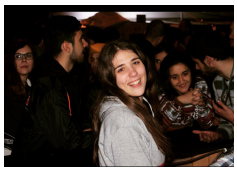
# Sistemas de Representação de Conhecimento e Raciocínio

MIEI - 3º ano - 2º semestre
Universidade do Minho

## Trabalho Prático III



Sara Pereira
A73700


Rui Vieira
A74658


Filipe Fortunato
A75008


Frederico Pinto
A73639

# *Conteúdo*

# 1. *Introdução*

O trabalho prático descrito neste relatório, o terceiro da unidade curricular de *Sistemas de Representação de Conhecimento e Raciocínio*, é um pouco diferente dos anteriores pois é abordado um tema ligeiramente diferente que é o *Conhecimento não Simbólico : Redes Neuronais Artificiais*.

Sendo assim, com base num conjunto de dados fornecidos pelo professor sobre Bank Marketing, este que contém diversas informações sobre campanhas de Bancos em Portugal (telefonemas), temos de ser capazes de fazer um pequeno estudo com o intuito de descobrir se o cliente irá ou não subscrever um depósito a prazo. A linguagem usada para realizar o estudo é o **R** e utilizamos o RStudio como ferramenta.

No seguinte relatório são apresentadas as várias etapas e decisões que grupo tomou durante a realização deste trabalho.

# 2. *Preliminares*

## 2.1 O que são RNA´s

As RNA's são um conceito de computação que tem como objetivo processar diferentes dados de uma forma parecida com o cérebro humano. O cérebro funciona de uma forma bastante parecida a um processador extremamente complexo que tem a capacidade de efetuar processamentos em paralelo a uma velocidade extremamente alta, mas ainda não existe nenhum computador capaz de fazer exatamente o que o cérebro é capaz de fazer.

A ideia deste processamento é a de que seja realizado tendo em conta os neurónios de um cérebro. Sendo assim, tal como no cérebro, as RNA's têm de ter a capacidade de aprendizagem e de tomada de decisões.

Desta forma, as RNA's podem ser vistas como um esquema que têm a capacidade de adquirir conhecimento através da aprendizagem/experiência.

## 2.2 Funcionamento

Uma RNA assemelha-se a um cérebro de duas formas: o conhecimento é adquirido através de processos de aprendizagem e este é armazenado nas conexões entre os nodos, mais concretamente nas sinapses. Desta forma, podemos afirmar que as Redes Neuronais têm presentes na sua constituição uma rede de neurónios artificiais que estão conectados entre si, formando assim uma rede com elementos de processamento.

A rede neuronal recebe então diversos parâmetros de um caso como *input* e percorrendo essa informação pela sua rede retorna um ou mais valores como *output*.

Na sua arquitetura é determinado o número de camadas usadas (neurónios), a quantidade de neurónios em cada camada, tipo de sinapse a utilizar, etc.

## 2.3 Aplicações

As Redes Neuronais têm um a capacidade de resolver diversos tipos de problemas. Um exemplo desta aplicação são os softwares de reconhecimento facial, pois estes têm a capacidade aprender a conhecer a cara de diversas pessoas, também são usadas para reconhecer vários padrões, usadas em robôs que desarmam bombas, etc. De forma geral, estas são usadas para problemas bastantes complexos como o mercado financeiro, em diversos ramos médicos , etc.

# 3. *Normalização e Análise dos dados*

O primeiro passo ,antes de passarmos à normalização e análise dos dados, é perceber profundamente as identidades e domínios dos conjuntos de dados.

Nos dados fornecidos pela equipa docente é nos fornecido diversa informação acerca de campanhas (telefónicas) de Bancos em Portugal, como por exemplo, dados sobre o cliente e os contactos feitos a este, dados sobre a situação económica que o pais se encontra, etc. Sendo que o ficheiro tem 20 atributos no total.

Após o grupo se reunir, achamos bem retirar alguns destes atributos pois estão não têm um peso suficiente para a decisão de subscrever ou não um depósito a prazo. Sendo assim ficamos com os seguintes atributos:

- Dados do cliente
  - **age** - idade do cliente
  - **job** - Área de trabalho do cliente
  - **marital** - Estado civil
  - **education** - Grau de educação do cliente
  - **default** - Se cliente já possui depósito
  - **housing** - Se o cliente tem um empréstimo para a casa
  - **loan** - Se o cliente tem um empréstimo pessoal
- Atributos relacionados com os contactos realizados ao cliente
  - **campaign** - número de contactos realizados durante esta campanha
  - **pdays** - número de dias passados desde o último contacto de uma outra campanha
  - **previous** - número de contactos efetuados para o cliente antes desta campanha
  - **poutcome** - resultado da última campanha de marketing
- Atributos sobre o contexto social e económico
  - **emp.var.rate** - variação da taxa de empregabilidade (trimestral)
  - **cons.price.idx** - índice do preço de consumidor (mês)
  - **cons.conf.idx** - índice da confiança do consumidor (mês)
  - **euribor3m** - taxa do euribor 3 meses (dia)

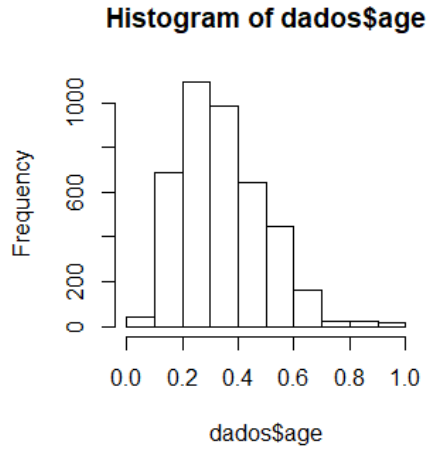
## 3.1 Normalização

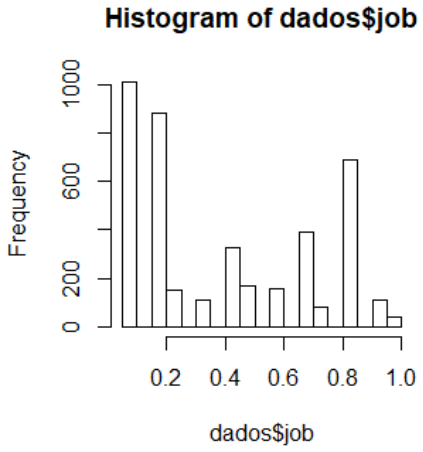
O nosso grupo teve a necessidade de normalizar os valores disponibilizados para podermos chegar ao objetivo final com melhores resultados. A maioria dos nossos parâmetros estavam definidos por strings, o que nos levou a normaliza-las para valores inteiros, sendo que para o parâmetro **job** usamos um intervalo entre [1,12] sendo que o 1 representa *'admin.'*, o 2 *'blue-collar'*, o 3 *'entrepreneur'*,o 4 *'housemaid'*, o 5 *'management'*,o 6 *'retired'*, o 7 *'self-employed'*, o 8 *'services'*, o 9 *'student'*, o 10 *'technician'*, o 11 *'unemployed'* e o 12 *'unknown'*. O parâmetro **marital** toma valores entre [1,4] sendo que o 1 representa o *'divorced'*, o 2 *'married'*, o 3 *'single'* e o 4 *'unknown'*. Para o parâmetro **education** o grupo decidiu usar um intervalo entre [1,8] sendo que o 1 representa *'basic.4y'*, o 2 *'basic.6y'*, o 3 *'basic.9y'*, o 4 *'high.school'*, o 5 *'illiterate'*,o 6 *'professional.course'*, o 7 *'university.degree'* e o 8 *'unknown'*. O parâmetro **default** está entre [1,3] e o 1 identifica o *'no'*, o 2 identifica o *'yes'* e o 3 identifica *'unknown'*. Para o parâmetro **housing** usamos o intervalo entre [1,3] sendo que o 1 identifica o *'no'*, o 2 identifica o *'yes'* e o 3 identifica *'unknown'*. Para parâmetro **loan** foi usado também um intervalo entre [1,3] com o 1 a identificar o *'no'*, o 2 identifica o *'yes'* e o 3 identifica *'unknown'*. Por último alteramos o parâmetro *poutcome* para estar entre [1,3] sendo que o 1 representa *'failure'*, o 2 identifica o *'nonexistent'* e o 3 identifica *'success'*.

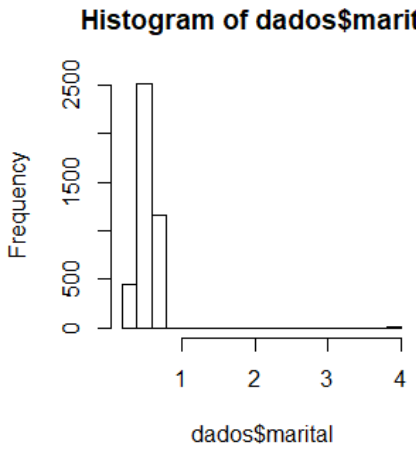
Após efetuarmos todas estas alterações, o grupo decidiu diminuir o tamanho dos intervalos para os diferentes parâmetros sendo que estes passaram a estar entre [0,1] e exclusivamente para os parâmetros **emp.var.rate** que tem um intervalo de [-1,1] e **cons.conf.idx** passou a estar entre [-1,0], de maneira a que a gama de valores dos diferentes parâmetros não seja tão grande e assim melhorar a solução final.

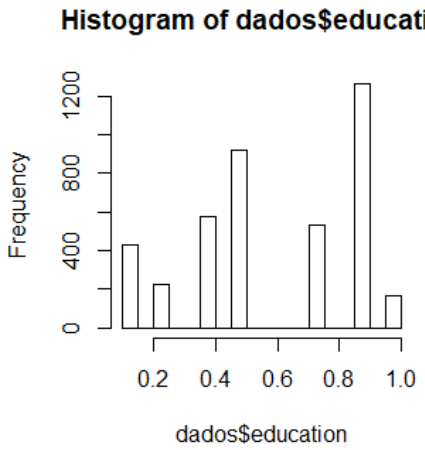
## 3.2 Análise dos dados

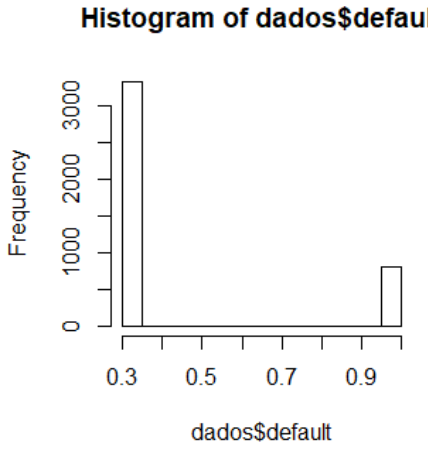
Neste trabalho é nos pedido para determinarmos se um cliente vai ou não realizar um depósito a prazo. Antes de realizarmos qualquer processo fizemos uma pequena análise aos dados fornecidos com o intuito de perceber melhor o problema em questão. Nas figuras seguintes é apresentado a variação dos nossos dados.

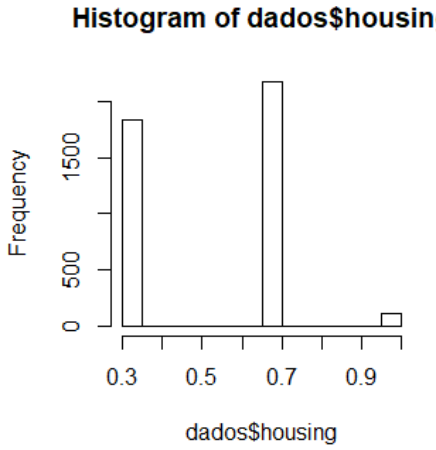


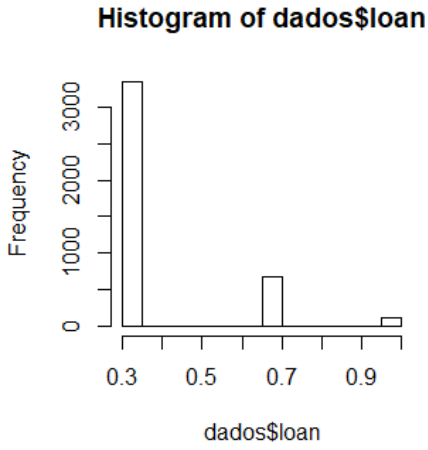



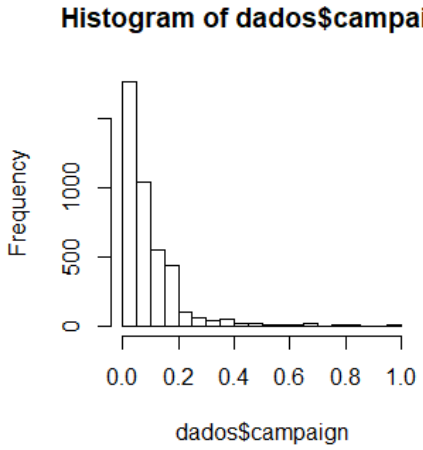

Histogram of dados$marital
Frequency
0
500
1500
2500
1
2
3
4
dados$marital

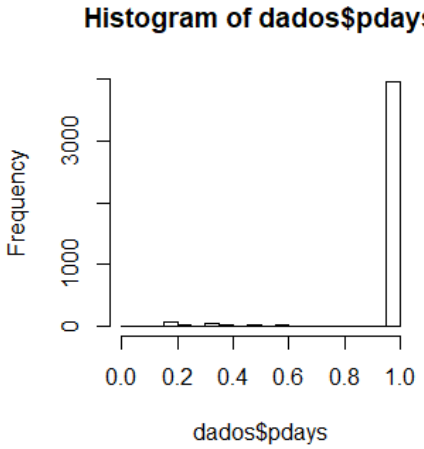

Histogram of dados$education
Frequency
0
400
800
1200
0.2
0.4
0.6
0.8
1.0
dados$education

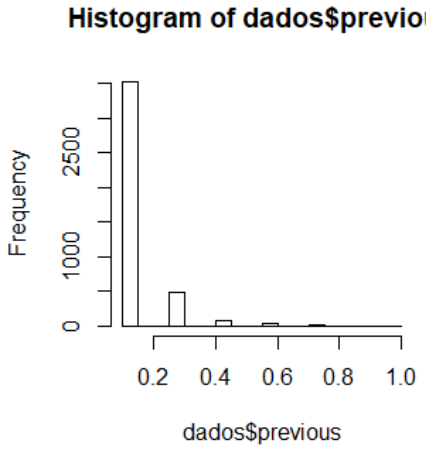

Histogram of dados$default
Frequency
0
1000
2000
3000
0.3
0.5
0.7
0.9
dados$default

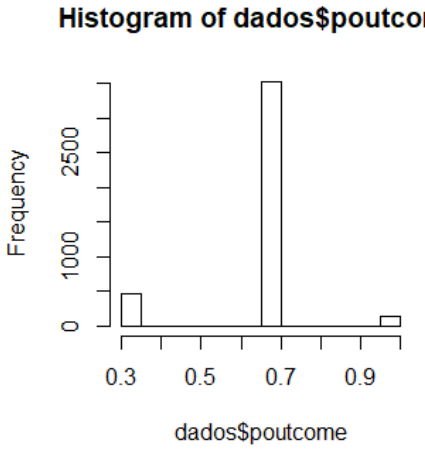

Histogram of dados$housing
Frequency
0
500
1500
0.3
0.5
0.7
0.9
dados$housing

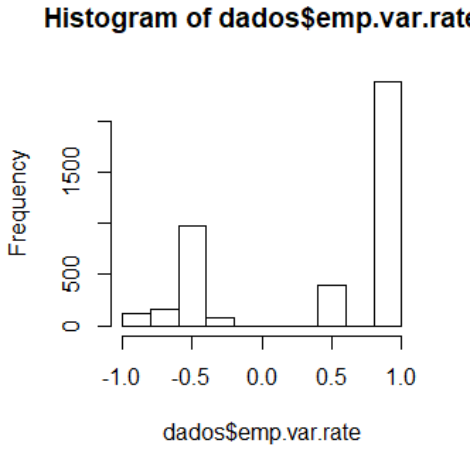

Histogram of dados$loan
Frequency
0
1000
2000
3000
0.3
0.5
0.7
0.9
dados$loan

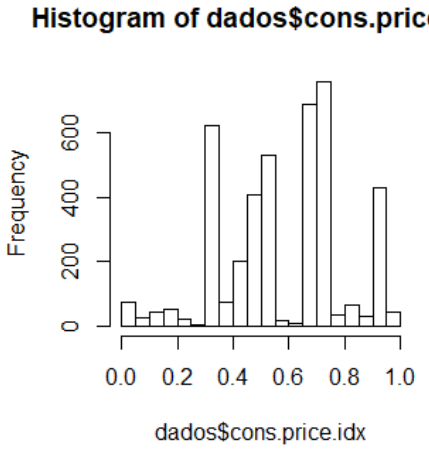

Histogram of dados$campaign
Frequency
0
500
1000
0.0
0.2
0.4
0.6
0.8
1.0
dados$campaign

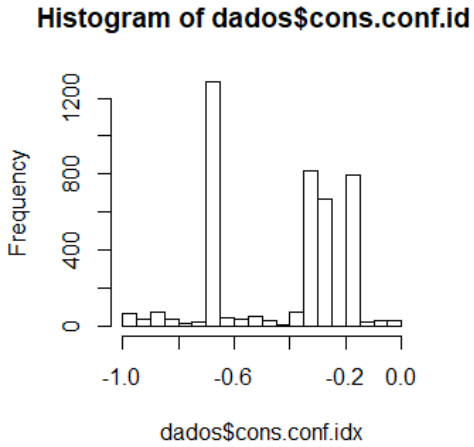

Histogram of dados$pdays
Frequency
0
1000
3000
0.0
0.2
0.4
0.6
0.8
1.0
dados$pdays

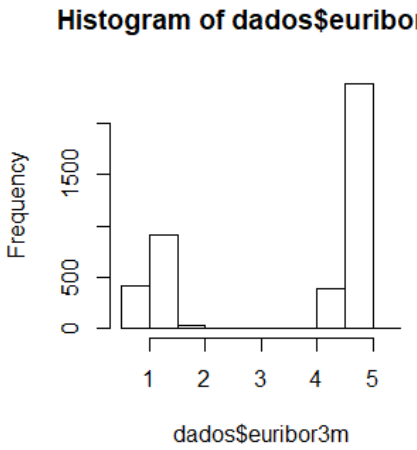

Histogram of dados$previous
Frequency
0
1000
2500
0.2
0.4
0.6
0.8
1.0
dados$previous


Histogram of dados$poutcome
Frequency
0
1000
2500
0.3
0.5
0.7
0.9
dados$poutcome


Histogram of dados$emp.var.rat
Frequency
0
500
1500
-1.0
-0.5
0.0
0.5
1.0
dados$emp.var.rate


Histogram of dados$cons.price.id
Frequency
0
200
400
600
0.0
0.2
0.4
0.6
0.8
1.0
dados$cons.price.idx


Histogram of dados$cons.conf.id
Frequency
0
400
800
1200
-1.0
-0.6
-0.2
0.0
dados$cons.conf.idx




Após efetuar essa análise, o grupo decidiu dividir o ficheiro em dois sets, um para treino e outro para testes. A amostra ficou dividida da seguinte forma

```
treino <- dados[1:3000,]

teste <- dados[3001:4120,]
```

# 4. *O cliente vai ou não realizar um depósito a prazo?*

## 4.1 Importância dos Atributos

No capitulo anterior é feito um estudo acerca da importância de cada atributo para o problema de questão. Numa primeira fase o grupo considerou que todos os parâmetros teriam a mesma importância e assim resultou a seguinte formula:

```
formula <- y ~ age+
               job+
               marital+
               education+
               default+
               housing+
               loan+
               campaign+
               pdays+
               previous+
               poutcome+
               emp.var.rate+
               cons.price.idx+
               cons.conf.idx+
               euribor3m
```

De seguida, aplicando o comando ***regsubsets*** conseguimos descobrir a relevância de cada atributo em relação ao y (variável alvo):

```
1 subsets of each size up to 15
Selection Algorithm: exhaustive
          age job marital education default housing loan campaign pdays previous poutcome emp.var.rate cons.price.idx
1  ( 1 )  " " " " " "     " "       " "     " "     " " " "      "*"   " "      " "      " "          " "
2  ( 1 )  " " " " " "     " "       " "     " "     " " " "      "*"   " "      " "      " "          " "
3  ( 1 )  " " " " " "     " "       " "     " "     " " " "      "*"   " "      " "      " "          "*"
4  ( 1 )  " " " " " "     " "       " "     " "     " " " "      "*"   " "      " "      " "          "*"
5  ( 1 )  " " " " " "     " "       " "     " "     " " " "      "*"   " "      "*"      " "          "*"
6  ( 1 )  " " " " " "     " "       " "     " "     " " " "      "*"   "*"      "*"      " "          "*"
7  ( 1 )  "*" " " " "     "*"       " "     " "     " " " "      "*"   " "      "*"      " "          "*"
8  ( 1 )  "*" " " " "     "*"       " "     " "     " " " "      "*"   "*"      "*"      " "          "*"
9  ( 1 )  "*" " " " "     "*"       " "     " "     " " "*"      "*"   "*"      "*"      " "          "*"
10 ( 1 )  "*" " " " "     "*"       " "     " "     " " "*"      "*"   "*"      "*"      "*"          "*"
11 ( 1 )  "*" " " " "     "*"       "*"     " "     " " "*"      "*"   "*"      "*"      "*"          "*"
12 ( 1 )  "*" " " "*"     "*"       "*"     " "     " " "*"      "*"   "*"      "*"      "*"          "*"
13 ( 1 )  "*" " " "*"     "*"       "*"     "*"     " " "*"      "*"   "*"      "*"      "*"          "*"
14 ( 1 )  "*" "*" "*"     "*"       "*"     "*"     " " "*"      "*"   "*"      "*"      "*"          "*"
15 ( 1 )  "*" "*" "*"     "*"       "*"     "*"     "*" "*"      "*"   "*"      "*"      "*"          "*"
          cons.conf.idx euribor3m
1  ( 1 )  " "           " "
2  ( 1 )  " "           "*"
3  ( 1 )  "*"           "*"
4  ( 1 )  "*"           "*"
5  ( 1 )  "*"           "*"
6  ( 1 )  "*"           "*"
7  ( 1 )  "*"           "*"
8  ( 1 )  "*"           "*"
9  ( 1 )  "*"           "*"
10 ( 1 )  "*"           "*"
11 ( 1 )  "*"           "*"
12 ( 1 )  "*"           "*"
13 ( 1 )  "*"           "*"
14 ( 1 )  "*"           "*"
15 ( 1 )  "*"           "*"
```

Figura 4.1: Resultado obtido pelo RStudio para os atributos com maior relevância

## 4.2 Fórmulas

De seguida a determinar quais os atributos com maior importância, decidimos desenvolver algumas fórmulas em que o número de atributos varia consoante a importância dos mesmos. Achamos bem começar fazer 7 fórmulas sendo que a primeira possui todos os atributos e as outras variam entre 4 e 14 no número de atributos.

```
formula <- y ~ age+
               job+
               marital+
               education+
               default+
               housing+
               loan+
               campaign+
               pdays+
               previous+
               poutcome+
               emp.var.rate+
               cons.price.idx+
               cons.conf.idx+
               euribor3m

formula2 <- y ~ pdays+
                cons.price.idx+
                cons.conf.idx+
                euribor3m
```

```
formula3 <- y ~ age+
                pdays+
                poutcome+
                cons.price.idx+
                cons.conf.idx+
                euribor3m

formula4 <- y ~ age+
                education+
                pdays+
                previous+
                poutcome+
                cons.price.idx+
                cons.conf.idx+euribor3m

formula5 <- y ~ age+
                education+
                campaign+
                pdays+
                previous+
                poutcome+
                emp.var.rate+
                cons.price.idx+
                cons.conf.idx+
                euribor3m

formula6 <- y ~ age+
                marital+
                education+
                default+
                campaign+
                pdays+
                previous+
                poutcome+
                emp.var.rate+
                cons.price.idx+
                cons.conf.idx+
                euribor3m

formula7 <- y ~ age+
                job+
                marital+
                education+
                default+
                housing+
                campaign+
```

```
pdays+
previous+
poutcome+
emp.var.rate+
cons.price.idx+
cons.conf.idx+
euribor3m
```

## 4.3 Treino

### 4.3.1 Neuralnet

Para conseguirmos realizar o treino da RNA usamos o comando **neuralnet**: neuralnet(formula,data,hidden,threshold, algoritmo, lifesign, linear.output) Os respetivos parâmetros são:

- formula : formula utilizada para treinar a rede
- data : dataset que contem os atributos da fórmula
- hidden : valor que define o número de nodos escondidos que será usado
- threshold : valor de erro que é responsável por parar a execução do comando
- algoritmo : algoritmo a ser usado no treino da RNA
- lifesign : especifica o que vai ser impresso durante o decorrer do comando
- linear.output : Booleano que especifica a utilização de nodos exteriores

## 4.4 Sets

Depois de realizado o treino da nossa RNA, é necessário a criação de *sets* para podermos testar a rede de maneira a que os atributos que vamos utilizar nas diferentes formulas estejam ao nosso alcance. Dessa forma criamos 7 subsets que correspondem as 7 formulas apresentadas anteriormente. Os sets são:

```
set <- subset(teste,select=c("age","job","marital","education","default",
    "housing","loan","campaign","pdays","previous","poutcome","emp.var.rate",
    "cons.price.idx","cons.conf.idx","euribor3m"))

set2 <-subset(teste,select=c("pdays","cons.price.idx","cons.conf.idx",
    "euribor3m"))

set3 <- subset(teste,select=c("age","pdays","poutcome","cons.price.idx",
"cons.conf.idx","euribor3m"))
```

```
set4 <- subset(teste,select=c("age","education","pdays","previous",
        "poutcome","cons.price.idx","cons.conf.idx","euribor3m"))

set5 <- subset(teste,select=c("age","education","campaign","pdays",
    "previous","poutcome","emp.var.rate","cons.price.idx","cons.conf.idx",
    "euribor3m"))

set6 <- subset(teste,select=c("age","marital","education","default",
    "campaign","pdays","previous","poutcome","emp.var.rate","cons.price.idx",
    "cons.conf.idx","euribor3m"))

set7 <- subset(teste,select=c("age","job","marital","education",
    "default","housing","campaign","pdays","previous","poutcome",
    "emp.var.rate","cons.price.idx","cons.conf.idx","euribor3m"))
```

## 4.5 Testes

Nesta parte do trabalho será apresentado os resultados dos testes realizados sobre se o cliente subscreve ou não um depósito a prazo. É apresentado de seguida uma tabela com esse resultados usando sempre parâmetros diferentes nos testes.

| Nº Teste | Fórmula | Hidden | Algoritmo | Threshold | Steps | Erro | RMSE |
|---|---|---|---|---|---|---|---|
| 1 | formula | (10) | rprop- | 0.1 | 66877 | 96.76929 | 0.3049150461 |
| 2 | formula | (6,3) | sag | 0.1 | 43605 | 89.63403 | 0.3171530395 |
| 3 | formula | (2,4) | rprop- | 0.1 | 844 | 118.54218 | 0.2750333298 |
| 4 | formula2 | (10) | slr | 0.1 | 3147 | 114.79008 | 0.2680516861 |
| 5 | formula2 | (4) | rprop- | 0.05 | 487 | 117.95369 | 0.2756427658 |
| 6 | formula2 | (4,2) | rprop+ | 0.05 | 992 | 117.48874 | 0.2733164828 |
| 7 | formula3 | (8) | slr | 0.05 | 11952 | 110.60865 | 0.2783470968 |
| 8 | formula3 | (6,4) | sag | 0.1 | 4831 | 112.21025 | 0.2771553325 |
| 9 | formula3 | (2,4) | rprop+ | 0.05 | 1421 | 117.81361 | 0.2721429009 |
| 10 | formula4 | (6) | sag | 0.05 | 2168 | 113.85243 | 0.2778901222 |
| 11 | formula4 | (10) | rprop+ | 0.1 | 23583 | 105.19786 | 0.2975717229 |
| 12 | formula4 | (2,4) | rprop- | 0.05 | 2380 | 118.1652 | 0.2773336133 |
| 13 | formula5 | (10) | slr | 0.1 | 11814 | 103.99809 | 0.2831475071 |
| 14 | formula5 | (8,4,2) | rprop+ | 0.05 | 83877 | 96.21859 | 0.2913025071 |
| 15 | formula5 | (4,2) | rprop- | 0.05 | 10702 | 112.54859 | 0.2838446477 |
| 16 | formula6 | (8) | sag | 0.1 | 12148 | 103.26072 | 0.2917744981 |
| 17 | formula6 | (4) | slr | 0.1 | 24282 | 112.3461 | 0.2810043144 |
| 18 | formula6 | (6,3) | rprop- | 0.05 | 10993 | 100.87138 | 0.2929273244 |
| 19 | formula7 | (10) | rprop- | 0.05 | 93619 | 93.89185 | 0.3014879192 |
| 20 | formula7 | (6,3) | rprop+ | 0.1 | 2481 | 104.48504 | 0.2946517924 |
| 21 | formula7 | (2,4) | rprop- | 0.05 | 17781 | 111.99895 | 0.2770847975 |

Tabela 4.1: Tabela de Treino e com os diferentes valores para o RMSE

# 5. *Conclusão*

A realização deste trabalho funcionou como uma consolidação da matéria lecionada nas aulas lecionadas na Unidade Curricular, mais concretamente sobre o tema de Redes Neuronais. Desta forma, com este exercício ignoramos o conhecimento não simbólico, baseado na lógica, e passamos ao estudo do conhecimento não simbólico baseado na capacidade de aprendizagem, algo que o grupo nunca tinha trabalhado. Posto isto, neste trabalho foi necessário tratar os diferentes dados fornecidos pelo docente, normalizando os mesmos de maneira a aumentar a eficiência e diminuirmos o RMSE. Para além disso fizemos diversos testes à nossa RNA e os diversos cálculos realizados para chegar aos resultados apresentados neste relatório. Em forma de conclusão, o grupo acha que todos os conceitos envolvendo Redes Neuronais com o passar do tempo passarão cada vez mais a ser um conceito muito importante no mundo da computação, pois este nos ajuda a resolver diversos problemas complexos.