Universidade do Minho
Mestrado em Engenharia Informática
Métricas e Machine Learning

T1- Dissemelhanças com Dados Binários

Lucas Mello - PG40158
Diogo Lopes - PG42823
Fábio Gonçalves - PG42827
Joel Carvalho - PG42837
Pedro Ribeiro - PG42848
Tiago Gonçalves - PG42851

Mestrado em Engenharia Informática
2021

Capítulo 1

Introdução

No âmbito da Unidade Curricular de Métricas em Machine Learning, no qual está enquadrado este Micro-Projeto, foi elaborado um estudo de métricas tendo por base as Bases de Dados Binárias, pois este é o principal objetivo do desenvolvimento do mesmo. Ao longo deste documento são apresentadas diversas métricas de Similaridade e Distância (Dissimilaridade), uma vez que são essenciais para resolver os diversos problemas de reconhecimento de padrões, como classificação e agrupamento.

Genericamente uma Base de Dados Binária é composta apenas por 0 e 1, representando a presença ou não de um determinado atributo de um certo objeto. Contextualizando, no caso da saúde, os tipos de dados binários podem indicar se uma pessoa possui ou não uma doença crónica, na química, pode determinar se um objeto possui ou não determinado composto, na psicologia, pode determinar se a pessoa possui ou não determinado traço psicológico.

Tendo em conta que o desempenho de um modelo de análise de Distância Binária depende da escolha apropriada de uma determinada medida, com este Projeto pretendemos apresentar uma análise crítica e uma comparação fundamentada em exemplos, sobre de 4 Métricas de Similaridade e 4 Métricas de Distância.

1.1 Objetivos

Para conseguir cumprir com tudo o que está estipulado para este Micro-Projeto, ao longo deste documento recolhemos diversas informações que nos permitiram fundamentar e apresentar a nossa opinião sobre os tópicos seguintes:

- Definição de uma Estrutura Binária
- Métricas de Similaridade associadas a dados binários;
- Métricas de Distância associadas a dados binários;
- Análise Crítica, Estudos e Implementação Prática;
- Aplicações das Métricas abordadas.

Capítulo 2

Métricas para Dados Binários

Sobre os dados binários, estes estão essencialmente relacionados nas três situações seguintes:

- Se uma ação é possível ou não;
- Se um objeto está ou não presente;
- Se uma informação é ou não verdadeira.

Devido à diversidade, de seguida são apresentadas diversas representações de dados binários.

2.1 Definição de uma Estrutura Binária

Considerando o seguinte problema: Um ambiente ecológico é caracterizado por várias `espécies` de gramíneas onde é reportado o resultado na tabela 2.1 onde: $(a_1, a_2, ..., a_n)$, em que `n` representa o numero de `espécies` de gramíneas. O espaço dos atributos é $\mathcal{A} = \{0, 1\}^n$ e $x = (x_1, x_2, ..., x_n)$,$y = (y_1, y_2, ..., y_n)$ onde $x_i, y_i \in \mathcal{A}$. Pode-se observar que os `n` espécies de gramíneas em conjuntos simples, em que é reunido em dois subconjuntos (x e y) com as gramíneas que pertencem aos ambientes ecológicos.

n	$espcie_1$ a_1	$espcie_2$ a_2	$espcie_3$ a_3	$espcie_4$ a_4	....	$espcie_n$ a_n
x	1	1	0	0	...	1
y	1	0	1	0	...	1

Tabela 2.1: Tabela de dados de sítio ecológico

Os valores 1 e 0 representam, respectivamente, a presença ou ausência de um atributo. Em termos linguísticos, podem caracterizar `Verdadeiro` ou `Falso`, `Sim` ou `Não` e `Tem` ou `Não tem`.

Para que seja possível a aplicação das métricas de semelhanças e dissemelhanças é essencial que à posteriori seja apresentada uma ferramenta para extrair os dados da nossa tabela 2.1, e subsequentemente a devolução em um formato que possa ser utilizado corretamente, desta forma, elaborando uma matriz de confusão. A `matriz de confusão` tem o objetivo

de contabilizar o número total de iterações de cada situação possível entre 0 e 1, e se tratando de dados binários é fácil perceber que existem apenas 4 situações em que os dados se divergem, por consequência foram definidos como a,b,c e d.

Foi desenvolvida uma matriz de confusão 2x2 entre dois vetores $x, y \in \mathcal{A}$ de tal forma:

- **a - 'Correspondências positivas'**: Número total de iterações onde os valores de x_i e y_i são ambos 1 (ou presença).
  - $a = |\{i \in 1, ..., N; x_i = 1 \wedge y_i = 1\}| = \sum_{i=1}^{N} (x_i)(y_i);$
- **b - 'Ausência de incompatibilidades'**: Número total de iterações onde os valores de $x_i = 1$ e $y_i = 0$, o que significa 'y_i ausência de incompatibilidades'.
  - $b = |\{i \in 1, ..., N; x_i = 1 \wedge y_i = 0\}| = \sum_{i=1}^{N} (x_i)(1 - y_i);$
- **c - 'Ausência de incompatibilidades'**: Número total de iterações onde os valores de $x_i = 0$ e $y_i = 1$, o que significa 'x_i ausência de incompatibilidades'.
  - $c = |\{i \in 1, ..., N; x_i = 0 \wedge y_i = 1\}| = \sum_{i=1}^{N} (1 - x_i)(y_i);$
- **d - 'Correspondências negativas'**: Número total de iterações onde os valores de x_i = 0 e y_i = 0 (ou ausência).
  - $d = |\{i \in 1, ..., N; x_i = 0 \wedge y_i = 0\}| = \sum_{i=1}^{N} (1 - x_i)(1 - y_i);$

x/y	1	0
1	a	c
0	b	d

Tabela 2.2: Matriz de Confusão

Com a matriz de confusão exemplificada foi possível produzir várias métricas de semelhanças e dissemelhanças combinando os seus coeficientes, estas métricas serão apresentadas nos sub-capítulos seguintes. É importante notar que a matriz não é simétrica e $b(x, y) = c(y, x)$ enquanto $d(x, y) = d(y, x)$ e $a(x, y) = a(y, x)$.

2.2 Métricas associadas a Dados Binários

Existe uma panóplia de medidas desenvolvidas com o propósito de trabalhar com os dados binários. Estas encontram-se essencialmente divididas em medidas similaridade e de distância, representadas por um S e um D, respetivamente. Com o objetivo de efetuar um pequeno estudo sobre as cada uma delas, de seguida encontram-se discriminadas 8 medidas (4 de similaridade e 4 de distância).

2.2.1 Similaridade Binária

De modo a exemplificar como é que é caracterizada uma similaridade binária, seja $\mathcal{A} = \{0,1\}^4$ o nosso conjunto de atributos, para $x = (x_1, x_2, x_3, x_4)$ e $y = (y_1, y_2, y_3, y_4)$, definimos a similaridade da seguinte forma:

$$x, y \in \mathcal{A}$$
$$x, y \Rightarrow s(x, y) \in \mathbb{R}$$

Em particular, consideramos as semelhanças aditivas por:

- $s_i(x, y) = 1$ se $x_i = y_i, i = 1, 2, 3, 4$
- $s(x, y) = \frac{1}{4} \sum_{i=1}^{4} s_i(x, y)$;

A função s_i é uma função de semelhança parcial que envolve apenas o componente i, enquanto que a função s é a função de semelhança total. Por construção temos as seguintes propriedades satisfeitas:

- $s(x, y) \in [0, 1]$
- Simetria $s(x, y) = s(y, x)$
- Normalização $s(x, x) = 1$
- *Definiteness* $s(x, y) = 1 \Rightarrow x = y$

De modo a ilustrar alguns dos possíveis cenários, de seguida apresentamos 3 exemplos, nos quais dois vetores serão distintos, parcialmente semelhantes e idênticos, respetivamente.

Para o primeiro exemplo seja $x = (1, 0, 0, 1)$, $y = (0, 1, 0, 0)$. Temos a similaridade dada por $s(x, y) = \frac{1}{4} * 0 = 0$, tendo este um grau de semelhança inexistente, já que x e y não possuem valores iguais correspondentes em nenhuma posição do vetor.

Para o segundo exemplo seja $x = (1, 0, 0, 1)$, $y = (1, 1, 0, 1)$. Temos a similaridade dada por $s(x, y) = \frac{1}{4} * 2 = 0.50$, tendo este um grau de semelhança parcial, onde x e y tem valores iguais a 1 na primeira e última posição do vetor.

Para o terceiro exemplo seja $x = (1, 1, 1, 1)$, $y = (1, 1, 1, 1)$. Temos a similaridade dada por $s(x, y) = \frac{1}{4} * 4 = 1$, tendo este um grau de semelhança total, onde x e y tem valores iguais em todas as posições do vetor, ou seja são iguais pode definição.

De seguida vamos definir 4 tipos de semelhanças usando os valores de uma tabela confusão, o `a, b, c` e `d`. Para exemplificar cada tipo de semelhança vamos recorrer ao exemplo apresentado na Tabela 2.3. Como se tratam de atributos não exclusivos, temos que: `a = 1, b = 1, c = 2 e d = 1`.

Objetos	Esfera	Doce	> 8cm	Crocante	Pesado
x = Maçã	1	1	0	1	0
y = Banana	0	1	1	0	0

Tabela 2.3: Exemplo geral para o cálculo da similaridade

Similaridade de Sokal-Michener

Esta medida de similaridade, também designada por "Simple matching", consiste na proporção de correspondências com o número total de valores. O peso é atribuído de igual forma a correspondências e não correspondências. Esta medida é bastante útil quando os valores positivos e negativos carregam informação simétrica/igual, como o género (masculino e feminino).

$$S_{SM} = \frac{a+d}{a+b+c+d} = \frac{atributos(correspondentes)}{atributos(total)}$$

Propriedades:

$S_{SM}(x,y) = 1 \Leftrightarrow x = y$ em particular temos que $S_{SM}(x,x) = 1$.

Como $b+c$, automaticamente esta propriedade é independente da ordem de (x,y) então faz com que a semelhança seja simétrica sabendo que $S_{SM}(x_i,y_i) = S_{SM}(y_i,x_i)$, uma vez que ao adicionar FP (Falsos Positivos) e FN (Falsos Negativos) estamos a considerar que são iguais.

Exemplificando:

Através da aplicação de Sokal-Michener, concluímos que a similaridade entre uma maçã e uma banana é igual a 0.4.

Similaridade de Jaccard

O quociente de $Jaccard(x,y)$ apenas deve ser utilizado quando se tratam de atributos que são exclusivos entre si. Temos $d(x,y) < I$, logo $a+b+c = I-d > 0$.

Analisando a fórmula é fácil de perceber que, para a nossa similaridade ser igual a 1 (valor máximo), todos os valores de x_i e y_i tem que ser iguais a 1, ou seja $a = I$ e $b, c = 0$. Para a similaridade ser igual a 0, somente é necessário $a = 0$.

Relacionando-a com a S_{SM}, a S_{SM} caso `d = 0`, este valor não é contabilizado para variar a distância entre 2 objetos, apenas é valorizado quando os objetos são presentes.

$$S_{Jaccard} = \frac{a}{a+b+c}$$

Propriedades:

Da mesma forma que S_{SM}, esta semelhança também contém $b+c$, então é simétrica sabendo que $S_{Jaccard}(x_i,y_i) = S_{Jaccard}(y_i,x_i)$.

Exemplificando:

Através da aplicação da Similaridade de Jaccard, concluímos que a similaridade entre uma maçã e uma banana é igual a 0.25.

Similaridade de Dice

Analisando a fórmula vemos que é semelhante a Jaccard, porém multiplicamos o valor de a por 2, e com isto estamos a duplicar a importância dos TP (True Positives). Genericamente se o valor em frente do a seja > 1, significa que havendo TP damos-lhe muita importância, caso o valor de a seja pequeno, como 0.1, o b, c vão continuar a ter preponderância, sendo assim para obtermos um valor próximo de 1 é necessário que haja bastantes TP. Comparativamente com Jaccard, quando $a \neq 0$, Dice dá mais peso aos casos positivos, com isto Dice

irá crescer mais rapidamente que Jaccard. Em termos textuais, Dice diz que 2 eventos são muito próximos, enquanto que Jaccard diz que os eventos não são muito próximos.

$$S_{Dice} = \frac{2a}{2a + b + c}$$

Exemplificando:

Através da aplicação da Similaridade de Dice, concluímos que a similaridade entre uma maçã e uma banana é igual a 0.4.

Similaridade de Russel and Rao

Esta consiste numa mistura entre Jaccard e Sokal-Michener, onde o numerador é igual à primeira e o denominador igual à segunda. O peso é atribuído de igual forma a correspondências e não correspondências.

$$S_{Russel_Rao} = \frac{a}{a + b + c + d}$$

Exemplificando:

Através da aplicação de Russel and Rao, concluímos que a similaridade entre uma maçã e uma banana é igual a 0.2.

Ao contrário da métrica de Jaccard, caso $d = 1$, ou seja todos valores de x_i e y_i são iguais a 0, o valor é indeterminado, na métrica de Russel-Rao não existe valores indeterminados pela presença do atributo d na fórmula.

Comparação entre os valores finais das diferentes métricas

Observando a Tabela 2.4, podemos concluir, através das métricas abordadas, que a semelhança entre uma maçã e uma banana é relativamente baixa. Contudo, as similaridades de Sokal-Michener e Dice, contém o mesmo valor, isto deve-se ao facto do de Dice conter o dobro das correspondências positivas, como $a = 1$ e $d = 1$ então nesse caso em particular $2a = a + d$.

Sokal-Michener	Jaccard	Dice	Russel and Rao
0.4	0.25	0.4	0.2

Tabela 2.4: Comparação entre os valores finais das diferentes similaridades

2.2.2 Distância Binária

Para exemplificar o que é caracterizado por distância binária utilizamos o mesmo exemplo utilizado anteriormente, sendo $\mathcal{A} = \{0,1\}^4$ o nosso conjunto de atributos. Para $x = (x_1, x_2, x_3, x_4)$ e $y = (y_1, y_2, y_3, y_4)$ definimos a distância por:

$$\forall x, y \in \mathcal{A}$$
$$d(x,y) \to \left[0, +\infty\right]$$

Em particular consideramos distância aditivas por:

- $d_i(x,y) = 0$ se $x_i = y_i, i = 1,2,3,4$
- $d(x,y) = 1 - \sum_{i=1}^{4} \frac{d_i(x,y)}{n}$;

A função d_i é uma função de distância parcial que envolve apenas o componente i, enquanto a função d é a função de distância total. Por construção temos as seguintes propriedades satisfeitas:

- $d(x,y) \in \left[0, +\infty\right]$
- Simetria $d(x,y) = d(y,x)$
- *Definiteness* $d(x,y) = 0 \to x = y$

Para ilustrar alguns dos possíveis cenários de se alcançar com a distância tomaremos três exemplos, no qual 2 vetores serão distintos, parcialmente distantes e com distância nula (idênticos) respetivamente, também tomaremos o valor de distância como $[0,1]$ para simplificar o entendimento .

Para o primeiro exemplo seja $x = (1,0,0,1)$, $y = (0,1,0,0)$. Temos a distância dada por $d(x,y) = 1 - \frac{0}{4} = 1$, tendo este o maior valor de distância que podemos obter com a fórmula aditiva, o que representa que os vetores são distintos já que x e y não possuem valores iguais correspondentes em nenhuma posição do vetor.

Para o segundo exemplo seja $x = (1,0,0,1)$, $y = (1,1,0,1)$. Temos a distância dada por $d(x,y) = 1 - \frac{2}{4} = 0.5$, tendo este valor de distância "média"onde x e y tem valores iguais a 1 na primeira e última posição do vetor.

Para o terceiro exemplo seja $x = (1,1,1,1)$, $y = (1,1,1,1)$. Temos a distância dada por $d(x,y) = 1 - \frac{4}{4} = 0$, tendo este a menor distância que podemos alcançar com a fórmula aditiva, que representa que x e y possuem valores iguais em todas as posições do vetor, ou seja são iguais por definição.

Quanto mais próximas forem as amostras, i.é.,quanto menor a distância métrica entre os pontos representativos dessas duas amostras, maior será a similaridade entre eles.

Distância Sokal-Michener

Esta distância simétrica é bastante usada, onde é dada a mesma importância aos Falsos Positivos e Falsos Negativos, contudo nas aplicações isso pode não acontecer.

$$D_{SM} = 1 - S_{SM} = \frac{(b+c)}{n} = \left[0 - 1\right]$$

Propriedades:
Baseado no estudo da S_{SM} mostramos que caso a $D_{SM} = 0$ então a $S_{SM} = 1$.

$$\frac{1-(a+b)}{n} porque \frac{n}{n} = 1$$

Em casos aplicacionais poderá haver a necessidade de atribuir pesos aos 2 tipos de erros, isso será capaz de ter impacto na segurança, por exemplo. De seguida exemplificamos onde atribuímos maior importância aos Falsos Negativos.

$$D_{(X,Y)} = \frac{(b*1+c*10)}{n}$$

Com esta atribuição de pesos garantimos que não há simetria: $D(X,Y) <> D(Y,X)$, ou seja, ao atribuir primeiramente o peso de 1 ao b e 10 ao c, o seu resultado será diferente quando atribuímos 10 ao b e 1 ao c, podemos evidenciar este caso no exemplo abaixo.

Exemplificando:
Através da aplicação da Distância de Sokal-Michener, concluímos que a distância entre uma maçã e uma banana é igual a 0.6. Caso apliquemos a fórmula de atribuição de pesos então, D(X,Y) = 4.2 e D(Y,X) = 2.4, o que comprova a não simetria entre as distâncias.

Distância de Hamming

O cálculo matemático é bastante próximo da D_{SM}, esta métrica pode tornar-se muito grande à medida que o número de atributos aumenta. Apesar disto, a diferença entre estas métricas recaí sobretudo na identificação de atributos não definidos. Isto é, caso uma base de dados contenha 10 atributos e algum evento não possui nenhum destes atributos, então não haverá valor para `b` ou `c`, não sendo possível alcançar o valor de `n`. De modo conclusivo, D_{SM} é normalizada e não é possível identificar atributos não definidos, enquanto que a $D_{Hamming}$ consegue fazer essa distinção.

$$D_{Hamming} = b + c$$

Exemplificando:
Através da aplicação da Distância de Hamming, concluímos que a distância entre uma maçã e uma banana é igual a 3.

Distância Euclidiana

Nota-se que a que diferença entre a Distância Euclidiana e a Distância de Hamming é representada da seguinte forma:

$$D_{Euclid} = \sqrt{b+c} = \sqrt{D_{Hamming}}$$

Uma vez que a Distância Euclidiana é a raíz quadrada da Distância de Hamming, logo tudo o que vai ser detectado por Hamming vai ser detetado pela Euclidiana. Em suma, podemos concluir que a Euclidiana irá crescer mais lentamente do que a Hamming, com isto os valores falsos positivos e falsos negativos irão ter uma importância inferior em comparação da distância de Hamming.

Exemplificando
Através da aplicação da Distância Euclidiana, concluímos que a distância entre uma maçã e uma banana é aproximadamente 1.73.

Distância do Produto

$$D_{Produto} = D(X, Y) = \sqrt{b.c}$$

Propriedades:
Garantimos que $X = Y = 0$, então são considerados iguais, uma vez que não existem falsos positivos nem falsos negativos, porém poderá haver a possibilidade da existência dos mesmos mas estes terão de ser considerados como fatores não importantes. Nesta métrica importante destacar que, sendo uma multiplicação de falsos positivos com falsos negativos, as noções de igualdade e diferença entre elementos são diferentes do comum. Qualquer situação que não tenha falsos positivos ou não tenha falsos negativos tem um resultado de 0, logo os elementos são considerados iguais, apesar de eventualmente até nem terem nenhum atributo em comum.

Exemplificando
Através da aplicação da Distância do Produto, concluímos que a distância entre uma maçã e uma banana é igual a 1.4.

Comparação entre os valores finais das diferentes métricas

Observando a Tabela 2.5, podemos concluir, através das métricas abordadas, que a distância entre a maçã e uma banana é relativamente alta, contudo o valor difere em todas elas. A métrica de distancia que apresenta menor valor é de Sokal-Michener pois relaciona as variáveis negativas consoante o espaço amostral. Hamming teve o maior valor uma vez que relaciona as variáveis negativas. Já a métrica Euclidiana dá menos importância a variáveis negativas repetidas, enquanto o métrica dos Produto dá menos importância às variáveis negativas que causam menos impacto no *outcome*.

Sokal-Michener	Hamming	Euclidiana	Produto
0.6	3	1.73	1.4

Tabela 2.5: Comparação entre os valores finais das diferentes distâncias

Capítulo 3

Análise e Estudos

Após a análise das funções acima mencionadas, procedemos então ao seu desenvolvimento na plataforma *cocalc*. Para além do seu desenvolvimento, efetuámos também testes com vetores sintéticos de modo a apurar a veracidade do desempenho das formulas desenvolvidas.

3.1 Implementação

O *input* de todas estas fórmulas desenvolvidas irá ser uma matriz confusão construída a partir de dois vetores binários (v1 e v2). Cada vetor binário é um *array* de dimensão 9, ou seja, possui 9 atributos, onde cada valor pode ser 0 ou 1.

```
1 CM=confusion_matrix(v1,v2)
```

Quanto ao *output*, dependendo do tipo de fórmula, similaridade ou distância, irá devolver o resultado da comparação dos dois vetores, ou seja, no caso da similaridade, irá retornar um valor de semelhança dos dois vetores, no caso da distância, irá retornar o valor da distância dos dois vetores inseridos. Neste estudo, todas estas métricas estão representadas no anexo: A.1 com as respetivas linhas associadas.

Relativamente às métricas de similaridades, foram implementadas as seguintes: Sokal-Michener (linhas: 16 a 19), Jaccard (linhas: 11 a 14), Dice (linhas: 6 a 9), Russel and Rao (linhas: 1 a 4).

Quanto às distâncias foram implementadas as seguintes: Sokal-Michener (linhas: 21 a 24), Hamming (linhas: 26 a 28), Euclidiana (linhas: 30 a 33) e, a dissemelhança do produto (linhas: 35 a 38).

3.2 Benchmark

Neste tópico, iremos abordar os resultados obtidos, das similaridades e distâncias, das comparações dos vetores introduzidos. Para tal, começámos por definir 4 vetores sintéticos, denotados por `x, y, z` e `w`.

```
1 x = (1 1 1 0 0 0 0 0 0)
2 y = (1 1 1 1 0 0 0 0 0)
3 z = (0 0 0 1 1 1 1 0 0)
4 w = (0 0 0 1 1 0 0 0 1)
```

O vetor principal é o `x` e, de seguida, a este vetor iremos fazer 3 comparações diferentes, ou seja, iremos comparar o `x`, com o `y`, com o `z` e com o `w`. As respetivas matrizes confusão estão representadas desde a Tabela 3.1 à Tabela 3.3.

x/y	1	0
1	3	1
0	0	5

Tabela 3.1: M.Conf. x/y

x/z	1	0
1	0	4
0	3	2

Tabela 3.2: M.Conf. x/z

x/w	1	0
1	0	3
0	3	3

Tabela 3.3: M.Conf. x/w

3.2.1 Similaridades

Na Tabela 3.4 dispomos todos os valores obtidos com base nas funções de similaridade.

Matrizes	Sokal-Michener	Jaccard	Dice	Russel and Rao
(x,y)	0.8889	0.75	0.8571	0.3333
(x,z)	0.2222	0	0	0
(x,w)	0.3333	0	0	0

Tabela 3.4: Benchmarking das Similaridades

3.2.2 Distâncias

Na Tabela 3.5 dispomos todos os valores obtidos com base nas funções de distância.

Matrizes	Sokal-Michener	Euclidiana	Hamming	Produto
(x,y)	0.1111	1	1	0
(x,z)	0.7778	2.6458	7	3.4641
(x,w)	0.6667	2.4495	6	3

Tabela 3.5: Benchmarking das Distâncias

Capítulo 4

Aplicações

4.1 Dados da Base de Dados SCADI

Este conjunto de dados contém 206 atributos de 70 crianças com deficiência física e motora com base na International Classification of Functioning Child and Youth (ICF-CY). Sendo este um projeto relacionado com dados binários, então para a implementação das métricas apenas valorizámos as colunas destacadas a negrito.

Colunas	Atributo/Valor
Género	**Masculino - 1**
	Feminino - 0
Idade	5,6,7 ...
205 Atividades na ICF-CY	**Tem - 1**
	Não Tem - 0
Classes	Classe 1 até Classe 7

Tabela 4.1: Esquema SCADI

4.1.1 Significado das Classes

- **Classe 1** - Problemas em Cuidar de Partes do Corpo;
- **Classe 2** - Problemas de Higiene;
- **Classe 3** - Problemas em Vestir;
- **Classe 4** - Problemas em Vestir, Cuidar de Partes do Corpo e Lavar;
- **Classe 5** - Problemas em Vestir, Cuidar de Partes do Corpo, Lavar e ir ao WC;
- **Classe 6** - Problemas em Vestir, Cuidar de Partes do Corpo, Lavar, ir ao WC e Cuidar da Saúde;
- **Classe 7** - Sem Problemas.

4.2 Dados da Base de Dados de Emojis

Esta base de dados consiste em 30 *bitmaps* correspondentes a imagens de diferentes emojis. As imagens foram escaladas para 15x15 de modo a facilitar o manuseamento das mesmas. Sendo que cada imagem é um *bitmap*, os atributos de cada elemento da base de dados correspondem aos valores binários de cada coordenada do mapa de 15x15, que indicam se a coordenada está ou não pintada de preto, com isto o valor 0 corresponde a branco e o valor 1 corresponde a preto.

4.3 Métricas a utilizar

Após o benchmarking, feito anteriormente, decidimos escolher apenas 4 métricas das 8 testadas. Essas métricas escolhidas são a similaridade de Sokal-Michener e similaridade Jaccard, pois das métricas de similaridade foram as que obtiveram resultados mais diferenciados das outras. Em relação da às métricas de distância escolhemos a distância de Sokal-Michener pois é a complementar da similaridade e a distância Euclidiana, pois também obteve resultados diferenciados e é uma métrica de distância muito comum. Para testar a aplicação destas métricas às bases de dados, selecionámos o primeiro elemento de cada base dados, representado nas figuras por `x` e o segundo elemento de cada base dados representado nas figuras por `y`.

4.3.1 SCADI

Matriz	Sokal-Michener	Jaccard
(x,y)	0.9508	0.7059

Tabela 4.2: Semelhanças entre x e y

Matriz	Sokal-Michener	Euclidiana
(x,y)	0.0492	3.1623

Tabela 4.3: Distâncias entre x e y

Vizinhanças

Aplicadas as métricas e comparando o primeiro elemento com os restantes elementos desta base de dados, concluímos que apenas 20% da base dados se aproxima ao dito elemento. De seguida introduzimos a filosofia do "vizinho do meu vizinho, é meu vizinho", obtendo então uma percentagem de 42% para os valores que se aproximam ao dito elemento.

4.3.2 Emojis

Matriz	Sokal-Michener	Jaccard
(x,y)	0.8311	0.5422

Tabela 4.4: Semelhanças entre x e y

Matriz	Sokal-Michener	Euclidiana
(x,y)	0.1689	6.1644

Tabela 4.5: Distâncias entre x e y

Vizinhanças

Aplicadas as métricas e comparando o sexto elemento com os restantes elementos desta base de dados, concluímos que apenas 7% da base dados se aproxima ao dito elemento.

Ao aplicar a filosofia do "vizinho do meu vizinho, é meu vizinho", a percentagem não foi alterada, ou seja, os valores mantiveram-se. Estes 7% dizem respeito apenas a um outro elemento presente na base de dados, para além do elemento referência, e, de um modo visual, conseguimos confirmar que ambas as fotografias, ou seja a fotografia 12 (sexto elemento) e 45, são de facto muito parecidas.

4.4 *Clusters*

Os *clusters* são implementados em diversos passos, isto é, após a carregamento da base de dados e da criação da tabela de similaridade/dissimilaridade, dependendo da métrica usada, tabela que contém os valores de similaridade/dissimilaridade entre cada par de elementos da base de dados. Começando por um dado elemento, compara-se com todos os outros elementos existentes na base de dados e, verificamos se a distância entre os pares comparados é menor que o *epsilon* definido, no caso das similaridades verificamos se a distância entre os pares comparados é maior que o *epsilon* definido. *Epsilon* esse que determina o valor de similaridade/dissimilaridade limite para os elementos de uma dada vizinhança.

Depois de comparar o primeiro elemento com todos os restantes, guardamos aqueles que pertencem à vizinhança do elemento numa lista secundária, repete-se o processo com os elementos restantes as vezes necessárias até que, todos os elementos estejam incluídos numa lista, sendo essas listas os *clusters*.

4.4.1 SCADI

Relativamente a esta base de dados, tentámos obter exatamente o mesmo número de *clusters* para cada métrica, sendo ela similaridade ou distância, e encontrámos um número de *clusters* ótimo, que se situa nos 7. Posto isto, obtemos então para a similaridade de Sokal-Michener um *epsilon* de 0.9 e, para a similaridade de Jaccard um *epsilon* de 0.48.

Quanto às métricas das distâncias utilizadas neste exercício, procedemos da mesma forma, ou seja, tentámos encontrar o mesmo número ótimo de *clusters* e, obtemos então, para a distância de Sokal-Michener, um *epsilon* de 0.1, o que representa o inverso da similaridade e, para a distância Euclidiana um *epsilon* de 4.5.

4.4.2 Emojis

Relativamente a esta base de dados, seguimos a mesma abordagem, ou seja, tentámos obter exatamente o mesmo número de *clusters* para cada métrica, sendo ela similaridade ou distância, e encontrámos um número de *clusters* ótimo, que se situa nos 11. Posto isto, obtemos então para a similaridade de Sokal-Michener um *epsilon* de 0.9 e, para a similaridade de Jaccard um *epsilon* de 0,645. Para esta última apesar de termos exatamente o mesmo número de *clusters*, algumas das fotograficas agrupadas dentro dos *clusters* eram diferentes.

Quanto às métricas das distâncias utilizadas neste exercício, procedemos da mesma forma, ou seja tentámos encontrar o mesmo número ótimo de *clusters* e obtemos então, para a distância de Sokal-Michener, um *epsilon* de 0.1, o que representa o inverso da similaridade e, para a distância Euclidiana um *epsilon* de 4.7.

Capítulo 5

Conclusão

Com o desenvolvimento e implementação deste Micro-Projeto ficou claro para todos os elementos do grupo de trabalho que a aplicação de métricas é um tema bastante importante no estudo da área de *Machine Learning*. Através da realização deste Projeto, foi possível aplicar, explorar e consolidar os diversos conhecimentos que nos foram sendo passados ao longo de todo o semestre, nomeadamente nas aulas teórico-práticas da Unidade Curricular de Métricas em Machine Learning. Essencialmente os diversos conceitos investigados estão relacionados com a Análise de Dados Binárias, mais especificamente, Similaridade, Distância entre Dados Binários. Recorrendo à plataforma *cocalc* foi possível implementar e testar às formulações matemáticas de Similaridade e Distância estudadas.

Como já foi referido, durante todo o desenvolvimento, foi necessário realizar estudos e pesquisas de modo a selecionar e interpretar as melhores métricas para as contextualizar com as Bases de Dados Binárias, sendo que as maiores adversidades encontradas foram na aquisição de conceitos, propriedades e conclusões de algumas das métricas, pois por vezes existiam métricas com demasiada informação, tendo sido fundamental selecionar a informação mais relevante e acertada de acordo com o contexto em estudo.

Apesar de este ser considerado um Micro-Projeto, como qualquer outro Projeto, independentemente da sua dimensão, este não é exceção e contém possíveis melhorias. Na nossa opinião e muito devido ao facto de termos um limite de páginas, algumas das métricas aqui descritas podiam ser mais esmiuçadas e exploradas ao nível de propriedades, contudo e observando os objetivos propostos consideramos ter concluído os demais.

De forma a concluir e reiterar, a elaboração deste Micro-Projeto permitiu a todos os constituintes do grupo alargar os seus conhecimentos relativamente a conceitos elaborados em contexto de aula e extra-aula.

Apêndice A

Código

```
1
2  def RusselRao_similarity(CM):
3      total=CM[0,0]+CM[1,1]+CM[0,1]+CM[1,0]
4      simil=(CM[0,0])/total
5      return simil
6
7  def Dice_similarity(CM):
8      total=2*CM[0,0]+CM[0,1]+CM[1,0]
9      simil=(2*CM[0,0])/total
10     return simil
11
12 def Jaccard_similarity(CM):
13     total=CM[0,0]+CM[0,1]+CM[1,0]
14     simil=(CM[0,0])/total
15     return simil
16
17 def SokalMichener_similarity(CM):
18     total=CM[0,0]+CM[1,1]+CM[0,1]+CM[1,0]
19     simil=(CM[0,0]+CM[1,1])/total
20     return simil
21
22 def SokalMichener_distance(CM):
23     total=CM[0,0]+CM[1,1]+(CM[0,1]+CM[1,0])
24     dist=(CM[0,1]+CM[1,0])/total
25     return dist
26
27 def Hamming_distance(CM):
28     dist=CM[0,1]+CM[1,0]
29     return dist
30
31 def Euclidean_distance(CM):
32     x=CM[0,1]+CM[1,0]
33     dist=math.sqrt(x)
34     return dist
35
36 def Produto_distance(CM):
37     x=CM[0,1]*CM[1,0]
38     dist=math.sqrt(x)
39     return dist
```

Listing A.1: Similaridades e Distâncias Implementadas