



# UTILIZANDO RESERVOIR COMPUTING PARA REALIZAR DIAGNÓSTICO PRECOCE DE MAL DE ALZHEIMER

Trabalho de Conclusão de Curso  
Engenharia da Computação

**Lara Dantas Coutinho**

**Orientador:** Prof. Dr. Mêuser Jorge Silva Valença



Lara Dantas Coutinho

***UTILIZANDO RESERVOIR COMPUTING  
PARA REALIZAR DIAGNÓSTICO  
PRECOCE DE MAL DE ALZHEIMER***

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia de Computação pela Escola Politécnica de Pernambuco - Universidade de Pernambuco

Orientador:

Prof. Dr. Mêuser Jorge Silva Valença

UNIVERSIDADE DE PERNAMBUCO  
ESCOLA POLITÉCNICA DE PERNAMBUCO  
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

Recife - PE, Brasil

20 de Novembro de 2012

## MONOGRAFIA DE FINAL DE CURSO

### Avaliação Final (para o presidente da banca)\*

No dia 5 de 12 de 2012, às 8:00 horas, reuniu-se para deliberar a defesa da monografia de conclusão de curso do discente LARA DANTAS COUTINHO, orientado pelo professor Mêuser Jorge Silva Valença, sob título Utilizando Reservoir Computing para realizar o diagnóstico precoce de Mal de Alzheimer, a banca composta pelos professores:

**Sérgio Murilo Maciel Fernandes**

**Mêuser Jorge Silva Valença**

Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

Aprovada       Aprovada com Restrições\*       Reprovada

e foi-lhe atribuída nota: 10 ( Dez )

\*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O discente terá 03 dias para entrega da versão final da monografia a contar da data deste documento.

SÉRGIO MURILO MACIEL FERNANDES

MÊUSER JORGE SILVA VALENÇA

Dedico à minha mãe Sônia de Sousa Dantas, à minha irmã Jéssica Dantas, à minha avó Sônia Barros de Sousa Dantas, ao meu namorado Breno Inojosa e às minhas amigas Bruna Galle e Gabriela Araripe.

# *Agradecimentos*

Em primeiro lugar, agradeço à minha mãe, Sônia de Sousa Dantas, que tão incansavelmente se esmerou para estar sempre presente em minha vida e por ser a principal responsável pela minha criação, educação, e formação do meu caráter. Sem ela esse dia não seria possível.

Agradeço à minha avó, Sônia Barros de Sousa Dantas, e à minha irmã, Jéssica Dantas, que também fizeram parte do meu aprendizado e fazem parte desta vitória.

Agradeço ao meu namorado, Breno Costa Inojosa, que sempre me apoiou e esteve ao meu lado no decorrer deste curso. Sua companhia, dedicação, paciência, amor e suporte foram componentes essenciais para a conclusão da minha graduação da melhor maneira possível.

A uma das pessoas mais presentes em minha vida ao longo desses 5 anos, pois sua amizade dentro e fora da Universidade foi responsável por muitos momentos de alegria e nunca me desamparou quando eu mais precisei. À Bruna Galle o meu sincero agradecimento.

Agradeço à minha amiga Gabriela Araripe que em 6 anos de amizade nunca deixou de me incentivar, esteve sempre ao meu lado e foi responsável por muitos momentos de alegria e descontração.

Aos amigos que começaram a fazer parte da minha vida desde o começo dessa jornada, Breno Soares, Breno Menezes e Carlos Castor, o meu muito obrigada.

Agradeço também a todos os professores que fizeram parte da minha formação desde o ensino fundamental até o fim da graduação. Em especial, meu orientador Mêuser Valença com quem tive a honra de trabalhar nos últimos 3 anos.

Por fim, porém não menos importante, agradeço a Deus por possibilitar a minha existência com saúde e disposição para enfrentar todos os obstáculos que precisei superar.

# *Resumo*

A expectativa de vida da população dos países mais desenvolvidos está crescendo a cada dia e, conseqüentemente, há o aumento de várias doenças relacionadas à idade. No Brasil, cerca de 1,2 milhão de pessoas possuem a Doença de Alzheimer (DA), hoje considerada o tipo de demência mais frequente na população brasileira. Apesar de ser uma doença degenerativa e irreversível, se diagnosticada precocemente, medicamentos podem ser administrados a fim de retardar a progressão dos sintomas e uma melhor qualidade de vida para o paciente. Ray et al. e Moscato realizaram estudos com classificadores contidos no software Weka e, a partir de uma base de dados com valores de 120 proteínas sanguíneas, conseguiram classificar se o paciente podia ou não ser diagnosticado com DA com uma taxa de acerto de 91% e 93% respectivamente. Dessa forma, esse trabalho visa utilizar uma nova abordagem conexionista denominada *Reservoir Computing* (RC) para realizar o diagnóstico precoce de um paciente com ou sem DA e também comparar esses resultados com os obtidos através de uma Rede Neural *Multi-Layer Perceptron* (MLP) e com os disponíveis na literatura.

**Palavras-chave:** Doença de Alzheimer, *Reservoir Computing*, Redes Neurais Artificiais.

# *Abstract*

The life expectancy of the population in most developed countries is growing every day and hence there is an increase of several age-related diseases. In Brazil, about 1.2 million people have Alzheimer's disease (AD), now considered the most common type of dementia in the population. Although a degenerative and irreversible disease, if diagnosed early, medications may be administered to slow the progression of symptoms and provide a better quality of life for the patient. Ray et al. and Moscato conducted studies with binders contained in the software Weka using a database with values of 120 blood proteins, and they notice that they could classify the patient may or may not be diagnosed with AD with an accuracy rate of 91% and 93% respectively . Thus, this study aims to use a new connectionist approach called Reservoir Computing (RC) to perform early diagnosis of a patient with or without AD, also to compare these results with those obtained using a Neural Network Multi-Layer Perceptron (MLP) and with those available in the literature.

**Keywords:** Alzheimer Disease, Reservoir Computing, Artificial Neural Networks.

# *Sumário*

<b>Lista de Figuras</b>	p. ix
<b>Lista de Tabelas</b>	p. xi
<b>Lista de Abreviaturas e Siglas</b>	p. xiv
<b>1 Introdução</b>	p. 15
1.1 Motivação . . . . .	p. 15
1.2 Objetivos . . . . .	p. 17
1.2.1 Objetivo Geral . . . . .	p. 17
1.2.2 Objetivos Específicos . . . . .	p. 17
1.3 Estrutura da Monografia . . . . .	p. 17
<b>2 Fundamentação Teórica</b>	p. 19
2.1 Mal de Alzheimer . . . . .	p. 19
2.2 Redes Neurais Artificiais . . . . .	p. 22
2.2.1 Multi-Layer Perceptron . . . . .	p. 24
2.2.1.1 Algoritmo Back-propagation . . . . .	p. 25
2.3 <i>Reservoir Computing</i> . . . . .	p. 26
2.3.1 Construção do RC . . . . .	p. 27
2.3.2 Simulação do RC . . . . .	p. 29
<b>3 Metodologia</b>	p. 31
3.1 Base de Dados . . . . .	p. 31



3.1.1	Pré-processamento dos dados . . . . .	p. 32
3.2	Implementação do RC . . . . .	p. 33
3.3	Simulação com o RC . . . . .	p. 36
3.4	Simulação com a MLP . . . . .	p. 36
3.5	Análise estatística . . . . .	p. 37
3.5.1	Teste Shapiro-Wilk . . . . .	p. 38
3.5.2	Teste F . . . . .	p. 38
3.5.3	Teste T-Student . . . . .	p. 39
3.5.4	Teste da Soma dos Postos de Wilcoxon . . . . .	p. 39
<b>4</b>	<b>Resultados</b>	p. 41
4.1	Base de dados com 5 proteínas e conjunto de testes DA . . . . .	p. 41
4.2	Base de dados com 10 proteínas e conjunto de testes DA . . . . .	p. 42
4.3	Base de dados com 5 proteínas e conjunto de testes MCI . . . . .	p. 43
4.4	Base de dados com 10 proteínas e conjunto de testes MCI . . . . .	p. 44
<b>5</b>	<b>Conclusões e Trabalhos Futuros</b>	p. 46
	<b>Referências</b>	p. 48
	<b>Apêndice A – Resultados da Análise Estatística no Software R</b>	p. 50
	<b>Apêndice B – Script utilizado para a execução da Análise Estatística no Software R</b>	p. 54

## *Lista de Figuras*

1	Ilustração do cérebro de um portador de Alzheimer, à esquerda, e de uma pessoa sem essa deficiência, à direita. . . . .	p. 20
2	Ilustração do modelo matemático criado por McCulloch e Pitts para representar o neurônio artificial. . . . .	p. 23
3	Ilustração de uma MLP com 3 camadas. . . . .	p. 25
4	Arquitetura de um <i>Reservoir Computing</i> . As linhas tracejadas representam quais os pesos que deverão ser ajustadas durante o treinamento da rede. . . . .	p. 28
5	Execução do teste de Shapiro-Wilk nos resultados das execuções com o RC utilizando a base de dados com 5 proteínas e o conjunto de testes DA. . . . .	p. 50
6	Execução do teste de Shapiro-Wilk nos resultados das execuções com a MLP utilizando a base de dados com 5 proteínas e o conjunto de testes DA. . . . .	p. 50
7	Execução do teste da Soma dos Postes de Wilcoxon para comparar o desempenho do RC e da MLP utilizando a base de dados com 5 proteínas e o conjunto de testes DA. . . . .	p. 51
8	Execução do teste de Shapiro-Wilk nos resultados das execuções com o RC utilizando a base de dados com 10 proteínas e o conjunto de testes DA. . . . .	p. 51
9	Execução do teste de Shapiro-Wilk nos resultados das execuções com a MLP utilizando a base de dados com 10 proteínas e o conjunto de testes DA. . . . .	p. 51
10	Execução do teste da Soma dos Postes de Wilcoxon para comparar o desempenho do RC e da MLP utilizando a base de dados com 10 proteínas e o conjunto de testes DA. . . . .	p. 51

11	Execução do teste de Shapiro-Wilk nos resultados das execuções com o RC utilizando a base de dados com 5 proteínas e o conjunto de testes MCI. . . . .	p. 52
12	Execução do teste de Shapiro-Wilk nos resultados das execuções com a MLP utilizando a base de dados com 5 proteínas e o conjunto de testes MCI. . . . .	p. 52
13	Execução do teste da Soma dos Postes de Wilcoxon para comparar o desempenho do RC e da MLP utilizando a base de dados com 5 proteínas e o conjunto de testes MCI. . . . .	p. 52
14	Execução do teste de Shapiro-Wilk nos resultados das execuções com o RC utilizando a base de dados com 10 proteínas e o conjunto de testes MCI. . . . .	p. 52
15	Execução do teste de Shapiro-Wilk nos resultados das execuções com a MLP utilizando a base de dados com 10 proteínas e o conjunto de testes MCI. . . . .	p. 53
16	Execução do teste da Soma dos Postes de Wilcoxon para comparar o desempenho do RC e da MLP utilizando a base de dados com 10 proteínas e o conjunto de testes MCI. . . . .	p. 53

# *Lista de Tabelas*

1	Representação das proteínas contidas em cada uma das assinaturas utilizada. . . . .	p. 32
2	Representação dos parâmetros utilizados para as simulações com o RC	p. 36
3	Resumo das simulações realizadas com o RC e com a MLP. . . . .	p. 36
4	Representação dos parâmetros utilizados para as simulações com a MLP	p. 37
5	Hipóteses nula e alternativa definidas na aplicação do Teste de Shapiro-Wilk . . . . .	p. 38
6	Hipóteses nula e alternativa definidas na aplicação do Teste F . . . . .	p. 38
7	Hipóteses nula e alternativa definidas na aplicação do Teste T-Student	p. 39
8	Hipóteses nula e alternativa definidas na aplicação do Teste da Soma dos Postos de Wilcoxon . . . . .	p. 40
9	Representação da média das taxas de acerto após os 30 experimentos com cada topologia utilizando a base de dados com 5 proteínas e o conjunto de testes DA. . . . .	p. 41
10	Hipóteses nula e alternativa definidas na aplicação do Teste de Shapiro-Wilk na amostra de dados do RC . . . . .	p. 42
11	Hipóteses nula e alternativa definidas na aplicação do Teste de Shapiro-Wilk na amostra de dados da MLP . . . . .	p. 42
12	Hipóteses nula e alternativa definidas na aplicação do Teste da Soma dos Postos de Wilcoxon para as amostras de ambas as topologias de rede neural utilizadas. . . . .	p. 42
13	Representação da média das taxas de acerto após os 30 experimentos com cada topologia utilizando a base de dados com 10 proteínas e o conjunto de testes DA. . . . .	p. 43

14	Representação da média das taxas de acerto após os 30 experimentos com cada topologia utilizando a base de dados com 5 proteínas e o conjunto de testes MCI. . . . .	p. 43
15	Representação da média das taxas de acerto após os 30 experimentos com cada topologia utilizando a base de dados com 10 proteínas e o conjunto de testes MCI. . . . .	p. 44
16	Comparações dos resultados obtidos com a MLP, RC e com os disponíveis na literatura. . . . .	p. 45

## *Lista de Abreviaturas e Siglas*

DA	<i>Doença de Alzheimer</i>
RNA	<i>Rede Neural Artificial</i>
MLP	<i>Multi-Layer Perceptron</i>
RC	<i>Reservoir Computing</i>
MCI	<i>Mild Cognitive Impairment</i>
ABRAz	<i>Associação Brasileira de Alzheimer</i>
OMS	<i>Organização Mundial de Saúde</i>
IBGE	<i>Instituto Brasileiro de Geografia e Estatística</i>
ADI	<i>Alzheimer's Disease International</i>
DMDA	<i>Dia Mundial da Doença de Alzheimer</i>
RNR	<i>Redes Neurais Recorrentes</i>
LSM	<i>Liquid State Machine</i>
ESN	<i>Echo State Networks</i>
ESP	<i>Echo State Property</i>
EPMA	<i>Erro Percentual Médio Absoluto</i>
EMQ	<i>Erro Médio Quadrático</i>

# 1 *Introdução*

## 1.1 *Motivação*

Os países mais desenvolvidos estão passando por uma grande mudança demográfica. Os segmentos mais velhos da população estão crescendo a uma taxa mais rápida, e por isso, há um aumento das doenças relacionadas à idade, principalmente os distúrbios demenciais progressivos. Descrita pela primeira vez pelo psiquiatra Alois Alzheimer em 1907, a Doença de Alzheimer (DA), é, hoje, a causa mais comum de demência da população idosa. (GIUSTI; SURDO, 2010)

Por ser considerada uma patologia neuro-degenerativa, ou seja, progressivamente e irreversivelmente os neurônios do sistema nervoso central se deterioram, e o paciente sofre de depressão, isolamento social, ansiedade e perda de memória.

A DA pode atingir tanto homens quanto mulheres e em qualquer época da fase adulta. Na maioria dos casos, os portadores começam a desenvolver a doença após os 65 anos, ou seja, a forma senil. Porém, existem casos de indivíduos que desenvolveram precocemente a DA, antes dos 65 anos, forma pré-senil, muitas vezes por sofrerem algum tipo de traumatismo craniano ou possuírem parentes próximos que já desenvolveram a doença.

No caso da forma senil, a evolução da doença é mais lenta e os sintomas permanecem por um tempo mais prolongado. Ao contrário disso, na forma pré-senil, a DA tem uma evolução bastante rápida e agressiva.

Segundo a Organização Mundial de Saúde (OMS), estima-se que, no Brasil, há em média 1,2 milhões de pessoas com DA e que esse número dobrará a cada 20 anos. Por faixa etária, estima-se que após os 65 anos de idade, entre 1 e 6% da população de idosos sofre com DA e, após os 85 anos, entre 10 e 30% possuem a doença. (SAÚDE, 2012)

Apesar de ser uma doença irreversível, se descoberta em sua fase inicial, medicamentos podem ser administrados a fim de retardar a progressão dos sintomas e prolongar o bem-estar do paciente. Dessa forma, nota-se que é de extrema importância que sejam

desenvolvidos mecanismos para a previsão de DA em toda a população.

Na literatura, Ray et al. desenvolveu um estudo utilizando uma base de dados com 120 proteínas contidas no plasma. Ele concluiu em sua pesquisa que uma combinação de 18 das 120 disponíveis conseguia realizar um diagnóstico precoce de DA com uma taxa de acerto de 91% utilizando um conjunto de testes com dados de 92 pacientes que foram diagnosticados ou não com DA. (GÓMEZ RAVETTI; MOSCATO, 2008)

Além disso, ele utilizou também outro conjunto de testes contendo dados de 47 pacientes com diagnóstico de Déficit Benigno da Memória ou *Mild Cognitive Impairment* (MCI). Para esse conjunto, a taxa de acerto foi de 81%. Esses valores foram calculados a partir da média das taxas de acerto encontradas por todos os classificadores utilizados na realização dos testes para ambos os conjuntos.

Posteriormente, Moscato realizou um estudo utilizando 20 classificadores diferentes disponíveis no software Weka e criou uma assinatura de 5 proteínas contidas nas 18 definidas por Ray et al. Esse trabalho possibilitou uma taxa de acerto de 93% utilizando o conjunto de testes de DA e 65% para o conjunto de testes de MCI. Essas taxas de acerto também foram calculadas a partir da média dos valores encontrados pelos 20 classificadores utilizados. (GÓMEZ RAVETTI; MOSCATO, 2008)

Uma vez que os resultados disponíveis na literatura utilizam apenas técnicas de classificação disponibilizadas pelo software Weka, neste trabalho vamos utilizar uma nova abordagem conexionista denominada *Reservoir Computing* (RC) para realizar a classificação de um paciente, dado o conjunto das 5 proteínas definidas por Moscato.

Para efeito de comparação, foi utilizada também uma rede neural *Multi-Layer Perceptron* (MLP), ou rede neural de múltiplas camadas, topologia essa amplamente utilizada e com um bom desempenho para problemas de classificação (VALENÇA, 2005). Além dessa metodologia, como explicitado anteriormente foi utilizado o RC desenvolvido por Verstraeten em sua tese de doutorado (VERSTRAETEN, 2009).

O RC é uma rede neural recorrente e, diferentemente da MLP que possui pelo menos 1 camada de neurônios intermediária que propagam o sinal da camada de entrada para a camada de saída, o RC contém um reservatório de neurônios intermediários que permitem conexão em todas as direções. Esta camada de neurônios escondidos do RC tem por objetivo construir um modelo matemático de uma rede neural artificial com uma maior aproximação do comportamento do cérebro humano (VERSTRAETEN, 2009).

Por fim, as duas metodologias de rede neural utilizadas tiveram seus resultados para a



classificação de DA comparados entre si e também com os resultados obtidos por Moscato e por Ray et al.

## 1.2 **Objetivos**

Nessa sessão serão expostos os objetivos gerais e em seguida os específicos.

### 1.2.1 **Objetivo Geral**

Verificar o desempenho do RC e da MLP para a classificação de DA a partir de uma base de dados de proteínas do plasma. Além disso, objetiva-se comparar os resultados obtidos pelas duas topologias escolhidas com os resultados disponíveis na literatura.

### 1.2.2 **Objetivos Específicos**

- Implementar na linguagem Java o RC;
- Explorar alguns dos parâmetros de configuração do RC, como, por exemplo, o limite do raio espectral da geração dos pesos do reservatório e conectividade;
- Comparar os resultados com os obtidos utilizando uma Rede Neural de Múltiplas Camadas (MLP);
- Comparar os resultados obtidos pelas duas topologias e comparar com o que há disponível na literatura.

## 1.3 **Estrutura da Monografia**

Este trabalho está organizado em 5 capítulos: Introdução, Fundamentação Teórica, Metodologia, Análise dos Resultados e Conclusões. O capítulo da Fundamentação Teórica irá descrever os conceitos necessários para o entendimento deste trabalho, como a descrição do Mal de Alzheimer, da MLP e do RC.

O capítulo de Metodologia irá descrever todo o procedimento utilizado para a realização desse trabalho, desde a preparação dos dados até a análise estatística realizada.

No capítulo de Análise dos Resultados, serão descritos os testes estatísticos utilizados para realizar a comparação entre as técnicas utilizadas para realizar a classificação de DA

em pacientes.

Por fim, no capítulo de Conclusões, serão expostos todos os resultados desse trabalho bem como as dificuldades encontradas durante a sua execução. Além disso, será reservada uma sessão que listará todos os trabalhos futuros e melhorias que serão realizadas.

## 2 *Fundamentação Teórica*

### 2.1 Mal de Alzheimer

Em 1907, o médico Alois Alzheimer descreveu em um congresso científico na Alemanha o cérebro de uma paciente com 51 anos, Auguste D., e que apresentava sintomas característicos de demência, como comprometimento da linguagem e perda de memória.

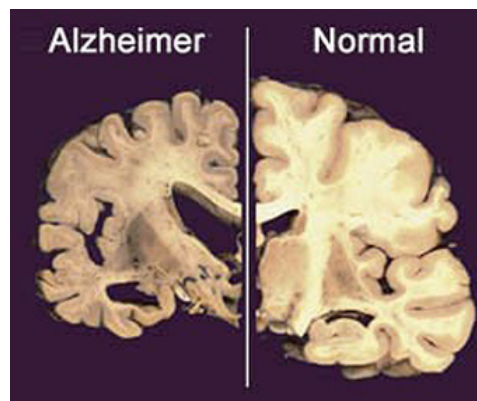
Após quatro anos e meio em tratamento, Auguste D. faleceu com os sintomas de demência bastante avançados. Ao realizar a autópsia, Alois Alzheimer notou lesões nos neurônios que nunca tinham sido observadas antes. As células nervosas estavam atrofiadas em vários locais do cérebro e estavam cobertas por fibras retorcidas e algumas placas estranhas. Essas características notadas pelo médico deram origem a um novo tipo de demência denominada Mal de Alzheimer ou Doença de Alzheimer (DA).

A DA é uma doença degenerativa e irreversível que causa a morte de várias células cerebrais, os neurônios. O paciente que sofre dessa doença possui um cérebro com microscópicas lesões anatomopatológicas, denominadas placas neuríticas, e emaranhados neurofibrilares (GREEN, 2001). Além disso, o cérebro de uma pessoa com Mal de Alzheimer é bem menor que o cérebro de uma pessoa saudável, como pode ser visto na Figura 1.

Essa doença se desenvolve em cada paciente de forma única, porém, há vários sintomas comuns a todos eles, como por exemplo, a perda de memória, distúrbios na linguagem, depressão, agressividade, entre outros.

Inicialmente, o paciente perde a memória episódica, ou seja, a memória que contém informações de eventos e suas relações espaço-temporais. Dessa forma, os fatos mais antigos e os que acabaram de acontecer são facilmente esquecidos.

Com o avanço da doença, a memória semântica também é perdida, ou seja, o conhecimento vocabular, regras, símbolos são esquecidos e o paciente começa a perder a sua identidade cultural.



**Figura 1: Ilustração do cérebro de um portador de Alzheimer, à esquerda, e de uma pessoa sem essa deficiência, à direita.**

[Fonte: Retirado do site <http://clinicadralexandreduzeiro.webnode.com.br/>]

Mesmo a doença se reproduzindo de maneira diferente para cada paciente, ela pode ser dividida em três fases: inicial, intermediária e terminal (MARQUES ROBERTO, 1997).

A fase inicial tem uma duração média de 2 a 4 anos e a patologia se manifesta, primeiramente, com leves esquecimentos, sendo muitas vezes confundidos pelo paciente e pelos familiares como sintomas naturais da velhice. Posteriormente, essa perda de memória se intensifica juntamente com as dificuldades de locomoção e até mesmo para realizar funções comuns do dia-a-dia, como preparar o café da manhã.

Ainda na primeira fase, o paciente começa a apresentar quadros depressivos, mudanças de humor e de personalidade. Há também uma dificuldade de o portador da DA armazenar e aprender novas informações, já que sua memória recente é perdida. Com isso, ele também apresenta sintomas de desorientação espaço-temporal e muitos dos pacientes já nessa fase encontram problemas de se reconhecer diante de um espelho.

A fase intermediária tem uma duração variável entre 2 e 10 anos a depender do paciente. Nesse estágio, há uma piora nas falhas funcionais, e uma intensificação da perda de memória e de funções cognitivas. O doente já tem dificuldades de reconhecer pessoas próximas e até mesmo de identificar graus de parentesco.

Nessa fase o paciente necessita de cuidados e não consegue mais desempenhar de forma satisfatória hábitos comuns como a higiene pessoal e alimentação. Há também o início da incontinência urinária e o aparecimento de pequenos surtos psicóticos.

Na terceira e última fase, o paciente já não consegue mais se locomover nem falar. Ele desconhece a sua própria identidade e depende totalmente de um cuidador em tempo integral. Por permanecer sempre na cama, o paciente está sujeito a outras doenças como

desnutrição, úlceras e pneumonias. Essa última fase tem uma duração de 1 a 3 anos, dependendo também de cada paciente com DA.

A DA pode ser desenvolvida tanto em homens quanto em mulheres. Não há nenhum estudo que comprove que o sexo influencie na predisposição da doença, porém, como as mulheres possuem uma expectativa de vida maior que a do homem, há um número maior de casos de mulheres com Alzheimer.

Além de ser subdividida em fases, a DA é dividida em dois grupos que levam em consideração a idade do paciente em que a doença começou a se desenvolver. Há a forma pré-senil, onde os pacientes desenvolvem a doença antes dos 65 anos, e a forma senil que ocorre após os 65 anos de idade.

A forma pré-senil, ou DA com início precoce, apresenta ocorrências mais raras do que a forma senil. No primeiro caso, o paciente tem suas funções neurológicas comprometidas mais rapidamente e a DA avança de uma forma mais rápida e agressiva (GIUSTI; SURDO, 2010).

A maioria dos casos de desenvolvimento precoce do Mal de Alzheimer é devida à existência de casos da doença na família por algumas gerações. Além disso, pessoas que possuem Síndrome de Down também tem uma maior tendência a desenvolver a DA precocemente.

Já a forma senil da doença tem uma evolução mais lenta, ou seja, a perda das capacidades neurológicas acontece aos poucos e por um período de tempo prolongado. Nesses casos, nem sempre há alguma ligação com o aparecimento da doença em familiares.

O aumento da expectativa de vida nos países mais desenvolvidos acarreta em um aumento da população idosa e, conseqüentemente, o aumento do número de portadores de DA. Segundo o Instituto Brasileiro de Geografia e Estatística (IBGE) e a OMS, há 1,2 milhões de pessoas no Brasil com DA. Dessas pessoas, acredita-se que apenas 5% dos portadores desenvolveram a doença precocemente, ou seja, antes dos 65 anos de idade. Nos portadores em que a DA iniciou após os 65 anos, estima-se que entre 10 e 30% dos casos iniciaram após os 85 anos (SAÚDE, 2012).

O diagnóstico de DA é muitas vezes realizado tardiamente visto que essa doença pode ser confundida com vários outros tipos de demências e até mesmo com os sintomas normais da velhice. Por isso, o paciente que apresentar falhas na memória deve realizar exames neurológicos com imagem, neuropatológicos, além de exames laboratoriais.

Quanto mais cedo for realizado o diagnóstico da doença, melhor será a qualidade de

vida do paciente. Isso acontece visto que medicamentos podem ser administrados com antecedência e, conseqüentemente, o progresso dos sintomas é retardado (GREEN, 2001).

Não há ainda no campo da ciência nada que comprove como prevenir o surgimento da DA. Porém, algumas práticas de saúde são recomendadas a fim de retardar ao máximo o aparecimento da doença. São elas:

- Ter uma dieta com baixo nível de gordura;
- Manter uma pressão arterial normal;
- Permanecer ativo mentalmente e socialmente;
- Praticar exercícios físicos regularmente.

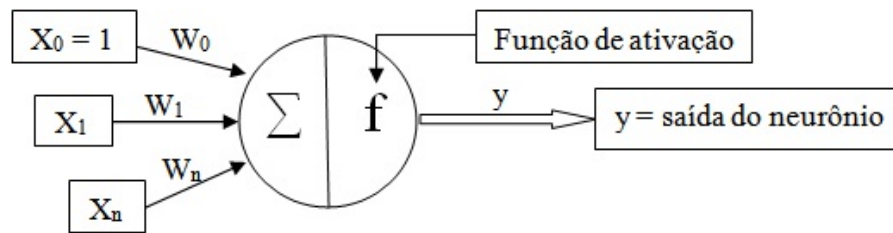
Mesmo sendo uma doença irreversível, os cuidadores de um portador de Alzheimer devem estar sempre buscando meios de integrá-los na sociedade. A Associação Brasileira de Alzheimer (ABRAZ) está há 20 anos trabalhando para oferecer uma melhor qualidade de vida para o portador de DA (ABRAZ, 2012).

Presente nas cinco regiões do país, a ABRAZ conta com o trabalho de voluntários envolvendo profissionais da área de saúde e também familiares de pessoas com Alzheimer. Na associação encontram-se grupos de apoio e atividades para os pacientes, aconselhamento familiar, cursos e treinamentos e desenvolvimento de projetos sociais.

A *Alzheimer's Disease International* (ADI) declarou o Dia Mundial da Doença de Alzheimer (DMDA) em 21 de setembro. Esse dia é marcado por diversos movimentos no mundo inteiro de conscientização da população e palestras informativas, a fim de diminuir o preconceito e a discriminação com os portadores de DA (ADI, 2012).

## 2.2 Redes Neurais Artificiais

As redes neurais artificiais (RNA) são modelos que procuram simular o comportamento e funcionamento do cérebro humano. Assim como existem os neurônios biológicos, componentes essenciais para o processamento das informações no cérebro, a RNA é formada por unidades que objetivam realizar as mesmas funções do neurônio biológico. Esses componentes são denominados neurônios artificiais e foram propostos em 1943 por McCulloch e Pitts (MCCULLOCH; PITTS, 1943). Seu modelo pode ser visualizado na Figura 2.



**Figura 2: Ilustração do modelo matemático criado por McCulloch e Pitts para representar o neurônio artificial.**

[Fonte: elaboração própria.]

O neurônio artificial é composto por um conjunto de entradas ( $X_i$ ) conectadas através de ligações que possuem seus respectivos pesos ( $W_i$ ). Cada entrada do neurônio é multiplicada pelo seu peso e o somatório de todos esses produtos representa a entrada do neurônio artificial, como pode ser visualizado na Equação 2.1. Pode-se notar na Figura 2 que o valor de  $X_0$  é fixado em 1, pois ele representa o limiar excitatório do neurônio (HAYKIN, 2007)

$$net_i = \sum_{i=0}^n x_i \cdot W_i \quad (2.1)$$

O valor de  $net_i$  será utilizado para encontrar a saída do neurônio, que deverá ser calculada de acordo com a função de ativação utilizada.

Existem na literatura diversas possibilidades de funções de ativação, porém as mais utilizadas são a linear, sigmoide logística e tangente hiperbólica (VALENÇA, 2005). As Equações 2.2, 2.3 e 2.4 representam os possíveis cálculos da função de ativação.

$$f(net_i) = net_i \quad (2.2)$$

$$f(net_i) = \frac{1}{1 + e^{-net_i}} \quad (2.3)$$

$$f(net_i) = \frac{e^{net_i} - e^{-net_i}}{e^{net_i} + e^{-net_i}} \quad (2.4)$$

Ao longo da evolução das redes neurais, surgiram o *Perceptron* e a *Adaline*. A diferença entre essas duas arquiteturas de RNA estava na função de ativação dos seus neurônios, sendo a primeira delas com uma função linear e a segunda com uma função sigmoide.

Posteriormente, surgiu a rede *Multi-Layer Perceptron*, que será melhor descrita na próxima sessão.

O processo de aprendizagem de uma RNA é considerado supervisionado, visto que para cada entrada inserida na rede neural, ela irá conhecer a respectiva saída desejada.

Essa ferramenta possui um alto poder de generalização, pois consegue prever valores que não foram apresentados durante o seu treinamento. Além disso, devido a sua estrutura se assemelhar à arquitetura cerebral, ela possui a capacidade de aprendizagem e também de processar informações paralelamente.

### 2.2.1 Multi-Layer Perceptron

Diferentemente do Perceptron e da Adaline, a rede *Multi-Layer Perceptron* (MLP) funciona como um *Perceptron* de muitas camadas, ou seja, ela possui pelo menos uma camada de neurônios intermediária ou escondida. A vantagem de se ter camadas intermediárias é que a rede neural passa a resolver problemas que não são linearmente separáveis, possibilitando, assim, a aproximação de qualquer função contínua, com apenas uma camada, e qualquer função matemática, quando utilizado mais de uma camada (HAYKIN, 2007).

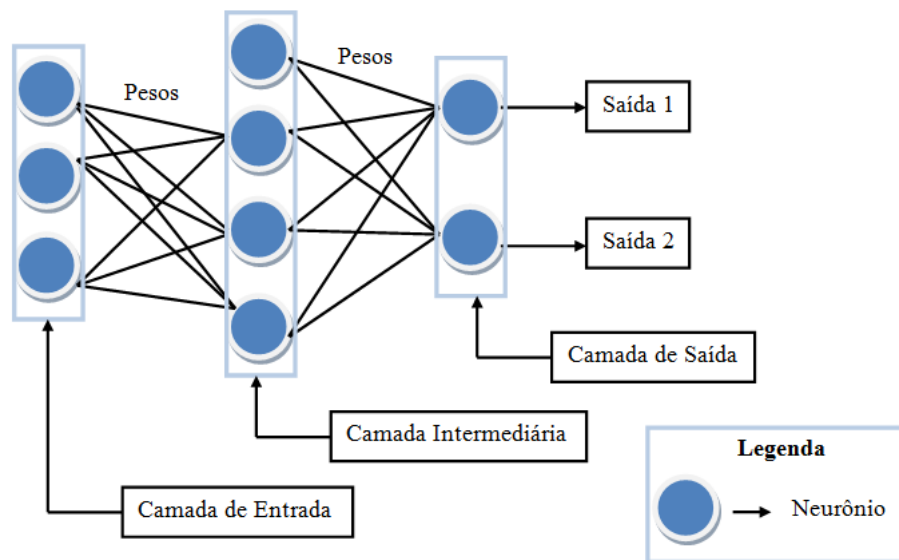
A rede MLP é composta por neurônios conectados através de ligações com pesos, inicialmente aleatórios, mas que, ao longo do treinamento, serão otimizados. Esses neurônios estão dispostos em três tipos de camadas:

- **Camada de entrada:** representam as variáveis de entrada do problema;
- **Camada intermediária ou escondida:** Cada rede MLP possui, pelo menos, uma camada intermediária. Essa parte da rede neural é o que a torna capaz de resolver problemas que não são linearmente separáveis;
- **Camada de saída:** Os neurônios dessa camada representam as variáveis de saída do sistema, ou seja, eles são equivalentes às saídas da rede neural.

A Figura 3 mostra um exemplo da MLP com 3 camadas, sendo uma de entrada, uma intermediária e uma de saída.

Assim como as outras topologias de rede neural, a MLP também precisa de um algoritmo de treinamento para calcular os seus pesos ótimos. Na literatura, o algoritmo mais utilizado atualmente é o *Back-propagation* desenvolvido por Paul Werbos em 1974





**Figura 3: Ilustração de uma MLP com 3 camadas.**

[Fonte: elaboração própria.]

(WERBOS J., 1974). Esse algoritmo trata-se de uma generalização da regra delta utilizada no treinamento da Adaline e utiliza a técnica de gradiente-descendente.

A maior dificuldade para o treinamento da MLP está em encontrar os pesos ótimos para a camada intermediária. Esse problema é solucionado com o *Back-propagation*, já que seu mecanismo utiliza uma retropropagação do erro encontrado na saída. Esse método será melhor descrito na próxima sessão.

### 2.2.1.1 Algoritmo Back-propagation

Esse algoritmo de treinamento da MLP é dividido em duas fases. Na primeira delas, fase *forward*, acontece a propagação progressiva do sinal, ou seja, da camada de entrada para a camada de saída. Nessa fase, os pesos não são alterados e a saída da rede neural e o erro são calculados.

Nesse momento, a segunda fase desse algoritmo, fase *backward*, é iniciada. É nessa fase que ocorre a retropropagação do erro, ou seja, baseado no erro da saída calculado, os pesos que conectam a camada intermediária à camada de saída são ajustados.

A fórmula utilizada para realizar esse ajuste pode ser visualizada na Equação 2.5:

$$W_{i,j}^m(t+1) = W_{i,j}^m(t) + \alpha \delta_i^m f^{m-1}(net_j^{m-1}) + \beta \Delta W_{i,j}^m(t-1) \quad (2.5)$$

Onde,

- $\alpha$ : representa a taxa de aprendizagem do algoritmo;
- $\beta$ : representa a taxa de momento e
- $\delta$ : representa a sensibilidade que pode ser calculada segundo a Equação 2.6 para os neurônios na camada de saída ou de acordo com a Equação 2.7 para os neurônios nas camadas escondidas:

$$\delta_i^m = (d_i - y_i)f'(net_i) \quad (2.6)$$

sensibilidade nas outras camadas

$$\delta_j^{m-1} = f'^{m-1}(net_j^{m-1}) \sum_{i=l}^N w_{ij}^m \delta_i^m \quad (2.7)$$

Mais informações detalhadas sobre a execução desse algoritmo podem ser encontradas na literatura (HAYKIN, 2007) e (VALENÇA, 2009).

## 2.3 Reservoir Computing

Como foi descrito na sessão anterior, a rede MLP tem uma topologia *feedforward*, ou seja, o sinal é propagado da camada de entrada para a saída. Porém, esses tipos de redes neurais são ideais para resolver problemas estáticos.

As Redes Neurais Recorrentes (RNR) foram criadas a fim de possibilitar a solução de problemas dinâmicos. Isso é obtido através de uma realimentação de um neurônio de uma camada  $i$  para um neurônio que se encontra em alguma camada anterior,  $i - j$ . Essa topologia de rede neural apresenta uma melhor semelhança com o funcionamento e comportamento do cérebro humano.

Apesar do grande potencial das RNR e de um grande sucesso em aplicações práticas e acadêmicas, essas redes tiveram seu desenvolvimento limitado por um grande tempo. Isso se deve ao fato de as RNR serem difíceis de treinar por métodos baseados na técnica de gradiente-descendente, como é o caso do algoritmo *Backpropagation* descrito na seção anterior.

Em 2001, surgiu uma nova abordagem para o projeto de treinamento de uma RNR proposta por Wolfgang Mass denominada *Liquid State Machine* (LSM). Ao mesmo tempo, porém independentemente, a mesma abordagem foi descrita por Herbert Jaeger e foi denominada de *Echo State Networks* (ESN) (JAEGER, 2010).

Ambas as redes ESN e LSM possuem a Propriedade do Eco, do inglês *Echo State Property* (ESP), ou seja, devido às conexões recorrentes da rede, as informações das entradas anteriores são armazenadas. Porém, esses dados não ficam guardados por um período de tempo infinito, assim como o cérebro humano, as informações mais antigas devem ser esquecidas com o tempo. Dessa forma, a rede neural possui um rico conjunto de informações do presente e do passado reforçando assim, a sua aplicabilidade para sistemas dinâmicos.

Em 2007, Verstraeten criou o termo *Reservoir Computing* (RC) que unificava os conceitos descritos na LSM e ESN. Desde então, esse termo passou a ser utilizado na literatura para exemplificar sistemas de aprendizagem que são representados por uma rede neural recorrente dinâmica.

O RC é composto por três partes: uma camada de entrada que, assim como a MLP, representa as variáveis de entrada do problema, um *reservoir*, que pode ser entendido como uma grande RNR distribuída e dinâmica com pesos fixos, e uma camada de saída linear denominada *readout*.

A Figura 4 representa a topologia RC com dois neurônios na camada de entrada, três no *reservoir*, e um neurônio na camada de saída.

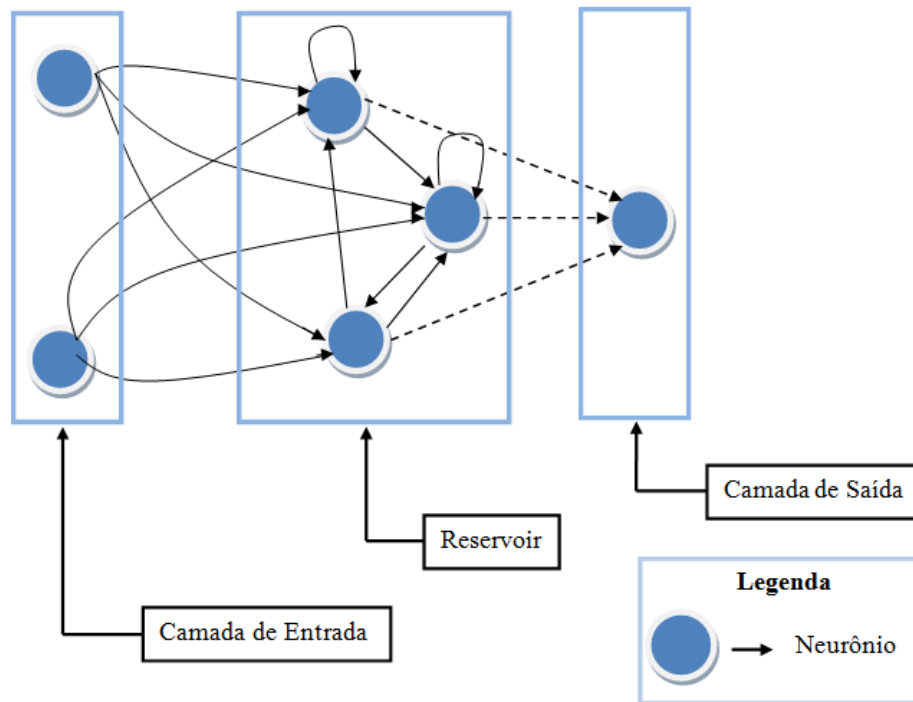
Nota-se na Figura 4 que todos os neurônios da camada de entrada devem ser conectados com todos os que compõem o *reservoir*, assim como todo o *reservoir* deve ser conectados com os neurônios na camada de saída.

As próximas seções irão descrever em detalhes como o RC deve ser criado e como funciona o seu algoritmo de treinamento.

### 2.3.1 Construção do RC

Inicialmente, deve-se construir a camada de entrada utilizando uma quantidade de neurônios que representem as variáveis de entrada do problema acrescido do neurônio que representará o bias.

Em seguida, deve-se estabelecer a quantidade de neurônios que será utilizada no *reser-*



**Figura 4:** Arquitetura de um *Reservoir Computing*. As linhas tracejadas representam quais os pesos que deverão ser ajustadas durante o treinamento da rede.

[Fonte: elaboração própria.]

*voir*. Como essa é uma área com poucas pesquisas, esse valor ainda é definido empiricamente, ou seja, ainda não há estudos que comprovem qual o número de neurônios ideal no *reservoir* para que se tenha um melhor desempenho da rede neural.

Feito isso, deve-se gerar a matriz de pesos que conecta a camada de entrada com o *reservoir* ( $W_{in}$ ). Essa matriz terá como número de linhas a quantidade de neurônios na camada de entrada e como número de colunas a quantidade de neurônios no *reservoir*. Esses pesos são gerados aleatoriamente e estão contidos no intervalo entre -1 e 1.

Além dessa matriz de pesos  $W_{in}$ , deve ser gerada também a matriz de pesos do *reservoir* ( $W_{res}$ ). Esses valores que irão compor a  $W_{res}$  também devem ser gerados aleatoriamente e devem estar contidos no intervalo entre -1 e 1. Nesse ponto, deve ser definida qual a taxa de conectividade do *reservoir*, ou seja, qual a porcentagem de neurônios que possuem os pesos das suas ligações diferentes de 0. Esse também é um parâmetro que necessita ser valorado empiricamente, já que não há teorias que comprovem qual a melhor taxa de conectividade.

Pesquisas afirmam que a matriz  $W_{res}$  deve ter um raio espectral igual a 1 (FERREIRA ARAÚJO, 2001), ou seja, ao inicializar a  $W_{res}$ , ela deve ter seus valores alterados da seguinte

forma:

- Inicialmente ela deverá ser decomposta em valores singulares;
- Em seguida,  $W_{res}$  deverá ter seus valores alterados até que o máximo valor da diagonal principal da matriz de autovalores seja menor ou igual a 1.

A matriz de pesos que conecta o *reservoir* à camada de saída não necessita ser gerada nesse momento já que ela será calculada durante o treinamento pelo método da Pseudo-inversa de Moore-Penrose.

### 2.3.2 Simulação do RC

Para iniciar o treinamento do RC é necessário realizar o pré-processamento da base de dados e dividi-la em 3 subconjuntos: treinamento, validação cruzada e teste. O conjunto de treinamento é utilizado para realizar a atualização dos estados do *reservoir* e ajustar os pesos que conectam o *reservoir* à camada de saída.

O conjunto de validação cruzada é utilizado para parar o treinamento da rede neural. A literatura afirma que essa é a melhor forma de parar o treinamento e evitar o superajustamento da rede ou que ela fique presa em mínimos locais (VALENÇA, 2005).

Por fim, o conjunto de testes será utilizado para calcular a taxa de classificação da rede neural. Para esse conjunto, não deve haver alteração nos estados dos neurônios do *reservoir* nem nos pesos da matriz  $W_{res}$ .

Os estados dos neurônios do *reservoir* devem ser inicializados com o valor zero. Já que se trata de uma rede recorrente e o RC armazena em uma matriz todos os seus estados ( $M_{est}$ ), é necessário que os valores finais encontrados pela rede não seja tão influenciado por essa inicialização.

Dessa forma, a literatura sugere que antes de iniciar o treinamento, um conjunto de ciclos de *warm up* sejam executados a fim de realizar atualizações de estados dos neurônios do *reservoir*.(VERSTRAETEN, 2009)

Finalizado o período de *warm up*, o treinamento do RC pode ser inicializado. O primeiro passo deve ser carregar o conjunto de treinamento e realizar a atualização dos estados do *reservoir*, ressaltando que as matrizes  $W_{in}$  e  $W_{res}$  não devem ser alteradas. Elas são geradas aleatoriamente durante a construção do RC, conforme foi descrito na seção anterior, e não devem ser ajustadas.

Ainda no período de treinamento, a matriz de pesos  $W_{out}$  deve ser calculada através da pseudo-inversa de Moore-Penrose. Como  $W_{out}$  se trata de uma matriz não quadrada e seu determinante pode se aproximar de zero, se faz necessário o cálculo da pseudo-inversa.

Sabe-se que uma matriz  $A$  pode ser decomposta em valores singulares conforme mostra a Equação 2.8:

$$A = U \Sigma V^T \quad (2.8)$$

Onde,  $U$  e  $V$  são matrizes cujas colunas são ortonormais, ou seja, sua norma é igual a 1 e o produto interno é diferente de 0.  $\Sigma$  representa a matriz diagonal, cujos elementos de sua diagonal principal são os valores singulares de  $A$ .

Logo, é dado que a pseudo-inversa de Moore-Penrose da matriz  $A$  é definida por:

$$A^+ = V \Sigma^{-1} U^T \quad (2.9)$$

A pseudo-inversa pode ser calculada desde que a matriz  $A$  satisfaça as seguintes condições:

- $AXA = A$
- $XAX = X$
- $(AX)^T = AX$
- $(XA)^T = XA$

Onde  $X$  é uma única matriz  $n \times m$  que satisfaz essas condições e  $A$  é uma matriz  $n \times m$ .

Ao fim de cada ciclo de treinamento, deve ser iniciado um ciclo de validação cruzada. Esse processo deve se repetir até que os critérios de parada definidos sejam atingidos e o treinamento seja finalizado. Durante o processo de validação cruzada, a matriz  $W_{out}$  deve continuar sendo reajustada.

Finalizado o processo de treinamento, é iniciada a etapa de testes. O conjunto de testes é apresentado ao RC e, nesse momento, todas as matrizes de pesos,  $W_{in}$ ,  $W_{res}$  e  $W_{out}$ , devem permanecer inalteradas, assim como a matriz  $M_{est}$ . Nesse momento, é calculado o erro de classificação ou predição. Esses valores serão utilizados no futuro para realizar as comparações necessárias.

## 3 *Metodologia*

### 3.1 Base de Dados

A base de dados utilizada no desenvolvimento deste trabalho foi a mesma exposta por Moscato em sua publicação (GÓMEZ RAVETTI; MOSCATO, 2008). Ela contém os valores de 120 proteínas encontrados através da análise de amostras sanguíneas de diversos pacientes. O objetivo final da base é classificar se um paciente pode ser diagnosticado ou não com DA ou MCI.

Em seu trabalho, Moscato subdividiu a base de dados em 3 conjuntos. O primeiro deles continha 83 amostras sanguíneas com os valores das 120 proteínas. Dessa quantidade de amostras, 68 delas foram reservadas para o treinamento dos classificadores utilizados. As 15 restantes foram destinadas para o processo de validação cruzada, critério de parada utilizado para finalizar o treinamento das técnicas de classificação.

Segundo Valença, a validação cruzada é o método mais eficiente para parar o treinamento de vários classificadores e técnicas baseadas em aprendizado, a exemplo da rede neural. Essa ferramenta evita que esses modelos fiquem superajustados (VALENÇA, 2005).

O segundo conjunto da base de dados continha informações de 92 pacientes diagnosticados com ou sem DA. Já o terceiro, continha amostras sanguíneas de 47 pacientes que apresentavam ou não MCI.

Além disso, conforme citado na Introdução, Moscato realizou testes com assinaturas de 18, 10, 6 e 5 proteínas, contidas nas 120 disponíveis inicialmente, e chegou a conclusão que essa última era suficiente para realizar o diagnóstico de DA e MCI. A Tabela 1 exhibe quais dessas proteínas estavam contidas nas assinaturas.

Neste trabalho, foram preparadas 4 bases de dados para reproduzir os experimentos, utilizando a MLP e o RC, as simulações descritas por Moscato. Foram elas:

- 1 base para testar a assinatura de 5 proteínas com o conjunto de testes DA;

**Tabela 1: Representação das proteínas contidas em cada uma das assinaturas utilizada.**

[Fonte: elaboração própria]

Quantidade de proteínas	Proteínas
5	IL-1a, IL-3, EGF, TNF-a and G-CSF
10	CCL7/MCP-3, CCL15/MIP-1d, EGF, G-CSF, IL-1a, IL-3, IL-6, IL-11, PDGF-BB, TNF-a

- 1 base para testar a assinatura de 5 proteínas com o conjunto de testes MCI;
- 1 base para testar a assinatura de 10 proteínas com o conjunto de testes DA;
- 1 base para testar a assinatura de 10 proteínas com o conjunto de testes MCI;

Todas as bases descritas acima mantiveram a organização utilizada por Moscato no que concerne a divisão de valores para o conjunto de treinamento, validação cruzada e testes.

### 3.1.1 Pré-processamento dos dados

Para realizar o treinamento da rede neural se faz necessário que os seus dados estejam normalizados, ou seja, os valores que serão inseridos na rede neural devem estar contidos em um mesmo intervalo numérico. Isso é importante visto que valores muito diferentes podem influenciar no treinamento e causar uma perda na capacidade de generalização da rede neural. (VALENÇA, 2009)

Uma das técnicas de normalização mais utilizadas na literatura é a transformação linear e foi ela a escolhida para este trabalho. A Equação 3.1 representa a fórmula utilizada para normalizar os valores da base de dados.

$$y = ((b - a) \times (\frac{x - x_{min}}{x_{max} - x_{min}})) + a \quad (3.1)$$

Na Equação 3.1, a e b representam os valores máximos e mínimos que os dados devem assumir. Neste trabalho, foi utilizado o valor de 0,15 para a e 0,85 para b.

Finalizada a etapa de normalização, a base de dados está pronta para ser inserida na técnica de classificação. Ressalta-se que, ao fim dos treinamentos, a taxa de acerto da rede neural deve ser calculada após o valor ser desnormalizado, ou seja, voltar para o seu intervalo inicial.



## 3.2 Implementação do RC

Conforme foi descrito anteriormente, há na literatura apenas uma implementação do RC em Matlab desenvolvida por Verstraeten (VERSTRAETEN, 2009). Dessa forma, um dos objetivos desse trabalho foi desenvolver uma ferramenta em JAVA que possibilitasse a simulação do RC.

Essa ferramenta abrange tanto problemas de previsão quanto os de classificação e, inicialmente, ela não possui recorrência entre neurônios da camada de saída. A recorrência nessa ferramenta está apenas nos neurônios do *reservoir* que, além de se conectarem com outros neurônios contidos no próprio *reservoir*, eles podem possuir a conexão com eles mesmos.

Quanto maior for o grau de recorrência e conectividade, maior a capacidade de memória da rede neural e, conseqüentemente, uma menor capacidade de aproximação não-linear. Como este trabalho trata da resolução de um problema estático, acredita-se que quanto menos dinâmico for o RC, ou seja, quanto menor o nível de recorrência e conectividade, melhor o resultado.

Antes de ter o seu treinamento iniciado, o RC desenvolvido neste trabalho necessita ter a sua topologia definida. Dessa forma, o usuário deve escolher a quantidade de neurônios que usará na camada de entrada, quantos neurônios no *reservoir* e quantos serão utilizados na camada de saída. Os neurônios do *reservoir* utilizam como função de ativação a sigmoide logística e os da saída, a função linear.

Os pesos utilizados no início da execução foram gerados aleatoriamente utilizando a função *random* da linguagem JAVA. Esses valores ficaram compreendidos no intervalo entre -1 e 1.

Segundo Verstraeten, para que haja uma estabilidade numérica, os pesos da matriz  $W_{res}$  devem ser normalizados até que o seu raio espectral seja menor ou igual a 1. Ainda não se sabe se essa técnica é a melhor para os problemas dinâmicos, porém, neste trabalho, manteve-se essa normalização a fim de validar o comportamento do RC desenvolvido.

Assim, com a topologia definida e as matrizes de pesos iniciais geradas aleatoriamente também calculadas, o RC tem o seu treinamento iniciado. O primeiro passo é realizar uma quantidade de ciclos de *warm up* definida pelo usuário apenas para atualizar os estados dos neurônios do *reservoir*.

Finalizada essa etapa, o treinamento, juntamente com a validação cruzada, são ini-

ciados. Durante o treinamento, os estados dos neurônios do *reservoir* são novamente atualizados e a matriz de pesos  $W_{out}$  é calculada através da pseudo-inversa de Moore-Penrose. Neste caso, para encontrar a pseudo-inversa, foi utilizada a biblioteca Jama disponível na linguagem Java.

Para o procedimento da validação cruzada, foi utilizado o Erro Médio Quadrático (EMQ) para verificar se o treinamento da rede neural já poderia ser finalizado.

Com o término do processo de treinamento, é apresentado à rede neural um conjunto de valores para testes, que não foram utilizados anteriormente, e a taxa de acerto ou o Erro Percentual Médio Absoluto (EPMA), para os casos de classificação e previsão respectivamente, são calculados.

O pseudocódigo que resume todo o algoritmo desenvolvido nesse trabalho encontra-se em Algoritmo 1.

---

**Algoritmo 1:** Pseudocódigo do RC

---

```
1 Definir a quantidade de neurônios na camada de entrada ;
2 Definir a quantidade de neurônios no reservoir ;
3 Definir a quantidade de neurônios na camada de saída ;
4 Gerar aleatoriamente os pesos da matriz Win entre -1 e 1;
5 Gerar aleatoriamente os pesos da matriz Wres entre -1 e 1;
6 Normalizar os pesos de Wres para que o raio espectral da matriz fique menor ou
  igual a 1;
7 while até o final da quantidade de ciclos de warm up do
8   | atualiza os estados dos neurônios do RC;
9 end
10 while até que o critério de parada seja atingido do
11   | for cada valor do conjunto de entrada do
12     | atualiza os estados dos neurônios do RC;
13   end
14   Calcula a pseudo-inversa de Moore-Penrose para encontrar a matriz de pesos
     que conecta o RC à camada de saída;
15   for cada valor do conjunto de validação cruzada do
16     | atualiza os estados dos neurônios do RC;
17   end
18   Calcula os valores de saída do RC;
19   Calcula o EMQ;
20   Verifica se o critério de parada já foi atingido;
21 end
22 for cada valor do conjunto de testes do
23   | atualiza os estados dos neurônios do RC;
24 end
25 Calcula os valores de saída do RC;
26 Calcula o EPMA // para o caso de previsão;
27 Calcula a taxa de acerto //para classificação;
```

---

Por se tratar de uma metodologia ainda recente e em estudos, vários dos parâmetros do RC são definidos empiricamente, como por exemplo, o grau de conectividade do RC, quantos neurônios são necessários no reservatório, a quantidade de ciclos de *warm up*, entre outros.

### 3.3 Simulação com o RC

Para realizar as simulações com as 4 bases de dados descritas na Seção 3.1 foram utilizadas as configurações do RC descritas na Tabela 2.

**Tabela 2: Representação dos parâmetros utilizados para as simulações com o RC**

[Fonte: elaboração própria]

Parâmetro	Valor
Conectividade do RC	20%
Quantidade de neurônios na camada de entrada	5 ou 10 (a depender da assinatura de proteínas que estiver sendo utilizada)
Quantidade de neurônios no RC	4
Quantidade de neurônios na camada de saída	2
Quantidade de ciclos de <i>warm up</i>	100
Função de ativação dos neurônios do <i>reservoir</i>	Sigmoide Logística
Função de ativação dos neurônios da camada de saída	Linear

Após definir as configurações do RC, foram realizadas 30 simulações com cada uma das bases de dados. Esse número é considerado ideal para realizar comparações estatísticas mais significantes (JURISTO; MORENO M., 2001). A Tabela 3 mostra detalhadamente a sequência das simulações.

**Tabela 3: Resumo das simulações realizadas com o RC e com a MLP.**

[Fonte: elaboração própria]

Quantidade de proteínas da assinatura utilizada	Conjunto de testes utilizado	Quantidade de simulações
5	Conjunto de testes DA	30
5	Conjunto de testes MCI	30
10	Conjunto de testes DA	30
10	Conjunto de testes MCI	30

Finalizadas as simulações, as taxas de acerto encontradas pelo RC foram armazenadas para a posterior comparação estatística com a MLP.

### 3.4 Simulação com a MLP

Assim como foi descrito na seção anterior, também é necessário definir alguns parâmetros da MLP antes de iniciar o treinamento. Foi utilizada uma MLP previamente desenvolvida e validada com *benchmarks* e outras bases de dados (SANTOS DE OLIVEIRA, 2012).

A geração dos pesos iniciais, tanto os que ligam a camada de entrada à camada escondida quanto os que conectam a camada escondida à camada de saída, foi aleatória e os valores ficaram dentro do intervalo entre -1 e 1. A Tabela 4 exibe os parâmetros e os seus valores utilizados para as execuções com a MLP.

**Tabela 4: Representação dos parâmetros utilizados para as simulações com a MLP**

[Fonte: elaboração própria]

Parâmetro	Valor
Quantidade de neurônios na camada de entrada	5 ou 10 (a depender da assinatura de proteínas que estiver sendo utilizada)
Quantidade de neurônios na camada escondida	20
Quantidade de neurônios na camada de saída	2
Função de ativação dos neurônios da camada escondida	Sigmoide Logística
Função de ativação dos neurônios da camada de saída	Linear
Taxa de aprendizagem	0,7
Taxa de momento	0,4

Foram realizadas, assim como foi com o RC, 30 simulações para cada uma das bases de dados. A sequência dessas execuções é a mesma que foi descrita na Tabela 3.

Ao final de todas as simulações, as taxas de acerto também foram armazenadas e o processo de análise estatística foi iniciado. Todos os testes utilizados estarão descritos na próxima seção.

## 3.5 Análise estatística

Finalizadas todas as simulações, fez-se necessário realizar uma sequência de testes estatísticos com o objetivo de validar cientificamente os resultados encontrados. Para isso, utilizou-se o software matemático R, já que ele contém todas as implementações dos testes que foram utilizados. Esse software utiliza como padrão um nível de significância ( $\alpha$ ) previamente definido no valor de 0,05.

As próximas seções descrevem quais os objetivos de cada um dos testes e como ele analisa os resultados encontrados.

### 3.5.1 Teste Shapiro-Wilk

Antes de utilizar um teste paramétrico em um conjunto de dados é necessário verificar se as amostras são normalmente distribuídas e se elas possuem variâncias estatisticamente iguais. Caso essas duas premissas sejam validadas, pode-se aplicar um teste paramétrico, caso contrário, deverá ser utilizado um teste não paramétrico.

O Teste de Shapiro-Wilk é utilizado com o objetivo de verificar se as amostras são normalmente distribuídas. Para isso, é necessária a formulação das hipóteses nula e alternativa conforme descreve a Tabela 5.

**Tabela 5: Hipóteses nula e alternativa definidas na aplicação do Teste de Shapiro-Wilk**

[Fonte: elaboração própria]

Hipótese	Descrição
<b>Hipótese nula</b>	A amostra provém de uma população normal
<b>Hipótese alternativa</b>	A amostra não provém de uma população normal

Dessa forma, ao se executar o teste para cada conjunto, deve-se analisar o *p-value* encontrado e comparar com o nível de significância adotado. Caso o *p-value* calculado seja menor que  $\alpha$ , a hipótese nula deve ser rejeitada e a amostra não deve ser considerada normalmente distribuída.

### 3.5.2 Teste F

Dando continuidade ao procedimento de verificação da possibilidade de aplicação de um teste paramétrico, o Teste F foi empregado com o objetivo de verificar se as variâncias entre dois conjuntos de amostras eram estatisticamente iguais, ou seja, oriundas de uma mesma população.

Da mesma forma que o teste anterior, faz-se necessário também a formulação das hipóteses para avaliar o resultado encontrado. Elas podem ser encontradas na Tabela 6.

**Tabela 6: Hipóteses nula e alternativa definidas na aplicação do Teste F**

[Fonte: elaboração própria]

Hipótese	Descrição
<b>Hipótese nula</b>	As amostras possuem variâncias estatisticamente iguais
<b>Hipótese alternativa</b>	As amostras não possuem variâncias estatisticamente iguais

Após a realização deste teste, caso o *p-value* calculado seja menor que o nível de significância adotado a hipótese nula deverá ser rejeitada e as amostras não podem ter

suas variâncias consideradas estatisticamente iguais.

Com o resultado deste teste e com o do descrito na seção anterior, pode-se tomar a decisão de aplicar ou não um teste paramétrico.

### 3.5.3 Teste T-Student

Esse teste estatístico faz a análise a partir das médias dos valores de duas amostras independentes. Por se tratar de um teste paramétrico, para que ele seja utilizado corretamente, é necessário ter validado anteriormente que as amostras são normalmente distribuídas e que elas são oriundas de uma mesma população, ou seja, suas variâncias são estatisticamente iguais.

Assim como todos os outros, para analisar o resultado obtido com sua execução, é necessário ter formulado previamente as hipóteses nulas e alternativas. O objetivo desse teste é verificar se as médias de cada uma das amostras são estatisticamente iguais, ou se há alguma diferença entre elas. Nesse último caso, a amostra que tiver a maior média, pode ser considerada com um melhor desempenho.

A Tabela 7 mostra as hipóteses nulas e alternativas utilizadas nesse trabalho.

**Tabela 7: Hipóteses nula e alternativa definidas na aplicação do Teste T-Student**

[Fonte: elaboração própria]

Hipótese	Descrição
<b>Hipótese nula</b>	As médias das amostras analisadas são estatisticamente iguais
<b>Hipótese alternativa</b>	As médias das amostras analisadas não são estatisticamente iguais

A análise deve ser feita através do *p-value* encontrado após a execução do teste. Caso ele seja menor que o nível de significância adotado, a hipótese nula é rejeitada e a amostra que tiver a maior média é considerada melhor.

No caso deste trabalho, ele deverá ser aplicado para comparar o desempenho entre o RC e a MLP e, caso a hipótese nula seja rejeitada, a topologia que apresentar a maior média de taxa de acerto será considerada melhor para o problema escolhido.

### 3.5.4 Teste da Soma dos Postos de Wilcoxon

Apesar de ser um teste não paramétrico e com um poder estatístico mais baixo que o Teste T-Student, o Teste da Soma dos Postos de Wilcoxon é bastante útil quando

se necessita comprovar os resultados e eles não são normais ou não possuem variâncias estatisticamente iguais.

Diferentemente do Teste T-Student, esse teste descrito nesta seção faz a sua análise baseado na mediana de cada uma das duas amostras independentes que estão sendo analisadas.

As hipóteses formuladas para a execução dos testes podem ser encontradas na Tabela 8.

**Tabela 8: Hipóteses nula e alternativa definidas na aplicação do Teste da Soma dos Postos de Wilcoxon**

[Fonte: elaboração própria]

Hipótese	Descrição
Hipótese nula	As medianas das amostras analisadas são estatisticamente iguais
Hipótese alternativa	As medianas das amostras analisadas não são estatisticamente iguais

Assim como nos testes anteriores, o *p-value* calculado deve ser comparado com o nível de significância adotado. Caso esse valor seja menor que alfa, a hipótese nula deve ser rejeitada implicando que há diferença significativa entre as duas amostras.

O próximo capítulo irá exibir os resultados das simulações efetuadas nesse trabalho bem como as análises realizadas a partir das execuções dos testes estatísticos.



## 4 *Resultados*

Este capítulo irá apresentar todos os resultados obtidos com este trabalho a partir da aplicação da metodologia descrita no Capítulo 3.

Como foi definido no capítulo anterior, 4 bases de dados foram definidas a fim de replicar o resultado encontrado por Moscato em (GÓMEZ RAVETTI; MOSCATO, 2008). Deve-se verificar o desempenho do RC para diagnóstico de DA e comparar esse resultado com o encontrado pela MLP.

### 4.1 Base de dados com 5 proteínas e conjunto de testes DA

Inicialmente foram realizadas as 30 simulações com o RC e MLP e a Tabela 9 exibe a média das taxas de acerto encontradas por cada uma das topologias utilizando a base de dados de 5 proteínas e o conjunto de testes DA.

**Tabela 9: Representação da média das taxas de acerto após os 30 experimentos com cada topologia utilizando a base de dados com 5 proteínas e o conjunto de testes DA.**

[Fonte: elaboração própria]

Base de dados com 5 proteínas e conjunto de testes DA	
Topologia da rede neural	Média da taxa de acerto (30 simulações)
RC	93,22%
MLP	95,61%

A partir desses resultados, a sequência de testes estatísticos foi realizada. Para cada conjunto de 30 simulações, foi realizado o teste de Shapiro-Wilk a fim de verificar a normalidade dessas amostras. Para isso, as hipóteses nula e alternativa foram definidas conforme mostra as Tabelas 10 e 11.

Através do software R, cujo nível de significância é definido com o valor de 0,05, esse teste demonstrou que ambas as amostras não eram normalmente distribuídas, visto

**Tabela 10: Hipóteses nula e alternativa definidas na aplicação do Teste de Shapiro-Wilk na amostra de dados do RC**

[Fonte: elaboração própria]

Amostra com os dados do RC – Teste de Shapiro-Wilk	
<b>Hipótese nula</b>	A amostra com as taxas de acerto do RC é normalmente distribuída.
<b>Hipótese alternativa</b>	A amostra com as taxas de acerto do RC não é normalmente distribuída.

**Tabela 11: Hipóteses nula e alternativa definidas na aplicação do Teste de Shapiro-Wilk na amostra de dados da MLP**

[Fonte: elaboração própria]

Amostra com os dados da MLP – Teste de Shapiro-Wilk	
<b>Hipótese nula</b>	A amostra com as taxas de acerto da MLP é normalmente distribuída.
<b>Hipótese alternativa</b>	A amostra com as taxas de acerto da MLP não é normalmente distribuída.

que o *p-value* calculado foi muito menor que 0,05. Dessa forma, foi aplicado o teste não-paramétrico Soma dos Postos de Wilcoxon.

O conjunto de hipóteses nula e alternativa definido para caracterizar o *p-value* encontrado na aplicação desse último teste está descrito na Tabela 12.

**Tabela 12: Hipóteses nula e alternativa definidas na aplicação do Teste da Soma dos Postos de Wilcoxon para as amostras de ambas as topologias de rede neural utilizadas.**

[Fonte: elaboração própria]

Amostra com os dados do RC e da MLP – Teste da Soma dos Postos de Wilcoxon	
<b>Hipótese nula</b>	As redes neurais RC e MLP apresentam desempenho estatisticamente iguais.
<b>Hipótese alternativa</b>	A rede MLP possui um melhor desempenho.

O resultado obtido através desse último teste foi um *p-value* muito inferior ao nível de significância. Dessa forma, pode-se concluir que a rede MLP para essa base de dados possui um desempenho melhor.

## 4.2 Base de dados com 10 proteínas e conjunto de testes DA

Finalizadas as simulações e testes estatísticos com a base de dados anterior, foram iniciadas as 30 simulações com cada uma das topologias com a base de dados com o conjunto de testes DA porém, levando-se em consideração 10 proteínas. As médias das taxas de acerto podem ser visualizadas na Tabela 13.

De posse dos 2 conjuntos referentes às 30 taxas de acerto de cada topologia, o teste de Shapiro-Wilk foi aplicado a partir do mesmo conjunto de hipóteses descrito nas Tabelas 10

**Tabela 13: Representação da média das taxas de acerto após os 30 experimentos com cada topologia utilizando a base de dados com 10 proteínas e o conjunto de testes DA.**

[Fonte: elaboração própria]

Base de dados com 10 proteínas e conjunto de testes DA	
Topologia da rede neural	Média da taxa de acerto (30 simulações)
RC	89,37%
MLP	93,44%

e 11 anteriormente.

Os valores para o *p-value* encontrados indicaram que a amostra de dados referente à MLP era normalmente distribuída ao contrário do RC que não foi considerada uma amostra normal.

Contudo, como pelo menos uma das amostras foi considerada não normal, o teste paramétrico T-Student não pode ser aplicado e, novamente, utilizou-se o teste Soma dos Postos de Wilcoxon. As hipóteses utilizadas para esse teste são as mesmas descritas na Tabela 12 da seção anterior.

Após a aplicação desse último teste, foi constatado que a MLP possui um desempenho melhor que o RC, visto que o valor encontrado para o *p-value* foi igual a 0,00012, ou seja, menor que o  $\alpha$  adotado pelo software R igual a 0,05.

### 4.3 Base de dados com 5 proteínas e conjunto de testes MCI

Essa seção irá descrever os resultados encontrados para a base de dados com 5 proteínas e conjunto de testes de MCI. Esse conjunto é naturalmente mais difícil de classificar, o que explica uma média da taxa de acerto mais baixa para ambas as topologias de rede neural. Elas podem ser visualizadas na Tabela 14.

**Tabela 14: Representação da média das taxas de acerto após os 30 experimentos com cada topologia utilizando a base de dados com 5 proteínas e o conjunto de testes MCI.**

[Fonte: elaboração própria]

Base de dados com 5 proteínas e conjunto de testes MCI	
Topologia da rede neural	Média da taxa de acerto (30 simulações)
RC	65,44%
MLP	69,14%

Assim como se decorreram os testes das seções anteriores, o mesmo procedimento

foi repetido para essa base de dados. As hipóteses criadas para a aplicação do teste de normalidade foram criadas e estão descritas nas Tabelas 10 e 11 da Seção 4.1.

Ao aplicar o teste de Shapiro-Wilk, foi constatado que a amostra dos valores das simulações com o RC foi considerada normalmente distribuída, diferentemente do conjunto da MLP que teve seu *p-value* calculado menor que 0,05, sendo considerada não normal.

Com isso, o teste não paramétrico foi aplicado e foi verificado, a partir da análise das hipóteses nula e alternativa descritas na Tabela 12, que a diferença entre as medianas da taxa de acerto era significativa. Portanto, a MLP apresenta um desempenho melhor do que o RC para esse conjunto de dados.

## 4.4 Base de dados com 10 proteínas e conjunto de testes MCI

Por fim, o último conjunto de dados foi simulado com ambas as topologias e a Tabela 15 mostra as médias das taxas de acerto calculadas após as 30 simulações com cada uma das topologias.

**Tabela 15: Representação da média das taxas de acerto após os 30 experimentos com cada topologia utilizando a base de dados com 10 proteínas e o conjunto de testes MCI.**

[Fonte: elaboração própria]

Base de dados com 10 proteínas e conjunto de testes MCI	
Topologia da rede neural	Média da taxa de acerto (30 simulações)
RC	67,23%
MLP	68,15%

O teste de Shapiro-Wilk foi executado para cada conjunto de amostras de cada uma das topologias. Seu resultado foi analisado a partir das hipóteses nula e alternativa definidas anteriormente nas Tabelas 10 e 11. Concluiu-se que a amostra referente à arquitetura RC foi considerada normalmente distribuída e à MLP não normal.

Já que o teste paramétrico necessita que as amostras sejam normalmente distribuídas e com variâncias estatisticamente iguais, ele não pode ser executado. Dessa forma, a utilização do teste Soma dos Postos de Wilcoxon, a partir das hipóteses nula e alternativas definidas na Tabela 12, resultou em um *p-value* igual a 0,7191.

Esse resultado implica que a hipótese nula não foi refutada já que *op-value* foi maior do que o nível de significância adotado pelo R que é igual a 0,05. Portanto, pode-se

concluir a partir desse resultado que, para essa base de dados, o RC e o MLP apresentam desempenhos estatisticamente iguais.

Os resultados de todos os testes estatísticos realizados podem ser encontrados no Apêndice A, assim como o script que foi escrito para possibilitar a execução desses testes no software R pode ser encontrado no Apêndice B.

A Tabela 16 resume os resultados encontrados pelas duas topologias de rede neural utilizadas nesse trabalho, juntamente com os valores encontrados por Moscato (GÓMEZ RAVETTI; MOSCATO, 2008).

**Tabela 16: Comparações dos resultados obtidos com a MLP, RC e com os disponíveis na literatura.**

[Fonte: elaboração própria]

Base de Dados	RC	MLP	Moscato
<b>5 proteínas e conjunto de testes DA</b>	93,22%	95,61%	93%
<b>10 proteínas e conjunto de testes DA</b>	89,37%	93,44%	89%
<b>5 proteínas e conjunto de testes MCI</b>	65,44%	69,44%	65%
<b>10 proteínas e conjunto de testes MCI</b>	67,23%	68,25%	66%

A partir da Tabela 16, pode-se concluir que as redes neurais obtiveram resultados próximos entre si e coerentes com os resultados descritos por Moscato. O próximo capítulo irá listar as conclusões obtidas com esse trabalho.

## 5 *Conclusões e Trabalhos Futuros*

A Doença de Alzheimer é hoje uma das doenças mais frequentes na população idosa mundial. Nos últimos anos, o número de pacientes cresceu bastante, visto que a expectativa de vida na maioria dos países desenvolvidos tem aumentado.

A DA é uma doença degenerativa, ou seja, as células cerebrais vão se deteriorando e não há como reverter o quadro da doença. Porém, quanto mais cedo os medicamentos forem administrados, melhor será a qualidade de vida do paciente, já que a medicação irá retardar a progressão dos sintomas.

Dessa forma, este trabalho teve como objetivo verificar o desempenho de uma nova abordagem conexionista de rede neural denominada *Reservoir Computing* para classificar, precocemente, se um paciente pode ser diagnosticado ou não com DA. Além disso, objetivou-se também realizar a comparação do desempenho do RC com a rede neural MLP e também com os resultados disponíveis na literatura.

Por ser uma metodologia recente, o estudo do RC se tornou a maior dificuldade para a execução deste projeto, já que há poucas fontes de informação disponíveis e a maioria dos seus parâmetros são configurados empiricamente.

A partir das simulações e testes estatísticos, pode-se concluir que a MLP apresentou desempenho superior na maioria dos casos. Isso pode ser explicado pelo fato de o RC ser mais adequado para o uso em sistemas dinâmicos já que ele apresenta uma boa capacidade de memorização.

Diferentemente do RC, a MLP apresenta uma melhor capacidade para aproximar funções não lineares e não contém a propriedade de armazenar os estados anteriores dos seus neurônios, ou seja, não possui recorrências.

Como trabalhos futuros, pretende-se realizar um estudo comparativo do RC entre a capacidade de aproximação não linear e a capacidade de memorização da rede, a fim de avaliar os parâmetros adequados para obtenção de melhores resultados.

Será realizado, também, um estudo aprofundado na parametrização do RC, como por exemplo, a definição do tamanho do raio espectral, quantos neurônios se devem colocar no *reservoir*, qual o grau de conectividade entre esses neurônios. Além disso, a fim de abordar problemas dinâmicos, será implementada a recorrência entre neurônios da camada de saída e desta com o reservatório.

Por fim, através desse trabalho, foi possível gerar uma ferramenta em JAVA para a simulação do comportamento do RC e ela deverá ser melhorada e utilizada para a execução desses trabalhos futuros.

## *Referências*

- ABRAZ. *Associação Brasileira de Alzheimer*. 2012. Acessado em 20 de Outubro de 2012. Disponível em: <<http://www.abraz.com.br/>>.
- ADI. *Alzheimer's Disease International*. 2012. Acessado em 15 de Outubro de 2012. Disponível em: <<http://www.alz.co.uk/>>.
- FERREIRA ARAÚJO, A. *Um Método para Design e Treinamento de Reservoir Computing Aplicado à Previsão de Séries Temporais*. Tese (Doutorado) — Universidade de Federal de Pernambuco, 2001.
- GIUSTI, E.; SURDO, V. *Alzheimer, cuidados e aconselhamento familiar: necessidades e tratamento psicológico da demência*. [S.l.]: Gryphus, 2010.
- GÓMEZ RAVETTI, M.; MOSCATO, P. Identification of a 5-protein biomarker molecular signature for predicting alzheimer's disease. *PLos One*, v3, p. 12p, 2008.
- GREEN, R. *Diagnóstico e Tratamento da Doença de Alzheimer e Outras Demências*. [S.l.]: Editora de Publicações Científicas Ltda., 2001.
- HAYKIN, S. *Redes Neurais: Princípios e Práticas*. [S.l.]: Bookman, 2007.
- JAEGER, H. *The echo state approach to analysing and training recurrent neural networks*. [S.l.], 2010.
- JURISTO, N.; MORENO M., A. *Basics of Software Engineering Experimentation*. [S.l.]: Kluwer Academic Publishers, 2001.
- MARQUES ROBERTO, d. B. P. *Demência tipo Alzheimer: diagnósticos, tratamento e aspectos sociais*. [S.l.]: Editora de Publicações Científicas Ltda., 1997.
- MCCULLOCH, W.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, p. p115–133, 1943.
- SANTOS DE OLIVEIRA, M. S. dos. *Uma Aplicação para Previsão de Recursos Hídricos Baseada em Redes Neurais e Enxames de Partículas*. Dissertação (Mestrado) — Universidade de Pernambuco, 2012.
- SAÚDE, P. *Portal da Saúde Brasileira*. 2012. Acessado em 15 de Julho de 2012. Disponível em: <<http://portal.saude.gov.br/saude/>>.
- VALENÇA, M. *Aplicando Redes Neurais: Um Guia Completo*. 1ªed.. ed. [S.l.]: Livro Rápido, 2005. 284p p.
- VALENÇA, M. *Fundamentos das Redes Neurais: exemplos em JAVA*. 2ªed.. ed. [S.l.]: Livro Rápido, 2009. 384p p.



- 
- VERSTRAETEN, D. *Reservoir Computing : computation with dynamical systems*. XXII, 178 p. Tese (Doutorado) — Ghent University. Faculty of Engineering, Ghent, Belgium, 2009.
- WERBOS J., P. *Beyond Regression: New tools for prediction and analysis in behavioral sciences*. Tese (Doutorado) — Harvard University, 1974.

## *APÊNDICE A – Resultados da Análise Estatística no Software R*

Neste Apêndice estão contidas as imagens referentes aos resultados encontrados após a execução dos testes estatísticos conforme foi descrito no Capítulo 4.

```
> shapiro.test(adRC5prot)

      Shapiro-Wilk normality test

data:  adRC5prot
W = 0.8977, p-value = 0.007365
```

Figura 5: Execução do teste de Shapiro-Wilk nos resultados das execuções com o RC utilizando a base de dados com 5 proteínas e o conjunto de testes DA.

[Fonte: Elaboração própria]

```
> shapiro.test(adMLP5prot)

      Shapiro-Wilk normality test

data:  adMLP5prot
W = 0.1796, p-value = 7.766e-12
```

Figura 6: Execução do teste de Shapiro-Wilk nos resultados das execuções com a MLP utilizando a base de dados com 5 proteínas e o conjunto de testes DA.

[Fonte: Elaboração própria]

```
> wilcox.test(adRC5prot,adMLP5prot)

      Wilcoxon rank sum test with continuity correction

data:  adRC5prot and adMLP5prot
W = 245, p-value = 0.0009168
alternative hypothesis: true location shift is not equal to 0
```

Figura 7: Execução do teste da Soma dos Postes de Wilcoxon para comparar o desempenho do RC e da MLP utilizando a base de dados com 5 proteínas e o conjunto de testes DA.

[Fonte: Elaboração própria]

```
> shapiro.test(adRC10prot)

      Shapiro-Wilk normality test

data:  adRC10prot
W = 0.4664, p-value = 2.406e-09
```

Figura 8: Execução do teste de Shapiro-Wilk nos resultados das execuções com o RC utilizando a base de dados com 10 proteínas e o conjunto de testes DA.

[Fonte: Elaboração própria]

```
> shapiro.test(adMLP10prot)

      Shapiro-Wilk normality test

data:  adMLP10prot
W = 0.9416, p-value = 0.1004
```

Figura 9: Execução do teste de Shapiro-Wilk nos resultados das execuções com a MLP utilizando a base de dados com 10 proteínas e o conjunto de testes DA.

[Fonte: Elaboração própria]

```
> wilcox.test(adRC10prot,adMLP10prot)

      Wilcoxon rank sum test with continuity correction

data:  adRC10prot and adMLP10prot
W = 192.5, p-value = 0.0001204
alternative hypothesis: true location shift is not equal to 0
```

Figura 10: Execução do teste da Soma dos Postes de Wilcoxon para comparar o desempenho do RC e da MLP utilizando a base de dados com 10 proteínas e o conjunto de testes DA.

[Fonte: Elaboração própria]

```
> shapiro.test(mciRC5prot)

Shapiro-Wilk normality test

data:  mciRC5prot
W = 0.9588, p-value = 0.288
```

Figura 11: Execução do teste de Shapiro-Wilk nos resultados das execuções com o RC utilizando a base de dados com 5 proteínas e o conjunto de testes MCI.

[Fonte: Elaboração própria]

```
> shapiro.test(mciMLP5prot)

Shapiro-Wilk normality test

data:  mciMLP5prot
W = 0.7099, p-value = 2.181e-06
```

Figura 12: Execução do teste de Shapiro-Wilk nos resultados das execuções com a MLP utilizando a base de dados com 5 proteínas e o conjunto de testes MCI.

[Fonte: Elaboração própria]

```
> wilcox.test(mciRC5prot,mciMLP5prot)

Wilcoxon rank sum test with continuity correction

data:  mciRC5prot and mciMLP5prot
W = 285, p-value = 0.01249
alternative hypothesis: true location shift is not equal to 0
```

Figura 13: Execução do teste da Soma dos Postes de Wilcoxon para comparar o desempenho do RC e da MLP utilizando a base de dados com 5 proteínas e o conjunto de testes MCI.

[Fonte: Elaboração própria]

```
> shapiro.test(mciRC10prot)

Shapiro-Wilk normality test

data:  mciRC10prot
W = 0.9327, p-value = 0.05777
```

Figura 14: Execução do teste de Shapiro-Wilk nos resultados das execuções com o RC utilizando a base de dados com 10 proteínas e o conjunto de testes MCI.

[Fonte: Elaboração própria]

```
> shapiro.test(mciMLP10prot)

      Shapiro-Wilk normality test

data:  mciMLP10prot
W = 0.8713, p-value = 0.001794
```

Figura 15: Execução do teste de Shapiro-Wilk nos resultados das execuções com a MLP utilizando a base de dados com 10 proteínas e o conjunto de testes MCI.

[Fonte: Elaboração própria]

```
> wilcox.test(mciRC10prot,mciMLP10prot)

      Wilcoxon rank sum test with continuity correction

data:  mciRC10prot and mciMLP10prot
W = 426, p-value = 0.7191
alternative hypothesis: true location shift is not equal to 0
```

Figura 16: Execução do teste da Soma dos Postes de Wilcoxon para comparar o desempenho do RC e da MLP utilizando a base de dados com 10 proteínas e o conjunto de testes MCI.

[Fonte: Elaboração própria]

*APÊNDICE B - Script utilizado para a  
execução da Análise  
Estatística no Software R*

```
dados<-read.csv("dados.csv")
adRC5prot<-c(dados[,1])
adMLP5prot<-c(dados[,2])
adRC10prot<-c(dados[,3])
adMLP10prot<-c(dados[,4])
mciRC5prot<-c(dados[,5])
mciMLP5prot<-c(dados[,6])
mciRC10prot<-c(dados[,7])
mciMLP10prot<-c(dados[,8])

shapiro.test(adRC5prot)
shapiro.test(adMLP5prot)
wilcox.test(adRC5prot,adMLP5prot)

shapiro.test(adRC10prot)
shapiro.test(adMLP10prot)
wilcox.test(adRC10prot,adMLP10prot)

shapiro.test(mciRC5prot)
shapiro.test(mciMLP5prot)
wilcox.test(mciRC5prot,mciMLP5prot)

shapiro.test(mciRC10prot)
shapiro.test(mciMLP10prot)
wilcox.test(mciRC10prot,mciMLP10prot)
```