestões foram utilizadas as regras obtidas da mineração da base de dados da Construmat. Dado o limite de tempo para uma implementação real no ERP da empresa e a indisponibilidade da licença para customizações do Protheus, o desenvolvimento desta rotina deu-se pela necessidade de demonstrar a relevância prática e útil do presente trabalho, oferecendo uma simulação da rotina de realização de vendas pelos vendedores que são os usuários que a utilizam para atendimento dos clientes.

Buscando praticidade e rapidez no desenvolvimento, foi escolhida a linguagem de programação *Personal Home Page* (PHP), que juntamente com a linguagem de marcação *HiperText Markup Language* (HTML), são utilizadas para gerar conteúdos e aplicações voltadas para a *Web*. Além disso, foi usado o *framework Bootstrap* para auxiliar no *design* da ferramenta.

Sabendo que a linguagem DMX é uma extensão do SQL nativa do *Analysis Services*, as consultas mostradas anteriormente não foram capazes de ser incorporadas ao código-fonte escrito em PHP, pois a linguagem oferece comunicação apenas com o mecanismo de banco de dados padrão simples do SQL. Nesse contexto, a solução encontrada para o acesso das regras pelo módulo desenvolvido foi o armazenamento destas em uma tabela relacional do banco de dados configurado na máquina local. A Figura 20 mostra a tabela criada e alguns registros presentes nela.

**Figura 20: Tabela de regras de associação.**

	REGRA	SUORTE	PROBABILIDADE	REGRA_NOME	PRODUTO_NOME
12	ARAME RECOZIDO 18 = Existente	1144	0,0093	91	ARAME RECOZIDO 18
13	TUBO AMANCO PVC 20MM = Existente	1113	0,0091	90	TUBO AMANCO PVC 20MM
14	TUBO AMANCO ESGOTO 100MM = Existente	1088	0,0089	89	TUBO AMANCO ESGOTO 100MM
15	LAMINA STARRET 12X300MM BS1218 = Existente	1049	0,0086	88	LAMINA STARRET 12X300MM BS1218
16	REJUNTE COLACERAMICA IMPER FL = Existente, ARGAMA...	978	0,008	119	ARGAMASSA COLACERAMICA AC2 15
17	REJUNTE COLACERAMICA IMPER FL = Existente, ARGAMA...	978	0,008	119	REJUNTE COLACERAMICA IMPER FL
18	TE AMANCO SOLDAVEL 25MM 11835 = Existente	936	0,0076	87	TE AMANCO SOLDAVEL 25MM 11835
19	LIXA PAREDE GR120 3M = Existente	928	0,0076	86	LIXA PAREDE GR120 3M
20	JOELHO AMANCO 90°SOLD. 25MM 11671 = Existente	877	0,0072	85	JOELHO AMANCO 90°SOLD. 25MM 11671
21	LAMPADA PHILIPS ELETR. ECO HOM = Existente	820	0,0067	84	LAMPADA PHILIPS ELETR. ECO HOM

Após a criação da tabela, o acesso aos seus registros foi obtido através do uso de funções do PHP próprias para a comunicação com o banco de dados SQL Server. Estabelecida a conexão com a base de dados, o passo seguinte foi desenvolver a interface visual de interação com o usuário, conforme apresentado na Figura 21.

**Figura 21: Formulário de busca de produtos.**

**Selecione a opção**

Código

Pesquisar

A primeira interface consiste em uma página inicial contendo um formulário para busca de produtos no banco de dados. Nesta, foi utilizada a tabela de cadastro de produtos SB1010 para retornar as informações acerca dos produtos reais da Construmat cadastrados na base de dados. Como mostra a Figura 21, na página é

possível selecionar o tipo de busca que se deseja fazer no banco, oferecendo três opções de consultas, sendo estas: a busca filtrada pelo código do produto, a busca pela descrição nominal do produto, além da opção de pesquisa pelo nome do fabricante do produto. A Figura 22 mostra as informações de alguns produtos filtrados através da pesquisa por descrição de um produto com o nome de “Anel Veda”.

**Figura 22: Pesquisa de produtos pela descrição.**

Selecione a opção

Descrição

anel veda

Pesquisar

026397	026396
ANEL VEDA CENSI C/GUIA 7490	ANEL VEDA CENSI S/GUIA 7480
CENSI	CENSI
Preço : R\$ 9,57	Preço : R\$ 6,79
<a href="#">Comprar</a>	<a href="#">Comprar</a>

Ao receber a entrada no formulário, a página irá retornar além dos nomes dos produtos desejados, as informações sobre os fabricantes, preços e códigos dos produtos. Para escolher um item para a compra, o usuário deve clicar na opção comprar e sem seguida o módulo irá encaminhar-se para uma página contendo as informações sobre a venda. Esta página é chamada de carrinho de compras e é nela que se apresentam as sugestões de produtos com base nos itens adicionados ao carrinho, como apresenta a Figura 23.

**Figura 23: Carrinho de compras com sugestão de produtos.**

Carrinho de Compras						
Código do Produto	Produto	Fabricante	Quantidade	Preço	SubTotal	Remover
026397	ANEL VEDA CENSI C/GUIA 7490	CENSI	<input type="text" value="1"/>	R\$ 9.57	R\$ 9.57	<a href="#">Remove</a>
<b>Quantidade Total</b>						<b>1 Unidades</b>
<b>Total</b>						<b>R\$ 9.57</b>
<input type="button" value="Atualizar Carrinho"/>						
<a href="#">Continuar Comprando</a>						
<input type="button" value="Encaminhar Pedido"/>						
<input type="button" value="Finalizar Venda"/>						
030863	BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA	LOGASA	Preço : R\$ 90,00 <a href="#">Comprar</a>	030864	CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA	LOGASA
				001290	ENGATE AMANCO 40CM 12834	AMANCO
						Preço : R\$ 4,45 <a href="#">Comprar</a>

Como mostrado na Figura 23, a página do carrinho além de apresentar as informações da compra do cliente, permite a atualização da quantidade de produtos e oferece também a opção de adicionar outros ao carrinho através opção “Continuar Comprando”, que retorna para a página do formulário de busca. Para facilitar a adição de produtos e evitar a pesquisa item por item, a ferramenta realiza consultas na tabela de regras de mineração e retorna os itens que formam regras de associação com o produto adicionado ao carrinho.

A sugestão de produtos é um artifício já utilizado e consolidado em vários sistemas de vendas, podendo ser aplicado em diversos cenários do comércio, sendo capaz de trazer notória otimização para o processo de venda, além possibilitar um relativo acréscimo de faturamento da empresa, já que a sugestão possibilita a adição de outros produtos que por ventura não seriam vendidos justamente pela falta de recomendação. Além do ganho para a companhia, a sugestão de produtos oferece vantagem também para o cliente, pois está é uma solução para os casos em que este acaba por várias vezes esquecendo de levar alguns itens, evitando desperdício de tempo tanto para quem compra, quanto para quem vende.

Portanto, com o módulo de simulação de vendas desenvolvido, é possível compreender como os resultados extraídos da etapa de mineração de dados podem auxiliar rotinas de vendas. A ferramenta desenvolvida foi apresentada aos vendedores como uma demonstração simples de como funcionará o processo de sugestão de produtos dentro do Protheus. Os códigos-fontes escritos para desenvolvimento das páginas do formulário e do carrinho de compras, estão presentes nos Apêndices C e D respectivamente.

## 5. CONSIDERAÇÕES FINAIS

No cotidiano de qualquer empresa, a responsabilidade de tomar decisões é uma atividade de suma importância para os seus gestores, pois é por meio desta que se define o futuro da companhia. Mediante a importância dessas escolhas, é mais que necessário se obter bases e informações seguras para uma decisão correta, caso contrário a ocorrência de erros nessa tarefa pode trazer grandes consequências. Buscando organizar as operações e apoiar estratégias, são usados sistemas de informações empresariais, que nem sempre oferecem o melhor suporte para todos os processos de uma corporação.

Trazendo este cenário para dentro do ambiente da Construmat, foi aplicado o estudo de KDD buscando minerar regras de associação. Esta aplicação foi dirigida pelo objetivo de garantir capacidade para amparar melhorias no ERP da empresa, mais especificamente dentro do escopo definido para o presente trabalho, isto é, no módulo de vendas para a recomendação de produtos, visando um aumento qualitativo na relação com o cliente e um incremento no faturamento da empresa.

Desempenhando o papel definido pela mineração de dados, ao analisar o histórico de compras presente no banco de dados foram encontrados diversos padrões de associação de produtos. Isto viabilizou a criação de um módulo genérico de sugestão de produtos desenvolvido para ambiente Web, que traz consigo o objetivo de simular a aplicação real na rotina de vendas da produção da companhia. A ferramenta desenvolvida demonstra de forma prática como o processo de atendimento ao cliente pelos vendedores pode ser melhorado, trazendo benefícios para ambas as partes envolvidas no comércio, isto é, a empresa e os clientes.

Ao encontrar maneiras de inovar e melhorar processos já consolidados dentro da empresa, a Construmat por meio da utilização de recursos e informações presentes em seu próprio repositório de dados, transforma em um grande passo para estar à frente de seus concorrentes, o desafio de analisar seus dados e dar novas utilidades a eles.

Todavia, com os resultados obtidos e havendo um conjunto de possíveis soluções para outros cenários, mostra-se bastante viável a realização de melhorias também em diversos outros módulos do sistema, assim como a colaboração em ações de gestão da empresa. Portanto, ao observar esse potencial da aplicação de mineração de dados, além da quantidade de informações relevantes que podem ser extraídas da base de dados da Construmat, há perspectivas futuras de implementação de projetos de customização do ERP Protheus na empresa.

## REFERÊNCIAS

- HAN, Jiawei e KAMBER, Micheline. **Predict Data Mining - Concepts and Techniques**. Morgan Kaufmann Publishers, Inc, 2001.
- O'LEARY, Daniel. **Enterprise Resource Planning Systems**. University of Southern California, 2000.
- BANCROFT, N., SEIP, H. e SPRENGEL, A. **Implementing SAP R/3**. 2. ed. Manning Publications, Greenwich, 1998.
- DAVENPORT, Thomas H. Ecologia da informação: Por que só a tecnologia não basta para o sucesso na era da informação. Tradução Bernadette Siqueira Abrão — São Paulo : Futura, 1998.
- FAYYAD, U. M.; PIATESKY, Shapiro, G.; SMYTH, P. **From Data Mining to Knowledge Discovery: An Overview: Advances in Knowledge Discovery and Data Mining**, AAAI Press, 1996.
- BERRY, Michael J. A., LINOFF, Gordon. **Data mining techniques:For marketing, sales and customer support**. 2. ed. USA: Wiley Computer Publishing, 1997.
- SILVA, M. P. S. **Mineração de Dados: Conceitos, Aplicações e Experimentos com Weka**. Livro da Escola Regional de Informática Rio de Janeiro - Espírito Santo. Porto Alegre: Sociedade Brasileira de Computação, 2004, v. 1, p. 1-20.
- FIGUEIRA, Rafael. **Mineração de dados e banco de dados orientados a objetos**. Dissertação, Mestrado em Ciência da Computação. Rio de Janeiro: UFRJ, 1998.
- TORRES, Noberto. **Competitividade empresarial com a tecnologia da informação**. São Paulo: Markrom Books, 1995.
- GONÇALVES, Eduardo C. **Mining de Regras de Associação**. Instituto Brasileiro de Geografia e Estatística – IBGE, 2013.
- FRACALANZA, Livia Fonseca. **Mineração de Dados voltada para Recomendação no Âmbito de Marketing de Relacionamento**. Pontifícia Universidade Católica do Rio de Janeiro, 2009.
- AGUIAR, Gustavo Maia. **História do SQL Server**. Fórum Social de Desenvolvedores Microsoft, Mar. 2008. Disponível em: <<https://goo.gl/D0vVgA>>. Acesso em: 10 Mar. 2017.
- Analysis Services. **Microsoft Developer Network**, Washington (U.S.A), dez. 2016. Disponível em: <<https://goo.gl/DVf9g0>>. Acesso em: 10 Mar. 2017.
- Mineração de dados.**Microsoft Developer Network**, Washington (U.S.A), dez. 2016. Disponível em: <<https://goo.gl/r1AXxw>>. Acesso em: 10 Mar. 2017.

Data Mining Algorithms: Microsoft Association Algorithm. **Microsoft Developer Network**, Washington (U.S.A), dez. 2016. Disponível em: <<https://goo.gl/9yRiWr>>. Acesso em: 27 Mar. 2017.

CORRÊA, Eduardo. **Mineração de Dados com Market Basket Analysis**: Revista SQL Magazine 111, 2013. Disponível em: <<https://goo.gl/USqn2m3>>. Acesso em 28 Mar. 2017.

Data Mining Queries.**Microsoft Developer Network**, Washington (U.S.A), dez. 2016. Disponível em: <<https://goo.gl/cBxxcr>>. Acesso em 28 Mar. 2017.

MEIRELLES, Fernando S. **Pesquisa Anual do Uso de TI**. Fundação Getúlio Vargas. 27ª Edição, 2016. Disponível em: <<https://goo.gl/HLtURw> >. Acesso em: 16 Abril. 2017.

## ANEXOS

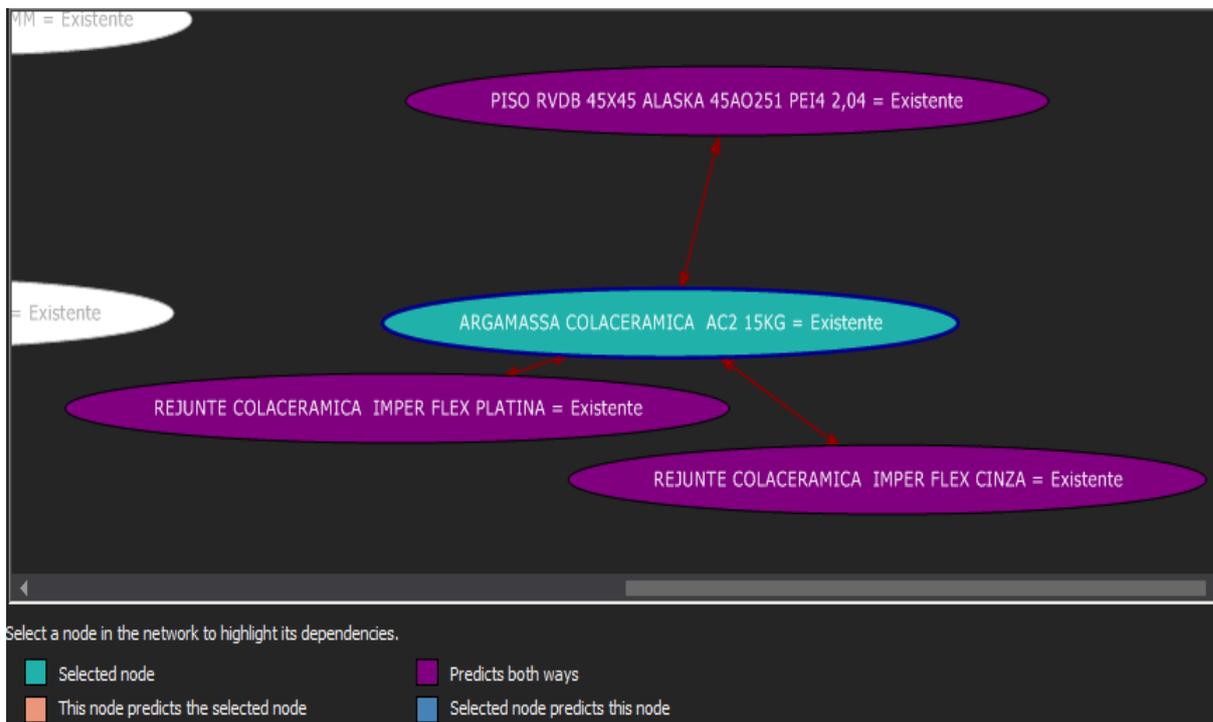
## Anexo A - Algumas regras encontradas com o segundo modelo de mineração.

↑	Probability	↑	Importance	Rule
	0,102		1,822	ARGAMASSA COLACERAMICA AC2 15KG = Existente -> PISO RVDB 45X45 ALASKA 45AO251 PEI4 2,04 = Existente
	0,162		2,092	ARGAMASSA COLACERAMICA AC2 15KG = Existente -> REJUNTE COLACERAMICA IMPER FLEX CINZA = Existente
	0,285		1,982	ANEL VEDA CENSI C/GUIA 7490 = Existente -> CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente
	0,286		1,985	ANEL VEDA CENSI C/GUIA 7490 = Existente -> BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente
	0,347		2,128	TUBO AMANCO ESGOTO 100MM = Existente -> TUBO AMANCO ESGOTO 50MM = Existente
	0,353		2,084	TUBO AMANCO PVC 20MM = Existente -> JOELHO AMANCO 90°SOLD. 20MM 11670 = Existente
	0,354		2,001	TUBO AMANCO PVC 25MM = Existente -> JOELHO AMANCO 90°SOLD. 25MM 11671 = Existente
	0,381		2,206	ANEL VEDA CENSI C/GUIA 7490 = Existente -> PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente
	0,385		2,017	TUBO AMANCO PVC 25MM = Existente -> TE AMANCO SOLDAVEL 25MM 11835 = Existente
	0,407		1,903	ANEL VEDA CENSI C/GUIA 7490 = Existente -> ENGATE AMANCO 40CM 12834 = Existente
	0,423		2,110	ARGAMASSA COLACERAMICA AC2 15KG = Existente -> REJUNTE COLACERAMICA IMPER FLEX PLATINA = Existente
	0,463		1,861	ENGATE AMANCO 40CM 12834 = Existente -> ANEL VEDA CENSI C/GUIA 7490 = Existente
	0,470		2,198	PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente -> CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente
	0,471		2,200	PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente -> BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente
	0,483		2,139	TE AMANCO SOLDAVEL 25MM 11835 = Existente -> JOELHO AMANCO 90°SOLD. 25MM 11671 = Existente
	0,492		1,988	ARGAMASSA COLACERAMICA AC2 15 = Existente -> REJUNTE COLACERAMICA IMPER FL = Existente
	0,505		2,169	CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente -> PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente
	0,506		2,171	BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente -> PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente

## Anexo B - Itemsets analisados no segundo modelo de mineração.

↓	Support	Size	Itemset
	375	1	COLA AMANCO PVC 850G = Existente
	374	1	BUCHA TRAM ALUM.1 56134/024 = Existente
	373	2	BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente, ANEL VEDA CENSI C/GUIA 7490 = Existente
	372	2	CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente, ANEL VEDA CENSI C/GUIA 7490 = Existente
	372	3	CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente, BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente, ANEL VEDA CENSI C/GUIA 7490 = Existente
	370	2	BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente, PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente
	369	2	CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente, PARAF.BOGNAR P/LOUCA 12CM CARTELA = Existente
	369	3	CX.ACOP.P/BACIA CELITE SAVEIRO BRANCA = Existente, BACIA CELITE SAVEIRO P/CX.ACOPL.BRANCA = Existente, PARAF.BOGNAR P/LOUCA 12CM CARTELA = ...
	364	1	ADAPT.AMANCO SOLD.LR 32X1 11509 = Existente
	364	1	CABO COND/NAMB/SIL/COBREM.FLEX = Existente
	360	1	VEDA ROSCA AMANCO 18X25 10435 = Existente
	359	1	ASSENTO AMANCO MUNDIAL BRANCO = Existente
	355	1	ELETRODUTO AMANCO SOLD.25MM = Existente
	352	1	ADAPT.AMANCO SOLD.LR 32X1 115 = Existente
	351	1	LUVA AMANCO LR 25X 3/4 11755 = Existente
	350	1	LIXA PAREDE GR 100 3M = Existente
	349	1	PREGO 3X8 GERDAU 110 = Existente

### Anexo C - Rede de dependências do segundo modelo de mineração.



## APÊNDICES

### **Apêndice A – Código do *script* da *view* SL1010**

```
SELECT R_E_C_N_O_, L1_NUM, L1_CLIENTE, L1_DOC  
FROM SL1010  
WHERE (L1_EMISSAO BETWEEN '20150101' AND '20151231') AND  
(D_E_L_E_T_ = '')
```

### **Apêndice B – Código do *script* da *view* SL2020**

```
SELECT L2_ITEM, L2_NUM, L2_DESCRI, L2_PRODUTO  
FROM SL2010  
WHERE (L2_EMISSAO BETWEEN '20150101' AND '20151231') AND  
(D_E_L_E_T_ = '')
```

## Apêndice C – Código-Fonte da página index.php

```

<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
    <link rel="stylesheet"
    href="http://localhost:8080/Monografia/bootstrap/css/bootstrap.css"/>
<title>Efetuando Venda</title>
    <link rel="stylesheet"
    href="//code.jquery.com/ui/1.11.4/themes/smoothness/jquery-ui.css">
<script src="//code.jquery.com/jquery-1.10.2.js"></script>
<script src="//code.jquery.com/ui/1.11.4/jquery-ui.js"></script>

<script>
$(function() {
    $( "#speed" ).selectmenu();});
</script>
</head>
    <body id="body_efetuar_venda">
        <div class="container">
            <div class="row">
                <div id="principal">
<div id="tipo_busca">
                <div class="col-md-8">
                    <form name="categoria" action="index.php" method="POST">
                        <div class="form-group">
                            <label for="speed">Selecione a opção</label>
                            <div class="input-group">
                                <select class="form-control" name="tipo_busca" id="speed">
                                    <option value="Codigo">Código</option>
                                    <option value="Descricao">Descrição</option>
                                    <option value="Fabricante">Fabricante</option>
                                </select>
                                <input id="box" class="form-control" type="text" name="entrada">
                            </div>
                        </div>
                    <button type="submit" id="botao" class="btn btn-info">Pesquisar</button>
                </div>
            </div>
        </div>
        </div>
        </div>
        </div>
</div>

<div id="produtos">
<?php
//CHAMA ARQUIVO DE CONEXÃO COM BANCO DE DADOS
require("conexao.php");
//VERIFICA SE HÁ OPÇÃO VÁLIDA ESCOLHIDA NO FORMULÁRIO DE BUSCA
if($_POST['tipo_busca']== ""){
$_POST['tipo_busca'] = isset($_POST['tipo_busca']) ? $_POST['tipo_busca'] : null;

//CONDIÇÃO CASO A OPÇÃO ESCOLHIDA SEJA CÓDIGO
}if($_POST['tipo_busca'] == "Codigo") {
    $codigo = $_POST['entrada'];

```

```

//CONSULTA AO BANCO ATRAVÉS DO FABRICANTE
$sql = "SELECT B1_COD, B1_DESC, B1_FABRIC, B1_PRV1 FROM SB1010 WHERE B1_COD = '".
$codigo . "'";
$qqr = sqlsrv_query($conn, $sql) or die( print_r( sqlsrv_errors(), true));
echo '<br /><br /><br /><br /><br />';
while($ln = sqlsrv_fetch_array($qqr, SQLSRV_FETCH_ASSOC)){
echo '<h3>'. $ln['B1_COD'].'</h3> <br />';
echo '<h3>'. $ln['B1_DESC'].'</h3> <br />';
echo '<h3>'. $ln['B1_FABRIC'].'</h3> <br />';
echo 'Preço : R$ '.number_format($ln['B1_PRV1'], 2, ',', '.').<br />';
//ADICIONANDO O PRODUTO CLICADO AO CARRINHO
echo '<a
href="/Monografia/carrinho.php?acao=add&id='.$ln['B1_COD'].'">Comprar </a>';
echo '<br /><hr />';
}

//CONDIÇÃO CASO A OPÇÃO ESCOLHIDA SEJA DESCRIÇÃO
}elseif($_POST['tipo_busca'] == "Descricao") {
    $descricao = $_POST['entrada'];
    $descricao = strtoupper($descricao);
//CONSULTA AO BANCO ATRAVÉS DA DESCRIÇÃO
$sql = "SELECT B1_COD, B1_DESC, B1_FABRIC, B1_PRV1 FROM SB1010 WHERE B1_DESC LIKE
'%" . $descricao . "%'";
$qqr = sqlsrv_query($conn, $sql) or die( print_r( sqlsrv_errors(), true));
echo '<br /><br /><br /><br /><br />';
?>
<div class="col-md-12">
<?php
while($ln = sqlsrv_fetch_array($qqr, SQLSRV_FETCH_ASSOC)){
?>
    <div class="col-md-4">
        <div class="panel panel-default">
<div class="panel-body">
<?php
echo '<h3>'. $ln['B1_COD'].'</h3></p>';
echo '<p><h3>'. $ln['B1_DESC'].'</h3></p>';
echo '<p><h3>'. $ln['B1_FABRIC'].'</h3></p>';
echo '<p>Preço : R$ '.number_format($ln['B1_PRV1'], 2, ',', '.').</p>';
$produto = $ln['B1_DESC'];
echo '<a
href="/Monografia/carrinho.php?acao=add&id='.$trim($ln['B1_COD']).'&produto='.$produ
to.'">Comprar</a>';
?>
        </div>
    </div>
</div>

<?php
}
//CONDIÇÃO CASO A OPÇÃO ESCOLHIDA SEJA FABRICANTE
}elseif($_POST['tipo_busca'] == "Fabricante") {
    $fabricante = $_POST['entrada'];
    $fabricante = strtoupper($fabricante);
//CONSULTA AO BANCO ATRAVÉS DO FABRICANTE
$sql = "SELECT B1_COD, B1_DESC, B1_FABRIC, B1_PRV1 FROM SB1010 WHERE B1_FABRIC =
'". $fabricante . "'";

```

```

$qqr = sqlsrv_query($conn, $ssql) or die( print_r( sqlsrv_errors(), true));
    echo '<br /><br /><br /><br /><br />';
while($ln = sqlsrv_fetch_array($qqr, SQLSRV_FETCH_ASSOC)){
    echo '<h3>'.$ln['B1_COD'].'</h3> <br />';
    echo '<h3>'.$ln['B1_DESC'].'</h3> <br />';
    echo '<h3>'.$ln['B1_FABRIC'].'</h3> <br />';
    echo 'Preço : R$ '.number_format($ln['B1_PRV1'], 2, ',', '.').'<br />';
echo '<a href="/Monografia/carrinho.php?acao=add&id='.$ln['B1_COD'].'">Comprar</a>';
echo '<br /><hr />';
}
}
?>

</div>

</div>

<!-- Bootstrap core JavaScript===== -->
<script src="../bootstrap/js/jquery.min.js"></script>
<script src="../bootstrap/js/bootstrap.min.js"></script>
<script src="../bootstrap/js/bootstrap.js"></script>
<!-- Colocado no final do documento para que as páginas carreguem mais rápido -->

</body>
</html>

```

## Apêndice D – Código-Fonte da página carrinho.php

```

<?php
    session_start();

    if(!isset($_SESSION['carrinho'])){
        $_SESSION['carrinho'] = array();
    }

    //adiciona produto

    if(isset($_GET['acao'])){

        //ADICIONAR CARRINHO
        if($_GET['acao'] == 'add'){
            $id = $_GET['id'];
            if(!isset($_SESSION['carrinho'][$id])){
                $_SESSION['carrinho'][$id] = 1;
            }else{
                $_SESSION['carrinho'][$id] += 1;
            }
        }

        //REMOVER CARRINHO
        if($_GET['acao'] == 'del'){
            $id = $_GET['id'];
            if(isset($_SESSION['carrinho'][$id])){
                unset($_SESSION['carrinho'][$id]);
            }
        }

        //ALTERAR QUANTIDADE
        if($_GET['acao'] == 'up'){
            if(is_array($_POST['prod'])){
                foreach($_POST['prod'] as $id => $qtd){
                    $id = $id;
                    $qtd = intval($qtd);
                    if(!empty($qtd) || $qtd <> 0){
                        $_SESSION['carrinho'][$id] = $qtd;
                    }else{
                        unset($_SESSION['carrinho'][$id]);
                    }
                }
            }
        }
    }

}

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <link rel="stylesheet"
        href="http://localhost:8080/Monografia/bootstrap/css/bootstrap.css"/>
<title>Carrinho de Compras</title>
</head>

<body id="body_carrinho1">
<div class="container">
<div class="row">
<div class="col-md-12">
<table class="table table-bordered" id="table">
    <caption><h3>Carrinho de Compras<h3></caption>
    <thead>

```

```

        <tr>
            <th width="244">Código do Produto</th>
            <th width="244">Produto</th>
            <th width="150">Fabricante</th>
            <th width="79">Quantidade</th>
            <th width="89">Preço</th>
            <th width="100">SubTotal</th>
            <th width="100">Remover</th>
        </tr>
    </thead>
    <form action="?acao=up" method="post">
    <tfoot>
        <tr>
            <td colspan="7"><input type="submit" value="Atualizar Carrinho" /></td>
        <tr>
            <td colspan="7"><a href="index.php">Continuar Comprando</a></td>
        </tr>
    </tfoot>

    <tbody>

<?php
require("conexao.php");
require("rules.php");

$nome = "";
if(count($_SESSION['carrinho']) == 0){
    echo '<tr><td colspan="7">Não há produto no carrinho</td></tr>';
}else{
    $total = 0;

foreach($_SESSION['carrinho'] as $id => $qtd){
    $sql = "SELECT B1_COD, B1_DESC, B1_FABRIC, B1_PRV1 FROM SB1010 WHERE B1_COD =
        '". $id . "'";
    $qr = sqlsrv_query($conn, $sql) or die( print_r( sqlsrv_errors(), true));
    $ln = sqlsrv_fetch_array($qr, SQLSRV_FETCH_ASSOC);

    $codigo = $ln['B1_COD'];
    //ARRAY QUE GUARDA OS CÓDIGOS DOS PRODUTOS DO CARRINHO
    $array_cod[] = $codigo;
    $fabricante = $ln['B1_FABRIC'];
    $guarda_fab[] = $fabricante;
    $nome = $ln['B1_DESC'];
    $guarda_nome[] = $nome;
    $preco = number_format($ln['B1_PRV1'], 2, '.', '');
    $sub = number_format($ln['B1_PRV1'] * $qtd, 2, '.', '');
    $total += $ln['B1_PRV1'] * $qtd;
    $guarda_qtd[] = $qtd;

echo '<tr>
    <td align="center">'. $codigo. '</td>
    <td>'. $nome. '</td>
    <td>'. $fabricante. '</td>
    <td><input type="text" size="3" name="prod['. $id. ']" value="'. $qtd. '" /></td>
    <td>R$ '. $preco. '</td>
    <td>R$ '. $sub. '</td>
    <td><a href="?acao=del&id='. $id. '">Remove</a></td>
</tr>';
}
?>
<div class="col-md-12">

<?php
if (isset($_GET['produto'])) {
    $query1 = "SELECT DISTINCT (PRODUTO_NOME)
    FROM SLM2015 WHERE REGRA LIKE '%" . $_GET['produto'] . "%'";
    $resultado = sqlsrv_query($conn, $query1) or die( print_r( sqlsrv_errors(),
    true));
}
}

```

```

while($row = sqlsrv_fetch_array($resultado, SQLSRV_FETCH_ASSOC)) {
?>
    <div class="col-md-3">
    <div class="panel panel-default">
    <div class="panel-body">
<?php
    $query2 = "SELECT B1_COD, B1_DESC, B1_FABRIC, B1_PRV1 FROM SB1010 AS B1
        WHERE B1_DESC = '".trim($row['PRODUTO_NOME'])."'
        GROUP BY B1_COD, B1_DESC, B1_FABRIC, B1_PRV1";
    $resultado2 = sqlsrv_query($conn, $query2) or die( print_r( sqlsrv_errors()
        true) ) ;
    $ln = sqlsrv_fetch_array($resultado2, SQLSRV_FETCH_ASSOC);
    echo '<h4>'. $ln['B1_COD'].'</h4> <br />';
    echo '<h4>'. $ln['B1_DESC'].'</h4> <br />';
    echo '<h4>'. $ln['B1_FABRIC'].'</h4> <br />';
    echo 'Preço : R$ '.number_format($ln['B1_PRV1'], 2, ',', '.').<br />';
    $produto = $ln['B1_DESC'];
    echo '<a
        href="/Monografia/carrinho.php?acao=add&id='.trim($ln['B1_COD']).'&produto='. $
        produto.'">Comprar</a>';
?>
    </div>
    </div>
    </div>
<?php
}
}
?>
</div>
<?php
    echo '<tr>
        <td colspan="6"><strong>Quantidade Total</strong></td>
        <td><strong>'.array_sum($guarda_qtd). ' Unidades</strong></td>
    </tr>';
    $total = number_format($total, 2, '.', '');
    echo '<tr>
        <td colspan="6"><strong>Total</strong></td>
        <td><strong>R$ '.$total.'</strong></td>
    </tr>';
}
?>

</tbody>
</form>
</table>
</div>
<div class="col-md-12">
<div id="encaminhar_pedido_entrega">
<input align="center" id="botao_1" class="btn btn-info" type="submit"
    value="Encaminhar Pedido">

</div>
</div>
<br>
<div id="finalizar_venda">
<input type="submit" id="botao_2" class="btn btn-info" value="Finalizar Venda">
>
</div>
</div>
</div>
</div>
<br>
<br>
</body>
</html>

```