

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

Tabela de conteúdos

PBI - Fase 2.....	2
Exportação conjuntos de dados	2
Transformação do CSV para ARFF	3
Utilização da API do Kettle.....	6
Howtos	9
JAVA_HOME em sistemas UNIX.....	9
Considerações adicionais.....	9
Software Data Integration em sistemas UNIX.	9
Considerações adicionais.....	10
Considerações MacOS:.....	10
Conexão à base de dados no data-integration	10
Registo de Trabalho	11

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

PBI - Fase 2

Exportação conjuntos de dados

Foi utilizado o software kettle referido na primeira fase, para gerar seis coleções de dados distintas com o objetivo de identificar regras de associação presentes nos dados. A imagem 1 demonstra a estrutura no kettle que realiza a query à base de dados Steelwheels e exporta para um ficheiro CSV o resultado.



Illustration 1: Estrutura Kettle para criação dos conjuntos de dados

Inicialmente foram elaboradas seis perguntas de negócio relevantes, uma para cada conjuntos de dados. Estas poderão ser alteradas futuramente, consoante os resultados obtidos nos próximos passos.

Algumas questões foram adaptadas ou removidas de modo a satisfazer as condições presentes na identificação das regras de associação. Segue-se uma listagem das algumas questões que não cumpriram estes requisitos.

- Vendedor de produtos que tem sucesso num dado país, consegue transmitir esse sucesso para os mesmos países desse território no ano 2005;
- Clientes com crédito limite elevado realizam compras com valores mais elevados na América do Norte;
- Relação entre quantidade em stock e soma do número de artigos em França para 2003;
- Cliente que compra grandes quantidades de “Classic Cars” compra menos quantidades “Trains”.

Segue-se as perguntas definidas tais como os campos seleccionados para fazer parte do ficheiro csv resultante:

1. Cliente que compra um produto também compra outro produto em 2004;

PRODUCTNAME	CUSTOMERNAME	YEAR_ID
-------------	--------------	---------

2. Cliente que compra uma linha de produtos também compra outra linha de produtos nos EUA;

PRODUCTLINE	CUSTOMERNUMBER	COUNTRY
-------------	----------------	---------

3. Clientes com crédito limite elevado realizam compras com valores mais elevados na América

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-
JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

do Norte;

- (a) Foram criados grupos de valores referentes ao BUYPRICE e ao CREDITLIMIT, nomeadamente <30, 30-100 e >100 para o BUYPRICE e <100k, 100k-200k e >200k para o CREDITLIMIT.

BUYPRICE	CUSTOMERNUMBER	CREDITLIMIT	TERRITORY
----------	----------------	-------------	-----------

4. Vendedor de produtos que realiza mais de 60 vendas num dado território, consegue transmitir esse sucesso para outros territórios;

PRODUCTVENDOR	TERRITORY	NMR_ORDERS
---------------	-----------	------------

5. Clientes com limite de crédito superior a 100.000 que compram uma linha de produtos também compram outra linha de produtos;

PRODUCTLINE	CUSTOMERNUMBER	CREDITLIMIT
-------------	----------------	-------------

6. Clientes que realizam mais de 5 compras numa dada linha de produtos realizam menos de 5 compras noutra linha de produtos.

- (a) Foram criados grupos de valores referentes ao NMR_ORDERS, nomeadamente <=5 e >5.

PRODUCTLINE	CUSTOMERNUMBER	NMR_ORDERS
-------------	----------------	------------

O grupo tentou criar uma amostra de perguntas de negócio diversificadas que demonstrem a flexibilidade e capacidades das ferramentas utilizadas. Futuramente estas serão testadas de forma a verificar se geraram regras de associação de sucesso.

Transformação do CSV para ARFF

Foi desenvolvido um script em Java que realiza uma pesquisa na diretoria que contém os ficheiros CSV, armazena o nome dos ficheiros num array e seguidamente converte-os para formato ARFF, com auxílio da Função ArffSaver().

Segue-se os blocos de código referentes ao main onde está a ser implementada uma console application, listagem de ficheiros csv, modificação dos ficheiros csv para a conversão e conversão para ARRF.

Classe CSVtoARFF

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

```
public static void ConvertFiles() {
    String folderPath =
"C:\\Users\\tadeu\\Desktop\\SAD2021\\SADG05\\PBI\\PBI2\\Current\\implementacao\\test_conv\\";

    ArrayList<String> csvFilesList = GetCSVFilesList(folderPath);
    for (int i = 0; i < csvFilesList.size(); i++){
        String fileName = csvFilesList.get(i);
        fileName = fileName.substring(0,fileName.length()-4);

        String filePath = folderPath + fileName;

        ConvertCSVtoARFF(filePath, fileName);
    }
}
```

```
private static ArrayList<String> GetCSVFilesList(String path) {
    File folder = new File(path);
    File[] listOfFiles = folder.listFiles();

    ArrayList<String> result = new ArrayList<String>();

    for(int i = 0; i < listOfFiles.length; i++){
        String fileName = listOfFiles[i].getName();
        String fileExtension = fileName.substring(fileName.length()-3);

        if( fileExtension.equals("csv")){
            result.add(fileName);
        }
    }

    return result;
}
```

```
private static void ConvertCSVtoARFF(String filePath, String fileName){

    try{
        ModifyCSVFile(filePath);

        //Load CSV
        CSVLoader loader = new CSVLoader();
        loader.setSource(new File(filePath+"_mod.csv"));
        Instances data = loader.getDataSet();

        // Save ARFF
        ArffSaver saver = new ArffSaver();

        // Save as ARFF
        saver.setFile(new
File("C:\\Users\\tadeu\\Desktop\\SAD2021\\SADG05\\PBI\\PBI2\\Current\\implementacao\\test_conv\\"+file
Name+".arff"));
        saver.setInstances(data);
        saver.writeBatch();
        System.out.println("SUCCESS: File Created -> "+fileName);
    }
    catch(IOException E){
        System.out.println("ERROR: The File "+ fileName +" cannot be converted");
    }
}
```

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

```
private static void ModifyCSVFile(String filePath) throws IOException {
    File csvFile = new File(filePath+".csv");

    try {

        BufferedWriter bw = new BufferedWriter(new FileWriter(filePath+"_mod.csv"));

        Scanner csvFileScanner = new Scanner(csvFile);
        csvFileScanner.useDelimiter("\n");

        while(csvFileScanner.hasNext()){
            String data = csvFileScanner.next();
            data = data.trim();
            data = data.replaceAll(" ","_");
            data = data.replaceAll("'", "_");
            bw.write(data);
            bw.newLine();
        }
        bw.close();
        csvFileScanner.close();
    } catch (FileNotFoundException ex) {
        //Se der erro
        System.out.println(ex);
    }
}
```

Classe WekaApriori

```
public static void RunAlgorithm() throws FileNotFoundException, IOException, Exception {

    String filePath =
"C:\\Users\\tadeu\\Desktop\\SAD2021\\SADG05\\PBI\\PBI2\\Current\\implementacao\\TASKDATA1.arff";

    // load data
    Instances data = new Instances(new BufferedReader(new FileReader(filePath)));

    System.out.println(data);
    // build model
    Apriori model = new Apriori();
    model.buildAssociations(data);
    System.out.println(model);

    FPGrowth fpgModel = new FPGrowth();
    fpgModel.buildAssociations(data);
    System.out.println(fpgModel);
}
```

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

Classe Main

```
public static void main(String[] args) throws Exception{
    Scanner scan = new Scanner(System.in);
    Boolean running = true;
    while(running){
        DisplayOption();
        int i = scan.nextInt();
        switch (i){
            case 1:
                CSVtoARFF.ConvertFiles();
                break;
            case 2:
                WekaApriori.RunAlgorithm();
                break;
            case 3:
                running = false;
                break;
            default:
                System.out.println("Keep runing");
        }
    }
}

private static void DisplayOption() {
    System.out.println("1- Convert Files");
    System.out.println("2- Apriori Algorithm");
    System.out.println("3- Exit");
}
```

Utilização da API do Kettle

Foi utilizada a API do Kettle de modo a executar as transformações desenvolvidas anteriormente através de código Java.

Primeiramente foi criado um projeto maven com um ficheiro pom.xml para importar as bibliotecas necessários com origem no repositório “<http://nexus.pentaho.org/content/groups/omni>”. Segue-se a listagem da sbibliotecas importadas.

- kettle-core
- commons-vfs
- kettle-engine
- hsqldb
- kettle-ui-swt
- mysql-connector-java

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

Os blocos de código que se seguem representam o ficheiro pom.xml e RunTransformation.java, respetivamente.

```
<dependencies>
  <dependency>
    <groupId>pentaho-kettle</groupId>
    <artifactId>kettle-core</artifactId>
    <version>7.1.0.32-246</version>
  </dependency>
  <dependency>
    <groupId>commons-vfs</groupId>
    <artifactId>commons-vfs</artifactId>
    <version>1.0</version>
  </dependency>
  <dependency>
    <groupId>pentaho-kettle</groupId>
    <artifactId>kettle-engine</artifactId>
    <version>7.1.0.32-246</version>
  </dependency>
  <dependency>
    <groupId>org.hsqldb</groupId>
    <artifactId>hsqldb</artifactId>
    <version>2.3.2</version>
  </dependency>
  <dependency>
    <groupId>pentaho-kettle</groupId>
    <artifactId>kettle-ui-swt</artifactId>
    <version>7.1.0.32-246</version>
  </dependency>
  <dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.23</version>
  </dependency>
</dependencies>
```

```
import org.pentaho.di.core.KettleEnvironment;
import org.pentaho.di.core.exception.KettleException;
import org.pentaho.di.core.logging.LogLevel;
import org.pentaho.di.trans.Trans;
import org.pentaho.di.trans.TransMeta;

public class RunTransformation {
  public static void main(String[] args) throws KettleException {

    KettleEnvironment.init();
    TransMeta transMeta = new TransMeta("../TASKDATA1.ktr");
    Trans trans = new Trans(transMeta);
    trans.setLogLevel(LogLevel.ERROR);
    trans.execute(null);
    trans.waitUntilFinished();

    if(trans.getErrors() > 0){
      System.out.println("Some errors occurred");
    }
  }
}
```

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

O projeto foi desenvolvido para a versão 7 do data integration, pois as dependências não ocorrem como esperado nas versões mais recentes.

De forma a resolver este problema, foi criada uma nova versão desta funcionalidade num projeto não maven. Desta forma importamos as dependências através da importação direta de ficheiros jar, nomeadamente todas os ficheiros da pasta lib no data-integration e os ficheiros jar da pasta plugins/pdi-core-plugins.

De seguida criamos uma pasta Files e adicionamos o ficheiro kettle-password-encoder-plugins.xml que se encontra na pasta classes. Na IDE Netbeans apenas necessitamos de adicionar esta pasta na source.

Relativamente à integração com a componente weka, esta foi feita através de uma nova classe acessível pelo menu de consola desenvolvido. Nesta realizamos um loop nos ficheiros ktr das transformações e executamo-os um a um.

```
import java.io.File;
import java.util.ArrayList;
import org.pentaho.di.core.KettleEnvironment;
import org.pentaho.di.core.exception.KettleException;
import org.pentaho.di.core.logging.LogLevel;
import org.pentaho.di.trans.Trans;
import org.pentaho.di.trans.TransMeta;

public class Transformation {
    static void RunTransformations() throws KettleException {
        File folder = new File("../");
        File[] listOfFiles = folder.listFiles();

        KettleEnvironment.init();

        for (File listOfFile : listOfFiles) {
            String fileName = listOfFile.getName();
            String fileExtension = fileName.substring(fileName.length()-3);
            if (fileExtension.equals("ktr")){
                TransMeta transMeta = new TransMeta("../" + fileName);
                Trans trans = new Trans(transMeta);
                trans.setLogLevel(LogLevel.ERROR);
                trans.execute(null);
                trans.waitUntilFinished();

                if(trans.getErrors() > 0){
                    System.out.println("Some errors occurred");
                }
            }
        }
    }
}
```


Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

Howtos

De forma a sistematizar os problemas encontrados e facilitar a execução de trabalho futuro usando as ferramentas do Pentaho, esta secção contém uma listagem de itens que, no momento da realização do trabalho, não constavam nos howtos online disponibilizados, de modo a que outros grupos possam usufruir destas novas soluções.

JAVA_HOME em sistemas UNIX.

1. Instalar JAVA 8 - algumas funcionalidades não funcionam em outras versões após consulta de documentação
 1. ***sudo apt install openjdk-8-jdk***
2. Descobrir caminho para a pasta JAVA
 1. ***dirname \$(dirname \$(readlink -f \$(which javac)))***
3. Adicionar o caminho anterior ao ambiente de desenvolvimento
 1. Editar o ficheiro /etc/profile
 1. Adicionar export **JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64**
 2. Reiniciar sessão do utilizador
4. Verificar sucesso da operação
 1. ***echo \$JAVA_HOME*** - deverá retornar o caminho detetado anteriormente
 2. ***java -version*** - deverá retornar a versão atualmente instalada do java

Considerações adicionais

1. Verificar a versão do java necessária para o software Pentaho aquando da instalação, pois a versão 8 poderá não ser a utilizada atualmente.
2. Como opção poderá definir o caminho da variável num ficheiro .bashrc ou .zprofile, dependendo do interpretador de comandos UNIX que esteja a utilizar.
3. Poderá ser necessário exportar a variável PATH
 1. ***export PATH=\${PATH}:\${JAVA_HOME}/bin***

Software Data Integration em sistemas UNIX.

1. Requisitos
 1. Java
 2. Variáveis de ambiente configuradas
2. Download do ficheiro zip
 1. ***<https://sourceforge.net/projects/pentaho/>***
3. Extrair ficheiros para uma pasta à sua escolha
4. Executar o ficheiro spoon.sh que se encontra dentro da pasta data-integration

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

1. ./spoon.sh

Considerações adicionais

1. Como opção poderá seguir o tutorial fornecido na documentação da plataforma Pentaho. O sistema pedirá que efetue o registo, mas na realidade pode fornecer dados aleatórios pois não é realizada uma verificação do email.

1. <https://www.hitachivantara.com/en-us/pdf/white-paper/pentaho-ce-installation-guide-on-linux-operating-system-whitepaper.pdf>

Considerações MacOS:

Ao tentar instalar o pentaho community edition, deparamo-nos com a informação que este ainda não estava disponível para MacOS, para tal teve-se de recorrer à versão enterprise.

Nota: A versão enterprise quando se desinstala e se volta a instalar efetua uma nova renovação de licença.

Pentaho from Hitachi Vantara Overview

End to end data integration and analytics platform

Pentaho tightly couples data integration with business analytics in a modern platform that brings together IT and business users to easily access, visualize and explore all data that impacts business results. Use it as a full suite or as individual components that are accessible on-premise in the cloud or on-the-go (mobile). Pentaho Kettle enables IT and developers to access and integrate data from any source, and deliver it to your business applications, all from within an intuitive and easy to use graphical tool.

Need help installing PDI? Access the installation guides for the following operations systems:

Windows: <https://www.hitachivantara.com/en-us/pdf/white-paper/pentaho-community-edition-installation-guide-for-windows-whitepaper.pdf>

Linux: <https://www.hitachivantara.com/en-us/pdf/white-paper/pentaho-ce-installation-guide-on-linux-operating-system-whitepaper.pdf>

Mac: Coming Soon

Figure 1: Mensagem suporte do Pentaho com a informação da ausência da versão community para MacOS.

Instalação plugin Saiku:

Nos sistemas MacOS tendo a versão enterprise instalada existe uma alteração na diretoria.

Sendo assim deve-se introduzir a pasta que está dentro do ficheiro zip (disponível [aqui](#)), na seguinte diretoria: **/Applications/Pentaho/server/pentaho-server/pentaho-solutions/system**, ficando assim **/Applications/Pentaho/server/pentaho-server/pentaho-solutions/system/saiku**. Os restantes passos podem ser seguidos no [Howto I](#).

Para iniciar o servidor local é executar o ficheiro **start.command**, disponível na diretoria **/Applications/Pentaho**, para desligar e salvar as alterações do servidor local é só executar o ficheiro **stop.command** disponível na mesma diretoria.

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

Conexão à base de dados no data-integration

1. *Ao configurar uma nova ligação à base de dados que contém a amostra Steelwheels a password da conexão poderá ser não só “pentaho_user”, mas também “password”.*

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

Registo de Trabalho

Global	Individual	Individual	Individual	
198,5	117,5	117,5	117,5	
# Participantes	Leonardo Abreu	José Freitas	João Franco	Descrição
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Alteração do java API para versao 9.1.0.0-324
1	O	X	O	Integração com o projeto java para implementação do weka
1	O	X	O	Integração com o projeto java para implementação do weka
1	O	X	O	Integração com o projeto java para implementação do weka
1	O	X	O	Processo iterativo para todos os ficheiros ktr
1	O	X	O	Processo iterativo para todos os ficheiros ktr
1	O	X	O	Processo iterativo para todos os ficheiros ktr
1	O	X	O	Escrita Relatório
1	O	X	O	Escrita Relatório
1	O	X	O	Escrita Relatório
1	O	O	X	Modificação das transformações no Kettle para a exportação ficheiro csv
1	O	O	X	Modificação das transformações no

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

				Kettle para a exportação ficheiro csv
1	O	O	X	Modificação das transformações no Kettle para a exportação ficheiro csv
1	O	O	X	Modificação das transformações no Kettle para a exportação ficheiro csv
1	O	O	X	Modificação das transformações no Kettle para a exportação ficheiro csv
1	O	O	X	Modificação das transformações no Kettle para a exportação ficheiro csv
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	O	O	X	Escrita do relatório
1	X	O	O	Estudo para o Miniteste 3
1	X	O	O	Estudo para o Miniteste 3
1	X	O	O	Estudo para o Miniteste 3
1	X	O	O	Estudo para o Miniteste 3
1	X	O	O	Modificação das transformações no Kettle para a exportação ficheiro csv
1	X	O	O	Modificação das transformações no Kettle para a exportação ficheiro csv
1	X	O	O	Modificação das transformações no Kettle para a exportação ficheiro csv
1	X	O	O	Exploração do possivel bug (que não era bug mas sim o formato do csv que estava incorreto)
1	X	O	O	Exploração do possivel bug (que não era bug mas sim o formato do csv que estava incorreto)
1	X	O	O	Exploração do possivel bug (que não era bug mas sim o formato do csv que estava incorreto)
1	X	O	O	Exploração do possivel bug (que não era bug mas sim o formato do csv que estava incorreto)
1	X	O	O	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	X	O	O	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão

Sistemas de Apoio à Decisão

SAD20202021-RT-N.07-GRUPO-N.A05-

JOAOFRANCO.e.JOSEFREITAS.e.LEONARDOABREU

				para arff
1	X	O	O	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	X	O	O	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	X	O	O	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	X	O	O	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff
1	X	O	O	Implementação do script em java para modificar os ficheiros csv para o formato correto para a conversão para arff