Ficha 3 - DC

João Nunes (A82300)
Luís Braga (A82088)

28/02/2020

Conteúdo

1 Exercício 1) Qual ou quais as diferenças entre uma base de dados, um datawarehouse e um dataset?

Uma base de dados é grupo organizado de informação, segundo uma estrutura específica. A esta, usualmente, são aplicadas técnicas de normalização de forma a não haver repetição de dados.

Um data warehouse é uma base de dados, que tende a ter grandes quantidades de informação, para além disto está normalmente desnormalizada e serve como arquivo.

Um dataset é uma amostra de uma base de dados ou de um warehouse.

2 Exercício 2) Quais são algumas das limitações do data mining e como podem ser ultrapassadas?

Algumas limitações do data mining podem estar relacionadas com os próprios dados a ser usados, uma vez que estes poderão estar desatualizados ou simplesmente desadequados o que irá afetar a qualidade dos resultados obtidos pela técnica de data mining. Como tal, é necessário ter em atenção os dados que estão a ser usados, podendo se aplicar técnicas, como por exemplo, de normalização de modo a melhorar a qualidade dos dados.

Existem também questões relacionadas com a privacidade dos dados, na medida em que podem estar a ser analisados dados pessoais, e como tal é necessário proteger a privacidade, por exemplo, dos clientes.

Outro problema que pode ser encontrado nesta área é que os dados relativos à conjuntura podem não ser suficientes para a representar e, posteriormente, analisar.

3 Exercício 3) Qual a diferença entre datawarehouse e data mart?

Um data mart é bastante similar a um data warehouse na medida em que se pode classificar como um subconjunto de um data warehouse mas orientado a uma linha de negócio tal como, por exemplo, a linha de serviço ao cliente de uma organização. Desta maneira no data mart define-se dados sumarizados e coletados com um fim específico para a análise de uma vertente do negócio da organização. Um data warehouse por sua vez possui os dados todos coletados e centralizados podendo depois se aplicar técnicas de data mining ou de análise de dados.

4 Exercício 4) Indique alguns constrangimentos éticos da utilização e aplicação do Data Mining.

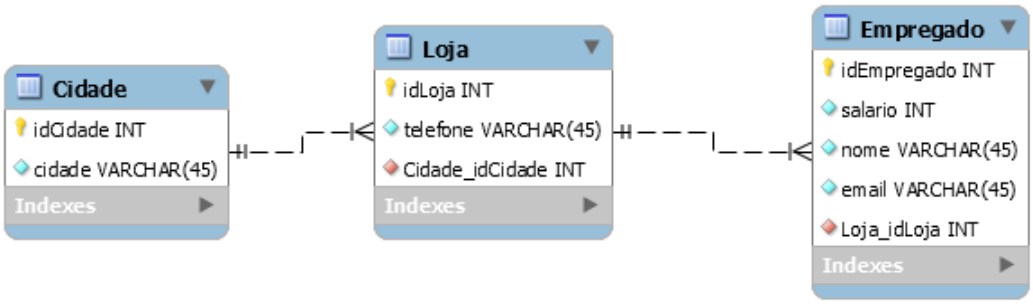
Na aplicação de técnicas de data mining, poderão ser analisados dados privados, ou seja, informação relativa a dados pessoais, como tal, é possível com esses dados efetuar previsões sobre esses mesmos, descobrindo até padrões que relacionam os mesmos dados. Desta maneira, é necessário assegurar a anonimização dos dados de modo a não ser possível identificar a pessoa em concreto.

5 Exercício 5) O que é a normalização de bases de dados e quais os impactos em sistemas OLTP e OLAP?

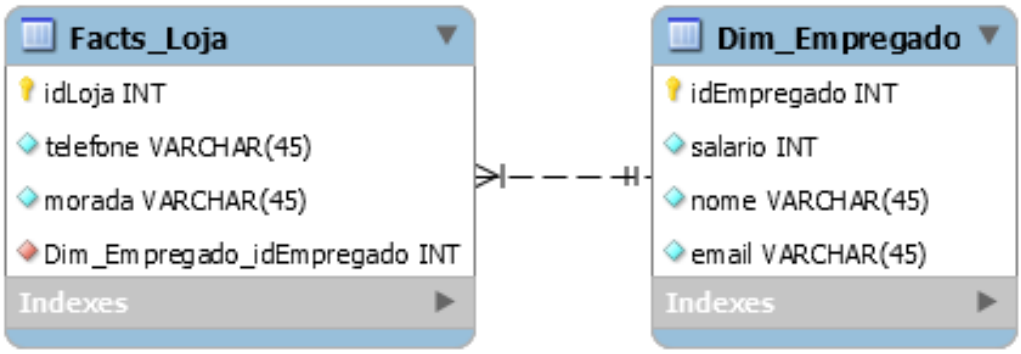
O OLTP (Online Transaction Processing) é uma categoria de software orientado ao suporte de aplicações orientadas à transacção, portanto o principal objectivo é o processamento de dados e não a sua análise.

Por sua vez o OLAP (Online Analytical Processing) é um categoria de software orientado à análise de dados e ao suporte de decisões de negócio (Business Inteligence), portanto o seu principal objetivo é a análise dos dados e não o seu processamento.

6 Exercício 6) Desenhe uma base de dados relacional com 3 tabelas. Garanta que cria o número de colunas adequadas para estabelecer relações entre as tabelas.



7 Exercício 7) Desenhe uma tabela datawarehouse com algumas colunas que normalmente seriam normalizadas. Explique porque faz sentido desnormalizar nesta situação.



8 Exercício 8) Faça uma pesquisa online e encontre 3 sites que contenham informação que pode ser aplicada ao processo de Data Mining.

https://opendata.socrata.com/

https://www.kaggle.com/datasets

http://archive.ics.uci.edu/ml/index.php

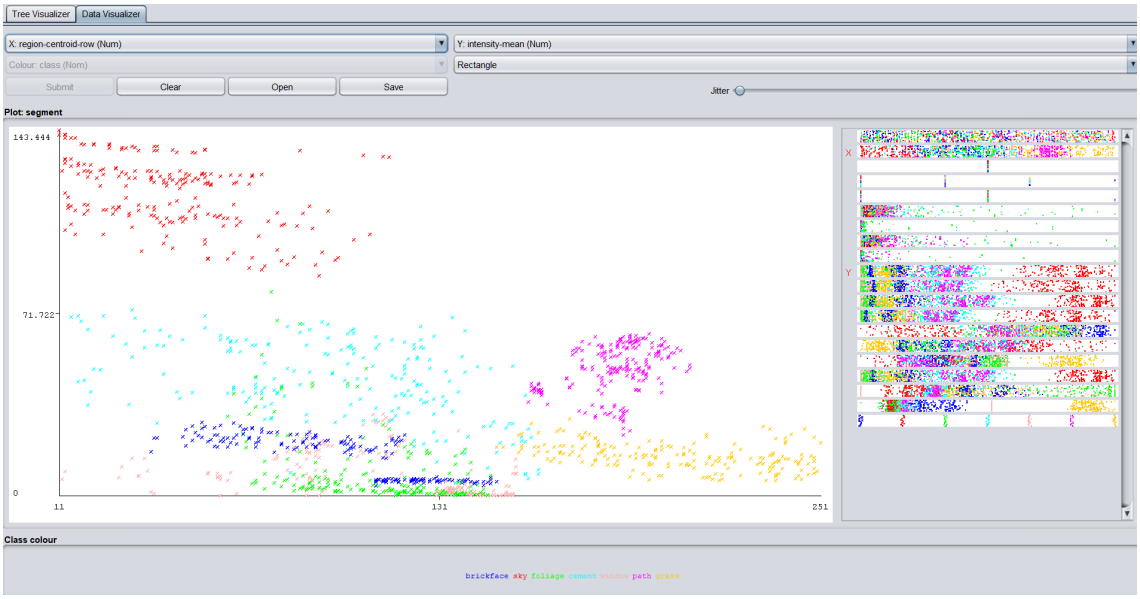
9 Exercício 9) Faça uma pequena pesquisa online e descubra um data set disponível para download. Descreva sucintamente o data set(conteúdo, propósito, tamanho, antiguidade).

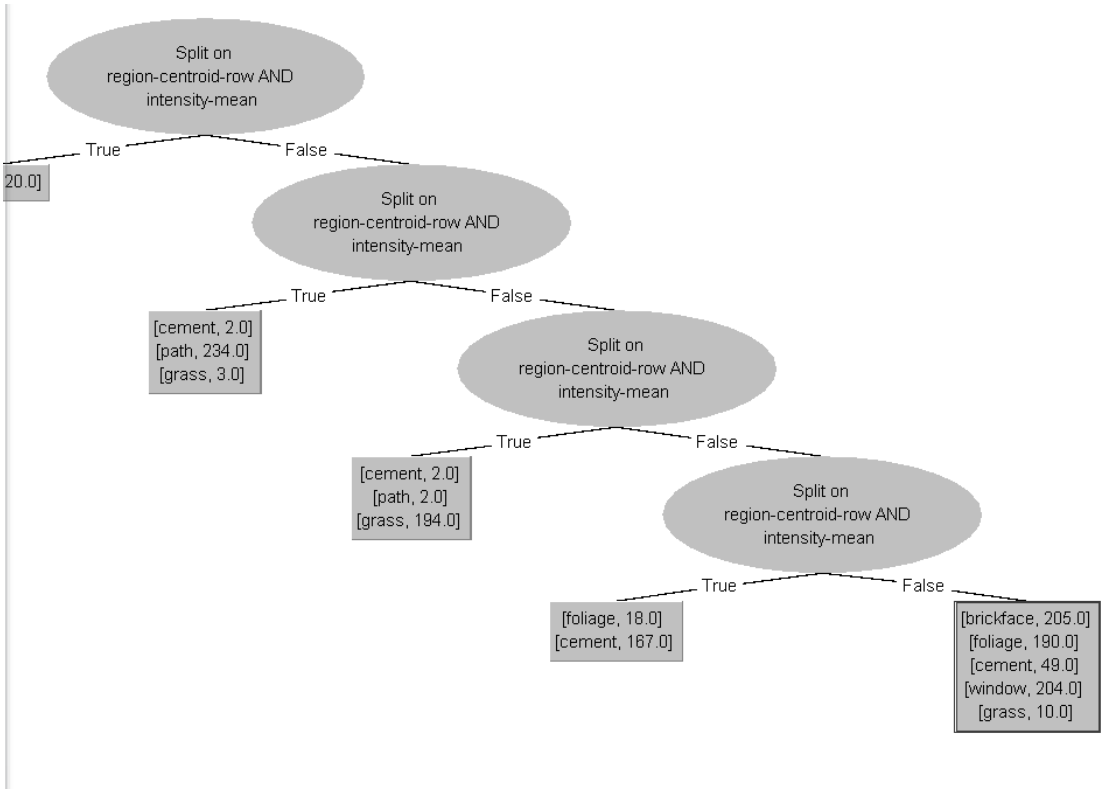
O dataset *Iris* disponibilizado no *UCI Machine Learning Repository* é um dos datasets mais utilizados para o reconhecimento de padrões, e possui informação relativa a três classes distintas de uma planta *iris* onde cada classe possui cerca de 50 instâncias distintas.

10 Exercício 10) Abrir o Weka / Explorer e carregar o data set "segment-challenge.arff". No separador Classify definir conjunto de dados"segmento-test.arff"como conjunto de teste.

10.1 a) Utilizar o trees -> UserClassifier;

Com o *UserClassifier* com os parâmetros disponibilizados, foi possível gerar a seguinte figura, onde é possível verificar a divisão por classes de cada um dos *targets*.





Dessa maneira é possível escolher cada uma das classes anteriores, adicionando folhas à árvore, de onde foi possível obter os resultados utilizando este método.

```
=== Summary ===

Correctly Classified Instances         537               66.2963 %
Incorrectly Classified Instances       273               33.7037 %
Kappa statistic                          0.6059
Mean absolute error                      0.1055
Root mean squared error                  0.2261
Relative absolute error                 43.0502 %
Root relative squared error             64.5518 %
Total Number of Instances              810

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    0,388    0,320      1,000    0,484      0,442    0,806     0,320     brickface
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     sky
                 0,000    0,000    ?          0,000    ?          ?        0,799     0,306     foliage
                 0,855    0,007    0,949      0,855    0,900      0,886    0,954     0,843     cement
                 0,000    0,000    ?          0,000    ?          ?        0,798     0,313     window
                 0,989    0,000    1,000      0,989    0,995      0,994    0,999     0,994     path
                 0,935    0,003    0,983      0,935    0,958      0,952    0,980     0,935     grass
Weighted Avg.    0,663    0,061    ?          0,663    ?          ?        0,899     0,652
```

Os resultados obtidos foram insatisfatórios, uma vez que não foi possível selecionar todas as classes para adicionar como folha à árvore, de onde se obteu apenas uma taxa de acerto de 66% e em cerca de 33% dos casos foi classificado com insucesso.

10.2 b) Comparar os resultados obtidos com este método de criação de árvore de decisão com os resultados do algoritmo J48.

O algoritmo de *J48* gerou bons resultados comparativamente ao *UserClassifier* no que toca a precisão de instâncias corretamente classificadas com cerca de 96.1728% e com 3.8272% de instâncias classificadas incorretamente, como se poderá verificar na seguinte figura.

```
=== Summary ===

Correctly Classified Instances         779               96.1728 %
Incorrectly Classified Instances        31                3.8272 %
Kappa statistic                          0.9553
Mean absolute error                      0.0127
Root mean squared error                  0.1005
Relative absolute error                  5.1771 %
Root relative squared error             28.6807 %
Total Number of Instances              810

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0,992     0,004     0,976       0,992   0,984      0,981    0,994     0,970     brickface
               1,000     0,000     1,000       1,000   1,000      1,000    1,000     1,000     sky
               0,975     0,019     0,902       0,975   0,937      0,926    0,986     0,926     foliage
               0,973     0,010     0,939       0,973   0,955      0,948    0,986     0,932     cement
               0,833     0,007     0,955       0,833   0,890      0,874    0,946     0,892     window
               1,000     0,003     0,979       1,000   0,989      0,988    0,999     0,979     path
               0,976     0,001     0,992       0,976   0,984      0,981    0,987     0,971     grass
Weighted Avg.  0,962     0,007     0,962       0,962   0,961      0,955    0,985     0,951
```

11 Execício 11) Abrir o Weka/Explorer e carregar o data set "segment-challenge.arff". Com este data set carregado responda às seguintes questões.

11.1 a) Usar o algoritmo J48 como classificador; Usar o data set "segment-test.arff" como conjunto teste. Qual o valor da classificação?

O algoritmo de J48 classifica corretamente em cerca de 96% dos casos e incorretamente em 4% dos casos.

```
=== Summary ===

Correctly Classified Instances         779               96.1728 %
Incorrectly Classified Instances        31                3.8272 %
```

11.2 b) Usando a opção "Use training set" determine o valor da classificação. Porque não deve ser usada esta opção para determinar a qualidade e a aplicabilidade dos algoritmos aos dados?

Usando apenas o *training set* não é possível produzir resultados coerentes, uma vez que se está a utilizar os próprios dados para avaliar o algoritmo produzido o que poderá incorrer em situações de *over fitting*.

```
=== Summary ===

Correctly Classified Instances        1485               99      %
Incorrectly Classified Instances        15                1      %
```

11.3 c) Escolha o J48 como classificador e vá alterando as percentagens de divisão (“Percentage Split”) dos grupos de treino e de teste em: 10%, 20%, 40%, 60% e 80%. O que observa?

Aumentando o "percentage split"é possível verificar que a percentagem de instâncias correctamente classificadas aumenta. Resultando numa taxa de acerto de cerca de 89% com "percentage split"de 10% e uma taxa de acerto de 97% com "percentage split"de 80%, aumentando sempre nos casos intermédios.

```
=== Summary ===

Correctly Classified Instances         290               96.6667 %
Incorrectly Classified Instances        10                3.3333 %
```

11.4 d) Repetir a questão anterior usando 90%, 95%, 98% e 99%. O que acontece ao número de instâncias corretamente classificadas? E o que acontece à percentagem de instâncias corretamente classificadas? Explicar esta variação.

Relativamente ao número de instâncias totais correctamente classificadas podemos referir que diminui. No entanto, a percentagem de instâncias correctamente classificadas mantem-se constante perto do 100%.

O "percentage split"define-se como sendo a divisão do dataset entre os dados de treino e teste. Por exemplo, num "percentage split"de 90% significa que se dividiu o dataset em 90% dos dados para treino e 10% para teste. Portanto, os valores da precisão são enganadores, uma vez que estamos a testar basicamente sobre os dados com os quais já se treinou.

11.5 e) Apesar de com uma percentagem de 98% para o treino e 2% para o teste dar uma classificação de 100% isto quer dizer que o modelo construído é o mais indicado para o problema apresentado?

Não quer dizer que o modelo construído é o mais indicado, muito pelo contrário uma vez que dessa maneira estamos a dividir o dataset em 98% dos dados para o treino e apenas 2% para o teste, o que efetivamente significa que estamos a testar sobre os dados já treinados, uma vez que com essas percentagens já é conhecido todo o dataset, não sendo necessário efetuar quase nenhuma previsão uma vez que os dados já são conhecidos.

11.6 f) Com base nas experiências acima, qual considera a melhor estimativa da verdadeira precisão de J48 para este data set?

A melhor precisão do *J48* para este dataset deverá rondar os 94% de taxa de acerto, tal como é obtido, por exemplo, para uma divisão mais justa do dataset como 60% para o treino e 40% para o teste.

12 Exercício 13) Abrir o Weka/Explorer e carregar o data set “diabetes.arff”. Com este data set carregado responda às seguintes questões.

12.1 a) Selecionando “Percentage Split” a 80% quantas instâncias serão usadas para treino e quantas serão usadas para teste? (O Weka arredonda ao número inteiro mais próximo).

O dataset anterior possui cerca de 768 instâncias, como tal, usando um "percentage split"de 80%, o dataset é dividido em 80% para teste, ou seja, $768 * 0.8 = 614$ instâncias para treino e $768 * 0.2 = 154$ instâncias para teste.

12.2 b) Mudando o “Random seed” entre 1,2,3,4 e 5, mantendo o “Percentage Split” a 80% indique o valor mínimo e máximo de instâncias incorretamente classificadas.

Utilizando novamente o algoritmo *J48*, com uma "random seed"de 4 foi possível obter o valor mínimo de instâncias incorretamente classificadas 20.13% dos casos incorretamente avaliados. O valor máximo dos casos incorretamente classificados foi com "random seed"de 5 onde 28.57% dos casos foram incorretamente avaliados.

```
=== Summary ===

Correctly Classified Instances         110               71.4286 %
Incorrectly Classified Instances        44               28.5714 %
```

12.3 c) Qual a média da percentagem de instâncias corretamente classificadas?

A média da percentagem de instâncias corretamente classificadas, poderá ser consultada através do valor da precisão, como tal a precisão do modelo é de 73.2%.

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0,740    0,340    0,819      0,740   0,778      0,384    0,681     0,772     tested_negative
                 0,660    0,260    0,550      0,660   0,600      0,384    0,681     0,511     tested_positive
Weighted Avg.    0,714    0,314    0,732      0,714   0,720      0,384    0,681     0,687
```

12.4 d) Se repetisse o exercício [13/b] com 10 “random seed” em vez de 5 qual seria o efeito na média?

A precisão do modelo com uma "random seed"de 10 aumenta ainda mais a precisão para cerca de 74.5%.

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0,873    0,481    0,781      0,873   0,824      0,423    0,716     0,796     tested_negative
                 0,519    0,127    0,675      0,519   0,587      0,423    0,716     0,516     tested_positive
Weighted Avg.    0,753    0,361    0,745      0,753   0,744      0,423    0,716     0,701
```

13 Exercício 14) Abrir o Weka/Explorer e carregar o data set “iris.arff”. Com este data set carregado responda às seguintes questões.

13.1 a) Este data set caracteriza 3 classes com 50 instâncias cada uma. Qual será a percentagem de acerto do algoritmo ZeroR quando aplicado ao training set?

Usando o "training set"a percentagem de acerto do algoritmo anterior é de apenas 33.33%.

```
=== Summary ===

Correctly Classified Instances          50               33.3333 %
Incorrectly Classified Instances       100               66.6667 %
```

13.2 b) Qual é o resultado da classificação base line quando é usado o método “Percentage Split” em 66%?

O resultado da classificação com um "percentage split"de 66% resulta numa percentagem de instâncias corretamente classificadas baixa de 29.41%, sendo que classificou incorretamente 70.49% das instâncias.

```
=== Summary ===

Correctly Classified Instances          15               29.4118 %
Incorrectly Classified Instances        36               70.5882 %
```

14 Exercício 15) Abrir o Weka/Explorer e carregar o data set “glass.arff”. Com este data set carregado responda às seguintes questões.

14.1 a) Qual é a percentagem de acerto do algoritmo ZeroR com 66% de “Percentage Split”?

Com o modelo *ZeroR* o algoritmo uma percentagem de acerto de 27.397%.

14.2 b) Qual o valor usando o J48 e os restantes parâmetros por defeito?

Utilizando o algoritmo *J48* com os parâmetros standard e com um "percentage split"de 66% foi possível obter uma taxa de acerto de 57.53%.

```
=== Summary ===

Correctly Classified Instances          42               57.5342 %
Incorrectly Classified Instances        31               42.4658 %
```

14.3 c) Qual a precisão (accuracy) do algoritmo NaiveBayes’ usando os parâmetros por defeito?

Com o algoritmo *NaiveBayes* com os parâmetros standard e mais uma vez um "percentage split"de 66% é possível obter uma precisão de 0.589.

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0,400    0,189    0,444      0,400    0,421      0,219    0,730     0,432     build wind float
               0,375    0,146    0,667      0,375    0,480      0,263    0,707     0,649     build wind non-float
               0,250    0,188    0,071      0,250    0,111      0,036    0,569     0,090     vehic wind float
               ?        0,000    ?          ?        ?          ?        ?         ?         vehic wind non-float
               0,750    0,058    0,429      0,750    0,545      0,535    0,951     0,448     containers
               1,000    0,042    0,400      1,000    0,571      0,619    0,986     0,583     tableware
               0,909    0,016    0,909      0,909    0,909      0,893    0,956     0,926     headlamps
Weighted Avg.  0,493    0,133    0,589      0,493    0,514      0,358    0,764     0,588
```

15 Exercício 16) Abrir o Weka/Explorer e carregar o data set “segment-challenge.arff”. Utilize o data set “segment-test.arff” para dataset de avaliação (teste). Com estes data sets carregados responda às seguintes questões.

15.1 a) Qual a precisão do algoritmo ZeroR?

A precisão do algoritmo anterior é de cerca de 11.6%.

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0,000    0,000    ?          0,000   ?          ?        0,500     0,154     brickface
                 0,000    0,000    ?          0,000   ?          ?        0,500     0,136     sky
                 0,000    0,000    ?          0,000   ?          ?        0,500     0,151     foliage
                 0,000    0,000    ?          0,000   ?          ?        0,500     0,136     cement
                 0,000    0,000    ?          0,000   ?          ?        0,500     0,156     window
                 1,000    1,000    0,116      1,000   0,208      ?        0,500     0,116     path
                 0,000    0,000    ?          0,000   ?          ?        0,500     0,152     grass
Weighted Avg.    0,116    0,116    ?          0,116   ?          ?        0,500     0,144
```

15.2 b) Qual a precisão do algoritmo IBk's, com todos os parâmetros por defeito?

A precisão do algoritmo IBk é de 95.8%, como se pode verificar no seguinte relatório.

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0,992    0,004    0,976      0,992   0,984      0,981    0,994     0,965     brickface
                 1,000    0,000    1,000      1,000   1,000      1,000    1,000     1,000     sky
                 0,951    0,012    0,935      0,951   0,943      0,933    0,968     0,900     foliage
                 0,936    0,013    0,920      0,936   0,928      0,917    0,963     0,864     cement
                 0,865    0,019    0,893      0,865   0,879      0,857    0,916     0,793     window
                 1,000    0,001    0,989      1,000   0,995      0,994    0,999     0,990     path
                 0,976    0,000    1,000      0,976   0,988      0,986    0,989     0,980     grass
Weighted Avg.    0,958    0,007    0,958      0,958   0,958      0,951    0,974     0,925
```

15.3 c) Qual a precisão do algoritmo PART, com todos os parâmetros por defeito?

O algoritmo anterior com os parâmetros por defeito possuirá uma precisão de 95.6%.

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0,968    0,009    0,953      0,968   0,960      0,953    0,982     0,956     brickface
                 1,000    0,000    1,000      1,000   1,000      1,000    1,000     1,000     sky
                 0,943    0,010    0,943      0,943   0,943      0,932    0,980     0,918     foliage
                 0,945    0,014    0,912      0,945   0,929      0,917    0,969     0,918     cement
                 0,881    0,013    0,925      0,881   0,902      0,885    0,948     0,878     window
                 0,989    0,006    0,959      0,989   0,974      0,970    0,993     0,970     path
                 0,976    0,000    1,000      0,976   0,988      0,986    0,988     0,979     grass
Weighted Avg.    0,956    0,008    0,956      0,956   0,955      0,948    0,979     0,944
```