

Sistemas de Representação de Conhecimento e Raciocínio

## **Trabalho de Grupo – 3º Exercício**

MIEI - 3º Ano, 2º Semestre

Universidade do Minho

Ano letivo 2016/2017

Ana Isabel Castro (a55522)

Joana Miguel (a57127)

Lúcia Abreu (a71634)

Braga, 21 de Maio de 2017

# Conteúdo

Lista de Figuras .....	4
Índice de Tabelas .....	5
Introdução .....	6
Preliminares.....	7
Dados Fornecidos .....	8
Parte 1: Estudo Inicial .....	10
1.1    Análise dos dados biométricos .....	10
1.2    Estudo dos atributos mais significativos.....	12
▪    Exaustão.....	12
▪    Tarefa.....	13
1.3    Planeamento dos Testes.....	14
▪    Planeamento dos testes da Parte 2: Identificação dos níveis de Exaustão.....	15
▪    Planeamento dos testes da Parte 3: Identificação das tarefas em execução .....	16
▪    Planeamento dos testes da Parte 4: Caracterização da Solução.....	17
Parte 2: Identificação dos níveis de Exaustão .....	18
1.    Cenário 1: Identificar os sete níveis de exaustão .....	18
1.1    Abordagem Supervisionada.....	19
1.1.1    Representação dos dados biométricos de forma contínua .....	19
1.1.1.1    Utilizando a tarefa como input .....	19
1.1.1.2    Não utilizando a tarefa como input.....	19
1.1.2    Representação dos dados biométricos de forma discreta .....	20
1.1.2.1    Utilizando a tarefa como input .....	20
1.1.2.2. Não utilizando a tarefa como input.....	20
1.1    Abordagem Não Supervisionada .....	21
1.2    Conclusões .....	21
2.    Cenário 2: Identificar a existência ou ausência de exaustão.....	22
2.1    Abordagem Supervisionada.....	22
2.1.1    Representação dos dados biométricos de forma contínua .....	22
2.1.1.1    Utilizando a tarefa como input .....	22
2.1.1.2    Não utilizando a tarefa como input .....	23
2.1.2    Representação dos dados biométricos de forma discreta .....	23
2.1.2.1    Utilizando a tarefa como input .....	23
2.1.2.2    Não utilizando a tarefa como input.....	24
2.2    Conclusões .....	24
3.    Cenário 3: Determinar a melhor escala de identificação de exaustão.....	25
3.1    Abordagem Supervisionada.....	26

3.1.1	Representação dos dados biométricos de forma contínua .....	26
3.1.1.1	Utilizando a tarefa como input .....	26
3.1.1.2	Não utilizando a tarefa como input .....	26
3.1.2	Representação dos dados biométricos de forma discreta .....	26
3.1.2.1	Utilizando a tarefa como input .....	27
3.1.2.2	Não utilizando a tarefa como input .....	27
3.2	Conclusões .....	27
Parte 3: Identificação da tarefa em execução .....		28
1.1	Abordagem Supervisionada .....	29
1.1.1	Tarefa normalizada com os valores 1, 2, 3 .....	29
1.1.2	Tarefa normalizada com os valores 0, 0.5, 1 .....	29
1.2	Conclusões .....	29
Parte 4: Caracterização da solução .....		30
1.	Testes à topologia .....	30
1.1	Conclusões .....	31
2.	Estudo dos atributos mais significativos .....	33
2.1	Estudo dos atributos mais significativos para identificação do nível de exaustão .....	33
2.2	Estudo dos atributos mais significativos para identificação da tarefa em execução .....	33
2.3	Conclusões .....	33
3.	Seleção dos algoritmos .....	34
3.1	Conclusões .....	34
4.	Conclusões da Caracterização da Solução .....	35
Conclusão .....		36
Anexo A .....		39
Anexo A.1 – Exaustão com valores não normalizados e aproximação não linear à função de ativação .....		39
Anexo A.1.1 – Cenário 1 .....		39
Anexo A.1.2 – Cenário 2 .....		39
Anexo A.1.3 – Cenário 3 .....		39
Anexo A.2 – Exaustão com valores não normalizados e aproximação linear à função de ativação .....		40
Anexo A.2.1 – Cenário 1 .....		40
Anexo A.2.2 – Cenário 2 .....		40
Anexo A.2.3 – Cenário 3 .....		40

## Lista de Figuras

Figura 1 – Estrutura de um neurónio e respetiva sinapse.....	7
Figura 2 – Distribuição estatísticas dos oito tipos de dados biométricos .....	10
Figura 3 – Amostra dos dados biométricos normalizados no intervalo [-1, 1] .....	11
Figura 4 – Representação gráfica estatística dos dados biométricos .....	12
Figura 5 – Representação gráfica estatística da exaustão.....	12
Figura 6 – Planeamento dos testes a realizar.....	14
Figura 7 – Topologia de rede para identificação do nível de exaustão .....	32
Figura 8 – Topologia de rede para identificação da tarefa em execução .....	32

## Índice de Tabelas

Tabela 1 – Representação das categorias e os respectivos valores discretos no intervalo $[-1, 1]$ .....	11
Tabela 2 – Atributos e respetiva significância para identificação da exaustão .....	13
Tabela 3 – Atributos e respetiva significância para identificação da tarefa em execução.....	13
Tabela 4 – Escala dos 7 níveis de exaustão e respetivos valores normalizados em $[0, 1]$ .....	18
Tabela 5 – Resultados dos testes com valores contínuos para o cenário 1 com tarefa como input .....	19
Tabela 6 – Resultados dos testes com valores contínuos para o cenário 1 sem tarefa como input .....	19
Tabela 7 – Resultados dos testes com valores discretos para o cenário 1 com tarefa como input.....	20
Tabela 8 – Resultados dos testes com valores discretos para o cenário 1 sem tarefa como input.....	20
Tabela 9 – Resultados dos testes com valores contínuos e discretos, com e sem a tarefa como input.....	21
Tabela 10 – Escala dos 7 níveis de exaustão e respetivos valores normalizados em $[0, 1]$ com dois níveis.....	22
Tabela 11 – Resultados dos testes com valores contínuos para o cenário 2 com tarefa como input .....	23
Tabela 12 – Resultados dos testes com valores contínuos para o cenário 2 sem tarefa como input .....	23
Tabela 13 – Resultados dos testes com valores discretos para o cenário 2 com tarefa como input.....	23
Tabela 14 – Resultados dos testes com valores discretos para o cenário 2 sem tarefa como input.....	24
Tabela 15 – Resultados dos testes com valores contínuos e discretos, com e sem a tarefa como input.....	24
Tabela 16 – Valores normalizados para todas as escalas utilizadas no cenário 3 .....	25
Tabela 17 – Resultados dos testes com valores contínuos para o cenário 3 sem tarefa como input .....	26
Tabela 18 – Resultados dos testes com valores discretos para o cenário 3 com tarefa como input.....	27
Tabela 19 – Resultados dos testes com valores discretos para o cenário 3 sem tarefa como input.....	27
Tabela 20 – Resultados dos testes com escala normalizada com valores 1,2,3 .....	29
Tabela 21 – Resultados dos testes com escala normalizada com valores 0, 0.5, 1.....	29
Tabela 22 – Topologias de rede utilizadas nos testes .....	30
Tabela 23 – Resultados de testes com diferentes topologias para a melhor escala da exaustão .....	30
Tabela 24 – Resultados de testes com diferentes topologias para a melhor escala de identificação da tarefa	31
Tabela 25 – Resultados de testes com diferentes topologias para a identificação da exaustão e da tarefa.....	31
Tabela 26 – Resultados de testes com diferentes números de atributos para a exaustão .....	33
Tabela 27 – Resultados de testes com diferentes números de atributos para a tarefa .....	33
Tabela 28 – Resultados de testes com diferentes algoritmos para identificação da exaustão .....	34
Tabela 29 – Resultados de testes com diferentes algoritmos para identificação da tarefa .....	34
Tabela 30 – Caracterização da solução para identificação do nível de exaustão e da tarefa .....	35

# Introdução

O terceiro trabalho de grupo da unidade curricular de Sistemas de Representação de Conhecimento e Raciocínio do 3º ano do Mestrado Integrado em Engenharia Informática da Universidade do Minho consiste na utilização de um sistema não simbólico para representação de conhecimento e desenvolvimento de mecanismos de raciocínio. Nomeadamente, ir-se-á utilizar Redes Neurais Artificiais (RNAs) para resolução de problemas.

Com este trabalho pretende-se realizar um estudo, envolvendo um conjunto dados biométricos para deteção de exaustão, recolhidos da interação humano-computador através de dispositivos físicos (rato e teclado). O estudo envolve a identificação da **exaustão** e da **tarefa** em execução, analisando a capacidade de resolução do problema por uma ou mais Redes Neurais Artificiais (RNAs).

Para identificação da **exaustão** serão estudados os seguintes cenários:

1. Identificação dos sete níveis de exaustão
2. Identificação da existência ou ausência de exaustão
3. Determinação da melhor escala de exaustão

Para identificação da **tarefa** em execução, deverão ser exploradas soluções para classificar as atividades como: *work*, *office* ou *programming*.

Este estudo será realizado recorrendo aos dados biométricos referidos anteriormente, recorrendo ao ambiente de análise de dados R e a Redes Neurais Artificiais (RNAs), sendo realizados:

- Estudo dos atributos mais significativos para a representação do conhecimento do problema em análise
- Identificação das topologias mais adequadas:
  - Número de camadas intermédias
  - Quantidade de neurónios:
    - na camada de entrada
    - na camada de saída
    - nas camadas intermédias
- Seleção das regras de aprendizagem para treinar as redes

*"The rung of a ladder was never meant to rest upon, but only to hold a man's foot long enough to enable him to put the other somewhat higher."*

- Thomas Huxley

# Preliminares

A abordagem da Inteligência Artificial à representação de conhecimento expõe-se através de dois paradigmas:

- **Simbólico** (abordado nos dois trabalhos anteriores)
  - Baseia-se na lógica para representar conhecimento
  - Fundamenta o raciocínio na construção de sistemas de inferência
- **Não simbólico**
  - Baseia o funcionamento do sistema na capacidade de aprender
  - Resolve problemas com base em conhecimento passado ou dados sobre a resolução de outros problemas

O objetivo deste trabalho prático é a utilização de sistemas não simbólicos na representação de conhecimento e no desenvolvimento de mecanismos de raciocínio, nomeadamente Redes Neurais Artificiais (RNAs).

Uma abordagem **não simbólica** a um problema, como por exemplo recorrendo às RNAs, é tem as seguintes vantagens:

- Capacidade de aprender a resolver problemas
- Capacidade de generalizar a resolução de problemas
- Tolerante a falhas

Uma **Rede Neuronal Artificial (RNA)** é um sistema computacional de base conexionista para a resolução de problemas. É um modelo simplificado do sistema nervoso central dos seres humanos.

É uma estrutura definida por uma estrutura interligada de unidades computacionais, designadas **neurónios**, com capacidade de aprendizagem.

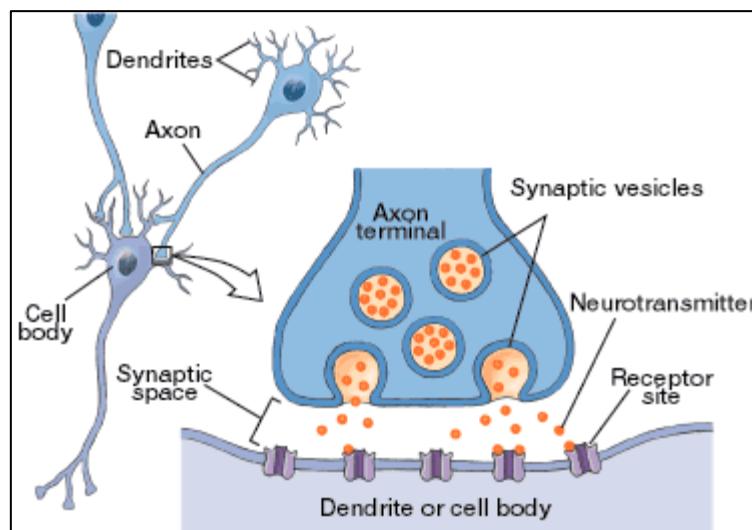


Figura 1 – Estrutura de um neurónio e respetiva sinapse

Uma rede neuronal artificial é constituída por elementos simples, semelhantes a neurónios biológicos chamados **nodos**. A rede é um conjunto ligações direcionadas, com **pesos** associados entre os nodos. Cada unidade processa uma função de um número limitado de **saídas** de outros nodos da rede. Essas saídas são pesadas e tornam-se as **entradas** da unidade.

As informações armazenadas por uma rede neuronal são compartilhadas por todas as suas unidades de processamento, daí possuir uma base **conexionista**.

Os neurónios estão ligados por topologias de redes constituídas por uma camada de **entrada**, uma ou mais camadas **intermédias** e uma camada de **saída**.

Para treinar a rede, é necessário fazer a escolha de um método para a treinar. O **treino** da rede consiste, basicamente, na apresentação de padrões de dados nas suas entradas, propagando a informação até se obter novos valores nas saídas. Por fim, tendo em atenção os *outputs* obtidos, alteram-se os pesos das ligações entre os neurónios da rede.

Existem duas formas opostas de treinar uma rede neuronal:

**Aprendizagem supervisionada:** Neste tipo de aprendizagem é avaliada a resposta da rede ao padrão atual de *inputs*. As alterações dos pesos são calculadas de forma a que a resposta da rede tenda a coincidir com a avaliação. É o tipo de aprendizagem em que nos é relativamente fácil avaliar o desempenho da rede para um determinado estado do nosso sistema.

**Aprendizagem não supervisionada:** Nesta forma de aprendizagem não existem casos de treino. A rede tem de descobrir sozinha relações, padrões, regularidades ou categorias nos dados que lhe vão sendo apresentados e codificá-las nas saídas.

## Dados Fornecidos

Neste trabalho prático, o objetivo é determinar o nível de exaustão e a tarefa em execução. Tal é possível devido aos valores das biométricas comportamentais que são fornecidas no *dataset*.

- **"Performance.KDTMean"** – tempo médio entre o momento em que a tecla é pressionada para baixo e o momento em que é largada;
- **"Performance.MAMean"** – aceleração do manuseamento rato em determinado momento. O valor da aceleração é calculado através da velocidade do rato (pixel/milissegundos) sobre o tempo de movimento (milissegundos);
- **"Performance.MVMean"** – velocidade do manuseamento do rato em determinado momento. A distância percorrida pelo rato (em pixeis) entre uma coordenada C1 (x1; y1) e uma C2 (x2; y2) correspondentes a time1 e time2, sobre o tempo (em milissegundos);
- **"Performance.TBCMean"** – tempo entre dois *clicks* consecutivos, entre eventos consecutivos MOUSE\_UP e MOUSE\_DOWN;
- **"Performance.DDCMean"** – período de tempo entre dois eventos MOUSE\_UP consecutivos;
- **"Performance.DMSMean"** – distância média em excesso entre o caminho de dois *clicks* consecutivos;



- **“Performance.ADMSLMean”** – distância média das diferentes posições do ponteiro entre dois pontos durante um movimento, e o caminho em linha reta entre esses mesmos dois pontos;
- **“Performance.AEDMean”** – esta métrica é semelhante à anterior, no sentido em calculará a soma da distância entre dois eventos MOUSE\_UP e MOUSE\_DOWN consecutivos;
- **“ExhaustionLevel”** – nível subjetivo de exaustão mental;
- **“Performance.Task”** – identificação da tarefa em execução no momento da recolha dos dados.

Estes parâmetros fornecidos estão normalizados no intervalo [-1 e 1]. Com estes parâmetros a rede deve ser capaz de calcular o nível de exaustão do utilizador e a tarefa em execução.

O *dataset* disponibilizado possui oito tipos de dados biométricos, que serão usados para treinar a rede e determinar o nível de exaustão do utilizador e a tarefa em execução.

Adicionalmente, no *dataset*, existem duas colunas, uma com a indicação do nível de exaustão (de 1 a 7) e outra coluna com a indicação da tarefa em execução: *work*, *office* ou *programming*.

Tabela 1.  
Excerto de dados biométricos para deteção de exaustão

KDT Mean	MA Mean	MV Mean	TBC Mean	DDC Mean	DMS Mean	AED Mean	ADMSL Mean	Exhaustion Level	Task
0,685	0,036	0,042	-0,086	0,099	0,000	-0,007	0,016	4	Work
0,022	0,038	0,042	-0,055	-0,016	0,002	0,014	0,058	4	Work
0,002	0,047	0,048	-0,095	0,524	0,008	-0,024	0,113	5	Work
0,000	0,052	0,058	-0,082	0,008	-0,011	0,023	-0,011	5	Work
0,008	0,077	0,081	-0,078	0,047	0,019	0,061	0,084	5	Work
0,002	0,031	0,036	-0,101	-0,009	-0,005	-0,021	0,000	5	Work
0,002	-0,029	-0,019	-0,099	-0,060	-0,007	0,046	-0,037	5	Work
0,074	0,018	0,032	-0,063	-0,045	0,000	-0,031	-0,001	3	Work
...	...	...	...	...	...	...	...	...	...

Cada um dos sete níveis de exaustão da escala, desenvolvida pelo Dr. Layne Perelli no USA Army, e a mais utilizada por investigadores de fatores humanos/ergonomia que estudam o fenómeno da exaustão mental, pode ser interpretado da seguinte forma:

1. Totalmente bem;
2. Responsivo, mas não no pico;
3. Ok, normal;
4. Em baixo de forma/do normal, a sentir-se em baixo;
5. Sentido moleza, perdendo o foco;
6. Muito difícil concentrar, meio tonto;
7. Incapaz de funcionar, pronto a “desligar”.

## Parte 1: Estudo Inicial

Antes de se proceder à identificação da **exaustão** e da **tarefa** em execução, recorrendo os dados biométricos fornecidos, realizou-se um estudo inicial, de forma a definir alguns parâmetros para treino das respetivas RNAs, de forma a que os testes possam ser fiáveis.

Começou-se por fazer um estudo inicial e estabelecer um planeamento para definir alguns parâmetros e ter testes *standard* que permitam que estes sejam comparáveis. Isto é, para o estudo da influência de uma determinada variável definiu-se o conjunto de testes que deve ser reproduzido quando se varia essa variável. Para tal, neste estudo inicial definiu-se:

- Como representar os dados biométricos
- Atributos (dados biométricos) a serem utilizados como *input* para a **exaustão** e para a **tarefa**
- Conjunto de tipologias teste a serem testadas

No entanto, depois de identificada a melhor forma de representar os níveis de **exaustão** (Parte 2) e de identificação da **tarefa** (Parte 3), na Parte 4 do trabalho estuda-se a influência destes parâmetros, variando-os.

Além disto, também se analisaram os dados biométricos fornecidos, para compreender como estão a ser representados e testar se, variando a forma de os representar, se obtêm melhores resultados.

### 1.1 Análise dos dados biométricos

Foram fornecidas 844 entradas no *dataset* em que cada um dos 8 atributos biométricos fornecidos está normalizado e representado com valores compreendidos no intervalo  $[-1, 1]$ . Este intervalo possui um conjunto infinito de elementos e por isso, decidiu-se analisar estaticamente a sua distribuição.

Para tal, recorreu-se ao *Weka*, *software* que disponibiliza um conjunto de algoritmos de *machine learning* para realização de *data mining*, para compreender como os dados dos oitos atributos de dados biométricos fornecidos se encontram distribuídos.

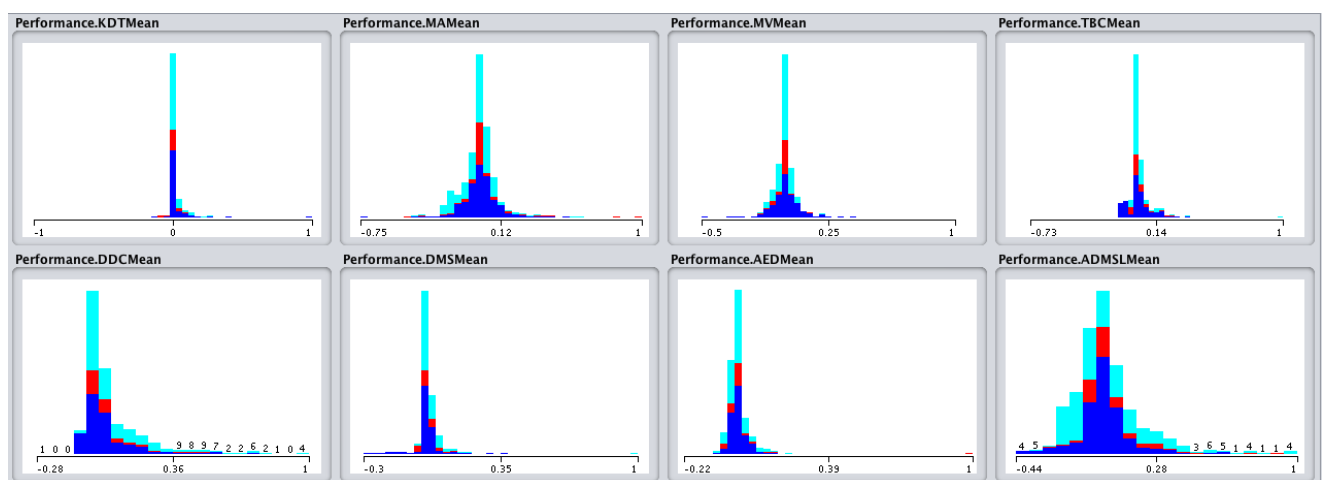


Figura 2 – Distribuição estatísticas dos oito tipos de dados biométricos

Observando a Figura 2 verifica-se que todos os dados biométricos apresentam uma distribuição normal, centrada na média, unimodal e simétrica. Portanto, os dados apresentam uma distribuição em que para as 844 entradas, os dados se concentram mais em determinados valores, para cada dado biométrico. Isto foi confirmado pelos valores obtidos para o desvio padrão.

Por esse motivo, decidiu-se discretizar estes valores recorrendo ao ambiente de análise de dados R para obter uma distribuição uniforme discreta. Optou-se por definir 10 categorias para cada tipo de dado biométrico, onde os resultados de cada categoria têm hipóteses iguais de ocorrerem.

Nesta forma de representação dos dados, optou-se por continuar a ter os dados normalizados no intervalo [-1, 1] contudo agora com valores discretos. Para tal, utilizou-se a seguinte fórmula:

$$x'' = 2 \frac{x - \min x}{\max x - \min x} - 1$$

Na Tabela 1 encontra-se a representação das categorias e os respetivos valores discretos no intervalo [-1, 1].

	Valores discretos
Categoria 1	-1
Categoria 2	-0,777777778
Categoria 3	-0,555555556
Categoria 4	-0,333333333
Categoria 5	-0,111111111
Categoria 6	0,111111111
Categoria 7	0,333333333
Categoria 8	0,555555556
Categoria 9	0,777777778
Categoria 10	1

**Tabela 1** – Representação das categorias e os respetivos valores discretos no intervalo [-1, 1]

Na Figura 3 encontra-se representado alguns dos dados.

ID	Performance.KDTMean	Performance.MAMean	Performance.MVMean	Performance.TBCMean	Performance.DDCMean	Performance.DMSMean	Performance.AEDMean	Performance.ADSLMean
1	-1.000000000	-1.000000000	-1.000000000	-1.000000000	-1.000000000	-1.000000000	-1.000000000	-1.000000000
2	-0.777777778	-0.777777778	-0.777777778	-0.777777778	-0.777777778	-0.777777778	-0.777777778	-0.777777778
3	-0.555555556	-0.555555556	-0.555555556	-0.555555556	-0.555555556	-0.555555556	-0.555555556	-1.000000000
4	-0.333333333	-0.333333333	-0.333333333	-1.000000000	-0.333333333	-0.333333333	-0.333333333	-0.555555556
5	-0.111111111	-0.111111111	-0.111111111	-0.777777778	-0.777777778	-0.111111111	-0.777777778	-0.333333333
6	-0.555555556	0.111111111	0.111111111	-0.333333333	-0.111111111	0.111111111	-0.111111111	-0.777777778
7	0.111111111	-0.111111111	0.333333333	-1.000000000	-1.000000000	0.111111111	0.111111111	-0.333333333
8	-0.333333333	0.111111111	0.555555556	-0.111111111	-0.333333333	0.333333333	-0.555555556	-0.111111111
9	0.111111111	0.333333333	0.777777778	0.111111111	0.111111111	0.555555556	0.333333333	-0.333333333
10	-0.555555556	0.555555556	0.777777778	0.333333333	0.111111111	0.777777778	0.111111111	-0.555555556
11	-0.333333333	0.333333333	0.777777778	0.555555556	0.333333333	-0.777777778	-0.555555556	-0.777777778
12	0.111111111	-0.555555556	-0.555555556	-0.333333333	-1.000000000	0.333333333	0.111111111	0.111111111
13	0.333333333	0.111111111	0.555555556	-0.333333333	0.555555556	-0.777777778	-0.555555556	0.111111111
14	-0.777777778	0.555555556	-0.111111111	-1.000000000	-1.000000000	0.111111111	0.555555556	-0.111111111
15	0.555555556	-0.555555556	0.333333333	0.333333333	-1.000000000	-0.111111111	0.111111111	0.111111111
16	0.111111111	-0.333333333	0.555555556	0.777777778	0.333333333	-0.333333333	-1.000000000	-0.555555556
17	-0.777777778	0.777777778	0.555555556	1.000000000	-0.111111111	0.777777778	0.111111111	0.333333333
18	0.777777778	0.555555556	-0.111111111	-0.333333333	0.777777778	-0.555555556	-0.777777778	-1.000000000
19	-0.555555556	-0.111111111	0.333333333	0.333333333	-0.111111111	0.111111111	-1.000000000	0.555555556
20	0.111111111	-0.555555556	-0.555555556	-0.333333333	-0.555555556	0.777777778	0.777777778	0.333333333
21	0.555555556	0.333333333	-0.333333333	0.333333333	0.777777778	0.111111111	-0.333333333	-0.777777778
22	-0.555555556	-0.333333333	0.555555556	0.555555556	0.333333333	0.555555556	-0.555555556	-0.333333333
23	-0.555555556	-1.000000000	-1.000000000	0.555555556	1.000000000	0.555555556	0.111111111	-0.333333333

**Figura 3** – Amostra dos dados biométricos normalizados no intervalo [-1, 1]

Observando a Figura 4, é possível confirmar que agora se obtém uma distribuição uniforme para cada um dos oito tipos de dados biométricos. Cada tipo encontra-se dividido em 10 categorias e para cada categoria tem-se um número aproximado de valores.

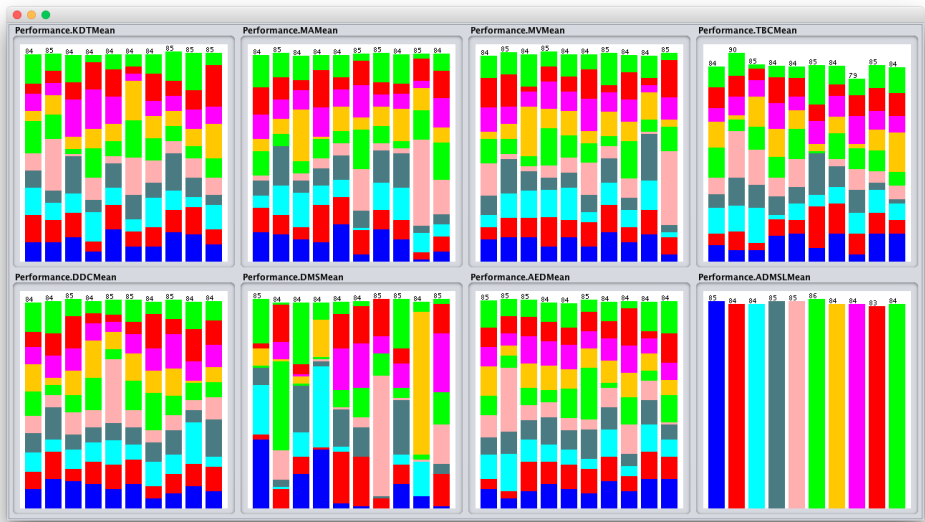


Figura 4 – Representação gráfica estatística dos dados biométricos

Representar os dados biométricos na forma como foram fornecidos e na forma discretizada são duas formas diferentes para representação do conhecimento. Na forma discretizada perde-se conhecimento, porém, espera-se que tal seja suportável neste tipo de ambiente dado a distribuição inicial dos valores.

Considerou-se interessante ter as duas formas de representação de conhecimento nos testes que se seguem de forma a analisar a sua influência nos resultados.

De referir, que inicialmente, ao observar os dados fornecidos, observou-se que os valores estavam demasiado concentrados nos níveis de menor exaustão. Havendo muito poucos casos para os maiores níveis de exaustão, decidiu-se distribuir aleatoriamente os dados no Microsoft Excel, para garantir a máxima representação dos dados nos casos de treino e de teste.

### 1.2 Estudo dos atributos mais significativos

- Exaustão

Numa primeira análise à representação da exaustão, na forma como os dados foram fornecidos, ou seja, através de 7 níveis, obteve-se a Figura 5.

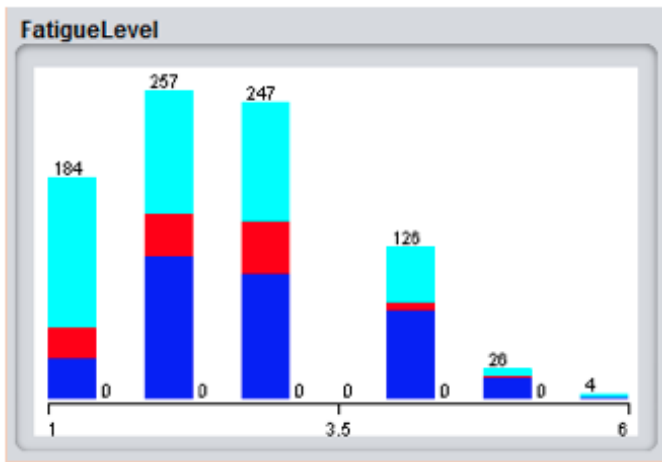


Figura 5 – Representação gráfica estatística da exaustão

Na Figura 5 verifica-se que mais de 80% dos casos de entrada correspondem aos três primeiros níveis de exaustão sendo que os restantes 20% se encontram nos últimos quatro níveis. Estes últimos quatro níveis estão pouco representados no conjunto de dados fornecidos, salientando que, **relativamente ao nível 7 não se possui qualquer dado**. Por esse motivo, um dos desafios é identificar a melhor forma de representar a exaustão.

É necessário, então, determinar quais os atributos que irão ser utilizados como *inputs* da RNA para identificar o nível de exaustão. Para tal, recorreu-se ao *regsubsets* do ambiente de análise de dados R e obteve-se os seguintes resultados, ordenados do atributo mais significativo para o menos significativo:

Nº	Atributo
1	Performance.Task
2	Performance.DDCMean
3	Performance.MAMean
4	Performance.MVMean
5	Performance.KDTMean
6	Performance.DMSMean
7	Performance.ADMSLMean
8	Performance.TBCMean
9	Performance.AEDMean

**Tabela 2** – Atributos e respetiva significância para identificação da exaustão

A tarefa é o atributo mais significativo para determinar a exaustão, pois, apresenta uma correlação muito forte. Uma vez que, devem estar diretamente relacionadas, decidiu-se testar esta influência na parte 2, fazendo teste com e sem a tarefa como atributo para determinar a exaustão.

#### ■ Tarefa

Determinou-se, também, quais os atributos que irão ser utilizados como inputs da RNA para a identificação da tarefa, recorrendo novamente ao *regsubsets*, e obteve-se os seguintes resultados, ordenados do mais significativo para o menos significativo:

Nº	Atributo
1	FatigueLevel
2	Performance.KDTMean
3	Performance.DMSMean
4	Performance.DDCMean
5	Performance.AEDMean
6	Performance.ADMSLMean
7	Performance.MAMean
8	Performance.MVMean
9	Performance.TBCMean

**Tabela 3** – Atributos e respetiva significância para identificação da tarefa em execução

Tal como se verificou anteriormente, a tarefa e a exaustão parecem estar diretamente relacionadas. A exaustão é o atributo mais significativo para determinar a tarefa, por isso, decidiu-se testar esta influência na Parte 3, fazendo testes com e sem a exaustão como atributo para determinar a tarefa.

Uma vez que o atributo tarefa é representado no *dataset* por uma *string* optou-se por normalizar o seu conjunto de valores. Numa primeira abordagem, realizou-se testes com o atributo **Performance.Task** (tarefa) normalizado de duas formas:

- **Escala de 1 a 3:**
  - 1 - *Work*
  - 2 - *Office*
  - 3 - *Programming*
- **Escala de 0 a 1:**
  - 0 - *Work*
  - 0.5 - *Office*
  - 1 - *Programming*

Sendo a escala normalizada com valores entre 0 e 1 a que obteve melhores resultados nos testes, foi a escolhida para utilizar nos testes das fases seguintes.

### 1.3 Planeamento dos Testes

Tendo já definido os atributos mais significativos e compreendido como estão representados os *inputs* e os *outputs*, definiu-se o conjunto de testes a ser realizado para cada estudo e como interligá-los. Na Figura 6 encontra-se o planeamento a seguir.

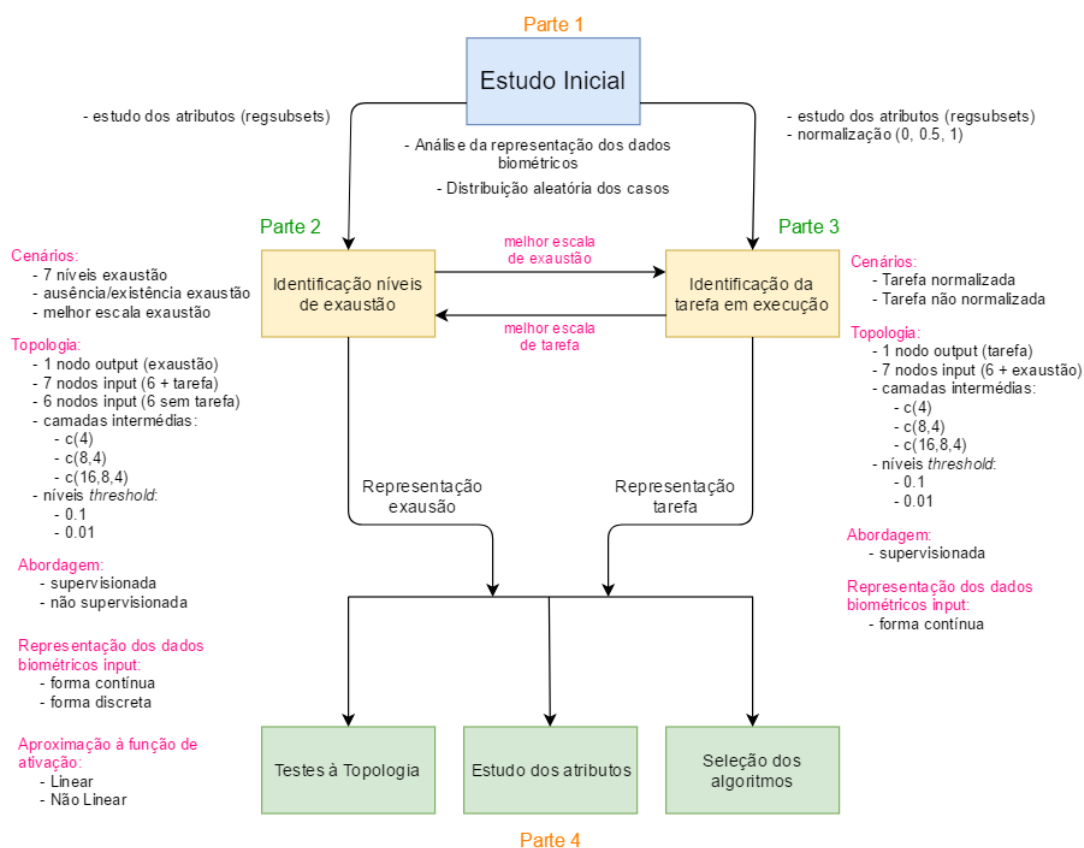


Figura 6 – Planeamento dos testes a realizar

- Planeamento dos testes da Parte 2: Identificação dos níveis de Exaustão

Inicialmente, para **identificação dos níveis de exaustão**, decidiu-se testar três cenários:

- **Cenário 1: identificar os 7 níveis de exaustão**
  - **Abordagem supervisionada**
    - Dados biométricos representados de forma contínua
    - Dados biométricos representados de forma discreta
  - **Abordagem não supervisionada**
    - Dados biométricos representados de forma discreta
- **Cenário 2: identificar, simplesmente, a existência ou ausência de exaustão**
  - **Abordagem supervisionada**
    - Dados biométricos representados de forma contínua
    - Dados biométricos representados de forma discreta
- **Cenário 3: encontrar a melhor forma de identificar a exaustão**
  - **Abordagem supervisionada**
    - Dados biométricos representados de forma contínua
    - Dados biométricos representados de forma discreta

Para os testes dos cenários abordados anteriormente, usando Redes Neurais Artificiais, propôs-se a sua realização em dois paradigmas principais: **Aprendizagem Supervisionada** e **Aprendizagem Não Supervisionada**. As RNAs, na resolução de determinado problema, passam por um processo de aprendizagem, onde a rede procura extrair informação relevante de padrões que lhe são apresentados. Com a Aprendizagem Supervisionada, as redes neuronais artificiais são treinadas antes de serem usadas, enquanto que na Aprendizagem Não Supervisionada a rede aprende sem a necessidade de um conjunto de dados de treino.

De forma a, apenas, compreender como varia a identificação da exaustão nos 3 cenários, definiram-se os seguintes casos teste:

- **Nodos *input*:**
  - atributos mais significativos (6 dos 8) + tarefa
  - atributos mais significativos (6 dos 8)
- **Nodo *ouput*:** FatigueLevel
- **Camadas intermédias:**
  - c(4)
  - c(8,4)
  - c(16,8,4)
- **Valores de *threshold*:**
  - 0.1
  - 0.01
- **Aproximação à função de ativação:**
  - linear
  - não linear

Optou-se por estas 3 topologias ( **c(4)**, **c(8,4)**, **c(16,8,4)** ) porque o objetivo neste ponto é encontrar a melhor forma para identificação da exaustão e não para estudar a melhor topologia. Espera-se que a melhor topologia esteja algures entre estas 3 topologias. Tal será estudado mais adiante, na Parte 4, após determinar a melhor forma de representar a exaustão (Parte 2). Também se optou por analisar apenas 2 valores para o *threshold*.

**Definiu-se, então, que para a abordagem supervisionada:**

- 2/3 de casos para treino
  - 1/3 de casos para teste
- 
- Planeamento dos testes da Parte 3: Identificação das tarefas em execução

Para a **identificação da tarefa em execução**, decidiu-se testar:

▪ **Abordagem supervisionada**

- Representar a tarefa na escala de 1-3
  - *Work*
  - *Office*
  - *Programming*
- Representar a tarefa na escala normalizada (0, 0.5, 1)
- Dados biométricos representados de forma continua
- Dados biométricos representados de forma discreta

**Definiu-se, então, que para a abordagem supervisionada:**

- 2/3 de casos para treino
- 1/3 de casos para teste

Da mesma forma que na exaustão, com o objetivo de apenas compreender como varia a identificação da tarefa, definiram-se os seguintes casos teste:

- **Nodos *input*:**

- atributos mais significativos (6 dos 8) + exaustão
- atributos mais significativos (6 dos 8)

- **Nodo *ouput*:** Performance.Task

- **Camadas intermédias:**

- c(4)
- c(8,4)
- c(16,8,4)

- **Valores de *threshold*:**

- 0.1
- 0.01

Da mesma forma que na exaustão, e para manter coerência, na tarefa optou-se por três topologias ( **c(4)**, **c(8,4)**, **c(16,8,4)** ) para identificar a melhor forma de representar a tarefa e os mesmos valores de *threshold*.



- [Planeamento dos testes da Parte 4: Caracterização da Solução](#)

Tendo sido determinado nas partes 2 e 3 qual a melhor forma de representar a exaustão e a tarefa, respetivamente, na parte 4 são realizados vários testes com essas representações, variando as variáveis que podem influenciar:

- Variação do número de atributos significativos
- Variação da topologia
- Testes de algoritmos

## Parte 2: Identificação dos níveis de Exaustão

Para identificar a melhor forma de definir os níveis de exaustão estudaram-se **três cenários**. Tal como definido na parte 1, vai-se sempre testar o seguinte conjunto de testes:

- **Nodos *input*:**
  - atributos mais significativos (6 dos 8) + exaustão
  - atributos mais significativos (6 dos 8)
- **Nodo *output*:** Performance.Task
- **Camadas intermédias:**
  - c(4)
  - c(8,4)
  - c(16,8,4)
- **Valores de *threshold*:**
  - 0.1
  - 0.01

Inicialmente os primeiros testes (que podem ser consultados no Anexo A.2) foram realizados com uma aproximação linear à função de ativação e com valores da exaustão nãoanexo normalizados. Contudo, após uma análise verificou-se que se obtinha melhores resultados com uma aproximação não linear à função de ativação. Portanto, todos os resultados que se seguem foram obtidos com uma aproximação não linear à função utilizada por defeito pela biblioteca *neuralnet*: **logistic**.

Concluiu-se também que se obteve melhores resultados quando se representa a exaustão normalizada com valores num intervalo de 0 a 1 (do que com valores discretos de 1 a 7, por exemplo). Por isso, em cada cenário, optou-se por fazer todos os testes representando a exaustão de forma normalizada.

### 1. Cenário 1: Identificar os sete níveis de exaustão

O objetivo do cenário 1 é identificar os 7 níveis de exaustão. Começou-se por normalizar os valores da exaustão para 7 níveis, tabela 4.

	Valores normalizados em [0, 1]
1	0
2	0.166667
3	0.333333
4	0.5
5	0.666667
6	0.833333
7	1

Tabela 4 – Escala dos 7 níveis de exaustão e respetivos valores normalizados em [0, 1]

Utilizaram-se os valores normalizados observados na tabela 4 em todos os testes do Cenário 1.

## 1.1 Abordagem Supervisionada

Na abordagem supervisionada definiu-se utilizar 2/3 dos casos do *dataset* para treino e 1/3 dos casos do *dataset* para testes.

### 1.1.1 Representação dos dados biométricos de forma contínua

Utilizando os 6 atributos mais significativos para a exaustão (identificados na Parte 1) como *input* com valores biométricos tal como foram fornecidos (representados de forma contínua), sem qualquer tratamento, fez-se o conjunto de testes definidos, testando duas situações: utilizando a tarefa como *input* e não utilizando.

#### 1.1.1.1 Utilizando a tarefa como *input*

Utilizando 7 atributos (os 6 mais significativos para a exaustão identificados na parte 1 + tarefa) como *input* com valores biométricos tal como foram fornecidos (representados de forma contínua), sem qualquer tratamento, fez-se o conjunto de testes definido. Na tabela 5 encontram-se os resultados obtidos.

Teste	Níveis	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
1	7			4	52	0.03 secs	8.56108	0.1965712775	0.1	Default	7
				4	1035	0.49 secs	7.54651	0.1986829698	0.01	Default	7
			8	4	19	0.02 secs	8.82652	0.1954172773	0.1	Default	7
			8	4	4109	4.31 secs	5.46483	0.2215649336	0.01	Default	7
		16	8	4	44	0.09 secs	8.46505	0.1978364743	0.1	Default	7
		16	8	4	28875	25.91 secs	1.92902	0.2812945544	0.01	Default	7

Tabela 5 – Resultados dos testes com valores contínuos para o cenário 1 com tarefa como *input*

#### 1.1.1.2 Não utilizando a tarefa como *input*

Tal como definido na parte 1, fez-se o conjunto de testes utilizando apenas os 6 atributos mais significativos (sem a tarefa como *input*). Contudo, como se pode observar, os resultados dos testes foram muito aproximados àqueles em que a tarefa foi usada como *input*. Visto que não se regista grande diferença, utilizando a tarefa como *input* para identificação do nível de exaustão, podemos afirmar que apesar da sua forte correlação, a identificação não é dependente da tarefa.

Teste	Níveis	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
1	7			4	7	0 secs	9.26763	0.1853925367	0.1	Default	6
				4	577	0.25 secs	8.00144	0.1953634552	0.01	Default	6
			8	4	44	0.12 secs	9.04565	0.1884221941	0.1	Default	6
			8	4	7063	6.99 secs	5.75266	0.2199391756	0.01	Default	6
		16	8	4	18	0.04 secs	9.42519	0.1853885862	0.1	Default	6
		16	8	4	33245	1.29 mins	2.50301	0.3237464956	0.01	Default	6

Tabela 6 – Resultados dos testes com valores contínuos para o cenário 1 sem tarefa como *input*

### 1.1.2 Representação dos dados biométricos de forma discreta

Como observado anteriormente, não existem dados suficientes para a rede treinar os casos em que exaustão está com níveis muito elevados (nível 7, nível 1 quando normalizado) pois os dados estão distribuídos de forma normal. Por esse motivo, decidiu-se testar se a representação de dados de forma discreta, com distribuição uniforme, poderia melhorar os resultados.

Utilizando os 6 atributos mais significativos (identificados na parte 1) como *input* com os valores biométricos tratados de forma a serem representados de forma discreta (Parte 1), fez-se o conjunto de testes definidos, testando duas situações: utilizando a tarefa como *input* e não utilizando.

#### 1.1.2.1 Utilizando a tarefa como *input*

Utilizando os 6 atributos mais significativos para a exaustão (identificados na Parte 1) como *input* com valores biométricos tratados de forma a serem discretos (Parte 1), e também utilizando como *input* a tarefa, fez-se o conjunto de testes de definido. Na Tabela 7 encontram-se os resultados obtidos.

Teste	Níveis		C		Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
1	7			4	13	0.01 secs	8.96007	0.1815113417	0.1	Default	7
				4	1084	0.62 secs	7.39702	0.1978130284	0.01	Default	7
			8	4	25	0.03 secs	8.77928	0.1803179442	0.1	Default	7
			8	4	3850	4.08 secs	5.72629	0.220817271	0.01	Default	7
		16	8	4	28	0.07 secs	8.96461	0.1800190981	0.1	Default	7
		16	8	4	22306	1.05 mins	2.20555	0.2764409903	0.01	Default	7

Tabela 7 – Resultados dos testes com valores discretos para o cenário 1 com tarefa como *input*

#### 1.1.2.2. Não utilizando a tarefa como *input*

Tal como anteriormente, fizeram-se testes utilizando apenas os 6 atributos mais significativos e sem a tarefa como *input*. Contudo os resultados foram aproximados aos anteriores (quando utilizando a tarefa como *input*), não se registrando grande diferença.

Teste	Níveis		C		Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
1	7			4	58	0.04 secs	8.98378	0.1940050837	0.1	Default	6
				4	256	0.12 secs	8.24067	0.197940611	0.01	Default	6
			8	4	31	0.04 secs	8.92007	0.195763893	0.1	Default	6
			8	4	3810	4 secs	5.80341	0.2207261921	0.01	Default	6
		16	8	4	26	0.06 secs	9.12409	0.2000611565	0.1	Default	6
		16	8	4	7597	19.61 secs	2.21093	0.2767716552	0.01	Default	6

Tabela 8 – Resultados dos testes com valores discretos para o cenário 1 sem tarefa como *input*

## 1.1 Abordagem Não Supervisionada

Inicialmente, o objetivo seria utilizar aprendizagem não supervisionada para identificar os 7 níveis de exaustão, desafio proposto neste Cenário 1. Porém, devido a falta de tempo disponível e dificuldade de acesso a ferramentas para a realização dos testes, optou-se por não abordar este tipo de aprendizagem, prosseguindo com os testes e cenários seguintes, e com as restantes partes (3 e 4).

## 1.2 Conclusões

Inicialmente, visto que o *dataset* não disponibiliza casos com um nível 7 de exaustão, é esperado que a rede não consiga aprender. Se a rede não treina, então não saberá responder a esses casos. Por esse motivo, optou-se por descobrir qual a influência de duas abordagens diferentes: aprendizagem supervisionada e não supervisionada.

### ▪ Aprendizagem Supervisionada

Nesta abordagem, e observando os resultados da Tabela 9, não se verificaram praticamente diferenças utilizando ou não a tarefa como *input* para determinar o nível de exaustão. A exaustão e a tarefa, apesar de estarem ambas muito relacionadas, não são assim tão relevantes para a identificação da outra, porque podem ser identificadas independentemente.

Também, nos resultados dos testes com valores contínuos e discretos não se observaram diferenças notórias.

Apesar de se ter duas formas distintas de se representar o conhecimento, com valores contínuos e discretos para os dados biométricos, e com valores discretos de certa forma se perder informação, os resultados dos testes mostram que não interfere praticamente na identificação dos níveis de exaustão.

No entanto, eram esperados melhores resultados para os testes que utilizam os valores discretos, uma vez que passamos a ter um menor número de valores distintos, visto que antes os intervalos possuíam valores infinitos.

Por fim, seria interessante testar uma distribuição uniforme para a escala dos 7 níveis de exaustão, como trabalho futuro. Com este tipo de distribuição, todos os níveis de exaustão possuiriam o mesmo número de casos no *dataset*, o que permitiria que a rede tivesse os casos necessários de cada nível para o seu treino.

RMSE da Aprendizagem Supervisionada			
Valores contínuos		Valores discretos	
Com Tarefa	Sem Tarefa	Com Tarefa	Sem Tarefa
0.1954172773	0.1853925367	0.1803179442	0.1940050837

Tabela 9 – Resultados dos testes com valores contínuos e discretos, com e sem a tarefa como input.

## ▪ Aprendizagem Não Supervisionada

Seria interessante seguir uma abordagem não supervisionada para identificar os sete níveis de exaustão. Contudo, e com muito pouco disponível, e dificuldade no acesso a ferramentas para tal, não foi possível prosseguir com esta abordagem.

Neste paradigma de aprendizagem, ao contrário do paradigma de Aprendizagem Supervisionada, a rede aprende sem a necessidade de um conjunto de dados de treino. O que seria ideal neste cenário pois não possuímos dados para o nível 7 de exaustão.

## 2. Cenário 2: Identificar a existência ou ausência de exaustão

No cenário 2 identifica-se a exaustão simplesmente em: existência ou ausência de exaustão. Começou-se por normalizar os valores da exaustão dos 7 níveis para apenas 2 níveis (0: ausência de exaustão e 1: exaustão), como pode ser observado na tabela 10. Utilizaram-se estes valores normalizados em todos os testes do cenário 2.

Níveis de a 1 a 7	Valores normalizados de [0, 1]
1+2+3	0
4+5+6+7	1

Tabela 10 – Escala dos 7 níveis de exaustão e respetivos valores normalizados em [0, 1] com dois níveis

### 2.1 Abordagem Supervisionada

Neste cenário 2, seguiu-se apenas a abordagem supervisionada, e definiu-se utilizar 2/3 dos casos do *dataset* para treino e 1/3 dos casos *dataset* para testes.

#### 2.1.1 Representação dos dados biométricos de forma contínua

Utilizando os 6 atributos mais significativos para a exaustão (identificados na Parte 1) como *input* com valores biométricos tal como foram fornecidos (representados de forma contínua), sem qualquer tratamento, fez-se o conjunto de testes definidos, testando duas situações: utilizando a tarefa como *input* e não utilizando.

##### 2.1.1.1 Utilizando a tarefa como *input*

Utilizando 7 atributos (os 6 mais significativos para a exaustão identificados na parte 1 + tarefa) como *input* com valores biométricos tal como foram fornecidos (representados de forma contínua), sem qualquer tratamento, fez-se o conjunto de testes definido. Na Tabela 11 encontram-se os resultados obtidos.

Teste	Níveis	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
2	2			4	40	0.02 secs	39.7696	0.3892641667	0.1	Default	7
				4	9334	4.97 secs	34.43372	0.4487835535	0.01	Default	7
			8	4	3888	4.65 secs	19.71552	0.5064126121	0.1	Default	7
			8	4	10518	10.72 secs	14.78645	0.4885312607	0.01	Default	7
		16	8	4	3673	8.69 secs	9.68748	0.5047691965	0.1	Default	7
		16	8	4	3902	9.21 secs	7.01301	0.4882466829	0.01	Default	7

Tabela 11 – Resultados dos testes com valores contínuos para o cenário 2 com tarefa como *input*

#### 2.1.1.2 Não utilizando a tarefa como *input*

Tal como definido na parte 1, fez-se o conjunto de testes utilizando apenas os 6 atributos mais significativos e sem a tarefa como *input*. Contudo os resultados foram aproximados aos anteriores (quando utilizando a tarefa como *input*), não se registando grande diferença.

Teste	Níveis	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
1	7			4	7	0 secs	9.26763	0.1853925367	0.1	Default	6
				4	577	0.25 secs	8.00144	0.1953634552	0.01	Default	6
			8	4	44	0.12 secs	9.04565	0.1884221941	0.1	Default	6
			8	4	7063	6.99 secs	5.75266	0.2199391756	0.01	Default	6
		16	8	4	18	0.04 secs	9.42519	0.1853885862	0.1	Default	6
		16	8	4	33245	1.29 mins	2.50301	0.3237464956	0.01	Default	6

Tabela 12 – Resultados dos testes com valores contínuos para o cenário 2 sem tarefa como *input*

#### 2.1.2 Representação dos dados biométricos de forma discreta

Também no cenário 2 decidiu-se testar se a representação de dados biométricos de *input* de forma discreta poderia melhor os resultados uma vez que a distribuição foi feita uniformemente.

Como observado anteriormente, não existem dados suficientes para a rede treinar os casos em que exaustão está com níveis muito elevados (nível 7, nível 1 quando normalizado) pois os dados estão distribuídos de forma normal. Por esse motivo, decidiu-se testar a representação de dados de forma discreta, com distribuição uniforme.

##### 2.1.2.1 Utilizando a tarefa como *input*

Utilizando os 6 atributos mais significativos para a exaustão (identificados na Parte 1) como input com valores biométricos tratados de forma a serem discretos (Parte 1), e também utilizando como input a tarefa, fez-se o conjunto de testes de definido. Na tabela 13 encontram-se os resultados obtidos.

Teste	Níveis	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
2	2			4	40	0.02 secs	39.7696	0.3892641667	0.1	Default	7
				4	9334	4.97 secs	34.43372	0.4487835535	0.01	Default	7
			8	4	3888	4.65 secs	19.71552	0.5064126121	0.1	Default	7
			8	4	10518	10.72 secs	14.78645	0.4885312607	0.01	Default	7
		16	8	4	3673	8.69 secs	9.68748	0.5047691965	0.1	Default	7
		16	8	4	3902	9.21 secs	7.01301	0.4882466829	0.01	Default	7

Tabela 13 – Resultados dos testes com valores discretos para o cenário 2 com tarefa como *input*

### 2.1.2.2 Não utilizando a tarefa como *input*

Tal como anteriormente, fizeram-se testes utilizando apenas os 6 atributos mais significativos e sem a tarefa como *input*. Contudo os resultados foram aproximados aos anteriores (quando utilizando a tarefa como *input*), não se registando grande diferença.

Teste	Níveis	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
2	2			4	211	0.09 secs	35.19575	0.4102233964	0.1	Default	6
				4	4084	1.84 secs	32.3039	0.4358470649	0.01	Default	6
		8		4	1386	1.44 secs	20.51701	0.4779748402	0.1	Default	6
		8		4	7653	7.8 secs	17.01362	0.493896644	0.01	Default	6
		16	8	4	728	1.67 secs	11.76038	0.5316820716	0.1	Default	6
		16	8	4	575	1.23 secs	13.01005	0.5376729514	0.01	Default	6

Tabela 14 – Resultados dos testes com valores discretos para o cenário 2 sem tarefa como *input*

## 2.2 Conclusões

Neste cenário, os 7 níveis de exaustão foram convertidos em dois níveis. Assim sendo, a rede treina com dois níveis de exaustão: ausência (0) e existência (1) de exaustão.

Neste cenário apenas se abordou a aprendizagem supervisionada.

### ▪ Aprendizagem Supervisionada

Nesta abordagem, e observando os resultados da Tabela 15, não se verificaram praticamente diferenças utilizando ou não a tarefa como *input* para determinar o nível de exaustão. Também, nos resultados dos testes com valores contínuos e discretos não se observaram diferenças notórias.

Aprendizagem Supervisionada			
Valores contínuos		Valores discretos	
Com Tarefa	Sem Tarefa	Com Tarefa	Sem Tarefa
0.3892641667	0.3965600082	0.3869134014	0.4102233964

Tabela 15 – Resultados dos testes com valores contínuos e discretos, com e sem a tarefa como *input*

Dividir os sete níveis de exaustão leva a que perca expressividade na representação do conhecimento. Limitando os níveis a apenas dois, espera-se um erro relativamente baixo, pois apenas temos um caso de fronteira entre o 0 e o 1, e o erro não é acumulado sucessivamente. Tal não foi observado nos nossos testes, comparativamente ao cenário 1 com 7 níveis. Dado isto, procurar-se-á encontrar uma escala ideal no cenário 3: um *tradeoff* entre a expressividade e o erro, balanceando estes dois parâmetros.



### 3. Cenário 3: Determinar a melhor escala de identificação de exaustão

No cenário 3 pretende-se encontrar a melhor escala para identificar a exaustão. Dado isto, começou-se por normalizar os valores da exaustão para 7, 6, 5, 4, 3 e 2 níveis, como é possível observar na tabela 16.

Número de Níveis	Níveis de 1 a 7	Valores Normalizados
7	1	0
	2	0.166667
	3	0.333333
	4	0.5
	5	0.666667
	6	0.833333
	7	1
6	1	0
	2	0.2
	3	0.4
	4	0.6
	5	0.8
	6+7	1
5	1	0
	2	0.25
	3	0.5
	4	0.75
	5+6+7	1
4 a)	1	0
	2	0.333333
	3	0.666667
	4+5+6+7	1
4 b)	1+2	0
	3	0.333333
	4+5	0.666667
	6+7	1
3 a)	1+2	0
	3	0.5
	4+5+6+7	1
3 b)	1+2+3	0
	4+5	0.5
	6+7	1
2	1+2+3	0
	1+2	1

Tabela 16 – Valores normalizados para todas as escalas utilizadas no cenário 3

Utilizaram-se estes valores normalizados em todos os testes do cenário 3.

### 3.1 Abordagem Supervisionada

Neste cenário 2, seguiu-se apenas a abordagem supervisionada, e definiu-se utilizar 2/3 dos casos do *dataset* para treino e 1/3 dos casos *dataset* para testes.

#### 3.1.1 Representação dos dados biométricos de forma contínua

Utilizando os 6 atributos mais significativos para a exaustão (identificados na Parte 1) como *input* com valores biométricos tal como foram fornecidos (representados de forma contínua), sem qualquer tratamento, fez-se o conjunto de testes definidos, testando duas situações: utilizando a tarefa como *input* e não utilizando.

##### 3.1.1.1 Utilizando a tarefa como *input*

Utilizando 7 atributos (os 6 mais significativos para a exaustão identificados na parte 1 + tarefa) como *input* com valores biométricos tal como foram fornecidos (representados de forma contínua), sem qualquer tratamento, fez-se o conjunto de testes definido. Na tabela 17 encontram-se os resultados obtidos, sendo que foi selecionando o melhor teste para cada nível, os restantes encontram-se em Anexo A.1.3.

Teste	Níveis		C		Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
1	7			4	52	0.03 secs	8.56108	0.1965712775	0.1	Default	7
2	2			4	40	0.02 secs	39.7696	0.3892641667	0.1	Default	7
3	6			4	19	0.02 secs	12.7396	0.2180555616	0.1	Default	7
4	5		8	4	16	0.02 secs	19.29557	0.2703944142	0.1	Default	7
5	4 a)		8	4	52	0.07 secs	29.00826	0.336372276	0.1	Default	7
6	4 b)			4	28	0.02 secs	18.30965	0.2535125513	0.1	Default	7
7	3 a)			4	562	0.31 secs	34.82558	0.4042677352	0.1	Default	7
8	3 b)			4	13	0.01 secs	11.46125	0.1926864733	0.1	Default	7

Tabela 17 – Resultados dos testes com valores contínuos para o cenário 3 sem tarefa como *input*

##### 3.1.1.2 Não utilizando a tarefa como *input*

Uma vez que os resultados obtidos sem a tarefa como *input* foram idênticos aos obtidos com a tarefa como *input* nos cenários anteriores, optou-se por não seguir e explorar este caminho para o cenário 3.

#### 3.1.2 Representação dos dados biométricos de forma discreta

Como já referido, os dados fornecidos não são suficientes para a rede treinar os casos em que exaustão está com níveis muito elevados (nível 7, normalizado nível 1), uma vez que os dados se encontram mal distribuídos. Dado isto, decidiu-se testar se a representação de dados de forma discreta poderia melhorar os resultados uma vez que a distribuição foi feita uniformemente.

### 3.1.2.1 Utilizando a tarefa como *input*

Utilizando os 6 atributos mais significativos para a exaustão (identificados na Parte 1) como *input* com valores biométricos tratados de forma a serem discretos (Parte 1), e também utilizando como *input* a tarefa, fez-se o conjunto de testes de definido. Na tabela 18 encontram-se os resultados obtidos, sendo que apenas foram realizados testes para uma escala, a 3 b), visto ser a escala que obteve melhores resultados com valores contínuos.

Teste	Níveis		C		Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
8	3 b)			4	7	0.01 secs	11.44348	0.197625949	0.1	Default	7
				4	477	0.32 secs	10.01822	0.2196235326	0.01	Default	7
			8	4	10	0.01 secs	11.5412	0.1948919237	0.1	Default	7
			8	4	36543	51.31 secs	4.74766	0.3140228769	0.01	Default	7
		16	8	4	10	0.03 secs	11.65782	0.1954225078	0.1	Default	7
		16	8	4	25390	1.2 mins	2.64924	0.36008082	0.01	Default	7

Tabela 18 – Resultados dos testes com valores discretos para o cenário 3 com tarefa como *input*

### 3.1.2.2 Não utilizando a tarefa como *input*

Tal como anteriormente, fizeram-se testes utilizando apenas os 6 atributos mais significativos e sem a tarefa como *input*. Contudo os resultados foram aproximados aos anteriores (quando utilizando a tarefa como *input*), não se registando grande diferença. Na tabela 19 encontram-se os resultados obtidos, sendo que apenas foram realizados testes para uma escala, a 3 b), visto ser a escala que obteve melhores resultados com valores contínuos.

Teste	Níveis		C		Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
8	3 b)			4	20	0.01 secs	11.39981	0.1935674366	0.1	Default	6
				4	1803	0.89 secs	8.82957	0.221302568	0.01	Default	6
			8	4	6	0.01 secs	11.65275	0.1945566703	0.1	Default	6
			8	4	17877	17.98 secs	5.00997	0.2801911764	0.01	Default	6
		16	8	4	12	0.02 secs	11.27518	0.1938398611	0.1	Default	6
		16	8	4	6878	17.87 secs	2.70006	0.3225046047	0.01	Default	6

Tabela 19 – Resultados dos testes com valores discretos para o cenário 3 sem tarefa como *input*

## 3.2 Conclusões

Neste cenário, pode-se concluir, após os testes realizados, que o melhor resultado obtido foi o teste 8, com 3 níveis para a exaustão, utilizando uma topologia com uma camada intermédia com 4 neurónios, quer para testes com valores contínuos ou discretos para os *inputs*, com ou sem tarefa como *input*.

A rede aprendeu melhor com 3 níveis com a seguinte junção **(1+2+3, 4+5, 6+7)**, uma vez que se conseguiu distribuir melhor os valores da exaustão fornecidos, obtendo-se melhor resultados.

Não há uma relação direta entre um aumento do número de camadas intermédias e um melhor resultado, o que foi verificado nos testes efetuados. A rede aprendeu melhor somente com uma rede intermédia de 4 neurónios.

## Parte 3: Identificação da tarefa em execução

Para identificar a melhor forma de definir como identificar a tarefa, tal como definido na parte 1, vai-se sempre testar o seguinte conjunto de teste:

- **Nodos *input*:**
  - atributos mais significativos (6 dos 8) + exaustão
  - atributos mais significativos (6 dos 8)
- **Nodo *output*:** Performance.Task
- **Camadas intermédias:**
  - c(4)
  - c(8,4)
  - c(16,8,4)
- **Valores de *threshold*:**
  - 0.1
  - 0.01

Na Parte 3, tal como definido no planeamento de testes da parte 1, apenas se seguiu uma abordagem supervisionada utilizando 2/3 dos casos do *dataset* para treino e 1/3 dos casos para teste.

Tal como na identificação da exaustão, após analisar verificou-se que se obtinha melhores resultados com uma aproximação não linear à função de ativação. Portanto, todos os resultados que se seguem foram obtidos com uma aproximação não linear à função utilizada por defeito pela biblioteca *neuralnet* do R, a função *logistic*.

Tal como planeado na Parte 1, optou-se apenas por estudar a tarefa com valores normalizados, com os valores (1, 2 e 3) e com os valores (0, 0.5 e 1). Não se estudou o caso de valores representados por *string*.

Ao contrário do que se planeou na Parte 1, uma vez que anteriormente (Parte 2) se verificou que a ter a tarefa como atributo para identificar a exaustão não era assim tão determinante, optou-se por apenas testar as redes com nodo *output* tarefa e nodos *inputs* com 6 atributos significativos para representar a tarefa (identificados na Parte 1) + exaustão. Não se testando a influência da exaustão para determinar a tarefa uma vez que não se testa com 6 atributos sem exaustão.

Além disso, vai-se apenas utilizar os dados biométricos com valores contínuos, tal como fornecidos, uma vez que não se verificou (na parte 2) que a discretização dos valores tenha melhorado os resultados.

## 1.1 Abordagem Supervisionada

### 1.1.1 Tarefa normalizada com os valores 1, 2, 3

- Representar a tarefa na escala normalizada

- 1 – *work*
- 2 – *office*
- 3 – *programming*

Teste	Normalização	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
1	1, 2, 3			4	26	0 secs	272.06675	1.010632779	0.1	Default	7
				4	25	0.02 secs	272.00738	1.010521981	0.01	Default	7
			8	4	19	0.1 secs	272.06429	1.010626386	0.1	Default	7
			8	4	11	0.01 secs	272.00613	1.010519451	0.01	Default	7
		16	8	4	17	0.05 secs	272.07073	1.010638201	0.1	Default	7
		16	8	4	29	0.06 secs	272.00892	1.010524565	0.01	Default	7

Tabela 20 – Resultados dos testes com escala normalizada com valores 1,2,3

### 1.1.2 Tarefa normalizada com os valores 0, 0.5, 1

- Representar a tarefa na escala normalizada

- 0 – *work*
- 0.5. – *office*
- 1 – *programming*

Teste	Normalização	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
2	0, 0.5, 1			4	826	0.51 secs	24.53786	0.3007821933	0.1	Default	7
				4	1729	0.85 secs	25.33834	0.3327459028	0.01	Default	7
			8	4	1483	1.7 secs	15.50651	0.3068178505	0.1	Default	7
			8	4	69634	1.39 mins	14.15912	0.3574028373	0.01	Default	7
		16	8	4	5135	13.79 secs	6.09858	0.369483813	0.1	Default	7
		16	8	4	140719	6.59 mins	5.55225	0.3817488399	0.01	Default	7

Tabela 21 – Resultados dos testes com escala normalizada com valores 0, 0.5, 1

## 1.2 Conclusões

Com os testes realizados anteriormente, pode-se concluir que a escala normalizada com os valores 0, 0.5, 1 é aquela que apresenta melhores resultados nos testes. Por esse motivo, é a escala que irá ser utilizada nas etapas seguintes.

De referir que não se testou com a tarefa com os seus valores no formato *string*, tal como foram fornecidos no *dataset*.

## Parte 4: Caracterização da solução

Para a melhor solução obtida tanto para a identificação da exaustão, assim como para a identificação da tarefa em execução, realizou-se os seguintes testes, a fim de refinar os resultados obtidos:

- Testes à topologia
- Estudo dos atributos mais significativos
- Seleção dos algoritmos

### 1. Testes à topologia

As topologias de rede testadas encontram-se na Tabela 22.

Número de Camadas	Número de Nodos		
1			2
1			3
1			4
1			5
2		6	2
2		8	4
3	16	8	4

Tabela 22 – Topologias de rede utilizadas nos testes

Pode-se verificar pela tabela que se variou o número de camadas intermédias da rede, assim como o número de neurónios de cada camada. Para além disto para cada topologia também se variou o parâmetro do *threshold*, utilizando os valores 0.1 e 0.01.

Para a melhor solução obtida para a identificação da exaustão, usando 3 níveis de exaustão, obteve-se os seguintes resultados para as topologias anteriormente definidas:

Nº Output	Nodo Output	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos	
1	Exaustão			2	22	0.01 secs	11.5819	0.1942093876	0.1	Default	7	
				2	273	0.17 secs	10.9484	0.1969935354	0.01	Default	7	
				3	18	0.01 secs	11.46489	0.1956520874	0.1	Default	7	
				3	1085	0.42 secs	9.92114	0.2104646219	0.01	Default	7	
				4	13	0.01 secs	11.46125	0.1926864733	0.1	Default	7	
				4	758	0.42 secs	9.52855	0.2053502723	0.01	Default	7	
				5	29	0.02 secs	10.7887	0.1960295138	0.1	Default	7	
				5	869	0.44 secs	8.64574	0.2109107623	0.01	Default	7	
			6	2	7	0.01 secs	11.4953	0.1957959037	0.1	Default	7	
			6	2	2953	2.32 secs	7.33386	0.2341398288	0.01	Default	7	
				8	4	8	0.01 secs	11.42704	0.1942215437	0.1	Default	7
				8	4	69700	1.38 mins	4.84043	0.2908377477	0.01	Default	7
			16	8	4	10	0.03 secs	11.72264	0.1961388872	0.1	Default	7
			16	8	4	80477	3.44 mins	2.2636	0.3384903099	0.01	Default	7

Tabela 23 – Resultados de testes com diferentes topologias para a melhor escala da exaustão

Para a melhor solução obtida para a identificação da tarefa em execução, usando os valores normalizados 0 0.5, 1 para a tarefa, obteve-se os seguintes resultados para as topologias anteriormente definidas:

Nº Output	Nodo Output	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos	
1	Tarefa			2	176	0.07 secs	27.87305	0.3443234473	0.1	Default	7	
				2	2024	0.71 secs	27.82715	0.3422942813	0.01	Default	7	
				3	151	0.06 secs	26.57648	0.3419093456	0.1	Default	7	
				3	1934	0.8 secs	25.5488	0.3511263559	0.01	Default	7	
				4	826	0.51 secs	24.53786	0.3007821933	0.1	Default	7	
				4	1729	0.85 secs	25.33834	0.3327459028	0.01	Default	7	
				5	359	0.33 secs	24.07192	0.3452435287	0.1	Default	7	
				5	832	0.44 secs	23.25	0.3716060321	0.01	Default	7	
			6	2	513	0.38 secs	22.42699	0.3624313243	0.1	Default	7	
			6	2	4439	3.35 secs	19.40931	0.3707507923	0.01	Default	7	
				8	4	1483	1.7 secs	15.50651	0.3068178505	0.1	Default	7
				8	4	69634	1.39 mins	14.15912	0.3574028373	0.01	Default	7
			16	8	4	5135	13.79 secs	6.09858	0.369483813	0.1	Default	7
			16	8	4	140719	6.59 mins	5.55225	0.3817488399	0.01	Default	7

Tabela 24 – Resultados de testes com diferentes topologias para a melhor escala de identificação da tarefa

Para a rede com 2 outputs (tarefa + exaustão) obteve-se os seguintes resultados:

							RMSE						
Nº Output	Nodos Output	C			Steps	Tempo	Erro	Exaustão	Tarefa	Threshold	Algoritmo	Atributos	
2	Exaustão + Tarefa		2		139	0.06 secs	41.00003	0.1932555881	0.3449233102	0.1	Default	7	
			2		1304	0.51 secs	40.00957	0.1945429234	0.3374685435	0.01	Default	7	
			3		254	0.12 secs	39.00901	0.1978938001	0.3589358516	0.1	Default	7	
			3		1069	0.57 secs	38.1754	0.1973592569	0.3665342703	0.01	Default	7	
			4		137	0.08 secs	38.01858	0.1965853115	0.3608732712	0.1	Default	7	
			4		979	0.61 secs	36.90506	0.1991648011	0.3691446559	0.01	Default	7	
			5		454	0.29 secs	36.48877	0.1954598281	0.3628988921	0.1	Default	7	
			5		1317	0.76 secs	35.50279	0.1962234765	0.3768368628	0.01	Default	7	
			6	2	359	0.29 secs	35.54	0.2021217786	0.3652714538	0.1	Default	7	
			6	2	2786	2.36 secs	32.97331	0.1976382695	0.3656361078	0.01	Default	7	
			8	4	506	0.59 secs	32.60976	0.2123955596	0.3833277484	0.1	Default	7	
			8	4	12370	13.13 secs	27.01682	0.2191765701	0.4231115458	0.01	Default	7	
			16	8	4	4818	10.68 secs	15.66266	0.2710814692	0.4675392039	0.1	Default	7
			16	8	4	12155	58.78 secs	17.78578	0.2672644564	0.4787452543	0.01	Default	7

Tabela 25 – Resultados de testes com diferentes topologias para a identificação da exaustão e da tarefa

## 1.1 Conclusões

Após os testes à topologia, pode-se concluir que a melhor topologia de rede para identificação dos níveis de exaustão é uma rede com um único output, *FatigueLevel*, e com uma apenas uma camada intermédia com quatro nodos. A rede pode ser observada na Figura 7.

Pode-se também concluir que a melhor topologia de rede para da tarefa em execução é uma rede com um único output, *Performance.Task*, e com uma apenas uma camada intermédia com quatro nodos. A rede pode ser observada na Figura 8.

Apesar dos resultados da rede com dois *outputs*, *FatigueLevel* e *Performance.Task*, serem razoáveis e parecidos com a outra topologia, duas redes com um nodo de output, *FatigueLevel* e *Performance.Task* respetivamente, conclui-se que a melhor topologia é o caso em que a Exaustão e a Tarefa são identificadas cada uma pela sua respetiva rede, em que cada uma é o seu único nodo de *output*. As topologias escolhidas podem ser observadas nas Figuras 7 e 8.

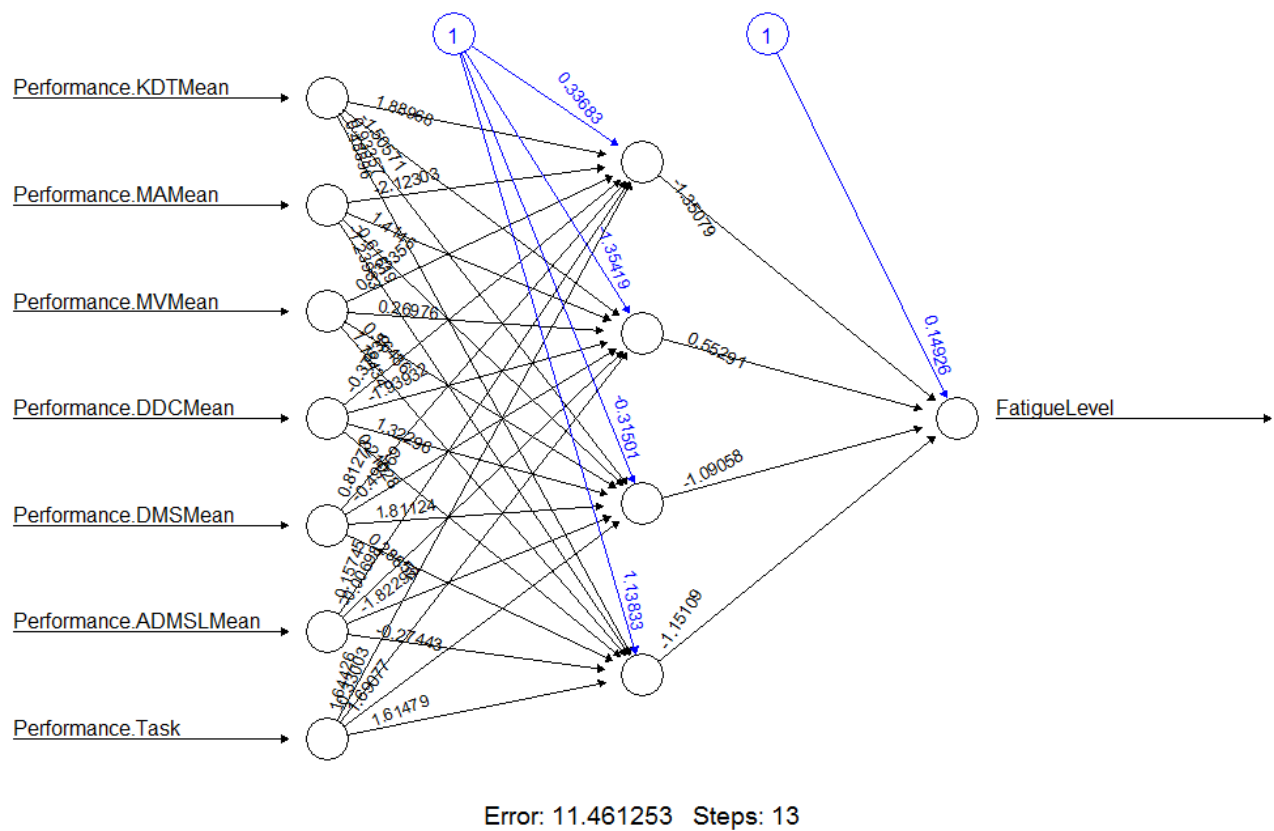


Figura 7 – Topologia de rede para identificação do nível de exaustão

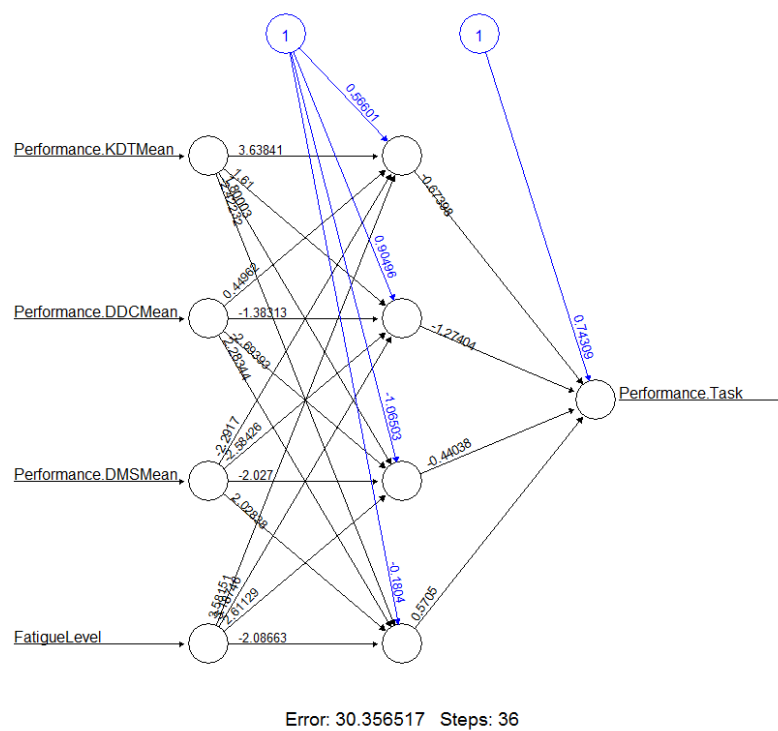


Figura 8 – Topologia de rede para identificação da tarefa em execução



## 2. Estudo dos atributos mais significativos

Após a identificação da melhor topologia, para a melhor escala de identificação da exaustão, como para a identificação da tarefa, procedeu-se agora ao estudo dos atributos mais significativos.

Após se ter identificado os atributos, dos mais significativos para os menos significativos, através do *regsubsets* da biblioteca *leaps* do ambiente R, na Parte 1, foram então realizados testes variando o número de atributos utilizados, sendo escolhidos sempre os mais significativos.

### 2.1 Estudo dos atributos mais significativos para identificação do nível de exaustão

Para as soluções encontradas até então para identificação do nível de exaustão, realizou-se os seguintes testes, que podem ser observados na Tabela 26, no qual se variou o número de atributos significativos.

Atributos	C	Steps	Tempo	Erro	RMSE	Threshold	Algoritmo
9	4	19	0.02 secs	11.24919	0.193139055	0.1	Default
8	4	13	0.01 secs	11.38585	0.192758276	0.1	Default
7	4	13	0.01 secs	11.46125	0.1926864733	0.1	Default
6	4	12	0.01 secs	11.37469	0.1935274929	0.1	Default
4	4	17	0 secs	11.39665	0.1932067852	0.1	Default

Tabela 26 – Resultados de testes com diferentes números de atributos para a exaustão

### 2.2 Estudo dos atributos mais significativos para identificação da tarefa em execução

Para as soluções encontradas até então para identificação do nível de exaustão, realizou-se os seguintes testes, que podem ser observados na Tabela 27, no qual se variou o número de atributos significativos.

Atributos	C	Steps	Tempo	Erro	RMSE	Threshold	Algoritmo
9	4	27	0.02 secs	30.26038	0.382427903	0.1	Default
8	4	34	0.02 secs	30.09656	0.380128595	0.1	Default
7	4	21	0.01 secs	30.20422	0.381913228	0.1	Default
6	4	39	0.02 secs	30.17193	0.3822535774	0.1	Default
4	4	36	0.02 secs	30.35083	0.379559832	0.1	Default

Tabela 27 – Resultados de testes com diferentes números de atributos para a tarefa

### 2.3 Conclusões

Neste subcapítulo, pode-se concluir, após os testes realizados, que para determinar o nível de exaustão, o ideal será utilizar 7 atributos, (*Performance.KDTMean*, *Performance.MAMean*, *Performance.MVMean*, *Performance.DDCMean*, *Performance.DMSMean*, *Performance.ADMSLMean* e *Performance.Task*).

Já para determinar qual a tarefa em execução, seria mais indicado utilizar apenas 4 atributos (*FatigueLevel*, *Performance.KDTMean*, *Performance.DMSMean* e *Performance.DDCMean*).

### 3. Seleção dos algoritmos

Após a identificação da melhor topologia e do número de atributos significativos a usar, segue-se o estudo do melhor tipo de algoritmo de aprendizagem. Os algoritmos de aprendizagem testados, tanto para a melhor solução para a identificação da exaustão e da tarefa, foram:

- rprop+
- backprop
- sag
- slr

Para a melhor solução obtida para a identificação da exaustão, até ao momento, obteve-se os seguintes resultados, observados na tabela 28.

Algoritmo	C	Steps	Tempo	Erro	RMSE	Threshold
rprop+	4	17	0,01 secs	11.39302	0.1943051698	0.1
backprop	4	2	0 secs	14.25	0.2180257124	0.1
sag	4	13	0.01 secs	11.47926	0.1972600594	0.1
slr	4	16	0.01 secs	11.68189	0.1964111101	0.1

Tabela 28 – Resultados de testes com diferentes algoritmos para identificação da exaustão

Para a melhor solução obtida para a identificação da tarefa em execução, até ao momento, obteve-se os seguintes resultados, observados na tabela 29.

Algoritmo	C	Steps	Tempo	Erro	RMSE	Threshold
rprop+	4	27	0.02 secs	30.26038	0.382427903	0.1
backprop	4	2	0 secs	71.25	0.2180257124	0.1
sag	4	25	0.01 secs	30.43165	0.382413336	0.1
slr	4	33	0.17 secs	30.28524	0.380619622	0.1

Tabela 29 – Resultados de testes com diferentes algoritmos para identificação da tarefa

#### 3.1 Conclusões

Podemos concluir que, em relação aos algoritmos mais indicados para treinar a rede, para a identificação do nível de exaustão deve ser utilizado o algoritmo *default*, *rprop+*, enquanto que para a identificação da tarefa em execução deverá ser utilizado o algoritmo *backprop*.

#### 4. Conclusões da Caracterização da Solução

Podemos concluir, que para a identificação do nível de exaustão e para a identificação da tarefa em execução, a solução deverá ser caracterizada pelos parâmetros indicados na Tabela 30.

	Exaustão	Tarefa
Topologia	uma rede com um nodo de output	uma rede com um nodo de output
	c(4)	c(4)
Atributos	7	4
Algoritmos	rprop+	backprop

Tabela 30 – Caracterização da solução para identificação do nível de exaustão e da tarefa

Conclui-se, então:

- Na Parte 2 obteve-se que a melhor forma de representar a exaustão foi representá-la em 3 níveis (**1+2+3, 4+5, 5+7**) de forma normalizada (**0, 0.5, 1**). Ainda se verificou que não existe uma grande diferença nos resultados na forma como se representam os 8 dados biométricos fornecidos e, por isso, na Parte 4 utilizaram-se exatamente como foram fornecidos (valores contínuos com distribuição normal).
- Na Parte 3 obteve-se a melhor forma de identificar a tarefa em execução foi representá-la de forma normalizada (0, 0.5, 1).
- Na Parte 4, utilizando a melhor representação da exaustão e da tarefa obtidas na Parte 2 e 3, após vários testes e estudos para obter a melhor rede:
  - Apesar de a diferença não ser acentuada, obteve-se melhores resultados quando se identifica a exaustão e a tarefa em **redes independentes com um único nodo de output**. Em vez de ter uma só rede com os dois *outputs*. Ambas as redes obtidas possuem 32 sinapses.
  - Tanto para a exaustão como para a tarefa a rede apresentou melhores resultados para **uma única camada intermédia com 4 nodos**.
  - Analisaram-se os atributos mais significativos para representar a exaustão e a tarefa com a nova forma de representar estes parâmetros, para confirmar se os primeiros testes realizados na parte 1 ainda se mantinham. Na exaustão mantiveram-se os mesmos 6 atributos significativos verificando-se que a nova forma de representar a exaustão aparentemente não influenciou muito os resultados. Contudo, na tarefa, verificou-se que apenas 4 dos atributos são significativos para a determinar.
  - Na determinação da exaustão obtiveram-se melhores resultados com o algoritmo **rprop+**, enquanto que na determinação da tarefa foi com o algoritmo **backprop** que se verificaram os melhores resultados.

De salientar que mesmo obtendo melhores resultados para os parâmetros descritos, a maioria dos testes deu resultados relativamente aproximados, sem muita discrepância.

# Conclusão

A elaboração deste trabalho visou a abordagem a um novo paradigma de representação de conhecimento: o paradigma não simbólico. Recorreu-se a Redes Neurais Artificiais (RNAs) para a resolução dos problemas apresentados.

Um *dataset* referente a dados recolhidos da interação humano-computador através dos dispositivos físicos rato e teclado, foi disponibilizado de modo a ser usado para a determinação do nível de exaustão e da tarefa em execução.

Inicialmente, realizou-se um estudo inicial, Parte 1, aos problemas a abordar de modo a *standardizar* os testes e poder concluir que parâmetros podem ou não realmente influenciar os resultados obtidos. Pelo que se decidiu:

- Representar dados biométricos de forma contígua e discreta;
- Analisar os atributos, ordenando do mais significativo para o menos significativo;
- O conjunto de topologias a serem testadas.

Na Parte 2, identificou-se os níveis de exaustão, usando os dados biométricos de forma discreta e contínua e os 6 atributos biométricos mais relevantes com e sem a tarefa e para as 3 topologias a ser testadas.

Esta Parte encontra-se dividida em 3 cenários:

- Identificação dos 7 níveis de exaustão
- Identificação da existência e não existência de exaustão
- Identificação da melhor escala para identificar a exaustão.

Para identificar a exaustão, o melhor resultado obtido foi utilizando uma rede neuronal representando a exaustão normalizada por 3 níveis (1+2+3, 4+5, 6+7). Com esta representação tentou-se obter um *tradeoff* entre a expressividade dos níveis de exaustão (que é baixa apenas com dois níveis) e valores de *rmse* e de *erro* relativamente baixos.

Conclui-se que o facto de discretizar os valores dos 8 parâmetros biométricos, distribuindo melhor os valores na forma contígua, deveria resultar em resultados melhores, contudo não se verificou, pelo que nas etapas seguintes apenas se realizaram testes com os valores representados de forma contínua, tal como foram fornecidos.

Na Parte 3 (Identificação da Tarefa em execução), utilizou-se os valores biométricos de forma contígua, uma vez que não se obteve melhores resultados na identificação da exaustão usando valores discretizados. Utilizou-se 2 escalas de normalização da tarefa: uma de 1 a 3 e outra de 0 a 1, sendo com a última que se obteve melhores resultados.

Após a identificação da melhor forma de representar a exaustão como a tarefa, procedeu-se na parte 4, à análise:

- Topologia da Rede
- Atributos
- Algoritmos de aprendizagem

Testaram-se 3 redes, uma com a exaustão como *output*, outra com a tarefa como *output* e outra com ambos exaustão e tarefa como *output*. Conclui-se que a melhor forma de identificar a exaustão

e a tarefa seria com redes distintas, isto é com um *output* dedicado somente a identificação da exaustão e outra com o *output* nodo para identificar apenas a tarefa.

Ambas as topologias obtiveram melhores resultados com apenas uma camada intermédia de 4 nodos.

O números de camadas intermédias não tem uma relação clara com melhores resultados, o que se verificou, sendo a rede com somente uma camada intermédia de 4 neurónios com melhores resultados.

Relativamente ao número de atributos usados para *input* na rede, testou-se de que forma, representando a exaustão e a tarefa de forma diferente (obtidas na parte 2 e 3), influencia os resultados. Conclui-se que para a exaustão o menor número seria a utilização dos 7 atributos mais significativos, já para a tarefa seria os 4 atributos mais significativos. Assim, pelo menos para identificar a tarefa, a representação de conhecimento dos atributos *inputs* influencia os resultados.

Para a identificação da exaustão o algoritmo de aprendizagem que obteve melhores resultados foi o rprop+ e para a tarefa o backprop.

Concluído, para representar a exaustão numa rede neuronal, o melhor resultado obtido foi com:

- Um nodo *output*: exaustão
  - Representada em 3 níveis (fazendo clusters dos níveis iniciais: 1+2+3, 4+5, 6+7)
  - Representada de forma normalizada (Nível 1 = 0, Nível 2 = 0.5, Nível 3 = 1)
- Com 7 nodos *inputs*
  - Todos os valores biométricos fornecidos exceto *Performance.TBCM* e *Performance.AEDM*.
  - Representados de forma discreta
  - Utilizando a tarefa como *input*
- Topologia c(4) apenas com uma camada intermédia de 4 neurónio
  - Capacidade de aprendizagem de 32 sinapses
- Abordagem: supervisionada
  - Aproximação não linear
  - Algoritmo rprop+
- Parâmetros de teste:
  - rmse: 0.1943051698
  - erro: 11.39302

Enquanto que, para a tarefa:

- Um nodo *output*: tarefa
  - Representada em 3 níveis (*work, office, programming*)
  - Representada de forma normalizada (Nível 1=0, Nível 2=0.5, Nível 3 = 1)
- Com 3 nodos *inputs*
  - Todos os valores biométricos fornecidos exceto *Performance.KDTM*, *Performance.TBCM*, *Performance.DMSM*, *Performance.AEDM*, *Performance.ADMSL*.
  - Utilizando a exaustão como *input*

- Topologia c(4) apenas com uma camada intermédia de 4 neurónios
  - Capacidade de aprendizagem: 20 sinapses
- Abordagem: supervisionada
  - Aproximação não linear
  - Algoritmo backprop
- Parâmetros de teste:
  - rmse: 0.2180257124
  - erro: 71.25

Por fim, seguimos uma metodologia que poderia ter outros caminhos, outros testes que poderiam ter sido realizados, e que poderão ser realizados no futuro, para que se possam tirar mais conclusões, obter novas informações e aprender ainda mais sobre representação de conhecimento, recorrendo ao conhecimento não simbólico e à utilização de redes neuronais artificiais.

## Anexo A

### Anexo A.1 – Exaustão com valores não normalizados e aproximação não linear à função de ativação

#### Anexo A.1.1 – Cenário 1

C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
		4	29	0.02 secs	972.09555	1.829976838	0.1	Default	7
		4	33	0.02 secs	972.00624	1.829919442	0.01	Default	7
	8	4	21	0.03 secs	972.05354	1.82996055	0.1	Default	7
	8	4	24	0.09 secs	972.00954	1.829923625	0.01	Default	7
16	8	4	27	0.05 secs	972.09225	1.829992695	0.1	Default	7
16	8	4	18	0.04 secs	972.00747	1.829921959	0.01	Default	7

#### Anexo A.1.2 – Cenário 2

C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
		4	40	0.02 secs	39.7696	0.3892641667	0.1	Default	7
		4	9334	4.97 secs	34.43372	0.4487835535	0.01	Default	7
	8	4	3888	4.65 secs	19.71552	0.5064126121	0.1	Default	7
	8	4	10518	10.72 secs	14.78645	0.4885312607	0.01	Default	7
16	8	4	3673	8.69 secs	9.68748	0.5047691965	0.1	Default	7
16	8	4	3902	9.21 secs	7.01301	0.4882466829	0.01	Default	7

#### Anexo A.1.3 – Cenário 3

Níveis	C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
6			4	31	0 secs	967.57873	1.838633833	0.1	Default	7
6			4	29	0.02 secs	967.50843	1.838562517	0.01	Default	7
6		8	4	26	0.02 secs	967.58141	1.838632123	0.1	Default	7
6		8	4	32	0.09 secs	967.50852	1.838562378	0.01	Default	7
6	16	8	4	22	0.05 secs	967.58008	1.838630461	0.1	Default	7
6	16	8	4	24	0.05 secs	967.56061	1.838611632	0.01	Default	7
5			4	31	0.03 secs	949.57837	1.83863384	0.1	Default	7
5			4	29	0.02 secs	949.50839	1.838562518	0.01	Default	7
5		8	4	26	0.01 secs	949.581	1.838632106	0.1	Default	7
5		8	4	32	0.12 secs	949.50846	1.838562365	0.01	Default	7
5	16	8	4	22	0.05 secs	949.57969	1.838630461	0.1	Default	7
5	16	8	4	24	0.06 secs	949.50844	1.838562317	0.01	Default	7
4			4	31	0.02 secs	879.57578	1.770335364	0.1	Default	7
4			4	29	0.01 secs	879.50817	1.770263389	0.01	Default	7
4		8	4	26	0.03 secs	879.57905	1.770333955	0.1	Default	7
4		8	4	32	0.04 secs	879.50826	1.770263242	0.01	Default	7
4	16	8	4	22	0.05 secs	879.57778	1.770332274	0.1	Default	7
4	16	8	4	24	0.05 secs	879.50824	1.770263201	0.01	Default	7
3			4	26	0.01 secs	287.59923	1.02109825	0.1	Default	7
3			4	22	0.01 secs	287.50619	1.020919555	0.01	Default	7
3		8	4	27	0.03 secs	287.59515	1.021074511	0.1	Default	7
3		8	4	17	0.02 secs	287.50958	1.020925075	0.01	Default	7
3	16	8	4	30	0.07 secs	287.57641	1.021038614	0.1	Default	7
3	16	8	4	26	0.07 secs	287.51	1.020921915	0.01	Default	7

## Anexo A.2 – Exaustão com valores não normalizados e aproximação linear à função de ativação

### Anexo A.2.1 – Cenário 1

C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
		4	8887	3.71 secs	266.46889	1.10016464	0.1	Default	7
		4	9966	4.05 secs	263.6696	1.079107056	0.01	Default	7
	8	4	41738	47.14 secs	166.59609	1.305418549	0.1	Default	7
	8	4				stepmax	0.01	Default	7
16	8	4				stepmax	0.1	Default	7
16	8	4				stepmax	0.01	Default	7

### Anexo A.2.2 – Cenário 2

C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos
		4	1527	0.81 secs	35.43462	0.4127708824	0.1	Default	7
		4	5259	2.21 secs	35.16348	0.4045474657	0.01	Default	7
	8	4	1601	1.61 secs	27.21937	0.5349130942	0.1	Default	7
	8	4	86441	1.54 mins	12.53702	0.5359504034	0.01	Default	7
16	8	4	6753	15.57 secs	6.16519	1.169609808	0.1	Default	7
16	8	4	16805	44.88 secs	2.49062	1.444187417	0.01	Default	7

### Anexo A.2.3 – Cenário 3

C			Steps	Tempo	Erro	RMSE	Threshold	Algoritmo	Atributos	Níveis
		4	1470	0.61 secs	280.49698	1.090381594	0.1	Default	7	6
		4	11051	4.55 secs	258.65079	1.098533438	0.01	Default	7	6
	8	4	33087	35.45 secs	180.30829	1.288096666	0.1	Default	7	6
	8	4				stepmax	0.01	Default	7	6
16	8	4	90253	3.68 mins	50.80171	1.970243806	0.1	Default	7	6
16	8	4				stepmax	0.01	Default	7	6
		4	2340	0.96 secs	275.96776	1.116468551	0.1	Default	7	5
		4	3662	1.79 secs	266.52314	1.092291579	0.01	Default	7	5
	8	4	11723	11.82 secs	178.57862	1.376972225	0.1	Default	7	5
	8	4				stepmax	0.01	Default	7	5
16	8	4				stepmax	0.1	Default	7	5
16	8	4				stepmax	0.01	Default	7	5
		4	2563	1.2 secs	238.85635	1.037017927	0.1	Default	7	4
		4	17411	7.36 secs	221.0758	1.006847712	0.01	Default	7	4
	8	4	9965	9.91 secs	147.79213	1.200607099	0.1	Default	7	4
	8	4				stepmax	0.01	Default	7	4
16	8	4	39863	1.44 mins	43.03244	1.817788091	0.1	Default	7	4
16	8	4				stepmax	0.01	Default	7	4
		4	3112	1.39 secs	141.37249	0.810162504	0.1	Default	7	3
		4	7987	3.36 secs	131.75169	0.7993400491	0.01	Default	7	3
	8	4	9393	9.04 secs	89.29182	0.8988766107	0.1	Default	7	3
	8	4				stepmax	0.01	Default	7	3
16	8	4	62421	2.6 mins	26.90252	1.28270152	0.1	Default	7	3
16	8	4				stepmax	0.01	Default	7	3