



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2017/05.12.20.32-TDI

CLASSIFICAÇÃO DE PREVISIBILIDADE DO MODELO GLOBAL DO CPTEC UTILIZANDO BREEDING E INTELIGÊNCIA COMPUTACIONAL

Luis Fernando Salgueiro Romero

Dissertação de Mestrado do
Curso de Pós-Graduação em
Computação Aplicada, orientada
pela Dra. Sandra Aparecida
Sandri, aprovada em 23 de maio
de 2017.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3NSDQ7P>>

INPE
São José dos Campos
2017

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

E-mail: pubtc@inpe.br

COMISSÃO DO CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELECTUAL DO INPE (DE/DIR-544):

Presidente:

Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação (CPG)

Membros:

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

Dr. André de Castro Milone - Coordenação de Ciências Espaciais e Atmosféricas (CEA)

Dra. Carina de Barros Melo - Coordenação de Laboratórios Associados (CTE)

Dr. Evandro Marconi Rocco - Coordenação de Engenharia e Tecnologia Espacial (ETE)

Dr. Hermann Johann Heinrich Kux - Coordenação de Observação da Terra (OBT)

Dr. Marley Cavalcante de Lima Moscati - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Silvia Castro Marcelino - Serviço de Informação e Documentação (SID)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon

Clayton Martins Pereira - Serviço de Informação e Documentação (SID)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Simone Angélica Del Duca Barbedo - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Marcelo de Castro Pazos - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2017/05.12.20.32-TDI

CLASSIFICAÇÃO DE PREVISIBILIDADE DO MODELO GLOBAL DO CPTEC UTILIZANDO BREEDING E INTELIGÊNCIA COMPUTACIONAL

Luis Fernando Salgueiro Romero

Dissertação de Mestrado do
Curso de Pós-Graduação em
Computação Aplicada, orientada
pela Dra. Sandra Aparecida
Sandri, aprovada em 23 de maio
de 2017.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3NSDQ7P>>

INPE
São José dos Campos
2017

Dados Internacionais de Catalogação na Publicação (CIP)

Salgueiro Romero, Luis Fernando.

Sa32c Classificação de previsibilidade do modelo global do CPTEC utilizando Breeding e Inteligência Computacional / Luis Fernando Salgueiro Romero. – São José dos Campos : INPE, 2017.
xxviii + 135 p. ; (sid.inpe.br/mtc-m21b/2017/05.12.20.32-TDI)

Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2017.

Orientadora : Dra. Sandra Aparecida Sandri.

1. Técnica de Breeding. 2. Redes neurais artificiais.
3. Classificadores Neuro-Difusod. 4. Classificadores binários hierárquico.5. Modelo global do CPTEC I.Título.

CDU 004.032.26:551.51



Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

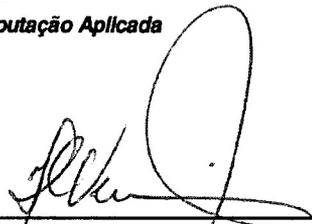
This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](https://creativecommons.org/licenses/by-nc/3.0/).

Aluno (a): *Luis Fernando Salgueiro Romero*

"CLASSIFICAÇÃO DE PREVISIBILIDADE DO MODELO GLOBAL DO CPTEC UTILIZANDO BREEDING E INTELIGÊNCIA COMPUTACIONAL"

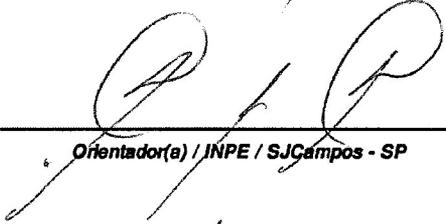
Aprovado (a) pela Banca Examinadora
em cumprimento ao requisito exigido para
obtenção do Título de *Mestre* em
Computação Aplicada

Dr. Haroldo Fraga de Campos Velho



Presidente / INPE / São José dos Campos - SP

Dra. Sandra Aparecida Sandri



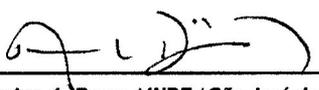
Orientador(a) / INPE / SJC Campos - SP

Dr. Stephan Stephany



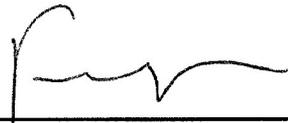
Membro da Banca / INPE / SJC Campos - SP

Dr. Antonio Divino Moura



Membro da Banca / INPE / São José dos Campos - SP

Dr. Fernando Antonio Campos Gomide



Convidado(a) / UNICAMP / Campinas - SP

Este trabalho foi aprovado por:

() *maioria simples*

() *unanimidade*

São José dos Campos, 23 de maio de 2017

"Quando nada parece ajudar, eu vou e olho o cortador de pedras martelando sua rocha talvez cem vezes sem que nem uma só rachadura apareça. No entanto, na centésima primeira martelada, a pedra se abre em duas e eu sei que não foi aquela martelada a que conseguiu, mas todas as que vieram antes. "

-JACOB A. RIIS

AGRADECIMENTOS

Agradeço a meus pais, Luis e Teresa, pelo apoio e suporte permanentes na minha vida toda. Por ser uma fonte de inspiração e sacrifício para mim e minha família. A minhas irmãs Angélica, Andrea e Araceli que souberam suprir minha ausência na família, pela compreensão e carinho de sempre.

Sou grato a minha namorada Ana Magdalena, que foi sempre um apoio constante, pela ajuda inestimável e pela compreensão da minha ausência em momentos difíceis da nossas vidas.

Agradeço à Dra. Sandra Sandri pela orientação e paciência, por mostrar-me o caminho certo em momentos complicados do trabalho, pela confiança depositada e pelo rigor adequado para fazer com que este trabalho seja o melhor possível. Agradeço também ao Dr. Haroldo Fraga de Campos Velho, por mostrar-me seu grande entusiasmo pela ciência, e por acreditar em mim em todo momento. Foi um honra para mim ter trabalhado com dois.

Agradeço a todos os professores da pós-graduação, que me ajudaram em todo momento que precisei. Sempre achei as portas abertas para tirar dúvidas. Um agradecimento especial a Dra. Rosângela Cintra, ao Dr. Celso Mendes e Dr. Paulo Kubota, que me ajudaram muito neste trabalho.

Também sou grato ao CNPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico, pelo apoio financeiro e ao Instituto Nacional de Pesquisas Espaciais (INPE) por ter me aberto as portas para esta oportunidade de crescimento acadêmico.

Finalmente, agradeço a todos os meus amigos, ao Diego Stalder e sua esposa Mariam, aos amigos da pós-graduação do INPE, em especial Luan Gaspar Pinto de Melo, Michael Douglas dos Santos Lemes, Raian Vargas Maretto, Reynier Hernandez Torres, Juliana Marino, Juliana Lacerda, Luciano Magrini e a toda a turma de 2015. Aos amigos da salinha 15 do LAC, a Sabrina Sambati, Juliana Anochi, Renata Ruiz, Ivana Sumida, José Renato Garcia, Marco Nardes, a secretária Jéssica e todos os que fizeram parte da minha vida nos momentos bons e ruins deste 2 anos de mestrado.

RESUMO

A Previsão Numérica de Tempo (PNT) teve um grande avanço nas últimas décadas devido ao aumento do poder computacional, novas parametrizações de processos físicos e o aumento de dados de observação. A PNT tem enorme impacto positivo nas atividades humanas. Os fenômenos que descrevem a dinâmica da atmosfera são não lineares e podem apresentar comportamento caótico, sendo altamente sensíveis às condições iniciais. Incertezas inerentes ao modelo atmosférico limitam a qualidade da previsão. Uma abordagem para o estudo da evolução e propagação desses erros é a técnica de *Breeding*, que consiste em uma avaliação periódica de ciclos de duas simulações do mesmo modelo: uma simulação partindo das condições iniciais de referência e a outra com as condições iniciais perturbadas. *Bred Vectors* são diferenças entre as simulações anteriormente mencionadas, em que a magnitude do vetor pode ser utilizada como uma medida da incerteza da previsão. Esta dissertação tem por objetivo utilizar a técnica de *Breeding* com algoritmos de inteligência computacional para avaliar a qualidade da previsão do modelo de circulação atmosférica global do CPTEC-INPE. A primeira fase do trabalho consistiu na instalação do modelo global do CPTEC-INPE do supercomputador Cray XE6 (Tupã) - compilador Portland, para o cluster Lacibrido (LAC-INPE) - compilador GNU. A técnica de *Breeding* foi comparada com previsão por conjuntos (*Ensemble*). Ocorreu uma concordância nas regiões no planeta entre a maior magnitude do *Bred Vector* e maior variância do *Ensemble*. Classes de confiabilidade da previsão foram definidas e a classificação foi realizada utilizando sistemas neuro-difusos, redes neurais artificiais, classificadores hierárquicos binários e máquinas de comitê. Os resultados obtidos são promissores, com índices de classificação satisfatórios, que possibilitam a introdução da metodologia de *Breeding* como uma ferramenta de estudo de previsibilidade de modelos atmosféricos.

Palavras-chave: Técnica de *Breeding*. Redes Neurais Artificiais. Classificadores Neuro-Difusos. Classificadores Binários Hierárquicos. Método do Comitê. Modelo Global do CPTEC. Sistema de previsão por Conjuntos.

CLASSIFICATION OF PREDICTABILITY OF THE CPTEC GLOBAL MODEL USING BREEDING AND COMPUTATIONAL INTELLIGENCE

ABSTRACT

Numerical Weather Prediction (NWP) has made great progress in the last decades due to the increase of computational power, new parameterizations of physical processes and the increase of observational data. NWP has a huge positive impact on human activities. The phenomena that describe the dynamics of the atmosphere are non-linear and can present chaotic behavior, being highly sensitive to the initial conditions. Thus, uncertainties inherent to the atmospheric model limit the quality of the forecast. One approach to study the evolution and propagation of these errors is the Breeding method which consists of a periodic evaluation of two simulations using the same model: one simulation starting from initial reference conditions and the other with disturbed initial conditions. Bred vectors are differences between the simulations mentioned above, in which the magnitude of the vector can be used as a measure of the uncertainty of the prediction. This dissertation aims to use the breeding technique with artificial intelligence algorithms to evaluate the prediction quality of the CPTEC-INPE global atmospheric circulation model. The first stage of the work consisted in the installation of the CPTEC-INPE global model of the Cray XE6 supercomputer (Tupã) - Portland compiler - to the Lacibrido cluster (LAC-INPE) - GNU compiler. The breeding technique was compared to ensemble prediction. There was a good agreement in the regions of the planet between the greater magnitude of the bred vector and larger variance of the ensemble. Prediction reliability classes were defined and classification was performed using neuro-fuzzy systems, artificial neural networks, binary hierarchical classifiers and committee method. The results obtained are promising, with satisfactory indexes of classification, which makes possible the introduction of the Breeding methodology as a tool for studying the predictability of weather models.

Keywords:

Breeding Technique. Neuro-Fuzzy Classifiers. Artificial Neural Networks. Binary hierarchical classifier. committee method. CPTEC's Global Atmospheric Model. Ensemble forecast.

LISTA DE FIGURAS

	<u>Pág.</u>
2.1 Estrutura Esquemática do Modelo de Circulação Geral da Atmosfera do CPTEC.	13
2.2 Esquema de um sistema de previsão por Conjuntos ou metodologia <i>Ensemble</i>	14
2.3 Esquema de um sistema de previsão por Conjuntos ou metodologia <i>Ensemble</i>	15
2.4 Esquema de previsão utilizando Ensemble.	17
2.5 Previsão de 15 dias utilizando a metodologia <i>Ensemble</i> para a previsão da altura geopotencial (m).	18
3.1 Esquema da técnica de <i>Breeding</i> e criação dos <i>bred vectors</i>	20
3.2 Simulação do sistema de Lorenz 3D	21
3.3 <i>Bred vectors</i> aplicados ao atrator estranho do sistema dinâmico de Lorenz em 3D	22
3.4 <i>Bred vectors</i> aplicados ao sistema de Lorenz para a variável $x(t)$	22
3.5 Atrator do modelo de ondas 3D aplicados ao sistema dinâmico de 3 ondas acopladas	23
3.6 Atrator do modelo de ondas 3D com a codificação da taxa de crescimento do <i>Bred Vector</i>	24
3.7 Serie temporal $x(t)$ do modelo de ondas 3D com a codificação da taxa de crescimento do <i>Bred Vector</i>	24
4.1 Comparação entre o neurônio biológico e o neurônio artificial	28
4.2 Funções de ativação	29
4.3 Tipos principais de arquiteturas de RNAs	30
4.4 Arquitetura da rede neural.	33
4.5 Função de pertinência triangular	34
4.6 Termos difusos (<i>baixa, média, alta</i>) para a variável linguística <i>temperatura</i>	35
4.7 Esquema de um sistema de inferência difusa.	37
4.8 Arquitetura básica de um sistema Neuro-Difuso.	40
4.9 Desempenho da classificação com software GUAJE.	41
4.10 Esquema de classificador hierárquico binário com 5 classes	43
4.11 Método de Comitê Estático	45
4.12 Método de Comitê Dinâmico	45

5.1	Campo de diferença entre a condição inicial de controle e a condição inicial perturbada, no campo da temperatura virtual a nível de superfície.	50
5.2	<i>Bred vectors</i> da temperatura de superfície.	51
5.3	Espalhamento (desvio padrão) das previsões por <i>Ensemble</i> do modelo global no campo de temperatura de superfície.	52
5.4	<i>Bred vectors</i> da temperatura de superfície e gráficos espagete dos membros do <i>Ensemble</i> .	53
5.5	<i>Bred vectors</i> da temperatura de superfície e gráficos espagete dos membros do <i>Ensemble</i> .	54
5.6	<i>Bred vectors</i> da temperatura de superfície e a variância dos membros do <i>Ensemble</i> para a América do Sul.	55
5.7	<i>Bred vectors</i> da temperatura de superfície e gráficos de espagete dos membros do <i>Ensemble</i> para a região de América do Sul.	57
5.8	<i>Bred vectors</i> da temperatura de superfície e variância dos membros do <i>ensemble</i> para a região de América do Sul para a data de 17 de maio de 2012	58
6.1	Região de estudo na América do Sul marcada por um quadro vermelho, correspondente à latitude=[11'S - 20'S] e à longitude=[42'W - 50'W], tendo como fundo o desvio padrão da temperatura superficial dos membros do sistema de previsão por conjuntos (<i>Ensemble</i>) para 02 de maio de 2014 às 00 Hs	62
6.2	Histogramas da magnitude do desvio padrão da previsão de temperatura superficial do sistema de previsão por conjuntos, na região de estudo na América do Sul, correspondente à latitude=[11'S - 20'S] e à longitude=[42'W - 50'W].	63
6.3	Representação de dados em forma contínua e em pontos de grade.	64
6.4	Janelas para montagem da tabela de entrada-saída: 1 ponto e 9 pontos	65
6.5	Obtenção dos dados a partir de janela com 9 pontos em torno de (i, j)	65
6.6	Esquema final da tabela de dados com 15 dias de simulação e 100 pontos na grade.	66
6.7	Esquema de partição para geração de classes.	67
6.8	Histogramas da magnitude do desvio padrão da previsão de temperatura superficial e das classes correspondentes.	68
6.9	Esquema de geração dos dados de treinamento e teste para misturar os valores dos anos 2011 até 2014.	72
6.10	Comparação de performance de sensibilidade de classificação como rede Neuro-difusas	75
6.11	Comparação de performance de sensibilidade de classificação com RNA	77

6.12	Desempenho de cada fase do classificador hierárquico	81
6.13	Comparação de performance de sensibilidade de classificação das redes Neuro-difusas, RNA e o classificador Hierárquico para o conjunto de da- dos alternados	83
6.14	Histograma da magnitude do desvio padrão e o conjunto de dados con- siderado para geração das classes de confiabilidade	86
6.15	Histograma da magnitude do desvio padrão e as classes de confiabilidade gerada a partir desses dados	87
6.16	Comitê estático implementado, com unidade decisora	91
6.17	Comitê estático implementado, sem unidade decisora	93
6.18	Comitê dinâmico implementado, sem unidade decisora na fase especialista	95
6.19	Comitê dinâmico implementado, com um contraste entre classes	97
B.1	Janela Principal do software GUAJE	119
B.2	Histogramas da magnitude do desvio padrão do <i>Ensemble</i> e das classes geradas	120
B.3	Janela Expert , de configuração dos conjuntos de entrada-saída	121
B.4	Configuração dos conjuntos de entrada e saída	122
B.5	Janela de Processos de inferência	123
B.6	Janela de processo de escolha do algoritmo de partição difusa	124
B.7	Regras geradas com o algoritmo de Wang-Mendel	125
B.8	Regras geradas adicionadas à base de conhecimentos	126
B.9	Janela de resultados de classificação	127
B.10	Janela de resultados da classificação - Matrizes de confusão	128
C.1	Regras de inferência geradas	130
C.2	Classificação utilizado a o conjunto de treinamento	130
C.3	Resultados da classificação utilizando o software GUAJE com 3 atributos de entrada.	132
C.4	Regras de inferência geradas utilizando o software GUAJE	133
C.5	Resultados da classificação utilizando o software GUAJE com 16 atribu- tos de entrada	133
C.6	Variação da percentagem da acurácia global da classificação e o aumento das regras de inferência conforme o aumento de atributos	134
C.7	Variação da taxa de acerto para 3 e 16 atributos de entrada	135

LISTA DE TABELAS

	<u>Pág.</u>
4.1 Principais T-normas e T-Conormas duais	36
4.2 Principais operadores de implicação em aplicações difusas	36
4.3 Comparação entre os sistemas difusos, RNA e redes neuro-difusas	42
4.4 Matriz de confusão	46
6.1 Distribuição de amostras na amostragem cronológica	68
6.2 Matriz de confusão do conjunto de treinamento - amostragem cronológica	69
6.3 Matriz de confusão do conjunto de teste (2014) - amostragem cronológica	70
6.4 Matriz de confusão do conjunto de treinamento - amostragem cronológica	71
6.5 Matriz de confusão do conjunto de teste - amostragem cronológica	71
6.6 Distribuição de amostras na amostragem alternada	72
6.7 Matriz de confusão do conjunto de treinamento - amostragem alternada .	74
6.8 Matriz de confusão do conjunto de teste - amostragem alternada	74
6.9 Matriz de confusão do conjunto de treinamento - amostragem alternada	75
6.10 Matriz de confusão do conjunto de teste da amostragem alternada	76
6.11 Matriz de Confusão do conjunto de treinamento, fase HH vs \sim HH.	77
6.12 Matriz de Confusão do conjunto de teste, fase HH vs \sim HH.	78
6.13 Matriz de Confusão do conjunto de treinamento, fase H vs \sim H.	78
6.14 Matriz de Confusão do conjunto de teste, fase H vs \sim H.	79
6.15 Matriz de Confusão do conjunto de treinamento, fase M vs \sim M	79
6.16 Matriz de Confusão do conjunto de teste, fase M vs \sim M.	79
6.17 Matriz de Confusão do conjunto de treinamento, fase L vs LL.	80
6.18 Matriz de Confusão do conjunto de teste, fase L vs LL.	80
6.19 Desempenho da sensibilidade das classes no teste global do classificador hierárquico binário	82
6.20 Distribuição de amostras para os conjuntos de treinamento-teste para as redes especialistas	87
6.21 Matriz de confusão treinamento da rede especialista Classe 1 (HH)	88
6.22 Matriz de confusão teste - rede especialista classe 1 (HH)	88
6.23 Matriz de confusão treinamento da rede especialista Classe 2 (H)	88
6.24 Matriz confusão de teste da rede especialista Classe 2 (H)	88
6.25 Matriz de confusão treinamento da rede especialista Classe 3 (M)	89
6.26 Matriz de confusão teste da rede especialista Classe 3 (M)	89
6.27 Matriz de confusão treinamento da rede especialista Classe 4 (L)	89
6.28 Matriz de confusão teste da rede especialista Classe 4 (L)	89

6.29	Matriz de confusão treinamento da rede especialista Classe 5 (LL)	90
6.30	Matriz confusão teste da rede especialista Classe 5 (LL)	90
6.31	Matriz de confusão do conjunto de treinamento da RNA integradora	91
6.32	Matriz de confusão do conjunto de teste da RNA integradora	92
6.33	Matriz de confusão do conjunto de treinamento da RNA integradora sem unidade decisora	93
6.34	Matriz de confusão do conjunto de teste da RNA integradora sem unidade decisora	94
6.35	Matriz de confusão do conjunto de treinamento da RNA integradora dinâmica	95
6.36	Matriz de confusão do conjunto de teste da RNA integradora dinâmica	96
6.37	Matriz de confusão treinamento da rede especialista 1 - Classe 1 (HH) vs Classe 2 (H)	97
6.38	Matriz de confusão teste - rede especialista Classe 1 (HH) vs Classe 2 (H)	98
6.39	Matriz de confusão treinamento da rede especialista 2 - Classe 1 (HH) vs Classe 3 (M)	98
6.40	Matriz de confusão teste - rede especialista 2 - Classe 1 (HH) vs Classe 3 (M)	98
6.41	Matriz de confusão treinamento da rede especialista 3 - Classe 1 (HH) vs Classe 4 (L)	98
6.42	Matriz de confusão teste - rede especialista 3 - Classe 1 (HH) vs Classe 4 (L)	99
6.43	Matriz de confusão treinamento da rede especialista 4 - Classe 1 (HH) vs Classe 5 (LL)	99
6.44	Matriz de confusão teste - rede especialista 4 - Classe 1 (HH) vs Classe 5 (LL)	99
6.45	Matriz de confusão treinamento da rede especialista 5 - Classe 2 (H) vs Classe 3 (M)	99
6.46	Matriz de confusão teste - rede especialista 5 - Classe 2 (H) vs Classe 3 (M)	100
6.47	Matriz de confusão treinamento da rede especialista 6 - Classe 2 (H) vs Classe 4 (L)	100
6.48	Matriz de confusão teste - rede especialista 6 - Classe 2 (H) vs Classe 4 (L)	100
6.49	Matriz de confusão treinamento da rede especialista 7 - Classe 2 (H) vs Classe 5 (LL)	100
6.50	Matriz de confusão teste - rede especialista 7 - Classe 2 (H) vs Classe 5 (LL)	101
6.51	Matriz de confusão treinamento da rede especialista 8 - Classe 3 (M) vs Classe 4 (L)	101

6.52	Matriz de confusão teste - rede especialista 8 - Classe 3 (M) vs Classe 4 (L)	101
6.53	Matriz de confusão treinamento da rede especialista 9 - Classe 3 (M) vs Classe 5 (LL)	101
6.54	Matriz de confusão teste - rede especialista 9 - Classe 3 (M) vs Classe 5 (LL)	102
6.55	Matriz de confusão treinamento da rede especialista 10 - Classe 4 (L) vs Classe 5 (LL)	102
6.56	Matriz de confusão teste - rede especialista 10 - Classe 4 (L) vs Classe 5 (LL)	102
6.57	Matriz de confusão do conjunto de treinamento da RNA integradora dinâmica classe vs classe	103
6.58	Matriz de confusão do conjunto de teste da RNA integradora dinâmica classe vs classe	103
C.1	Matriz de Confusão do conjunto de treinamento com 16 atributos de entrada	134

LISTA DE ABREVIATURAS E SIGLAS

AGCM	–	<i>Atmospheric Global Circulation Model</i>
ANFIS	–	<i>Adaptative-Network-based Fuzzy Inference System</i>
BV	–	<i>Bred Vectors</i>
BAM	–	<i>Brazilian Atmospheric Model</i>
CFL	–	Courant-Friedrich-Lewy Condition
CPTEC	–	Centro de Previsão de Tempo e Estudos Climáticos
COLA	–	<i>Center for Ocean, Land and Atmosphere Studies</i>
ECMWF	–	<i>European Center for Medium-Range Weather Forecast</i>
EDP	–	Equações Diferenciais Parciais
EDF	–	Equações em diferenças finitas
ENIAC	–	<i>Electronic Numerical Integrator And Computer</i>
EOF	–	<i>Empiric Ortogonal Function</i>
FSU	–	<i>Florida State University</i>
GDAS	–	<i>Global Data Assimilation System</i>
GUAJE	–	<i>Java Environment for Generating Understandable and Accurate</i>
GFDL	–	<i>Geophysical Fluid Dynamics Laboratory</i>
INPE	–	Instituto Nacional de Pesquisas Espaciais
MRF	–	<i>Medium Range Forecasting Model</i>
SPCON	–	Sistemas de Previsão por Conjunto
PNT	–	Previsão Numérica de Tempo
RNA	–	Redes Neurais Artificiais

LISTA DE SÍMBOLOS

u	–	Velocidade do Vento (eixo x)
v	–	Velocidade do Vento (eixo y)
w	–	Velocidade do Vento (eixo z)
p	–	pressão
ρ	–	Densidade do fluido
T	–	Temperatura
q_v	–	Umidade Específica
Ω	–	Frequência de rotação da terra
ϕ	–	Latitude
a	–	Raio da Terra
k	–	Quantidade de números de ondas espectrais
γ	–	Taxa de lapso da temperatura
γ_d	–	Taxa adiabática seca de lapso
c_p	–	Calor específico do ar a pressão constante
g	–	Aceleração da gravidade
H	–	Ganho ou perda de calor
Q_v	–	Ganho ou perda de vapor d'água entre trocas de fases
Fr	–	Fricção genérica em cada direção
Λ	–	Universo de um conjunto difuso

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
1.1 Objetivos	5
1.2 Motivação	5
1.3 Organização da dissertação	6
2 FUNDAMENTOS TEÓRICOS DA PREVISÃO NUMÉRICA DE TEMPO	7
2.1 Equações que governam a dinâmica da atmosfera	7
2.2 Métodos Numéricos	8
2.3 Parametrizações	10
2.4 Assimilação de dados	10
2.5 Modelo AGCM-CPTEC	11
2.6 Metodologia do Sistema de Previsão por Conjuntos (<i>Ensemble</i>)	14
2.7 Sumário do capítulo	18
3 O MÉTODO DE <i>BREEDING</i>	19
3.1 Descrição do modelo e geração dos <i>Bred Vectors</i>	19
3.2 Sumário do capítulo	25
4 ALGORITMOS DE INTELIGÊNCIA COMPUTACIONAL UTILIZADOS	27
4.1 Conceitos básicos de Redes Neurais Artificiais (RNAs)	27
4.1.1 Arquitetura da rede	29
4.1.2 Processo de aprendizagem	30
4.2 Conceitos básicos de Sistemas Difusos	33
4.2.1 Operações com conjuntos difusos	35
4.2.2 Sistemas de inferência difusa	37
4.2.2.1 Modelo Takagi-Sugeno	37
4.2.2.2 Modelo Mamdani	38
4.3 Sistemas Neuro-Difusos	39
4.4 Diferenças entre sistemas difusos, redes neurais artificiais e sistemas neuro-difusos	41
4.5 Classificadores Hierárquicos	42

4.6	Máquinas de Comitê	44
4.7	Medidas de desempenho da classificação utilizadas	45
5	PREVISIBILIDADE: <i>BREEDING</i> APLICADO AO MODELO GLOBAL DO CPTEC	49
5.1	Sumário	59
6	PREVISIBILIDADE: IDENTIFICAÇÃO DE CLASSES	61
6.1	Região de Estudo e obtenção de dados de entrada-saída	61
6.1.1	Conjuntos de dados de treinamento e teste	64
6.1.2	Mapeamento de desvio padrão em classes de previsibilidade	66
6.2	Classificação com amostragem cronológica	68
6.2.1	Classificação com a rede neuro-difusa GUAJE	69
6.2.2	Classificação utilizando redes neurais	70
6.3	Classificação com amostragem alternada	72
6.3.1	Classificação utilizando uma rede Neuro-Difusa	73
6.3.2	Classificação com uma Rede Neural	75
6.3.3	Classificação utilizando uma árvore hierárquica binária	77
6.3.4	Fase 1: Classe HH vs \sim HH	77
6.3.4.1	Fase 2: Classe H vs \sim H	78
6.3.4.2	Fase 3: Classe M vs \sim M	79
6.3.4.3	Fase 4: Classe L vs \sim L	80
6.3.4.4	Desempenho das fases no treinamento e teste - resumo	80
6.3.4.5	Cálculo da taxa global de classificação do algoritmo hierárquico binário	81
6.3.5	Comparação dos classificadores	82
6.3.6	Método de Comitê	83
6.3.6.1	Conjuntos de treinamento e teste para o método de comitê - experimento grupal	84
6.3.6.2	Matrizes de confusão das distintas redes especialistas do método de comitê - Experimento grupal	88
6.3.6.3	Estratégias de treinamento-teste da rede integradora - experimento grupal	90
6.3.6.4	Máquinas de comitê dinâmicas com contraste entre duas classes - Experimento dual	96
6.4	Resumo do capítulo	104
7	CONCLUSÕES E TRABALHOS FUTUROS	105
7.1	Trabalhos Futuros	107

REFERÊNCIAS BIBLIOGRÁFICAS	109
APÊNDICE A - Método de <i>Breeding</i> aplicado ao modelo global	
SPEEDY	115
A.1 Descrição do modelo	115
A.2 Parametrizações Físicas	116
A.3 Condições de borda	116
A.4 Trabalho apresentado na Conferência de Ciências não lineares e Complexidade 2016	116
APÊNDICE B - Breve tutorial de uso do software GUAJE . .	119
APÊNDICE C - Primeiros testes de classificação realizados . .	129
C.0.1 Primeiros testes: Região de 100x100	129
C.0.2 Aumentando a quantidade de atributos	131

1 INTRODUÇÃO

A Previsão Numérica do Tempo (PNT) é uma disciplina jovem que utiliza técnicas matemáticas e computacionais para resolver equações que regem o comportamento dinâmico da atmosfera, procurando obter o estado futuro a partir de valores iniciais que são conhecidos por meio de observações (COIFFIER, 2011). As soluções destas sofisticadas equações são calculadas numericamente através de supercomputadores instalados em centros operacionais e de pesquisas. De fato, a operação e pesquisa em tempo e clima são tarefas que sempre demandaram sistemas de processamento de última geração, devido à complexidade e da busca de resolução (discretização espacial e temporal) sempre a mais refinada possível do modelo e da grande quantidade de dados manipulados nas operações de pré-processamento, execução e pós-processamento.

No início do século passado, iniciam-se as pesquisas sobre a possibilidade de previsão do estado da atmosfera, utilizando equações matemáticas que descrevem a física e dinâmica destes fenômenos. A PNT desde então veio ganhando preponderância não somente no ambiente acadêmico, mas também tornou-se um recurso significativo nas decisões econômicas dos países, assim como no cotidiano das pessoas.

A primeira tentativa de previsão continha erros grosseiros, mas a PNT aprimorou muito os resultados em termos de acurácia com os avanços da análise matemática, dos computadores, da parametrização dos processos físicos e dos sistemas de observação das últimas décadas. A PNT se encontra em constante desenvolvimento, levando em consideração cada vez fenômenos mais complexos, que governam a dinâmica da atmosfera.

São várias as ciências (meteorologia, oceanografia, física, matemática, computação) que contribuem no desenvolvimento destes modelos meteorológicos. Isto tipifica um caráter interdisciplinar de colaboração, com a finalidade de gerar modelos mais precisos e capazes de simular uma quantidade maior de processos.

Em 1904, Wilhelm Bjerknes publicou um trabalho sobre os fundamentos teóricos da previsão de tempo, resolvendo um sistema não linear de equações diferenciais parciais. Ele definiu o problema de previsão como um problema de valor inicial. Anos depois, Lewis Fry Richardson calculou uma previsão de 6 horas para as variações de pressão, com o uso de técnicas numéricas para resolver o sistema de equações diferenciais da dinâmica (LYNCH, 2014). Embora os resultados estivessem fora da realidade. A previsão de Richardson foi para o dia 20 de maio de 1910 e estimou

uma variação de pressão de 145 HPa em 6 horas (LYNCH, 2014). Entre as razões para a falha da previsão, pode-se citar o desbalanço entre os campos de vento e de pressão e a inadequada discretização do tempo – excessivamente grande, gerando instabilidade numérica. Mas, como menciona Kalnay (2003), o real valor do trabalho foi lançar as bases de como atacar o problema de prever o tempo e clima de uma forma mais metodológica.

Algumas décadas depois dos trabalhos de Richardson e Bjerknes, com um melhor entendimento dos processos numéricos e meteorológicos, uma equipe de destacados cientistas liderados por John Von Neumann e Jule Charney lograram realizar a primeira simulação com sucesso de uma *previsão a posteriori*¹ de tempo para um dia, utilizando o computador da época (ENIAC: Electronic Numerical Integrator and Computer). Com estes resultados, eles mostraram a viabilidade da utilização de modelos numéricos para fazer previsão de tempo (NCEP, 2007). J. Von Neumann também percebeu a oportunidade de atacar o problema da climatologia utilizando simulações por modelos globais da atmosfera e dos oceanos. É este trabalho pioneiro e entusiasta que dá início a uma era moderna da meteorologia (MOURA, 1996)

Conforme foram evoluindo as pesquisas nesta área, a qualidade da previsão foi melhorando ano a ano. São 4 fatores que possibilitaram esse avanço (KALNAY, 2003):

- o incremento da potência de cálculo com a chegada dos supercomputadores que permitem malhas mais finas e melhores aproximações,
- parametrizações de processos físicos, como por exemplo formações de nuvens e turbulência,
- melhoras nos métodos de obtenção de observações e a área conhecida como assimilação de dados, que permitem iniciar os modelos com mais dados precisos, e
- aumento da qualidade e quantidade de dados observacionais, através de dados de satélites e outras fontes.

No início dos anos 60, surge uma questão importante: o limite da previsibilidade do tempo e clima. A previsibilidade, segundo Lorenz (1984) é o grau de acurácia com que é possível prever o estado futuro da atmosfera. Edward Lorenz demonstrou que a dinâmica da atmosfera é sensível às condições iniciais (LORENZ, 1963). Lorenz,

¹Previsão *a posteriori* é uma previsão experimental de uma situação que já aconteceu.

considerado como um dos principais meteorologistas e um dos pais da ciência do caos, gerou evidências deste fato quando tentava replicar resultados de um experimento de previsão numérica de tempo. Ele notou que esta sensibilidade em modelos dinâmicos, como os atmosféricos, podem gerar resultados totalmente inesperados com variações mínimas nas condições iniciais. Este fenômeno ficou conhecido como *efeito borboleta*, em decorrência de um de seu mais célebres trabalhos, intitulado "*Predictability: Does the Flap of a Butterfly's Wings in Brazil Set Off a Tornado in Texas?*" (MIT-NEWS, 2008). Jule Charney já havia verificado um fato semelhante em 1951, devido a problemas com o modelo e as parametrizações. Mas, Lorenz asseverou que mesmo tendo um modelo perfeito esse limite ainda vai estar presente (KALNAY, 2003).

Segundo Palmer e Hagedorn (2006), os modelos de previsão são necessariamente incertos, já que são vários os tipos de incertezas inerentes aos modelos meteorológicos: erros de modelagem (processos com representação pobre, ou simplesmente não representados), incluindo a imprecisão do método numérico (erros numéricos: truncamento e arredondamento), bem como erros de medida das variáveis atmosféricas. Todas estas incertezas indicam que a previsibilidade da atmosfera tem limite.

Existem diversas metodologias e aproximações que se utilizam para mitigar esses erros com a finalidade de gerar previsões com boa acurácia, como por exemplo melhoras na assimilação de dados e resolução espaço-temporal mais fina, graças à contínua evolução dos computadores e das técnicas computacionais. Como técnicas avançadas de assimilação, pode-se citar o LETKF (*Local Ensemble Transform Kalman filter*) (KALNAY, 2003), 4D-var (*4 Dimensional Variational*) (KALNAY, 2003), redes neurais artificiais (HÄRTER; VELHO, 2008; CINTRA, 2010), entre outras.

Um dos métodos para quantificar a previsibilidade é a *previsão por conjuntos* (KALNAY, 2003), uma técnica de Monte Carlo, na qual são realizadas várias simulações (realizações). Um método distinto é conhecido como *Breeding*, que é a base deste trabalho. Os princípios da técnica de *Breeding* foram lançados no início dos anos 90 no NCEP (*National Centers for Environmental Prediction*). Esse método foi aplicado como processo de geração de perturbações para o modelo de previsão por conjuntos visando obter uma melhor amostragem dos erros (KALNAY, 2003).

A técnica de *Breeding* consiste em fazer duas simulações do mesmo modelo com condições iniciais levemente diferentes. Uma delas denomina-se *simulação de controle* e a outra *simulação perturbada*. Nesta última, perturbações são adicionadas às condições iniciais de controle e esta nova condição inicial é usada para uma nova simulação perturbada.

A partir das duas simulações citadas, é calculado um vetor com as diferenças entre as variáveis de estado do modelo numérico, denominado *bred vector*. Os *bred vectors* são gerados a intervalos de tempo fixos denominados *intervalo de breeding*. Neste trabalho, a previsibilidade quantificada pelo *bred vector* é comparada com métrica desvio padrão, obtida pelo método de previsão por conjuntos.

As magnitudes de *bred vectors* das variáveis do modelo de circulação global da atmosfera podem ser usadas para quantificar a previsibilidade do modelo, isto é, indicar em que regiões a previsão têm maior ou menor confiabilidade. A magnitude do *bred vector* também pode ser usada como variável de entrada para distintos classificadores.

Uma previsão correta e precisa é de fundamental importância em casos de eventos extremos, como inundações ou secas, o que ajuda a mitigar os efeitos negativos sobre a população e a economia. Por tal motivo, a Previsão Numérica de Tempo (PNT) ganhou relevância, fazendo que os países invistam grandes recursos em capacidade de computação. O CPTEC (Centro de Previsões de Tempo e Estudos Climáticos), uma divisão do INPE (Instituto Nacional de Pesquisas Espaciais), é um dos centros brasileiros de excelência em pesquisa nas áreas de modelagem atmosférica e previsão numérica de tempo. É também o órgão governamental com a missão de fornecer previsão de tempo de curto e médio prazo, bem como previsão climática sazonal.

O CPTEC-INPE adquiriu o supercomputador TUPÃ (Cray XE6) em 2010. À época de sua aquisição, este computador foi a segunda máquina no mundo em poder de cálculo dedicada à previsão meteorológica. A infra-estrutura computacional tem servido à previsão de curto e médio prazo, previsão climática sazonal, previsão de ondas do mar, pesquisas em circulação oceânica, modelos acoplados oceano-atmosfera, previsão ambiental – o CPTEC-INPE foi o primeiro centro no mundo a realizar esta atividade como uma ação operacional –, bem como os grupos de pesquisa em mudanças climáticas.

Como resultado de anos de pesquisas, é hoje o CPTEC-INPE quem desenvolve e mantém o Modelo Atmosférico Brasileiro (BAM: *Brazilian Atmospheric Model*), constituindo-se em um pilar de operação para previsões operacionais de tempo e previsão climática sazonal com alta confiabilidade. O desenvolvimento do BAM compreende o uso de novas técnicas/algoritmos computacionais e de modelagem, que permitam gerar previsões de qualidade com até 5 dias de antecedência (INPE, 2015).

1.1 Objetivos

Este trabalho tem dois objetivos:

- a) aplicar a técnica de *Breeding* como metodologia alternativa para avaliação de previsibilidade para o modelo Global do CPTEC-INPE,
- b) utilizar algoritmos de Inteligência Computacional para se obter um classificador a partir dos dados disponíveis, com o intuito de avaliar a qualidade da previsão de tempo gerada pelo modelo de circulação global da atmosfera (AGCM) do CPTEC-INPE.

Em relação ao item-(b), neste trabalho testamos duas metodologias de aprendizado de Inteligência Computacional, visando obter uma máxima acurácia na classificação: redes neurais artificiais e sistemas neuro-difusos.

1.2 Motivação

Na prática operacional atual do CPTEC-INPE são utilizados 15 simulações na previsão por conjunto, um número relativamente pequeno, se comparado a outras instituições similares no mundo. Como se trata de um sistema do tipo Monte Carlo, o que se espera é que este número aumente caso se queira melhorar a previsão do CPTEC-INPE.

A principal motivação deste projeto de mestrado reside no fato que mesmo com um número pequeno de membros, o método previsão por conjunto têm um alto custo computacional. É interessante pois buscar alternativas para avaliar a previsão gerada pelos modelos do CPTEC-INPE, que tenham um custo menor mas que apresentem a mesma qualidade, ou superior, ao método de previsão por conjunto. Neste contexto, é importante também gerar métricas que permitam avaliar a qualidade de métodos alternativos.

Diferentemente do sistema de previsão por conjunto, a metodologia por *Breeding* tem baixo custo computacional, já que necessita somente de duas simulações: previsão de referência (ou controle) e a previsão com condição inicial perturbada. A técnica de *Breeding* tem potencial para ser uma ferramenta útil e de auxílio ao *Ensemble* por seu baixo custo, que permita também avaliar a previsão gerada pelo modelo numérico do CPTEC-INPE no futuro.

1.3 Organização da dissertação

Os capítulos restantes desta dissertação estão organizados como segue:

- Capítulo 2: São descritos os principais fundamentos teóricos da previsão numérica do tempo necessários para o entendimento do trabalho. Também se descreve brevemente os modelos global e a metodologia de previsão por conjuntos.
- Capítulo 3: Apresenta-se a técnica de *breeding*, os fundamentos teóricos-práticos de como é aplicado ao modelo Global do CPTEC-INPE.
- Capítulo 4: É apresentada uma introdução das técnicas de inteligência computacional para gerar os classificadores: as redes neurais, os classificadores neuro-difusos, o classificador hierárquico binário e as máquinas de comitê.
- Capítulo 5: O capítulo mostra os resultados obtidos pelos distintos testes de aplicação de *breeding* ao modelo Global do CPTEC-INPE.
- Capítulo 6: O capítulo mostra os resultados obtidos pelos distintos testes de classificação obtidos.
- Capítulo 7: Apresenta as conclusões da dissertação e discute trabalhos futuros.

2 FUNDAMENTOS TEÓRICOS DA PREVISÃO NUMÉRICA DE TEMPO

A Previsão Numérica de Tempo (PNT) é um método que, a partir do conhecimento atual da atmosfera, procura predizer seu estado futuro, utilizando modelos físicos. Basicamente é um problema de valores inicial e de contorno de Equações Diferenciais Parciais (EDP) não lineares.

As equações que governam a dinâmica da atmosfera, assim como as leis da física que interagem entre os distintos processos envolvidos, são resolvidas por métodos numéricos, uma vez que estas equações não têm soluções analíticas. Os métodos aproximados (numéricos) requerem uma quantidade alta de cálculos. Nesse contexto, é necessário o uso de supercomputadores para obter uma solução com acurácia aceitável.

2.1 Equações que governam a dinâmica da atmosfera

Em 1904, V. Bjerkness inicialmente estabeleceu um conjunto de 7 equações com 7 variáveis como modelo para o estudo da evolução da atmosfera (KALNAY, 2003). Estas equações, conhecidas como *Equações Primitivas* (2.1–2.7), são baseadas nas leis de Newton de conservação de quantidade de movimento, nas equações de conservação de massa e energia, na equação dos gases ideais, nas leis da termodinâmica, que inclui transferência radiativa da energia do Sol através da atmosfera e das nuvens, e na equação de conservação de massa d'água.

O conjunto contém as equações da quantidade de movimento para uma terra esférica, as equações de energia termodinâmica, a equação de continuidade de massa— que estabelece o balanço de massa —, a equação de balanço de vapor d'água e a equação de gases ideais. O modelo utiliza as seguintes variáveis: u , v e w são as três componentes da velocidade do vento, p é a pressão, ρ é a densidade do ar, T é a temperatura absoluta, q_v é a umidade específica, Ω é a velocidade de rotação da terra, ϕ é a latitude, λ é a longitude, a é o raio da Terra, γ é a taxa de lapso da temperatura (gradiente vertical), γ_d é a taxa adiabática seca, c_p é o calor específico do ar a pressão constante, g é a aceleração da gravidade, H é o ganho ou perda de calor, Q_v é o ganho ou perda de calor do vapor d'água entre trocas de fases, e F_r é a fricção genérica em cada direção.

As equações em coordenadas esféricas são expressas por:

$$\frac{\partial u}{\partial t} = -\frac{u}{a \cos \phi} \frac{\partial u}{\partial \lambda} - \frac{v}{a} \frac{\partial u}{\partial \phi} - w \frac{\partial u}{\partial z} - \frac{uv \tan \phi}{a} - \frac{uw}{a} - \frac{1}{\rho a \cos \phi} \frac{\partial p}{\partial \lambda} - 2\Omega(w \cos \phi - v \sin \phi) + F_{r_x} \quad (2.1)$$

$$\frac{\partial v}{\partial t} = -\frac{u}{a \cos \phi} \frac{\partial v}{\partial \lambda} - \frac{v}{a} \frac{\partial v}{\partial \phi} - w \frac{\partial v}{\partial z} - \frac{u^2 \tan \phi}{a} - \frac{uv}{a} - \frac{1}{\rho a} \frac{\partial p}{\partial \phi} - 2\Omega u \sin \phi + F_{r_y} \quad (2.2)$$

$$\frac{\partial w}{\partial t} = -\frac{u}{a \cos \phi} \frac{\partial w}{\partial \lambda} - \frac{v}{a} \frac{\partial w}{\partial \phi} - w \frac{\partial w}{\partial z} - \frac{u^2 + v^2}{a} - \frac{1}{\rho} \frac{\partial p}{\partial z} + 2\omega u \cos \phi - g + F_{r_z} \quad (2.3)$$

$$\frac{\partial T}{\partial t} = -\frac{u}{a \cos \phi} \frac{\partial T}{\partial \lambda} - \frac{v}{a} \frac{\partial T}{\partial \phi} + (\gamma - \gamma_d)w + \frac{1}{c_p} \frac{dH}{dt} \quad (2.4)$$

$$\frac{\partial \rho}{\partial t} = -\frac{1}{a \cos \phi} \frac{\partial(u\rho)}{\partial \lambda} + \frac{1}{a \cos \phi} \frac{\partial(v\rho \cos \phi)}{\partial \phi} + \frac{\partial(w\rho)}{\partial z} \quad (2.5)$$

$$\frac{\partial q_v}{\partial t} = -\frac{u}{a \cos \phi} \frac{\partial q_v}{\partial \lambda} - \frac{v}{a} \frac{\partial q_v}{\partial \phi} - w \frac{\partial q_v}{\partial z} + Q_v \quad (2.6)$$

$$p = \rho RT \quad (2.7)$$

Em 1922, L. Richardson resolveu as equações por processos numéricos. Porém, os dados não estavam balanceados, um problema conhecido como de *inicialização*, e não havia conhecimento suficiente de *análise numérica* para se estabelecer um esquema numérico consistente e estável. Hoje em dia, a análise numérica é uma técnica bem estabelecida, e existem metodologias adequadas de inicialização e assimilação de dados.

Existem uma série de aproximações às equações primitivas¹ que tornam o problema mais tratável. Há também outros fenômenos associados como os modelos acoplados, que interagem com a atmosfera, como os processos da circulação oceânica e interação (*modelos*) atmosfera-superfície.

Em PNT, geralmente são utilizados dois tipos de modelos, os *Globais* e os *Regionais*, com propósitos diferentes. Nos modelos globais, as equações são resolvidas para o planeta inteiro e têm menor resolução espacial. Modelos regionais, ou de *área limitada*, tem resolução espacial mais fina e as condições de contorno são obtidas dos modelos globais. Uma tendência atual é a investigação de modelos únicos, com malhas não estruturadas, onde a malha é mais refinada na região de interesse – removendo a necessidade do uso de um modelo regional em separado.

2.2 Métodos Numéricos

Como menciona Kalnay (2003), um problema bem-posto (*well-posed problem*) de condições iniciais e de fronteira tem solução única que depende continuamente dos dados. Como o sistema de equações primitivas é um conjunto de Equações Dife-

¹ Dois exemplos: aproximação hidrostática e o ajuste geostrófico.

enciais Parciais (EDP), é essencial que as condições iniciais e da fronteira sejam apropriadas para produzir uma previsão de qualidade aceitável.

Existem diversos métodos para calcular de forma aproximada uma solução para uma EDP (diferenças finitas, elementos finitos, método espectral, etc). As discretizações numéricas geram erros, denominados erros de truncamento. Em PNT, primeiramente foram usadas as diferenças finitas, que transformam as EDP em espaço contínuo em Equações de Diferenças Finitas (EDF) em espaço discreto. Esta discretização é feita tanto no domínio temporal como o domínio espacial. Por exemplo, no domínio espacial, o problema assume valores discretos $x_j = j\Delta x$ que dependem do tamanho de passo Δx . O mesmo conceito se aplica aos demais eixos y e z , assim como o domínio temporal. O modelo global do CPTEC-INPE utiliza o método espectral para a discretização das equações.

A *condição de consistência* exige que, se as discretizações espaciais e temporal tendem para zero, as EDF devem gerar as EDP originais. Outro elemento importante é a estabilidade do método, isto é, se a EDP é limitada ($|\Phi(\vec{r}, t)| < \infty$) então a solução numérica também deve ser limitada quando $t \rightarrow \infty$. Um método numérico consistente e estável reúne requisitos para que a solução numérica possa *convergir* para a solução verdadeira das EDP. Para integração no tempo com métodos explícitos, uma condição para estabilidade é conhecida como critério CFL². A condição de estabilidade visa a escolha correta dos valores de passos de tempo, a fim de obter uma solução computacionalmente estável.

O uso de discretizações no espaço introduz erros de fase, que geram inconsistências para alguns fenômenos da atmosfera. O método espectral, onde a dependência espacial é assumida ser conhecida, reduz este tipo de problema numérico. De forma genérica, o método espectral faz uso da separação de variáveis: $U(\vec{r}, t) = \sum_{k=1}^m A_k(t)\psi_k(\vec{r})$, onde as funções base são assumidas conhecidas. Este método leva a um conjunto de equações diferenciais parciais em equações diferenciais ordinárias, para calcular os coeficientes da expansão que dependem do tempo. A projeção da estrutura horizontal sobre a(s) forçante(s) do sistema (*transformada de Hough*) leva a um problema de auto-valor que determina a representação da função vertical do modelo. A representação espectral exige que novas condições sejam satisfeitas, como por exemplo a quantidade k de números de ondas a utilizar (KALNAY, 2003).

Os modelos globais utilizam funções harmônicas esféricas como funções base, que são

²CFL:Courant–Friedrichs–Lewy em honra aos matemáticos que formularam esta condição.

as autofunções da equação de Laplace para as coordenadas de longitude e utilizam polinômios de Legendre para as condições de latitude. O modelo global do CPTEC-INPE usa um método semi-lagrangiano e semi-implícito, que permite utilizar um esquema de tempo 3 vezes maior que um método explícito (KUBOTA, 2012) como o método de Euler ou *Leap-frog*.

2.3 Parametrizações

A discretização das equações primitivas limitam a resolução do modelo; dependendo dos esquemas numéricos utilizados, sempre existirá uma escala de resolução. Os diferentes modelos climáticos associam essas resoluções como valores em distância-tempo. Para modelos globais, temos resoluções horizontais como distância de dezenas a centenas de quilômetros e uma distribuição vertical dependendo da quantidade de níveis definidos (KALNAY, 2003).

O poder computacional vem crescendo nos últimos anos, tornando factíveis soluções com alta resolução. Mas, ainda existirão fenômenos que serão mal representados nas escalas escolhidas por terem dinâmicas e tempos de ocorrência diferentes. Entre estes fenômenos podemos mencionar a turbulência, evaporação/condensação, entre outros, que podem ter escalas até nível molecular e/ou de curta duração, que afetam a dinâmica global de algum modo. Os processos e as considerações tomadas para não ignorar estes fenômenos são denominados parametrizações (KALNAY, 2003; WARNER, 2011).

2.4 Assimilação de dados

Assimilação de dados é um conjunto de procedimentos desenvolvidos com a finalidade de obter a representação dos estados da atmosfera de forma mais acurada possível, para que sirvam como condições iniciais e de fronteira às equações primitivas que governam a dinâmica climática.

Esta extrema dependência é vital a fim de que a solução (previsão) seja a mais representativa possível. Atualmente, os centros de operações e pesquisas utilizam uma combinação estatística de métodos com observações e previsões anteriores para produzir o que se denomina *análise*, um conjunto de dados que servirão como entrada para o modelo, como condição inicial.

2.5 Modelo AGCM-CPTEC

O modelo global AGCM (*Atmospheric Global Circulation Model*) do CPTEC-INPE teve sua origem no modelo de previsão de médio prazo MRF (*Medium Range Forecasting Model*). O MRF é uma combinação do código espectral desenvolvidos pelo NCEP/USA com as parametrizações físicas do *Geophysical Fluid Dynamics Laboratory* (GFDL) da Universidade de Princeton-USA . Esse modelo foi transferido ao *Center for Ocean, Land and Atmosphere Studies* (COLA/USA) onde foram adicionados cálculos diagnósticos para melhorar o entendimento físico dos processos simulados (BONATTI, 1996). Posteriormente, o CPTEC-INPE adota a versão 1.7 do COLA no ano de 1994, realiza mudanças significativas em relação ao antecessor como por exemplo a mudança de um truncamento romboidal a truncamento triangular, resoluções horizontais e verticais versáteis, melhoras no sistema de geração de arquivos de entrada-saída, entre outras, chamando assim a este novo modelo como CPTEC-COLA 1.0 (BONATTI, 1996).

Anos depois, o CPTEC realizou novas mudanças seguindo recomendações do COLA, e uma nova versão foi colocada em operação no ano de 1998. Entre as grandes mudanças encontram-se novas parametrizações, melhoras na interação atmosfera-biosfera, melhoras no pós-processamento para evitar incompatibilidades entre campos de vento na proximidade de topografia alta, e a introdução de uma malha gaussiana na horizontal (GLOBAL-CPTEC, 2013).

Operacionalmente, a resolução do modelo global utilizado na metodologia *Ensemble* é de T126L28, onde T refere-se ao tipo de truncamento espectral utilizado, denominado triangular, nas ondas zonais 126, e L refere-se ao número de camadas na vertical, neste caso, 28. A resolução horizontal T126 equivale a uma grade aproximada de 100x100 Km. (GLOBAL-CPTEC, 2013).

Os princípios físicos e equações (primitivas) que governam o modelo AGCM são:

- Leis de conservação de massa, de umidade, de energia e de momento angular.
- Equações de continuidade de massa para o ar seco e vapor d'água,
- Primeira lei da termodinâmica,
- Equações de movimento (segunda lei de Newton),

- Equações de movimento horizontal, transformadas nas equações da vorticidade e da divergência, facilitando tratamento numérico.
- Forças verticais consideradas em aproximado balanço hidrostático.
- Parametrizações como gelo, superfície do solo, radiação, entre outros.

O modelo é global em extensão e conforme a um sistema com geometria esférica, onde a altitude é suficientemente pequena para que a distância ao centro da Terra seja aproximadamente constante (BONATTI, 1996). A Figura 2.1 mostra os componentes principais do modelo, assim como as parametrizações implementadas, como exemplo os módulos de química, gelo marinho, de radiação, entre outros (KUBOTA, 2012).

As variáveis prognósticas do MCGA são: o logaritmo da pressão à superfície, a vorticidade (componente vertical do rotacional do vento), a divergência do vento horizontal, a temperatura virtual e a umidade específica. Além dessas, existem outras que são previstas nas parametrizações incluídas no modelo, tais como: temperatura de superfície, do interior do solo e do interior do dossel, umidade do solo, entre outras (BONATTI, 1996).

Como o modelo numérico é altamente não linear, é necessário discretizar as quatro dimensões. A integração do tempo é feita de forma separada com um tamanho de passo fixo e regular. A diferenciação é feita utilizando duas metodologias, métodos semi-implícitos para as equações de continuidade e as de divergência, enquanto que outras equações, como a conservação de umidade e as as equações da vorticidade, são utilizados métodos explícitos. A razão de se usar estratégias distintas é devido ao fato de que algumas variáveis como a vorticidade, por exemplo, são as responsáveis pela geração de ondas de alta frequência (BONATTI, 1996) e, portanto, requerem um passo de tempo menor – se utilizado um método explícito.

Nas duas dimensões horizontais, as variações são representadas por coeficientes harmônicos esféricos sobre uma base de funções que formam um conjunto ortonormal completo. A representação citada é adequada para movimentos de fluidos em coordenadas esféricas e permite o cálculo analítico das derivadas nas direções horizontais. As derivadas verticais são calculadas por diferenças finitas com espaçamento irregular sobre a coordenada de pressão normalizada: pressão do nível dividida pela pressão à superfície no ponto considerado, chamada coordenada σ (BONATTI, 1996). A coordenada σ tem a vantagem de permitir a introdução da topografia muito facilmente, pois σ tem valor constante igual a 1 acompanhando a superfície da Terra.

O espaçamento vertical é irregular pois é necessária uma resolução maior próximo à superfície, devido aos processos de camada limite e transferência de fluxos de energia entre solo e ar; também é necessária resolução maior na região de interface entre troposfera e estratosfera devido a variações verticais rápidas principalmente no campo de temperatura (BONATTI, 1996).

Figura 2.1 - Estrutura Esquemática do Modelo de Circulação Geral da Atmosfera do CPTEC.



Os módulos representativos dos processos de interação superfície-atmosfera no modelo AGCM, podem ser escolhidos pelo usuário mais facilmente, ativando o módulo por meio de chaveamento em uma nova compilação.

Fonte: Kubota (2012)

O modelo BAM (*Brazilian Atmosphere Model*, ficou em modo experimental durante um ano no CPTEC-INPE, três meses em pré-operação e, desde janeiro de 2016, é usado para previsão de tempo operacional no CPTEC-INPE, substituindo o modelo Global ou MCGA (Modelo Global de Circulação Atmosférica) anterior. As resoluções horizontais permitem configurações desde 200 km até 10 km (FIGUEROA et al., 2016) em truncamento triangular e 96 camadas na vertical. O BAM é executado para previsão de sete dias nos horários 0000 e 1200 UTC. Futuramente, será necessária execução nos horários intermediários, 0600 e 1800 UTC, porém com previsões de nove horas, para produzir um ciclo de assimilação de seis horas. O BAM apresentou uma boa correlação de suas previsões em relação às observações com até cinco dias de

antecedência, enquanto o antigo modelo mostrou uma queda de qualidade a partir do segundo dia das previsões (INPE, 2015). O modelo BAM utiliza o sistema de previsão por conjuntos (*Ensemble* – ver seção 2.6).

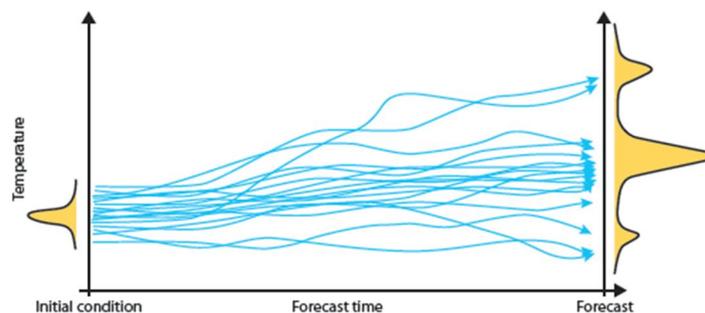
As condições iniciais do modelo são coeficientes espectrais GDAS (*Global Data Assimilation System*): ³ logaritmo da pressão à superfície, temperatura virtual, divergência horizontal, vorticidade vertical, umidade específica e topografia. As condições de contorno impõem uma restrição cinemática (conservação da massa), a velocidade vertical deve ser nula em todos os pontos do topo e na superfície do modelo (BONATTI, 1996). Mais informações de detalhes técnicos podem ser consultados em Bonatti (1996), Kubota (2012) e Figueroa et al. (2016).

2.6 Metodología do Sistema de Previsão por Conjuntos (*Ensemble*)

A técnica de previsão de tempo por conjunto (*Ensemble*, em inglês/francês) surgiu como uma tentativa de aumentar o prazo de previsão e avaliar a previsibilidade dos modelos dinâmicos. Modelos matemáticos são aproximações da natureza. Uma maneira de quantificar como se propagam as incertezas é gerar um *conjunto* (*Ensemble*) de condições iniciais. O conjunto de previsões é obtido integrando-se o modelo a partir de cada uma destas novas análises perturbadas (ENSEMBLE-CPTEC, 2013).

Com o conjunto de realizações/simulações, é possível calcular um *distribuição de probabilidade de máxima ocorrência da previsão, a partir de* uma distribuição inicial das várias condições iniciais como se visualiza na figura 2.2.

Figura 2.2 - Esquema de um sistema de previsão por Conjuntos ou metodologia *Ensemble*

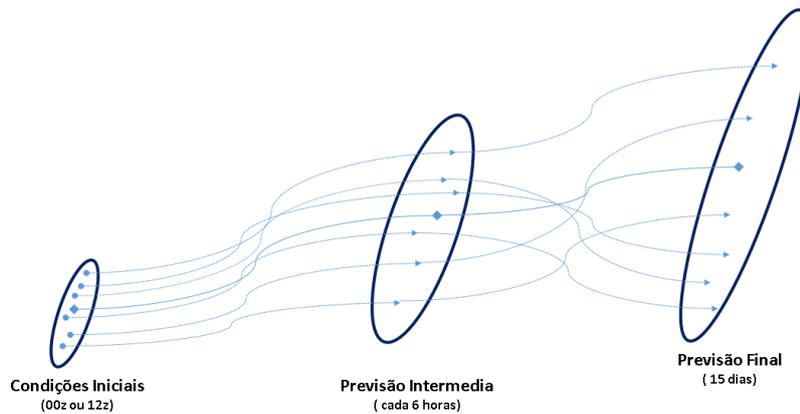


Fonte: ECMWF (2017)

³<https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-data-assimilation-system-gdas>

O SPCON (Sistema de Previsão por Conjuntos) do CPTEC implementa o método de *ensemble*, que mostrou ser uma ferramenta muito útil para a previsão operacional (MENDONÇA; BONATTI, 2002). Desde 2001, o CPTEC-INPE vem usando essa técnica operacionalmente. Atualmente, é utilizado um conjunto de 14 análises perturbadas e uma de controle (sem perturbação) na resolução T0126L028 (aproximadamente 100x100 km e 28 camadas verticais) e realizadas duas simulações por dia: 00 GMT e 12 GMT, respectivamente. Cada simulação ou membro do *Ensemble* é integrado para um período de 15 dias ou 360 hs, fornecendo saídas intermediárias a cada 6 hs, como se visualiza na figura 2.3.

Figura 2.3 - Esquema de um sistema de previsão por Conjuntos ou metodologia *Ensemble*



6

Fonte: Adaptado de Bustamante (2010)

As duas técnicas mais conhecidas de geração das condições iniciais perturbadas são o *Breeding of Growing Modes* empregado no *National Centers for Environmental Prediction* (NCEP), nos Estados Unidos, – ver capítulo 3 – e o *Singular Vectors* utilizado pelo *European Centre for Medium-Range Forecasts* (ECMWF), na Europa. Embora utilizem metodologias diferentes, ambas as técnicas buscam estimar os modos de crescimento mais rápido associados aos erros nas condições iniciais (MENDONÇA; BONATTI, 2002).

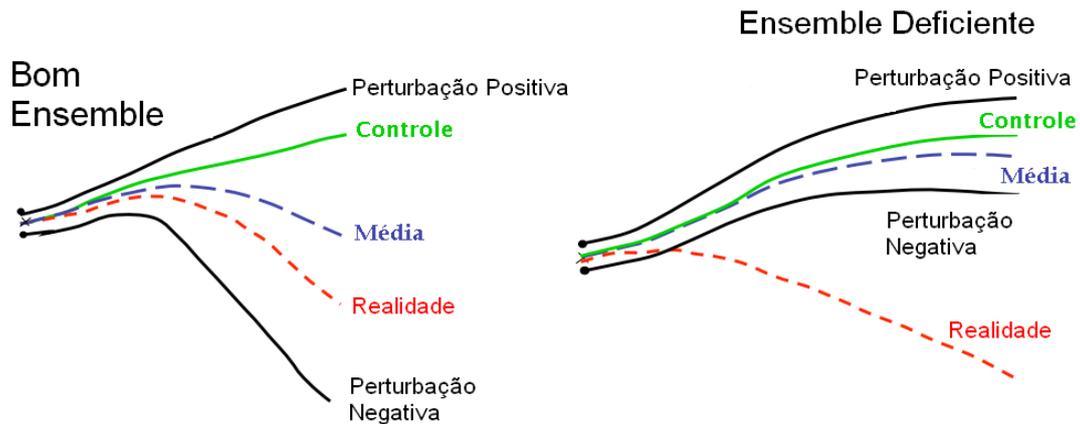
Conforme mencionado em Mendonça e Bonatti (2002), o mecanismo utilizado para a geração das condições iniciais perturbadas no CPTEC-INPE é baseado no algoritmo EOF (*Empirical Orthogonal Function*), utilizado inicialmente pelo modelo FSU (*Florida State University*) para previsão de furacões. O EOF consiste em:

- a) adicionar pequenas perturbações aleatórias, comparáveis aos erros de previsão, à análise de controle (não perturbada);
- b) integrar o modelo por 36 horas (intervalo ótimo) utilizando as análises perturbadas e de controle, com saídas em tempos fixos de 3 horas;
- c) subtrair a previsão de controle da previsão perturbada para cada horário de saída, obtendo uma série temporal de diferenças dos campos previstos;
- d) realizar uma análise de Funções Ortogonais Empíricas (EOF) para a série temporal de diferenças sobre uma região de interesse, obtendo os autovetores (modos) associados aos maiores autovalores, que são as perturbações ótimas;
- e) reescalonar as perturbações ótimas de forma que o desvio padrão dessas perturbações sejam da ordem das perturbações iniciais;
- f) construir um conjunto de estados iniciais, adicionando e subtraindo essas perturbações ótimas, após o reescalonamento, à análise de controle.

São várias as métricas que podem ser utilizadas para avaliar os resultados da previsão por *Ensemble*; parâmetros estatísticos, como média e espalhamento (desvio padrão ou variância), e métodos gráficos (gráficos espagete) são os mais utilizados nos principais centros de previsão.

A previsão é dita ser de *boa qualidade* quando um sistema de previsão por conjuntos envolve a evolução real das variáveis atmosféricas através das saídas de seus membros, e *deficiente* senão. Uma previsão deficiente indica que a evolução real da atmosfera se afasta das previsões obtidas dos membros. Esta dissidência é devido aos erros (tanto da condição inicial como do modelo) que crescem devido à natureza caótica do modelo e tornam-se significativos até prejudicar a previsão para a variável em questão. Ambos casos se visualizam na Figura 2.4.

Figura 2.4 - Esquema de previsão utilizando Ensemble.



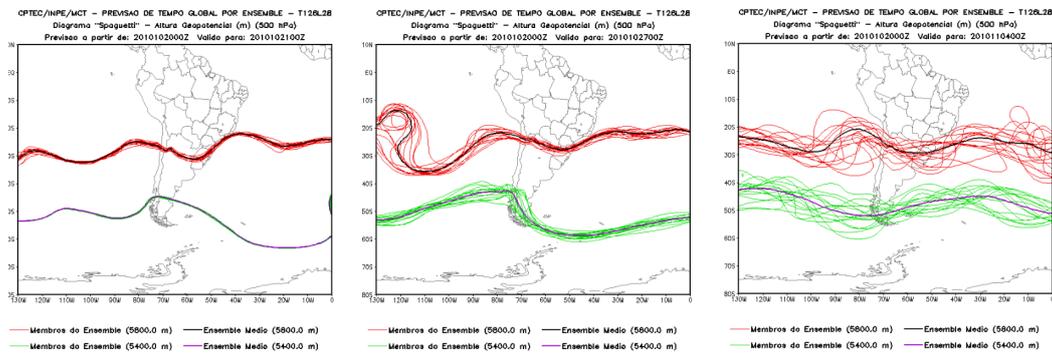
Fonte: Adaptado de Kalnay et al. (2007)

A quantidade de dados gerada pela previsão por *Ensemble* é muito grande, o que torna difícil a análise. Outra desvantagem é o custo computacional, já que várias simulações precisam ser feitas. Uma das técnicas de análise das informações fornecidas é a média do *Ensemble* e o desvio padrão ou espalhamento das previsões. O primeiro é a informação que é repassada ao público, e o segundo expressa uma medida da incerteza associada a cada variável (BUSTAMANTE, 2010).

A incerteza traduzida pelo desvio padrão entre os membros está em relação inversa com a previsibilidade; uma alta dispersão (variância) entre as previsões gera uma baixa *confiabilidade* da previsão.

Exemplos de discrepâncias entre as previsões para os distintos membros do *Ensemble* podem ser visualizados na Figura 2.5, que usa gráficos espaguete para mostrar a evolução da previsão da altura geopotencial (m) a 500 hPa com data de início em 20/Outubro/2010 às 00 Hs e data de finalização de 04/Nov/2010 às 00 Hs. Os gráficos spaghetti ilustram a evolução da incerteza inerente à simulação de previsão para um dada variável meteorológica, no caso, a altura geopotencial. Como o gráfico espaguete mostra o valor de uma variável em determinado nível para todos os membros do conjunto, tem-se um melhor panorama da propagação da incerteza na previsão.

Figura 2.5 - Previsão de 15 dias utilizando a metodologia *Ensemble* para a previsão da altura geopotencial (m).



Fonte: Bustamante (2010)

Utilizando o gráfico de spaghetti se visualizam as previsões para 1 dia, 7 dias e 14 dias correspondentes a uma simulação por *Ensemble* que inicia no dia 20 de outubro de 2010 as 00 Hs.

2.7 Sumário do capítulo

Neste capítulo foram apresentados os conceitos básicos da previsão numérica do tempo e uma breve descrição do modelo de previsão utilizado nesta dissertação. Um modelo de previsão é um modelo bastante complexo, que através do processo denominado assimilação de dados gera as condições iniciais para as variáveis de entrada do modelo. O modelo processa numericamente as equações com incertezas inerentes e como se trata de um modelo potencialmente caótico, estas incertezas diminuem a previsibilidade do sistema.

Metodologias como o sistema de previsão por conjuntos geram um conjunto de condições iniciais, que permitem avaliar a qualidade da previsão com o cálculo da média do ensemble, desvio padrão (espalhamento) e gráfico espaguete.

3 O MÉTODO DE *BREEDING*

Este capítulo aborda as características gerais do método de *Breeding*. A formulação apresentada foi desenvolvida pela Dra. Eugenia Kalnay e pelo Dr. Zoltan Toth no *National Centers for Environmental Prediction* (NCEP) no início dos anos 90. Foi aplicada em estudos de previsões por conjunto (*Ensemble Forecasting*) para gerar perturbações ótimas e agora também é utilizado para avaliar previsibilidades de sistemas dinâmicos caóticos (SANTOS, 2014)- (CINTRA; VELHO, 2008).

3.1 Descrição do modelo e geração dos *Bred Vectors*

A técnica de *Breeding*, proposta em (TOTH; KALNAY, 1997; KALNAY et al., 2007), consiste em fazer duas simulações do mesmo modelo. A primeira simulação, denominada *Simulação de Controle*, consiste na integração do modelo com as condições iniciais, por um intervalo de tempo definido. A segunda simulação, denominada *Simulação Perturbada*, consiste na integração a partir de condições iniciais com perturbações adicionadas, no mesmo intervalo de tempo. O processo de *Breeding* pode ser visualizado na Figura 3.1 e se resume nos seguintes passos:

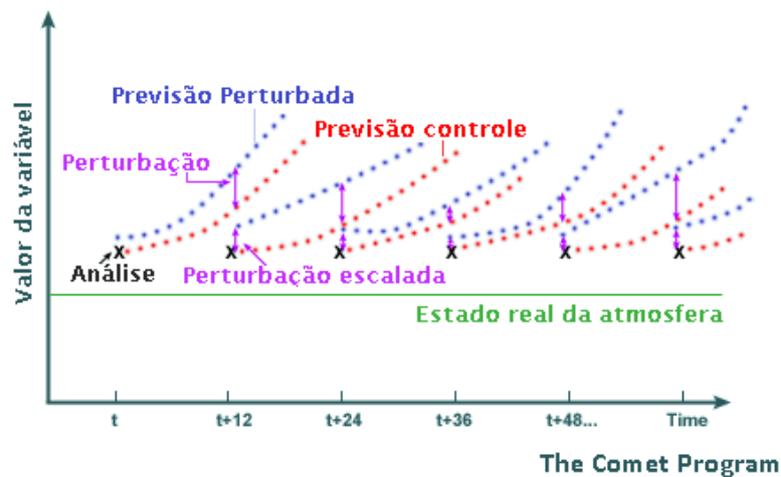
- (a) Adicionar uma perturbação pequena e arbitrária à análise do sistema (estado inicial), sendo t_0 o tempo inicial.
- (b) Integrar ambos modelos, o que se inicia com as condições de controle e o que se inicia com perturbação por um intervalo de período $(t_i - t_{i-1})$, denominado *Intervalo de Breeding*.
- (c) Subtrair os resultados obtidos entre os modelos, gerando os chamados *Bred Vectors*.
- (d) Reduzir os campos de diferenças das variáveis do *bred vector*, de modo que este tenha a mesma norma como perturbação inicial (normalização do *bred vector*).
- (e) Adicionar essa perturbação à próxima análise¹ (tempo t_i).
- (f) Repetir (b)–(e) nos tempos subsequentes.

As diferenças entre as variáveis de saída dos modelos podem divergir ou convergir, conforme o comportamento do sistema dinâmico em questão. Portanto, a magnitude

¹O campo de análise contém as variáveis iniciais para fazer a simulação.

do *bred vector* pode crescer ou diminuir em relação ao tamanho da perturbação inicial. A técnica de *Breeding* é cíclica e não depende da norma escolhida para avaliar as magnitudes dos vetores (TOTH; KALNAY, 1997). O tamanho do intervalo de *Breeding* e o tamanho da perturbação inicial são as únicas variáveis livres nesse método. No contexto de assimilação de dados, a perturbação inicial é feita na análise (combinação de observações e previsões anteriores) (CINTRA; VELHO, 2008).

Figura 3.1 - Esquema da técnica de *Breeding* e criação dos *bred vectors*



Fonte: Adaptado de The Comet Program.

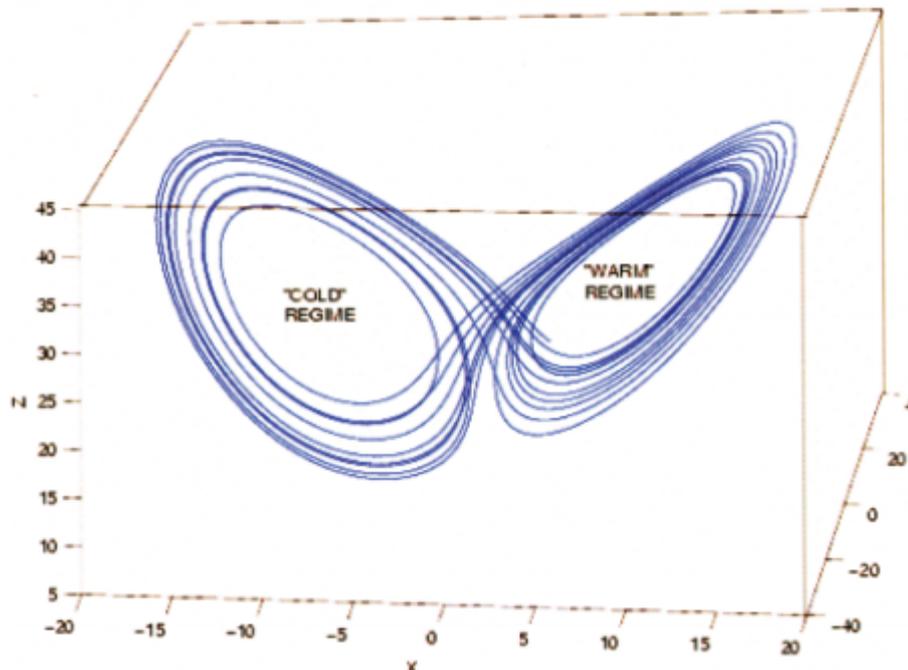
A taxa de crescimento do *bred vector* é uma medida de instabilidade do fluxo. Ela define a previsibilidade do sistema, já que a evolução dos erros iniciais determinam a dinâmica do sistema (CINTRA; VELHO, 2008). Abordagens para o estudo de estabilidade podem ser feitas por diferentes métodos, como *Vetores de Lyapunov*, *Exponentes de Lyapunov*, e o próprio *Breeding* (TOTH; KALNAY, 1997). Uma vantagem apresentada pelos *bred vectors* é o menor custo computacional. Além disso, conforme vários ciclos são executados, os resultados vão se aproximando principalmente aos vetores de Lyapunov (TOTH; KALNAY, 1997).

Evans et al. (2004) foram um dos pioneiros na aplicação de *bred vectors* em estudos de previsibilidade de sistemas caóticos. Nesse trabalho, foi demonstrada a possibilidade de prever a duração de troca regime do modelo do atrator estranho de Lorenz², utilizando regras simples baseadas na taxa de crescimento dos *bred vectors* com uma precisão de 90% para alguns testes. O sistema dinâmico de Lorenz é frequentemente

²Um atrator ou conjunto de atração é um conjunto de pontos que se aproximam uns a outros pelas trajetórias, depois de atingir o regime permanente (KALNAY, 2003).

citado em meteorologia como um *Toy Problem*, por ter 2 regimes que podem ser associados aos termos quente (*Hot*) e frio (*Cold*) (vide Figura 3.2).

Figura 3.2 - Simulação do sistema de Lorenz 3D



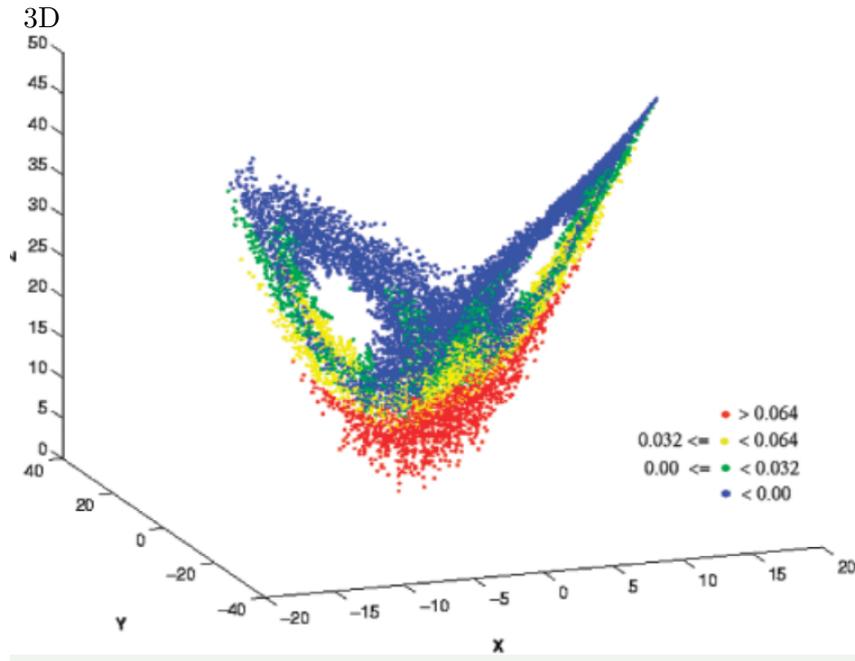
Fonte: Evans et al. (2004)

Em Evans et al. (2004) define-se a *Taxa de Crescimento de Bred Vector* como

$$g = \frac{1}{n} * \left(\frac{|\delta X|}{\delta X_0} \right),$$

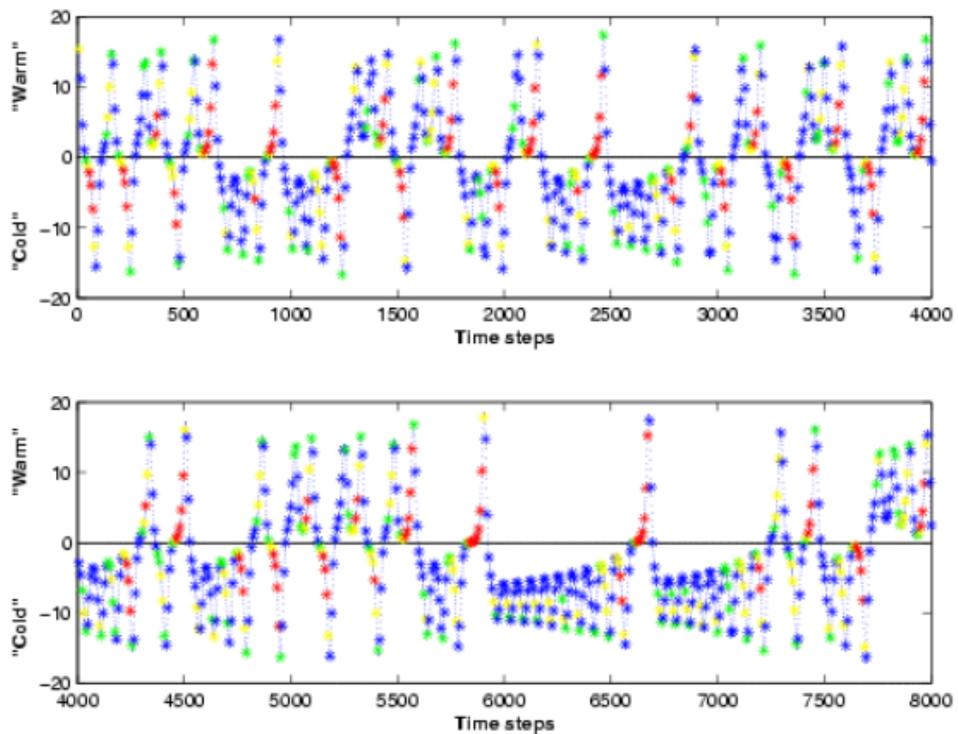
onde g depende de uma perturbação inicial $\delta X_0 = (\delta x_0, \delta y_0, \delta z_0)$ e de n , o intervalo de *Breeding*. Observando a evolução desta variável g , Evans et al. (2004) conseguiram gerar de forma empírica um conjunto de regras de previsibilidade para o sistema. A Figura 3.3 mostra as magnitudes da taxa de crescimento do *bred vector*, com $n = 8\Delta t$ como intervalo de *Breeding*. Nesta figura, adota-se um código de cores para as classes de tamanho da taxa de crescimento do *bred vector*; a cor vermelha indica que a região possui uma taxa g maior que 0.064 e a cor azul indica que a região tem uma taxa g negativa. As regiões com estrelas de cor vermelha e azul são regiões de alto crescimento e diminuição respectivamente, da magnitude do *bred vector*. Na figura 3.4, mostra-se a série temporal da variável $x(t)$ do modelo de Lorenz com as codificações das taxas de crescimento do *Bred Vector*.

Figura 3.3 - *Bred vectors* aplicados ao atrator estranho do sistema dinâmico de Lorenz em



Fonte: Evans et al. (2004)

Figura 3.4 - *Bred vectors* aplicados ao sistema de Lorenz para a variável $x(t)$



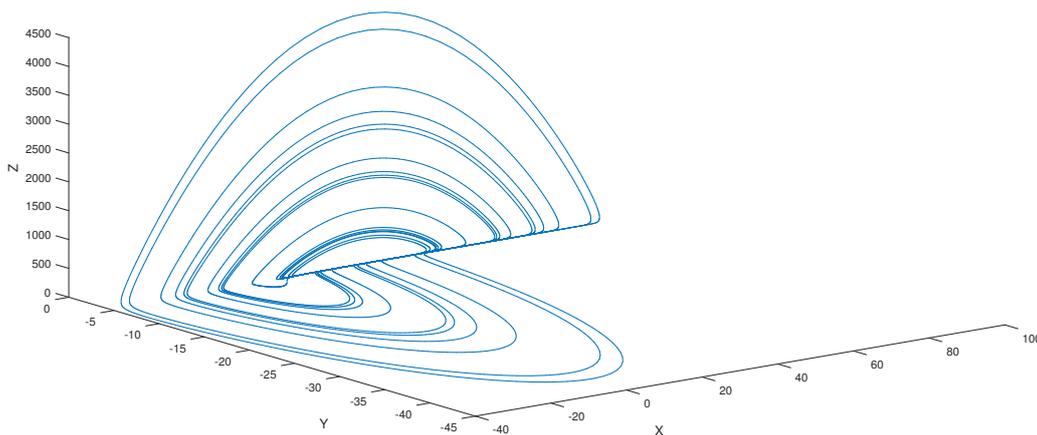
Fonte: Evans et al. (2004)

As regras geradas em Evans et al. (2004) são simples e se detalham à continuação:

- Quando a taxa de crescimento g for superior a 0,064 sobre um período de 8 passos, conforme indicado pela presença de uma ou mais estrelas vermelhas o atual regime terminará após completar a órbita atual.
- A duração do novo regime é proporcional ao número de estrelas vermelhas. Por exemplo, a presença de cinco ou mais estrelas no antigo regime, indicando um forte crescimento sustentado, implica que o novo regime vai durar quatro órbitas ou mais

Em Cintra e Velho (2008), além de estudar o modelo de Lorenz, também foi estudado o modelo de 3 ondas acoplado. Este modelo é de grande interesse em vários ramos da ciência, como fusão nuclear, geofísica, entre outros. Como no modelo de Lorenz, também possui 2 regimes mas, ao contrário do modelo de Lorenz, os regimes não são simétricos. Na Figura 3.5, mostra-se o atrator 3D do modelo. Um dos regimes do atrator consiste nas órbitas curvas formadas no plano XZ e segue no plano XY, enquanto o outro regime se localiza na interseção dos planos XZ e XY (SANTOS, 2014).

Figura 3.5 - Atrator do modelo de ondas 3D aplicados ao sistema dinâmico de 3 ondas acopladas

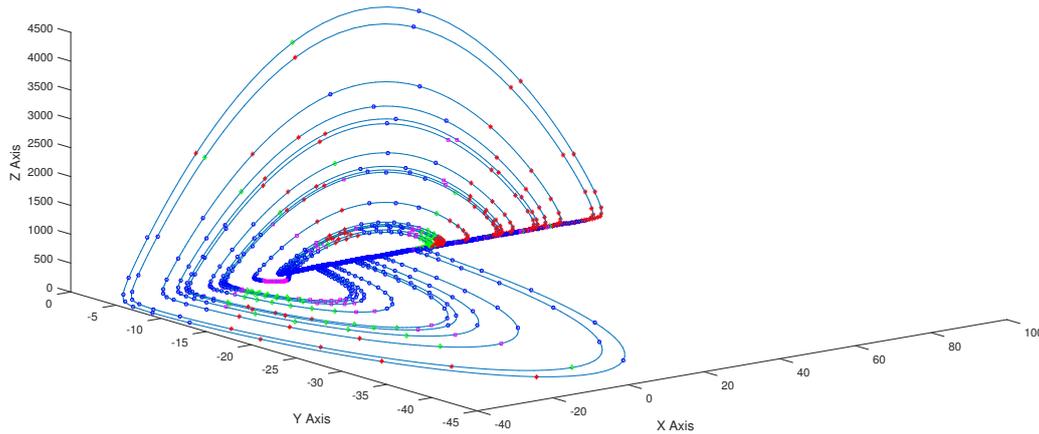


Fonte: Cintra e Velho (2008)

Aplicando *Breeding* ao modelo de 3 ondas, Cintra e Velho (2008) desenvolveram um sistema de regras de previsibilidade. Nas Figuras 3.6 e 3.7 são mostrados o modelo de 3 ondas com a codificação em cores da taxa de crescimento do *Bred Vector* e a série temporal de $x(t)$ respectivamente.

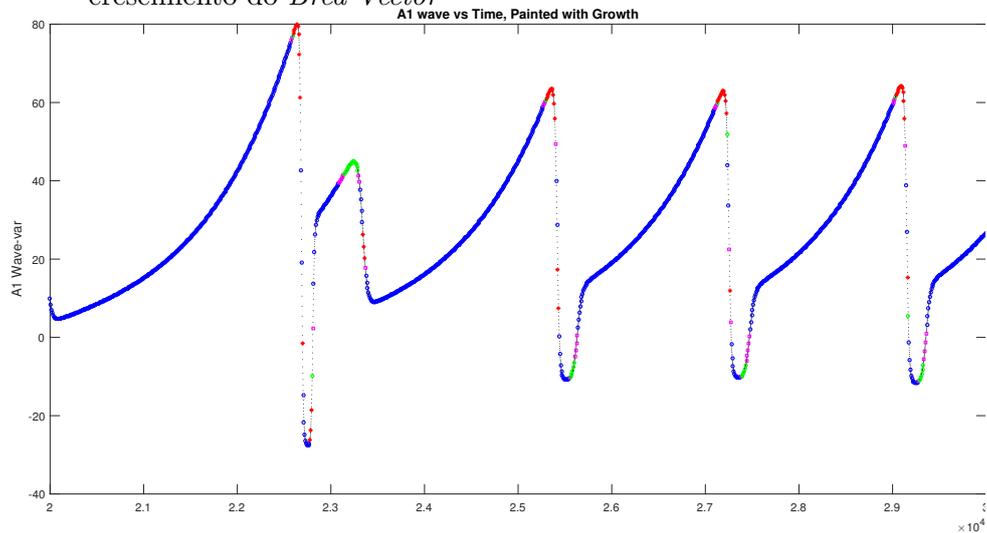
Em Cintra e Velho (2008), os limiares das taxas de crescimento g do modelo de 3 ondas varia em relação ao modelo de Lorenz; as estrelas vermelhas indicam uma taxa de crescimento $g > 0.035$, uma estrela verde indica uma taxa de $0.017 < g < 0.035$ e uma estrela azul indica uma taxa $g < 0.002$.

Figura 3.6 - Atrator do modelo de ondas 3D com a codificação da taxa de crescimento do *Bred Vector*



Fonte: Cintra e Velho (2008)

Figura 3.7 - Serie temporal $x(t)$ do modelo de ondas 3D com a codificação da taxa de crescimento do *Bred Vector*



Fonte: Cintra e Velho (2008)

As regras de previsibilidade foram geradas para testar a transição de um regime a outro. Por exemplo, a aparição de 2 ou mais estrelas verdes indicam uma transição de regime e a aparição de 2 estrelas amarelas predizem um período curto de tempo para a transição de período (SANTOS, 2014).

Em Santos (2014), foram utilizados redes neurais e redes neuro-difusas para classificar o comportamento dos sistemas de Lorenz e de 3 ondas. Uma das vantagens de utilizar redes neuro-difusas é que além de testar as regras de previsão, foram adicionas outras regras de inferência do tipo SE-ENTÃO, que adiciona informação interpretada para o sistema. Em alguns testes de classificação, a rede neural mostrou melhor desempenho que utilizando redes neuro-difusas.

3.2 Sumário do capítulo

A metodologia de *Breeding* consiste em simular o mesmo modelo duas vezes, partindo de condições levemente diferentes. Uma simulação parte das condições iniciais de controle e a outra parte de condições levemente perturbadas. A cada intervalo fixo de tempo, as saídas são re-escaladas de forma que possuam perturbação de mesma ordem que a perturbação inicial. A cada intervalo de tempo, são calculados os *bred vectors*. A taxa de crescimento dos *bred vectors* foi usada para gerar regras de previsão em alguns sistemas dinâmicos.

Para este projeto, foi implementado a metodologia de *Breeding* para gerar os *bred vectors* e avaliar a previsibilidade do modelo global do CPTEC.

4 ALGORITMOS DE INTELIGÊNCIA COMPUTACIONAL UTILIZADOS

Neste capítulo se apresentam breves introduções aos três paradigmas de Inteligência Computacional (IC) utilizados neste trabalho: Redes Neurais Artificiais (RNA), Sistemas Difusos e Redes Neuro-Difusas. Também são apresentados dois outros classificadores, um binária hierárquica e um de *Máquina de Comitê*, ambos implementados com aplicação de RNAs.

4.1 Conceitos básicos de Redes Neurais Artificiais (RNAs)

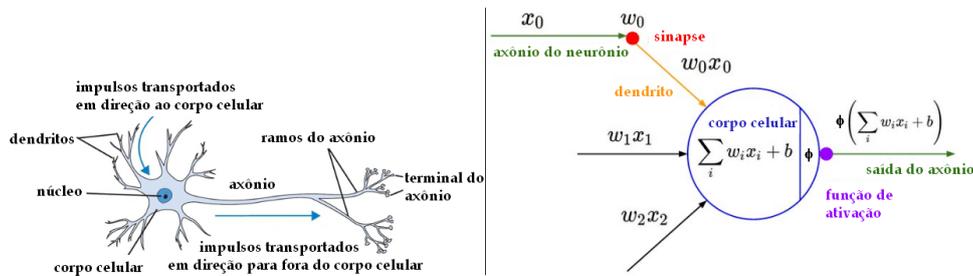
Uma das áreas de mais destaque da Inteligência Computacional é a de Redes Neurais Artificiais (RNA), com a qual é possível se automatizar uma série de processos de alto nível, como por exemplo agrupamento, classificação e reconhecimento de padrões e previsão a partir de séries temporais. O modelo matemático das RNAs é inspirado no comportamento do cérebro humano e em sua estrutura física (STERGIOU; SIGANOS, 1996).

Segundo Haykin (2009), o cérebro é um sistema altamente complexo, não linear e paralelo com a capacidade de adaptação e auto-organização das estruturas internas, conhecidas como neurônios. No contexto biológico, os neurônios exercem o papel da transmissão de informação na forma de impulsos elétricos. O lado esquerdo da Figura 4.1 ilustra a estrutura de um neurônio. Um neurônio biológico é composto por três estruturas principais: dendritos, corpo celular e axônio. Os dendritos são a estrutura de entrada do neurônio, responsáveis pela recepção dos impulsos elétricos. Esses impulsos elétricos são passados ao corpo celular que o processa e define um limiar de ação que permite a propagação do sinal elétrico através do axônio. Por sua vez, o axônio é a estrutura de saída do neurônio, que torna possível a propagação do impulso do sinal aos dendritos de outros neurônios em zonas ativas dos neurônios, chamadas de sinapses.

Os neurônios, interpretados como unidades de processamento, e suas interligações, são a chave do processo das RNAs, onde essas entidades individuais simples trabalham em uníssono na resolução de diferentes tipos de problemas complexos (STERGIOU; SIGANOS, 1996). O primeiro modelo de neurônio artificial, proposto por McCulloch e Pitts em 1943, e conhecido por MCP, consiste de três estruturas principais: as entradas, o somatório ponderado das entradas, e a função de ativação. O lado direito da Figura 4.1 ilustra essa estrutura: cada entrada x_i é multiplicada ao seu respectivo peso, w_i . Um peso é uma medida de ajuste aplicada a entrada para

adaptá-la ao propósito da rede. O somatório reúne todos os valores resultantes das operações realizadas nas entradas ponderadas ($x_i w_i + x_{i+1} w_{i+1} + \dots$) e repassa o valor resultante à função de ativação.

Figura 4.1 - Comparação entre o neurônio biológico e o neurônio artificial



Fonte: Adaptado de <http://cs231n.github.io/neural-networks-1/>

Mais formalmente, um neurônio artificial MCP pode ser implementado com as seguintes equações:

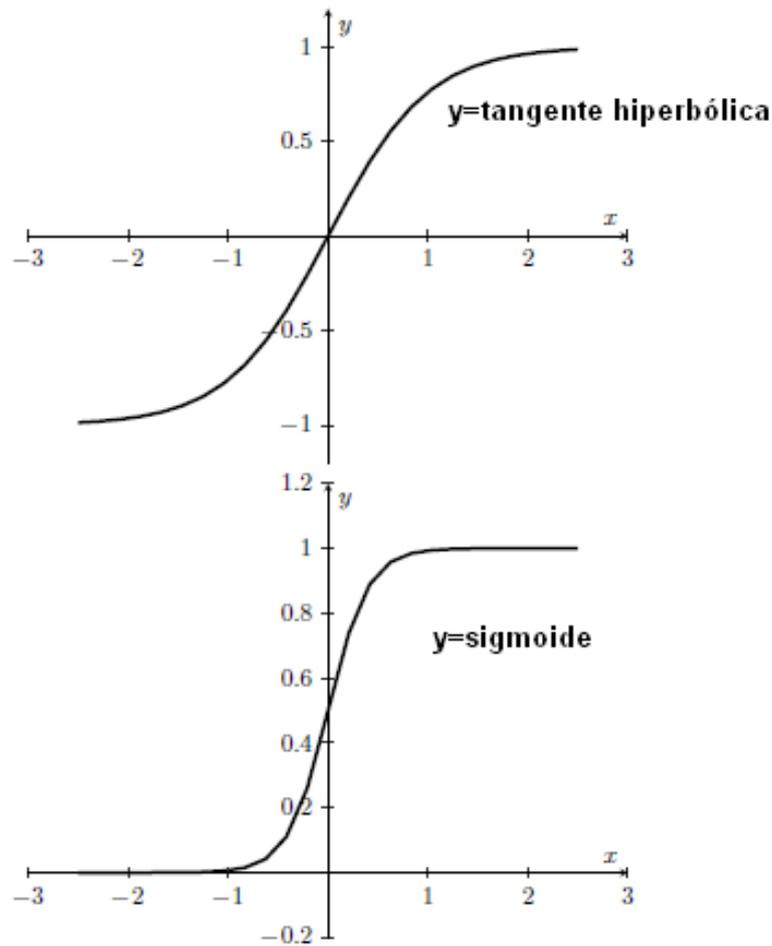
$$v_k = \sum_{j=1}^m w_{kj} x_j, \quad (4.1)$$

$$y_k = \phi(v_k + b_k), \quad (4.2)$$

onde x_1, x_2, \dots, x_m representam as variáveis de entrada, w_{kj} representa os respectivos pesos sinópticos. O valor obtido pela somatória ponderada é um dos argumentos da função de ativação $\phi(\cdot)$; o outro argumento b_k é denominado *bias* (viés) e permite calibrar o limiar de ativação da função $\phi(\cdot)$.

A função de ativação restringe a amplitude do sinal na saída do neurônio em função de seus argumentos. Na literatura, podem ser encontradas várias funções de ativação, mas não existe uma regra que indique que tipo de função de ativação utilizar para uma determinada tarefa. Na Figura 4.2 se visualiza duas funções mais utilizadas nas RNAs, a função tangente hiperbólica e a função sigmóide. Ambas funções são contínuas, o que é essencial para a aplicação do algoritmo de treinamento da rede, mas têm contradomínios diferentes.

Figura 4.2 - Funções de ativação



Fonte: Anochi (2015)

4.1.1 Arquitetura da rede

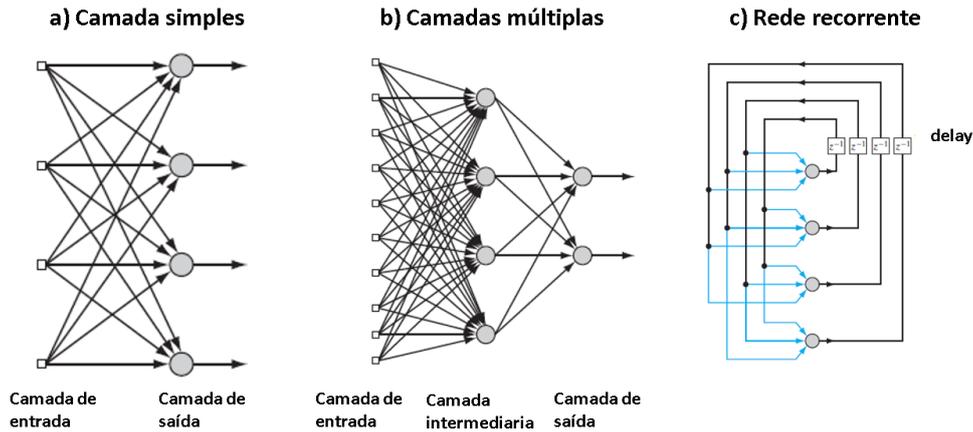
A arquitetura da rede consiste na forma em que os neurônios da rede neural estão estruturados e como estão interligados, e é essencial na resolução de um problema. A Figura 4.3 mostra os três principais tipos de arquiteturas definidos em Haykin (2009). Basicamente, uma rede neural possui uma camada de entrada e uma de saída, podendo ter também camadas intermediárias ou ocultas¹ (HAYKIN, 2009). A quantidade de neurônios na camada de entrada depende do tipo de problema.

Quanto mais camadas intermediárias a rede possui, maior é sua capacidade de extrair características de alta ordem das entradas apresentadas. Uma rede adquire uma perspectiva mais global devido às conexões extras entre os neurônios (HAYKIN, 2009).

¹As camadas ocultas são assim chamadas por seus neurônios não serem vistos nem pela entrada nem pela saída da rede.

A rede MLP (*MultiLayer Perceptron*), talvez a arquitetura mais empregada em aplicações, é um tipo de rede de várias camadas intermediárias é um tipo de rede com várias camadas intermediárias.

Figura 4.3 - Tipos principais de arquiteturas de RNAs



Fonte: Produção do autor, adaptado de Haykin (2009)

Tipicamente, as redes podem ter as saídas conectadas para as entradas dos neurônios das camadas seguintes (conhecidas como full-conectado). Em casos especializados, algumas ligações podem ser eliminadas e, dependendo do problema tratado, pode melhorar o desempenho da rede.

Outro tipo de arquitetura, conhecido como rede recorrente, possui uma retroalimentação das saídas para as suas entradas. A presença de ligações de retroalimentação ajudam no aprendizado e no desempenho da rede. Estas redes possuem elementos, chamados de *Time-delay*, que auxiliam o desempenho não linear da rede (HAYKIN, 2009). As redes recorrentes são muito utilizadas em processos de modelagem de séries temporais, reconhecimentos de padrões, entre outras áreas (ANOCHI, 2015).

4.1.2 Processo de aprendizagem

A principal característica da rede neural é sua capacidade de aprender através de exemplos (ANOCHI, 2015). Geralmente, o conjunto de dados é dividido em três subconjuntos: conjunto de treinamento, validação e teste. O primeiro consiste nos padrões fornecidos à rede para o ajustes dos seus parâmetros (aprendizagem), o segundo

conjunto tem a tarefa de verificar a capacidade de generalização da rede e o conjunto de teste tem a tarefa de testar o comportamento da rede neural com dados novos (ANOCHI, 2015).

O processo de aprendizagem se dá de forma iterativa, mediante mudanças dos valores dos pesos sinápticos e e do bias. As metodologias de treinamento se agrupam em duas grandes classes: treinamento supervisionado e treinamento não supervisionado (HAYKIN, 2009):

- Aprendizado supervisionado: neste método, a rede é treinada através de dados fornecidos por um especialista ou através de exemplos de (entrada, saída desejada). O objetivo é ajustar os parâmetros da rede, de forma a encontrar o melhor mapeamento entre as entradas e as saídas fornecidas, minimizando o erro produzido entre a saída desejada e a resposta da rede à entrada.
- Aprendizado não supervisionado: neste método, não é fornecida a saída desejada aos padrões de entrada. Os parâmetros são ajustados conforme a impulsos fornecidos pelo ambiente externo à rede e visam minimizar métricas estabelecidas previamente.

Neste trabalho, foi utilizado o algoritmo supervisionado de retro-propagação (*back-propagation*) para se treinar uma rede, mais especificamente o algoritmo de retro-propagação com taxa de aprendizagem adaptativo, com minimização pelo gradiente descendente (ANOCHI, 2015).

Segundo Haykin (2009), o algoritmo de retro-propagação é implementado em duas fases:

- a) Propagação (*forward*): consiste na ativação da rede; a resposta da rede é calculada utilizando pesos sinápticos fixos. Os sinais na camada de entrada são propagados camada por camada, até que o vetor de saída seja obtido na última camada.
- b) Retro-propagação (*backward*): os ajustes dos pesos sinápticos são calculados, visando minimizar a soma dos erros quadráticos $e_j(n)$ (o erro é a diferença entre a saída desejada e a resposta da rede). Este erro é retro-propagado da saída para cada elemento da rede (camada de entrada e intermediária). Este processo se repete e a partir desse sinal de erro; os

pesos sinápticos das conexões são ajustados para cada elemento, de modo a fazer a rede convergir para as saídas desejadas.

O sinal de erro $e_j(n)$ de cada neurônio j da camada de saída, para a iteração n , é dado por

$$e_j(n) = d_j(n) - y_j(n),$$

onde $d_j(n)$ é a saída desejada e $y_j(n)$ é a saída da rede. Com cada um desses erros $e_j(n)$, é calculado um parâmetro denominado valor instantâneo da energia total do erro ϵ_j , dado por

$$\epsilon(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n),$$

onde C é o conjunto de todos os neurônios da camada de saída da rede. Considerando-se os N padrões apresentados para a rede no treinamento, o erro quadrático médio é calculado utilizando-se as energias do erro para esse conjunto:

$$e_{med} = \frac{1}{N} \sum_{n=1}^N \epsilon(n).$$

Para um determinado conjunto de treinamento com N padrões de entrada-saída, e_{med} representa a função de custo como medida de desempenho de aprendizagem, cujo objetivo é minimizar esta função para ajustar os pesos sinápticos e bias da rede. (ANOCHI, 2015).

Tendo as saídas $y_i(n)$ retro-alimentadas na camada de entrada para os m neurônios, a somatória ponderada produz a variável $v_j(n)$, conhecida como campo local induzido $v_j(n)$, que é o argumento da função ativação $\phi(\cdot)$:

$$v_j(n) = \sum_{i=0}^m w_{ji}(n) y_i(n),$$

O algoritmo de retro-propagação aplica uma correção $\Delta w_{ji}(n)$ ao peso sináptico de cada neurônio, usando o gradiente da função custo com respeito ao peso w_{ji} para cada iteração n :

$$\Delta w_{ji} = -\eta \frac{\partial \epsilon(n)}{\partial w_{ji}(n)}.$$

A correção nos pesos é conhecida como regra *Delta* (Δ). Seu valor é negativo porque indica o sentido de descida do gradiente dentro do espaço de pesos, a fim de reduzir o valor de $\epsilon(n)$. O parâmetro η é conhecido como taxa de aprendizado do algoritmo de retro-propagação; quando possui um valor baixo o aprendizado é mais lento e o

processo não se torna instável (HAYKIN, 2009).

Adicionando um novo parâmetro α , é possível modificar a regra Delta de forma a evitar convergência para uma bacia de atração pouco efetiva. Este parâmetro é denominado taxa de momento (*momentum*):

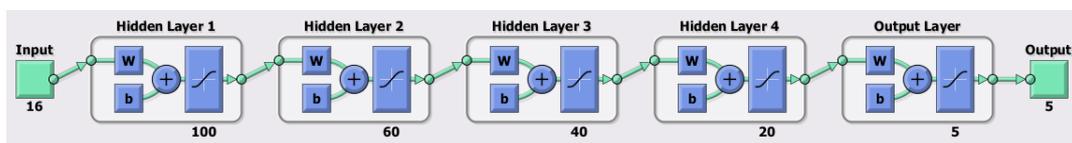
$$\Delta w_{ji} = \alpha \Delta w_{ji}(n-1) + \eta \delta_j(n) y_i(n).$$

Para que $\Delta w_{ji}(n)$ seja convergente, é sugerido que a constante α esteja no intervalo $0 \leq \alpha < 1$ (ANOCHI, 2015).

Neste projeto, foi utilizada a função *trainngdx* da plataforma Matlab, que combina o processo de retro-propagação com um treinamento adaptativo dos parâmetros η (taxa de aprendizagem) e α (taxa de *momentum*). Foram testados outros algoritmos de aprendizagem nativos de Matlab para treinamento de RNA mas os melhores resultados foram obtidos com esta função. Foi utilizado Matlab em estos testes iniciais devido a rápida implementação que oferece a plataforma e no futuro procura-se outras abordagens como *deep learning* e computação heterogênea.

A Figura 4.1.2 mostra a arquitetura da rede neural utilizada para os classificadores das Seções 6.2.2 e 6.3.2. Foi utilizada a estratégia *one-hot encoding* para codificar as classes de saída (1 neurônio por cada classe). A ativação de cada camada foi feita utilizando-se a função tangente hiperbólica e foi utilizado o valor 0,01 para as taxas de aprendizado e de momento. Nas seções 6.3.3 e 6.3.6, a estrutura interna da rede difere nas camadas de entrada e saída.

Figura 4.4 - Arquitetura da rede neural.



4.2 Conceitos básicos de Sistemas Difusos

Os sistemas difusos são aqueles baseados na teoria dos conjuntos difusos. Esta teoria, desenvolvida por Lotfi Zadeh (ZADEH, 1965), capta o aspecto vago da informação presente no discurso humano (SANDRI; CORREA, 1999).

Na teoria dos conjuntos clássica, dado um conjunto clássico A definido em um uni-

verso de discurso X , um elemento $x \in X$ pode ou não pertencer a um conjunto A . Esse caráter binário é representada matematicamente por uma função característica (TANSCHKEIT, 2004), que assume o valor 1 se x pertence a A e 0, em caso contrário:

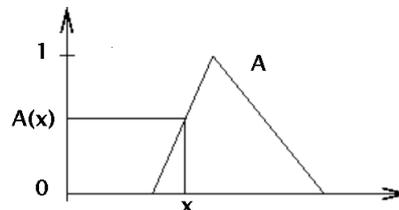
$$x_A(x) = \begin{cases} 1 & \text{se e somente se } x \in A \\ 0 & \text{se e somente se } x \notin A \end{cases}$$

Na teoria dos conjuntos difusos, dado um conjunto difuso A definido em X , a pertinência de um elemento $x \in X$ varia entre 0 e 1. Formalmente, um *Conjunto Difuso* A definido em um universo X é descrito através de uma *Função de Pertinência*² $A : X \rightarrow [0, 1]$, que mede a compatibilidade de x com o conjunto expresso por A .

$$A(x) = \begin{cases} 1, & x \text{ é completamente compatível com } A \\ 0, & x \text{ é completamente incompatível com } A \\ 0 < \alpha < 1, & x \text{ é compatível com } A \text{ com grau } \alpha \end{cases}$$

A Figura 4.5 ilustra um conjunto difuso, com função de pertinência triangular.

Figura 4.5 - Função de pertinência triangular



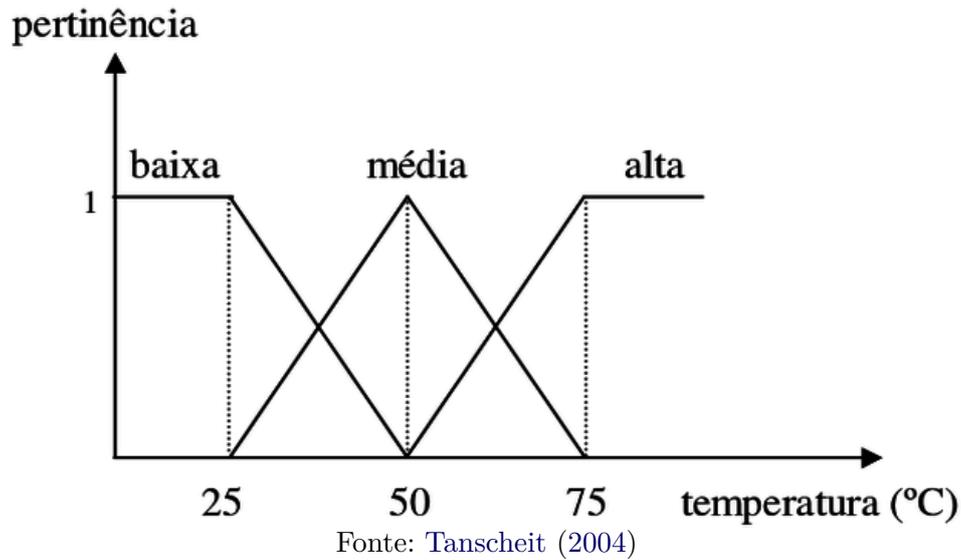
Fonte: Fispro (2003)

Um conjunto clássico é um caso particular de conjunto difuso, denominado nítido ou ordinário (*crisp* em inglês), assumindo valores exclusivamente em $\{0, 1\}$. O conjunto singular do caso clássico (*singleton* em inglês) é definido por uma função característica que assume o valor 1 para um elemento $x_0 \in X$ e 0 para todos os outros valores de X .

Uma *Variável Linguística* é uma variável à qual está associada a uma coleção de conjuntos difusos (uma *Partição Difusa*), chamados de *Termos Difusos*. Por exemplo, como mencionado em (TANSCHKEIT, 2004), a variável linguística *temperatura* pode estar associada à partição difusa $\{baixa, média, alta\}$ (vide Figura 4.6).

²A notação mais usual para a função de pertinência de A é μ_A . Seguindo a literatura mais recente, aqui adota-se o mesmo nome para um conjunto e sua função de pertinência associada.

Figura 4.6 - Termos difusos (*baixa*, *média*, *alta*) para a variável linguística *temperatura*



As funções de pertinência podem ser definidas a partir da experiência do usuário, ou aprendidas a partir dos dados de um problema. As formas mais usuais para modelar estas funções são gaussiana, triangular e trapezoidal.

4.2.1 Operações com conjuntos difusos

Sejam dois conjuntos difusos A e B definidos em X ; a interseção $A \cap B$ (respec. a união $A \cup B$) destes conjuntos é implementada por uma família de operadores denominada T -Normas (respec. T -Conormas) (SANDRI; CORREA, 1999).

Uma t -norma T e uma t -conormas S , são operadores $T, S: [0, 1]^2 \rightarrow [0, 1]$ comutativos, associativos e monotônicos, com elementos neutros 1 e 0, respectivamente.

O complemento de um conjunto difuso A , denotado por \bar{A} , é obtido pela aplicação de um operador de negação $n: [0, 1] \rightarrow [0, 1]$, monotônico, involutivo e tal que $n(0) = 1$ e $n(1) = 0$. A Tabela 4.1 traz as principais t -normas e t -conormas, com dualidade segundo as leis de De Morgan, com o operador de negação $n(x) = 1 - x$.

Tabela 4.1 - Principais T-normas e T-Conormas duais

t-norma	t-conorma	Nome
$\min(a,b)$	$\max(a,b)$	Zadeh
$a.b$	$a+b-ab$	probabilística
$\max(a+b-1,0)$	$\min(a+b,1)$	Lukasiewicz
$\begin{cases} a, & \text{se } b = 1 \\ b, & \text{se } a = 1 \\ 0, & \text{senão} \end{cases}$	$\begin{cases} a, & \text{se } b = 0 \\ b, & \text{se } a = 0 \\ 1, & \text{senão} \end{cases}$	Weber

Fonte: Sandri e Correa (1999)

Os operadores de implicação $I : [0,1]^2 \rightarrow [0,1]$ são a extensão difusa do operador de implicação da Lógica Clássica (\implies), e são utilizados para modelar regras do tipo *Se-Então* (SANDRI; CORREA, 1999). As propriedades destes operadores variam na literatura, mas elas têm em comum a monotonicidade à direita, além de $I(0;b) = 1$ (princípio da falsidade) e $I(1;b) = b$ (princípio da neutralidade). Dados dois conjuntos difusos A e B , a relação $A \rightarrow B$ é expressa como:

$$\mu_{A \rightarrow B}(x, y) = I(A(x), B(y))$$

A Tabela 4.2 traz um conjunto de operadores de implicação encontrados na literatura. As t-normas não são operadores de implicação, mas são muito empregadas na prática como nos sistemas de Mamdani ($\min(a, b)$) e Larsen ($a.b$).

Tabela 4.2 - Principais operadores de implicação em aplicações difusas

Implicação	Nome
$\max(1-a,b)$	Kleene-Dienes
$\min(1-a+b,1)$	Lukasiewicz
$\begin{cases} 1, & \text{se } a \leq b \\ 0, & \text{senão} \end{cases}$	Rescher-Gaines "Sharp"
$\begin{cases} 1, & \text{se } a \leq b \\ b, & \text{senão} \end{cases}$	Bower-Godel
$\begin{cases} \min(b/a), & \text{se } a \neq b \\ 1, & \text{senão} \end{cases}$	Goguen
$1-a+ab$	Reichenback "Estocástica"
$\max(1-a,\min(a,b))$	Zadeh-Wilmott

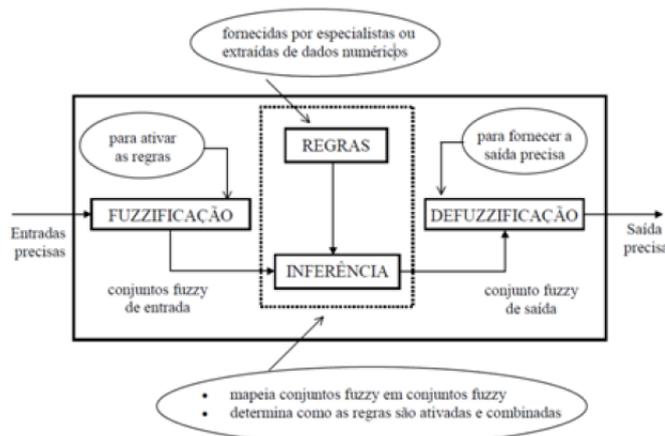
Fonte: Sandri e Correa (1999)

4.2.2 Sistemas de inferência difusa

O processo de inferência difusa é um mapeamento entre um conjunto de entradas e um conjunto de saídas utilizando um conjunto de regras de inferência que podem servir para tomadas de decisões ou inferir conhecimento do conjunto de dados (MATHWORKS, 2011).

Nos sistemas descritos como o exemplo da Figura 4.7, as entradas consistem de um conjunto de dados (por exemplo algumas medições) precisos (não difusos)(TANSCHAIT, 2004). As regra difusas, geralmente do tipo **Se** <premissa>, **Então** <conclusão>, definem o comportamento e ações do sistema conforme os valores que as variáveis podem assumir (SANDRI; CORREA, 1999).

Figura 4.7 - Esquema de um sistema de inferência difusa.



Fonte: Tanscheit (2004)

O processo denominado “Fuzzificação” basicamente codifica os dados em conjuntos difusos, que são então aplicados nas regras pela máquina de inferência, gerando um conjunto difuso como resultado. O processo de decodificação (“defuzzificação”) transforma o resultado difuso em um valor preciso.

Existem vários modelos que implementam esses sistemas de inferência, os mais conhecidos são os modelos de Takagi-Sugeno e o modelo Mamdani, principais representantes das classes de sistemas ditos clássico e interpolativo.

4.2.2.1 Modelo Takagi-Sugeno

Conhecido também como TSK (Takagi-Sugeno-Kang), o modelo Takagi-Sugeno é um sistema de inferência capaz de descrever sistemas dinâmicos não-lineares através

de sistemas lineares. É descrito por regras do tipo SE-ENTÃO que representam localmente relações lineares entre a entrada e a saída de um sistema (TANSCHKEIT, 2004). Dado um sistema com uma única variável de entrada x e uma variável de saída, tomando valores em X e Y , respectivamente, as regras deste modelo têm o seguinte formato:

Se x é A (premissa)

Então $y = ax + b$ (consequente),

onde A é um conjunto difusos definido em X .

O consequente pode ser modelado por um sistema polinomial (por exemplo, a regra acima é de ordem 1), o que o torna adequado para o tratamento de modelos numéricos. Os parâmetros a e b são ajustados conforme aos requerimentos do problema e, dependendo da quantidade de regras, o sistema pode se tornar computacionalmente caro. Este tipo de sistema não utiliza decodificação; cada regra produz uma saída e o resultado final é uma combinação linear destas saídas, usando como pesos, o grau de compatibilidade dos dados de entrada e as premissas das regras.

4.2.2.2 Modelo Mamdani

No modelo de Mamdani, o sistema de inferência é implementado utilizando-se conjuntos difusos tanto nos antecedentes como consequentes. Dado um sistema com uma única variável de entrada x e uma variável de saída, tomando valores em X e Y , respectivamente, as regras deste modelo têm o seguinte formato:

Se x é A (Premissa),

Então y é B (Consequente),

onde A e B são conjuntos difusos definidos em X e Y , respectivamente.

Segundo Knapp (2004), são 6 passos que se precisam para obter uma saída através do sistema de inferência:

- a) Determinar o conjuntos de regras difusas,
- b) Fuzzificar as entradas utilizando as funções de pertinência,
- c) Combinar as entradas fuzzificadas de acordo com as regras difusas para estabelecer uma importância associada a cada regra,

- d) Obter a consequência da regra combinando a importância recém-calculada e a função de pertinência,
- e) Combinar as consequências para obter um conjunto difuso da saída,
- f) Desfuzzificar o conjunto de saída.

No sistema de Mamdani, utiliza-se o operador min (uma t-norma) para implementar a inferência.

4.3 Sistemas Neuro-Difusos

Como é definido em Detlef (1997), um sistema Neuro-Difuso é um sistema difuso no qual um algoritmo de treinamento derivado de, ou inspirado em, redes neurais artificiais é utilizado para determinar os parâmetros (conjuntos difusos e regras difusas) através de dados de amostra de processamento.

As redes neurais são aproximadores universais e, utilizando o algoritmo de treinamento adequado, são excelentes para aprender informações através de conjuntos de entradas e saídas desejadas do problema abordado. O inconveniente é que a informação aprendida não é interpretável para o usuário, situação conhecida como “caixa preta”, ou seja, não é possível identificar como a rede gera os dados de saída a partir dos dados de entradas.

Por sua vez, estão os sistemas difusos usam regras do tipo SE-ENTÃO, o que facilita a um usuário do processo interpretar a geração de saída a partir dos dados de entrada. O problema é que essas regras devem ser fornecidas por um especialista e não são geradas automaticamente.

Portanto, a combinação de ambos paradigmas potencia ambas ferramentas, produzindo uma rede híbrida com capacidade de aprendizagem automática a partir de dados de um determinado problema e, ao mesmo tempo, gerando interpretabilidade para o usuário.

Nas redes neuro-difusas os pesos entre as conexões e as funções de ativação são diferentes das convencionais utilizadas para as redes neurais. Elas são baseadas em algum modelo de sistemas difusos. Um sistema neuro-difuso pode ser visto como uma rede neural de 3 camadas. A primeira camada representa as variáveis de entrada, a camada do meio (escondida) representa as regras difusas e a terceira camada representa as variáveis de saída. A Figura 4.8 ilustra a arquitetura de um sistema neuro-difuso baseado no modelo de Mamdani.

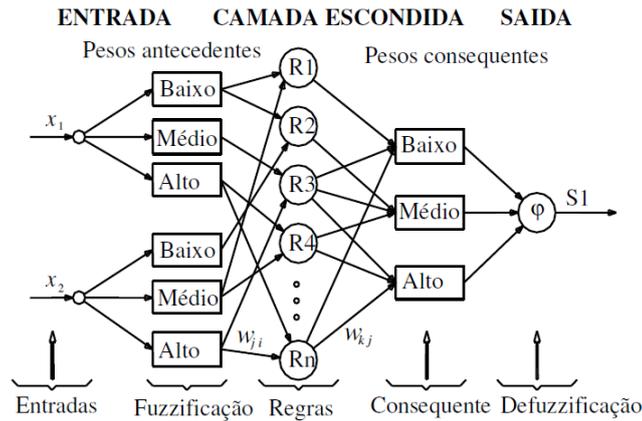


Figura 4.8 - Arquitetura básica de um sistema Neuro-Difuso.
 Fonte: Sanchez (2009)

A plataforma ANFIS (*Adaptive-Network-Based Fuzzy Inference System*) é um sistema neuro-difuso, baseado no modelo difuso de Takagi-Sugeno.

A plataforma GUAJE (*Generating Understandable and Accurate fuzzy models in a Java Environment*), implementado em um ambiente JAVA, é um software livre que implementa um algoritmo chamado HILK (de Highly Interpretable Linguistic Knowledge), que cria classificadores difusos do tipo clássico, como o de Mamdani. Todo o processo é implementado em 3 passos: definição das funções de pertinência, das variáveis de entrada, implementação de algoritmos para inferência de regras e um aprimoramento dos resultados através de um refinamento iterativo (PARRA, 2014).

Neste trabalho, foram utilizado o software GUAJE, com algoritmo de treinamento Wang-Mendel (vide Apêndice B), gerando regras com saída do tipo unitário (*singleton*). O software ANFIS foi testado, mas os resultados não foram satisfatórios. No Capítulo 6 são apresentados os resultados do uso do Guaje, exclusivamente.

Um exemplo do resultado do uso do sistema Guaje está na Figura 4.9. Alguns itens importantes são cobertura (*covering*), que indica a porcentagem de dados que ativam pelo menos uma regra, e a acurácia (*accuracy*).

Dois parâmetros importantes na configuração do GUAJE são os limiares BT (*BT-Threshold*) e AT (*Ambiguity Threshold*). O limiar BT define o valor mínimo, a partir do qual considera-se que uma amostra de entrada ativa pelo menos regra de inferência. O limiar AT é o valor mínimo de diferenciação entre duas conclusões com maior valor de pertinência, geradas pelas regras de inferência para uma amostra dada.

Figura 4.9 - Desempenho da classificação com software GUAJE.

Coverage (%) :	100
Accuracy :	0.911
Accuracy (CONS) :	0.911
Accuracy (BT=0) :	0.911
Average Confidence Firing Degree :	0.271
Minimum Confidence Firing Degree :	0
Maximum Confidence Firing Degree :	0.842
Max Error :	4
Mean Square Classification Error :	0.069

Label	Data	EC	AC (Total)	AC (Error)	UC	UC (Error)	TP	FP	TN	FN
1.0	13091	653	0	0	0	0	12438	725	4184	653
2.0	4239	760	0	0	0	0	3479	672	13089	760
3.0	534	131	0	0	0	0	403	153	17313	131
4.0	113	45	0	0	0	0	68	36	17851	45
5.0	23	10	0	0	0	0	13	13	17964	10
TOTAL	18000	1599	0	0	0	0	16401	1599	70401	1599

No exemplo da figura acima, a cobertura *coverage* de 100% significa que cada um dados testados para a classificação ativou pelo menos uma regra. A acurácia global da classificação foi de 91,1% para um total de 18.000 amostras de dados.

4.4 Diferenças entre sistemas difusos, redes neurais artificiais e sistemas neuro-difusos

Os sistemas difusos, redes neurais artificiais e sistemas neuro-difusos buscam aproximar uma função entre o espaço de entrada e o espaço de saída (KNAPP, 2004). Mas são vários os fatores que os diferenciam, tomados 2 a 2. É importante notar que numa rede neural, o mesmo sistema é utilizado para o aprendizado da resolução de um dado problema, e para a solução de uma particularização deste problema com um conjunto de dados qualquer. Por outro lado, o resultado do uso de um sistema neuro-difuso é um sistema difuso, e é este último que será utilizado na particularização. A Tabela 4.3 mostra um resumo comparando as características destes três algoritmos.

Tabela 4.3 - Comparação entre os sistemas difusos, RNA e redes neuro-difusas

	Sistemas difusos	RNA	Sistemas neuro-difusos
Conhecimento a priori	utilizado	não é utilizado	pode ser incorporado
Capacidade de aprendizado	não tem	tem	tem
Interpretação	fácil	não tem (caixa preta)	fácil
Algoritmos de aprendizado	simples	complexo	complexo

Fonte: Produção do autor, baseado em Knapp (2004)

4.5 Classificadores Hierárquicos

Uma grande quantidade de problemas de classificação envolvem um sistema hierárquico de classes, como árvores ou grafos direcionados DAG (*Directed Acyclid Graph*) (SILLA; FREITAS, 2011). Como mencionado em Wang (2016), os classificadores hierárquicos são utilizados para classificar as entradas em grupos ou *clusters*. Podem ser utilizados procedimentos do tipo *Top-Down* ou *Bottom-Up*, com aprendizagem supervisionada ou não.

Em uma aprendizagem supervisionada, com dados etiquetados fornecidos para o treinamento, ambas metodologias, tanto *top-down* como *bottom-up*, podem ser utilizadas; a diferença entre elas ocorre na fase de união/divisão.

Numa metodologia *top-down*, todas as entradas são tratadas como um agrupamento *cluster* simples, que ao longo do processamento será dividido em dois ou mais agrupamentos menores, cada qual representando um nó de uma estrutura do tipo árvore. O processo é cíclico e termina quando cada nó contém padrões únicos. A metodologia *bottom-up*, ao revés, se inicia com agrupamentos pequenos que vão sendo unidos, em um processo também cíclico.

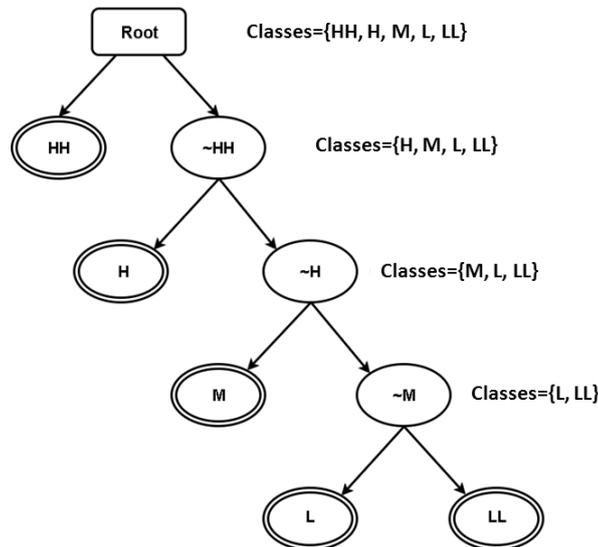
Em Silla e Freitas (2011), são propostos vários critérios de definição para os algoritmos hierárquicos, tais como tipo de estrutura do problema, a profundidade da classificação e os métodos de parada. Além disso, o trabalho menciona que a metodologia *top-down* não é um algoritmo hierárquico completo, sendo mais propriamente dito um método aplicado para evitar ou corrigir inconsistências nas predições das classes classificadas na fase do teste.

Neste trabalho foi implementado um classificador hierárquico binário, com uma metodologia *top-down*, com um esquema de árvore binária, como se visualiza na Figura

4.10, com 5 classes. O processo se inicia no nó raiz, utilizando o conjunto completo de classes na primeira fase, com o objetivo de identificar uma delas. A Figura 4.10 traz um exemplo com 5 classes: $\{HH, H, M, L, LL\}$. Na primeira fase, são identificadas se as amostras pertencem a HH, ou à classe (artificialmente criada) $\sim HH$, que denota o complemento das amostras classificadas em HH.

Para a seguinte fase, as amostras consideradas não pertencentes à classe HH ($\sim HH$), são utilizadas para distinguir entre as classes restantes. A seguinte classificação é então entre as H e ($\sim H$), que denota o complemento das amostras classificadas em H, já excetuadas aquelas classificadas como HH.

Figura 4.10 - Esquema de classificador hierárquico binário com 5 classes



Fonte: Produção do autor.

O processo é iterativo e termina quando as amostras restantes são atribuídas a uma das duas classes finais (no exemplo as classes L e LL). É preciso impor uma hierarquia entre as classes, de forma a guiar o processo. Neste trabalho, o critério escolhido para definir a hierarquia foi a distribuição de frequências das classes de saídas, relacionadas ao desvio padrão do sistema de previsão por conjuntos (quanto maior a frequência da classe, mais alta sua posição na hierarquia).

Neste trabalho, foram treinados classificadores RNA de múltiplas camadas em cada fase, utilizando-se o processo iterativo explicado anteriormente e os conjuntos de amostras correspondentes a cada fase do processo.

Como se menciona em Silla e Freitas (2011), uma desvantagem dos classificadores do tipo *top-down* é que o erro que surja em uma fase acaba sendo propagado para as fases posteriores. Para evitar que o erro continue crescendo devem ser adotadas estratégias que mitiguem esse erro.

Neste projeto, um classificador RNA foi implementado para cada fase, todas elas utilizando a mesma arquitetura de camadas ocultas da rede 4.1.2 e os mesmos parâmetros de treinamento. A única diferença entre elas está na camada de saída, onde somente 2 neurônios são utilizados para distinguir uma classe do resto (*One-against-all*).

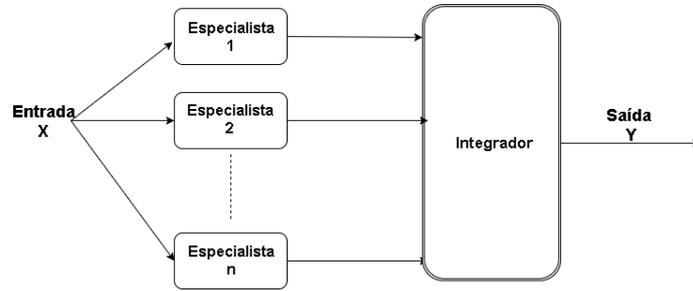
4.6 Máquinas de Comitê

As *máquinas de comitê* são um tipo algoritmo de aprendizado de máquinas que combina características de vários modelos, denominados *componentes* ou *especialistas*, visando assim melhorar o desempenho que se obtém com cada modelo separadamente. Geralmente, uma máquina de comitê tem duas fases, a primeira que computa os resultados da execução de cada componente, e a segunda que integra esses resultados.

Nesta estratégia, os componentes podem, por exemplo, ser especialistas na determinação de uma classe específica, ou serem especialistas em algum atributo. A integração dos resultados dos especialistas melhora a generalização do sistema e também sua tolerância a ruídos (VILLANUEVA, 2006). Na prática, é preciso fornecer variabilidades para cada componente ao sistema. Para isso, a codificação da saída é determinada de forma específica para cada componente. Os tipos de métodos de comitê se subdividem em dois grandes grupos, os estáticos e os dinâmicos.

Nas máquinas de comitê do tipo estático, as saídas das componentes (especialistas) são combinados utilizando um mecanismo no qual não intervêm as entradas iniciais (vide Figura 4.11). Estes métodos incluem tanto os métodos de *Ensemble* (nos quais as saídas dos especialistas são combinadas linearmente) quanto os métodos de *Boosting* (nos quais, modelos de aprendizado mais fracos são combinados para gerar um mais forte). No método de *Boosting* cada conjunto de dados de treinamento dos especialistas possui uma distribuição diferente e é treinado recursivamente visando diminuir a taxa de erro (BULLINARIA, 2004).

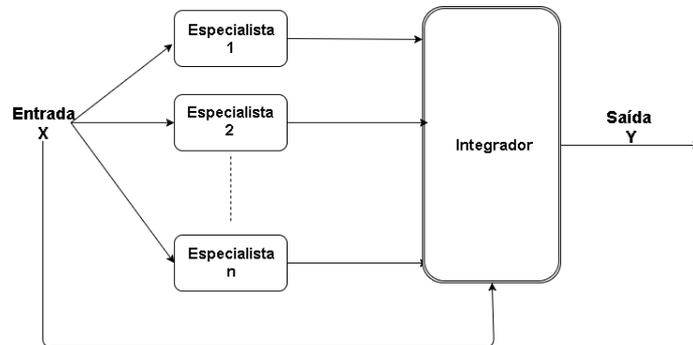
Figura 4.11 - Método de Comitê Estático



Fonte: Adaptado de Bullinaria (2004)

Nas máquinas de comitê do tipo dinâmico (vide Figura 4.12), os valores de entrada iniciais atuam também como uma das entradas da fase integradora, que combina a saída das componentes especialistas (BULLINARIA, 2004).

Figura 4.12 - Método de Comitê Dinâmico



Fonte: Adaptado de Bullinaria (2004)

4.7 Medidas de desempenho da classificação utilizadas

A matriz de confusão é um dos instrumentos mais utilizados para medir o desempenho de um classificador multi-classe. Um exemplo se visualiza na Tabela 4.4, as linhas da matriz representam as os valores verdadeiros ou observações de cada classe e as colunas representam as os resultados de inferência do classificador para cada classe. No exemplo, temos um total de 1.962 amostras distribuídas em 3 classes. A classe 1 possui 891 (690+152+49) amostras, a classe 2 possui 934 amostras e a classe 3 possui 137 amostras. O classificador infere que 817 (690+109+18) amostras pertencem à classe 1, 688 amostras pertencem à classe 2 e 457 amostras pertencem

cem à classe 3. Como menciona [Lever et al. \(2016\)](#), o entendimento do objetivo da classificação é a chave para escolher a métrica adequada para avaliar o classificador. Neste trabalho, foram escolhidas duas métricas a Sensibilidade, Precisão e a Acurácia Global para avaliar o classificador.

Tabela 4.4 - Matriz de confusão

		Inferido			
		Classe 1	Classe 2	Classe 3	
Obs	Classe 1	690	152	49	77,44
	Classe 2	109	501	324	53,64
	Classe 3	18	35	84	61,31
(%)		84,4	72,8	18,38	63,47

A *Precisão* da classe j mede a proporção de inferência correta dessa classe e se calcula como segue:

$$Prec_j = \frac{diagonal_j}{\sum(coluna(classe_j))}$$

A *Sensibilidade* (também conhecido como *True Positive Rate (TPR)* ou *Recall*) da classe i , mede a proporção de os dados observados da classe que foram inferidos corretamente, se calcula como segue:

$$Sens_i = \frac{diagonal_i}{\sum(linha(classe_i))}$$

A *Acurácia Global* mede a proporção de inferência correta de todas as classes e se calcula como segue:

$$Acurácia = \frac{\sum diagonal_j}{Total - de - dados}$$

Por exemplo, visualizando a Tabela 4.4 o classificador possui uma Acurácia global de $(690 + 501 + 84)/1.962 = 63,47\%$; a Precisão da classe 1 é de $84,4\%$ ($690/817$) e a Sensibilidade é de $77,44\%$ ($690/891$). Isto indica que precisão da inferência para essa classe é alta ($84,4\%$) e a capacidade do classificador para identificar corretamente essa classe é um pouco menor ($77,44\%$).

A classe 3 possui uma Precisão de $18,38\%$ e uma Sensibilidade de $61,31\%$, indicando

que o classificador identifica melhor a classe do que infere. Note que 324 amostras são inferidas como classe 3 mas na realidade pertencem a classe 2.

Usando uma métrica somente não se consegue distinguir todas as potencialidades e deficiências do classificador, inclusive se é utilizada uma métrica mais completa como a F_1 -score (LEVER et al., 2016).

5 PREVISIBILIDADE: *BREEDING* APLICADO AO MODELO GLOBAL DO CPTEC

Neste capítulo são apresentados os resultados da aplicação do processo de *Breeding* ao modelo global do CPTEC. Antes de iniciar com os testes de *Breeding*, foi necessário realizar a migração do código operacional do modelo global, para poder utilizá-lo no cluster Lacibrido, onde este trabalho foi desenvolvido.

O código operacional do modelo global é customizado para o compilador Portland, utilizado no supercomputador Tupã (XT6 da Cray) do CPTEC. Visando ter não só independência para pesquisas, mas também desenvolvimentos futuros, foi realizada a migração do código para o cluster Lacibrido¹ do Laboratório Associado de Computação e Matemática Aplicada (LAC) do INPE, utilizando compiladores *open-source* como Gfortran e MVAPICH-MPI.

Durante a fase de migração e debugging do código, em paralelo foram feitos testes iniciais com outro modelo denominado SPEEDY, um modelo de circulação global da atmosfera de menor complexidade que o do CPTEC. Isto ajudou na fase de capacitação inicial dos conceitos de modelagem atmosférica e a aplicação do *Breeding* a um modelo atmosférico. Os primeiros resultados obtidos com o modelo SPEEDY foram apresentados no Congresso de Ciências não Lineares e Complexidade realizado no INPE em maio de 2016 (ROMERO et al., 2016b). Mais detalhes sobre este trabalho se encontram no apêndice A.

Uma vez finalizada a fase de migração e debugging do modelo global do CPTEC, foram realizados os primeiros testes de *Breeding* com data inicial de 02 de dezembro de 2014 às 00 Hs. Foi gerada uma perturbação inicial com distribuição uniforme na camada da temperatura virtual de superfície, incrementando o valor desta variável inicial em 0.1% para todo o globo (vide Figura 5.1). Este método de perturbação foi escolhido por ser de simples implementação.

A norma escolhida para realizar o re-escalamento foi o RMS (*Root Mean Square*) dos campos de Temperatura de Superfície para todo o domínio:

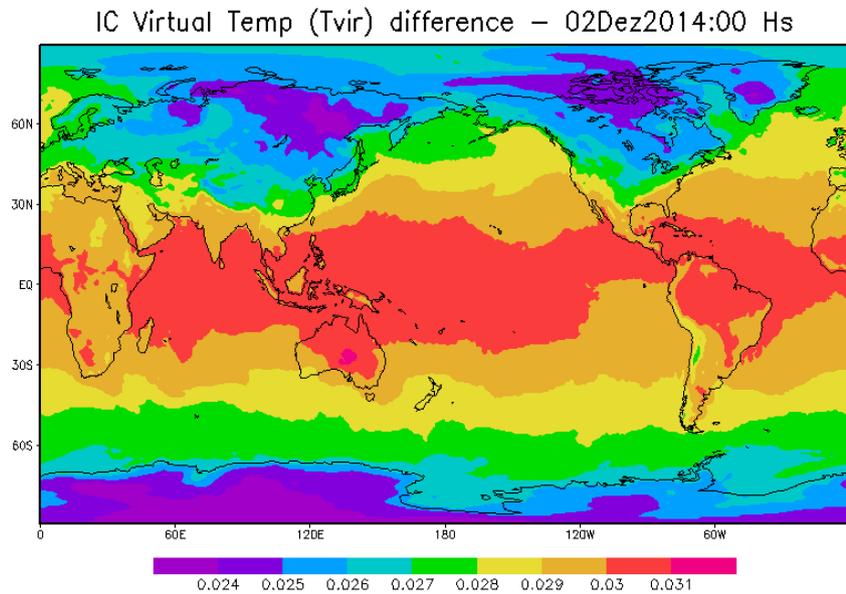
$$Norma = \frac{1}{N_{lat} \times N_{lon}} \sum_{i=1, j=1}^{N_{lat}, N_{lon}} [(\text{Temp}_{\text{Ctrl}})_{i,j} - (\text{Temp}_{\text{Pert}})_{i,j}]^2, \quad (5.1)$$

onde N_{lat} e N_{lon} são os números de pontos de discretização de latitude e longitude,

¹Mais detalhes técnicos do cluster se encontram em <http://www.lac.inpe.br/~celso/LAQUIBRIDO.html>.

respectivamente.

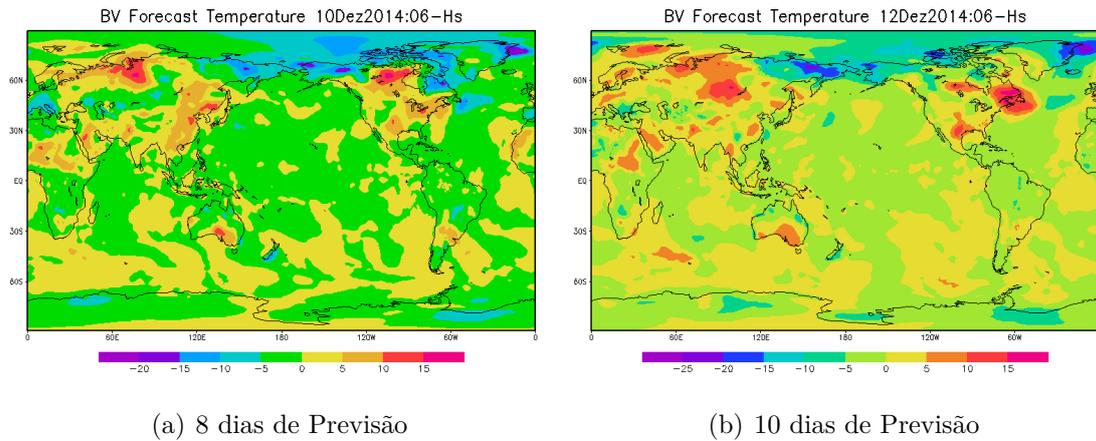
Figura 5.1 - Campo de diferença entre a condição inicial de controle e a condição inicial perturbada, no campo da temperatura virtual a nível de superfície.



Fonte: Produção do autor

A Figura 5.2 mostra os *bred vectors* para o campo de temperatura de superfície, depois de 8 e 10 dias de simulação do processo de *Breeding*. Pode-se verificar na mesma figura que regiões com maior magnitude de *bred vector* se localizam predominantemente na América do Norte, no norte da Rússia, e no sul de Austrália.

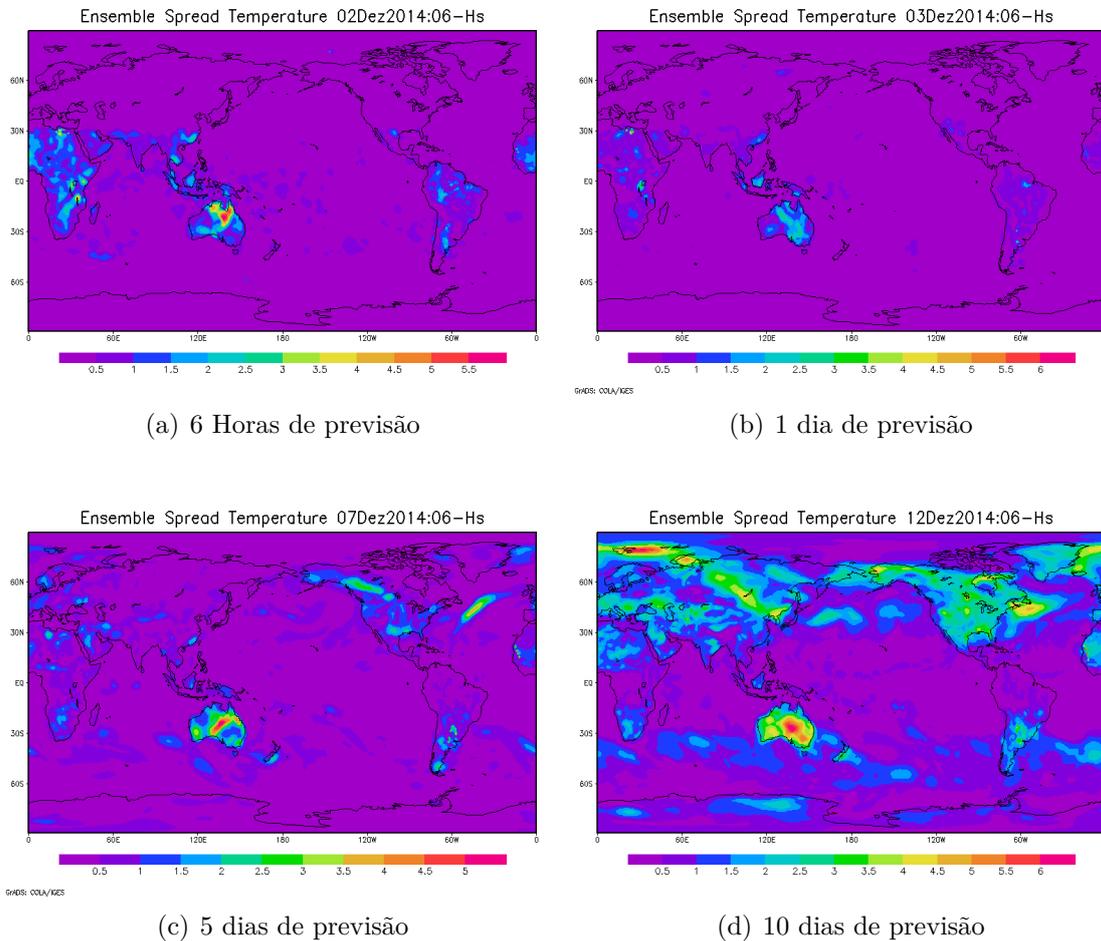
Figura 5.2 - *Bred vectors* da temperatura de superfície.



Fonte: Produção do autor

A metodologia *Ensemble* não foi simulada, somente foram utilizados os resultados operacionais do CPTEC. Na Figura 5.3, pode-se visualizar a evolução do espalhamento (desvio padrão) da previsão. Pode-se notar que o espalhamento cresce à medida que aumentam os dias de simulação. As regiões com maior crescimento se encontram na Austrália, no norte da Rússia e na América do Norte.

Figura 5.3 - Espalhamento (desvio padrão) das previsões por *Ensemble* do modelo global no campo de temperatura de superfície.



(a) 6 Horas de previsão

(b) 1 dia de previsão

(c) 5 dias de previsão

(d) 10 dias de previsão

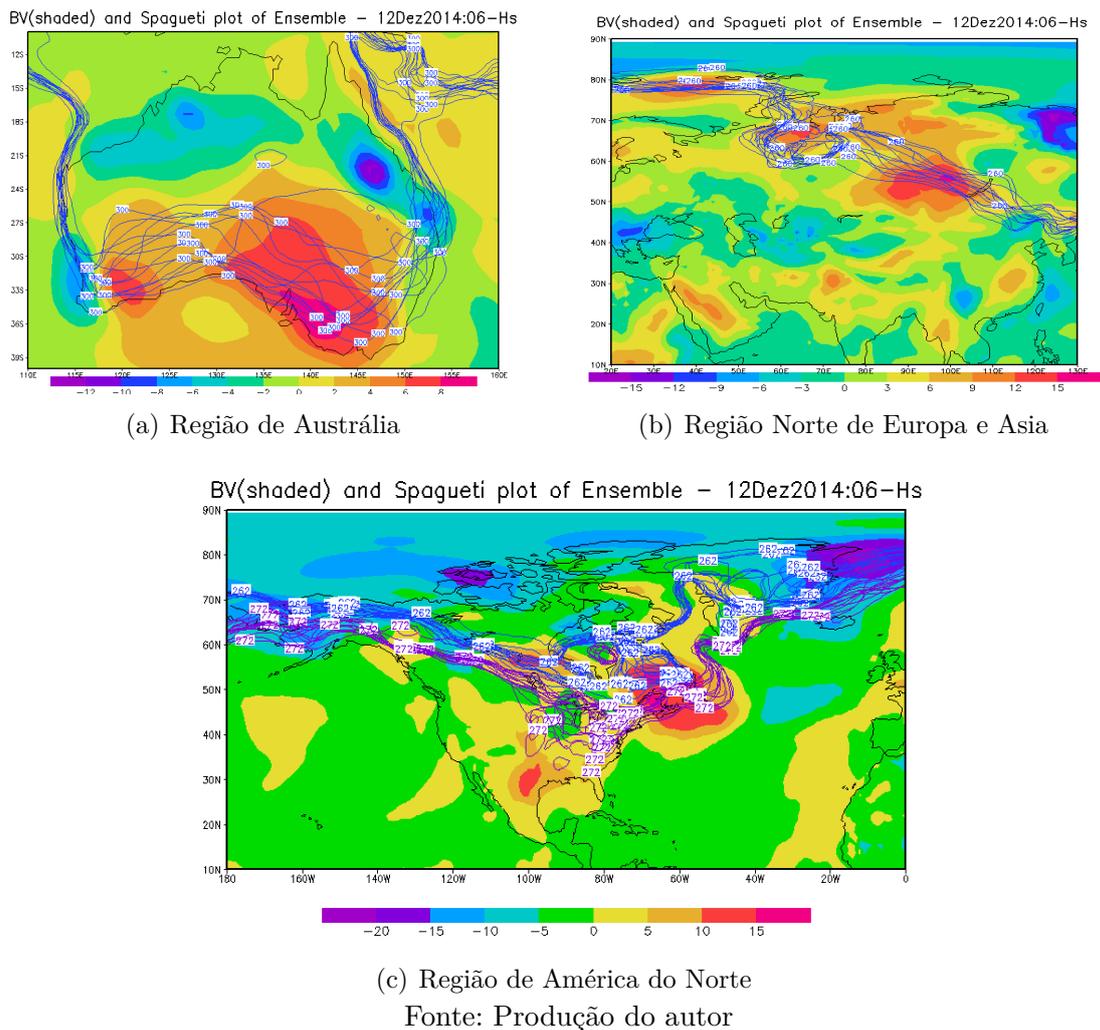
Fonte: Produção do autor

Comparando-se os resultados para a mesma data e tempo de simulação das técnicas de *Breeding* e *Ensemble* (vide Figuras 5.2b e 5.3d), nota-se que ambos mostram semelhanças nas regiões de maior crescimento dos erros na previsão. As regiões onde o *Ensemble* mostra uma maior perda da acurácia da previsão, indicado por um maior valor do espalhamento, coincidem com as regiões de grande magnitude do *bred vector*. Estes padrões são um bom indicativo de que o processo de *Breeding* pode servir para análises de previsibilidade.

A Figura 5.4 mostra os *bred vectors* da temperatura superficial em escala de cores (*shaded*) e os gráficos espaguete dos membros do *Ensemble*. O gráfico espaguete consiste em isolinhas para uma temperatura específica. Cada região de estudo mos-

tra como é a natureza do espalhamento e como as áreas de maior espalhamento coincidem com aquelas onde os *bred vectors* têm uma maior magnitude. Este tipo de análise, usando gráficos espaguete, sugere que informações proporcionadas pelo espalhamento da previsão por *Ensemble* são indicativos de previsibilidade, assim como a magnitude dos *bred vectors*.

Figura 5.4 - *Bred vectors* da temperatura de superfície e gráficos espaguete dos membros do *Ensemble*.

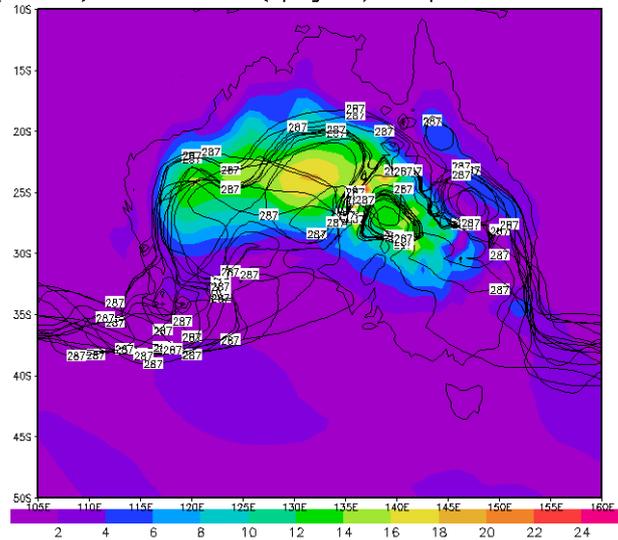


Os padrões de *Breeding* também podem ser estudados para outras datas. Por exemplo, na Figura 5.5 se visualizam padrões de *Breeding* e o gráfico espaguete do *Ensemble* para as datas de outubro de 2012. Padrões nas regiões de Austrália e em áreas de América do Norte, mostram um grande espalhamento nas isolinhas da temperatura

de superfície, assim como áreas de grande magnitude do *bred vector* da temperatura.

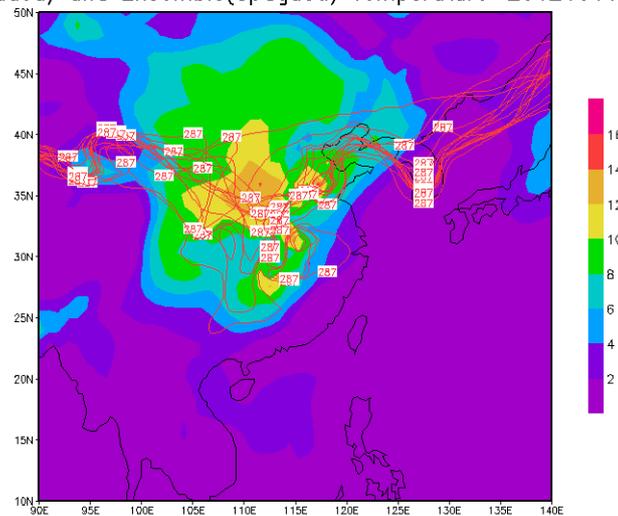
Figura 5.5 - *Bred vectors* da temperatura de superfície e gráficos espaguete dos membros do *Ensemble*.

BV(shaded) and Ensemble(spagueti) Temperature 2012101618



(a) Região de Australia

BV(shaded) and Ensemble(Spagueti) Temperature-2012101618



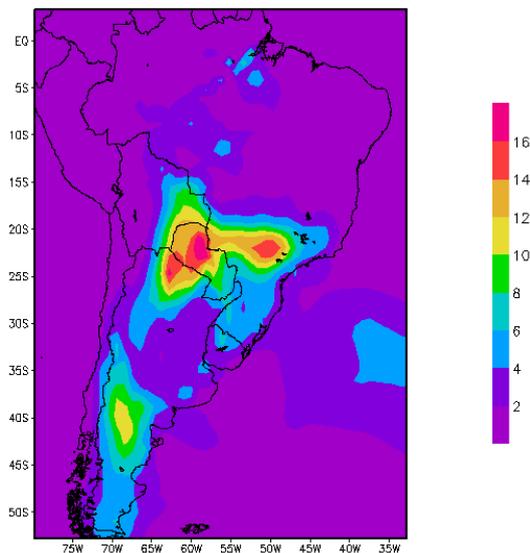
(b) Região de América do Norte

Fonte: Produção do autor

A Figura 5.6 mostra a simulação com data de início 02 de outubro de 2012, considerando-se a variável temperatura de superfície. Nesta figura, pode-se notar os mesmos padrões de comportamento da metodologia de *Breeding* e dos membros do *Ensemble* em áreas da América do Sul.

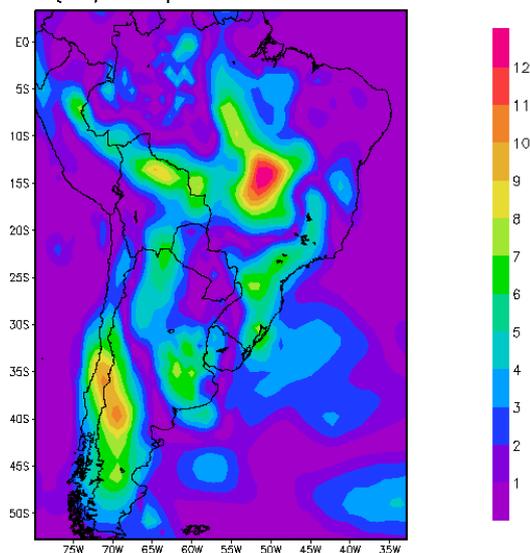
Figura 5.6 - *Bred vectors* da temperatura de superfície e a variância dos membros do *Ensemble* para a América do Sul.

Variância do Ensemble 2012101700



(a) Variância do Ensemble

Abs(BV) Temperatura 2012101700



(b) *Bred vectors* de temperatura

Fonte: Produção do autor

Podemos notar na Figura 5.6, que a variância dos membros do *Ensemble* têm o maior valor na região do norte do Paraguai e nos estados de São Paulo e Minas Gerais no Brasil. Também pode-se notar um valor médio da variância na região de fronteira

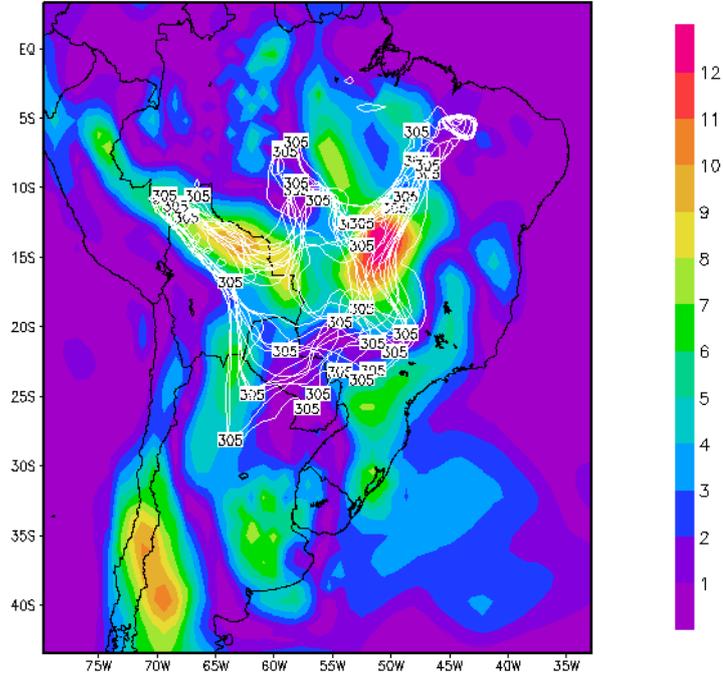
entre o Chile e a Argentina, perto da latitude 40°S . Por outro lado, a figura mostra que, nesta região, existem *bred vectors* cuja magnitude absoluta se encontra no valor médio da escala. No entanto, existe uma área no meio do Brasil, onde a magnitude do *bred vector* é máxima, situação que não ocorre com a variância do *Ensemble*.

Na Figura 5.7, se verifica que a região na qual a magnitude do *bred vector* alcança o valor máximo, o gráfico espaguete para temperatura nas isolinhas de 305 K também apresenta discrepância entre seus membros. No entanto, este comportamento não é predominante no *Ensemble* como um todo, e por isso ele não se traduz num aumento da variância. Outras isolinhas de temperatura, como por exemplo para a temperatura de 295 K, mostram discrepâncias que concordam com a valor máximo da variância do *Ensemble*.

Analisando o caso do mês de maio para o mesmo ano na Figura 5.8, vemos que a variância do *ensemble* atinge um valor máximo na região do norte argentino na fronteira com o Paraguai. A figura mostra também que o *bred vector* tem uma magnitude significativa na mesma região.

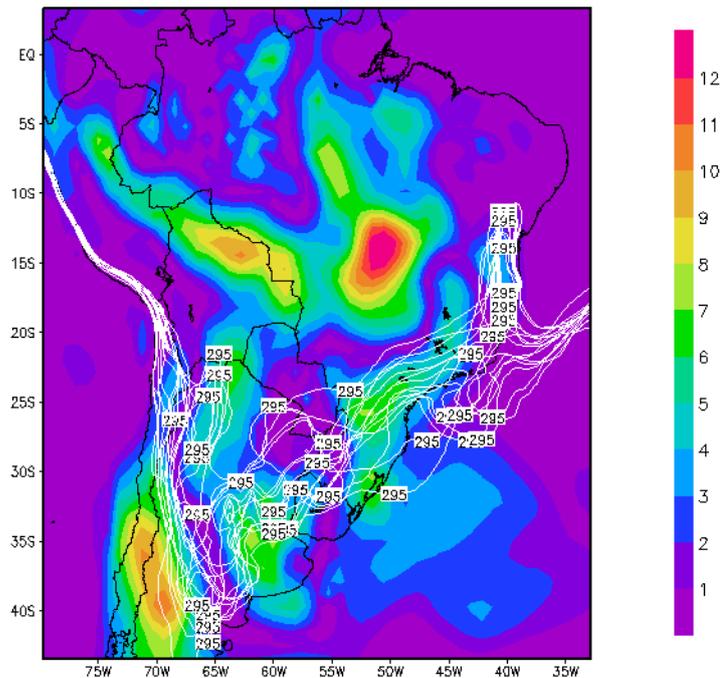
Figura 5.7 - *Bred vectors* da temperatura de superfície e gráficos de espaguete dos membros do *Ensemble* para a região de América do Sul.

Ensemble Method (spaguete) and Breeding (shaded) 2012101700



(a) Isolinhas de temperatura = 305K

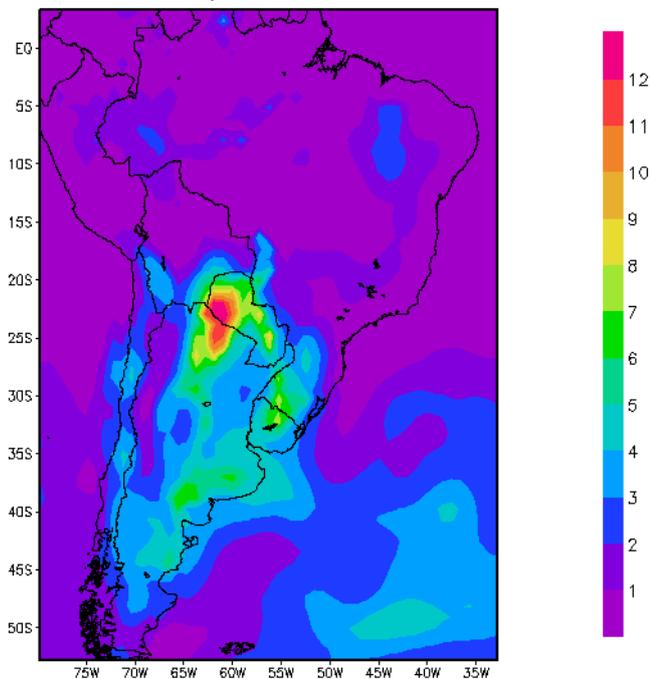
Ensemble Method (spaguete) and Breeding (shaded) 2012101700



(b) Isolinhas de temperatura=295K
Fonte: Produção do autor

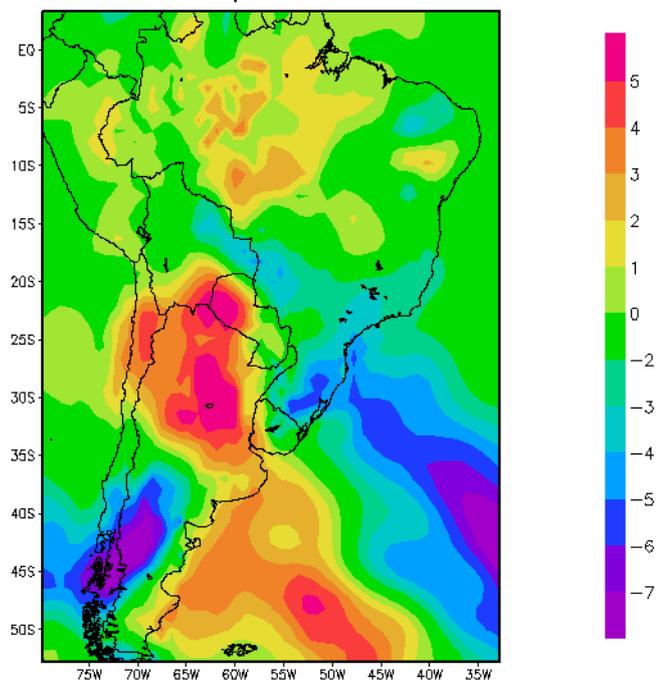
Figura 5.8 - *Bred vectors* da temperatura de superfície e variância dos membros do *ensemble* para a região de América do Sul para a data de 17 de maio de 2012

Variância Temperatura 2012051700



(a) Variância do *ensemble*

Bred vector Temperatura 2012051700



(b) *Bred vector* da temperatura

Fonte: Produção do autor

5.1 Sumário

Neste capítulo foram apresentados os experimentos de *Breeding* aplicados ao modelo global do CPTEC. Foram comparados os resultados do *ensemble* com o *Breeding* em diversos testes, considerando-se várias datas e regiões.

Foi verificado que em quase todos os casos analisados, as regiões onde a magnitude *bred vector* é grande coincidem com as regiões onde o *Ensemble* apresenta maior discrepância, o que pode ser visualizado tanto pela magnitude da variância quanto com o uso de gráficos espaguete.

Os resultados deste capítulo foram apresentados no Congresso de Ciências Interdisciplinares realizado no INPE-SJC (ROMERO et al., 2016a).

6 PREVISIBILIDADE: IDENTIFICAÇÃO DE CLASSES

Neste capítulo, apresentamos os sistemas desenvolvidos para classificar os níveis de confiabilidade da previsão obtidos pelo processo de *breeding*. Foram adotadas 5 classes de previsibilidade, rotuladas de 1 a 5, também referenciadas no texto com os rótulos $\{HH, H, M, L, LL\}$ e que é explicado na seção 6.1.2. A entrada dos classificadores é um conjunto de *bred vectors* associados às variáveis meteorológicas. Já a saída de cada classificador é uma classe de previsibilidade no conjunto, onde quanto menor o valor da classe, melhor é a previsibilidade.

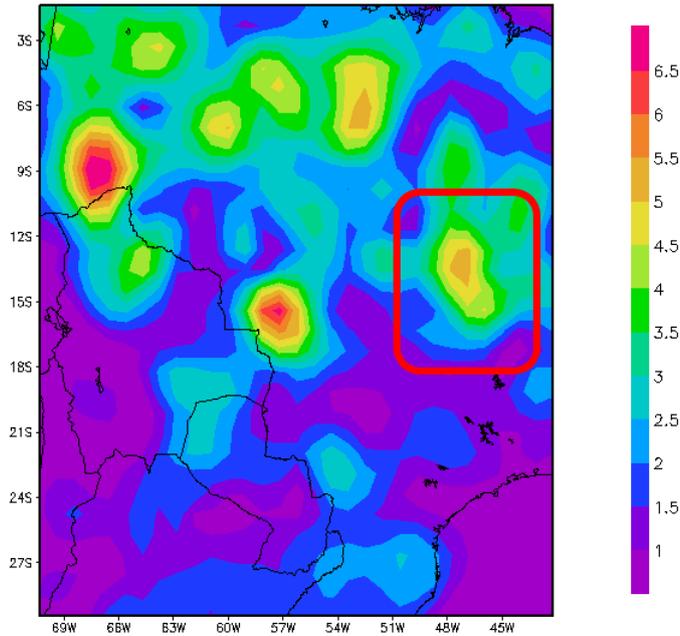
Foram realizados experimentos com previsões para uma data do mês de maio e uma data no mês de outubro, em 4 anos cronologicamente seguidos. Os resultados destas previsões formam o conjunto de dados para o estudo. Inicialmente, utilizou-se dados dos 3 primeiros anos para o treinamento e os dados do 4º ano usado para teste de verificação (amostragem cronológica). Entretanto, os experimentos revelaram que os resultados não são neutros em relação ao tipo de amostragem. Foi então gerado um outro conjunto de treinamento com dados dos 4 anos de previsão, tomando-se 3 dados consecutivos para treinamento e o 4º para teste (amostragem alternada).

6.1 Região de Estudo e obtenção de dados de entrada-saída

A previsão com modelo global gera uma enorme quantidade de dados. Para poder se adequar aos softwares disponíveis, definiu-se uma região de estudo, o que levou a uma quantidade menor de dados e também a uma melhora nos resultados. A região de estudo escolhida é uma parte da América do Sul correspondente às coordenadas latitude= $[11'S - 20'S]$ e longitude= $[42'W - 50'W]$, como se visualiza na Figura 6.1.

Figura 6.1 - Região de estudo na América do Sul marcada por um quadro vermelho, correspondente à latitude=[11'S - 20'S] e à longitude=[42'W - 50'W], tendo como fundo o desvio padrão da temperatura superficial dos membros do sistema de previsão por conjuntos (*Ensemble*) para 02 de maio de 2014 às 00 Hs

Desvio Padrao (Spread) – Região de America do Sul



Fonte: Produção do autor

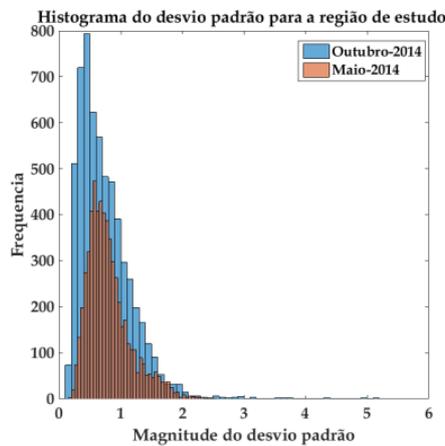
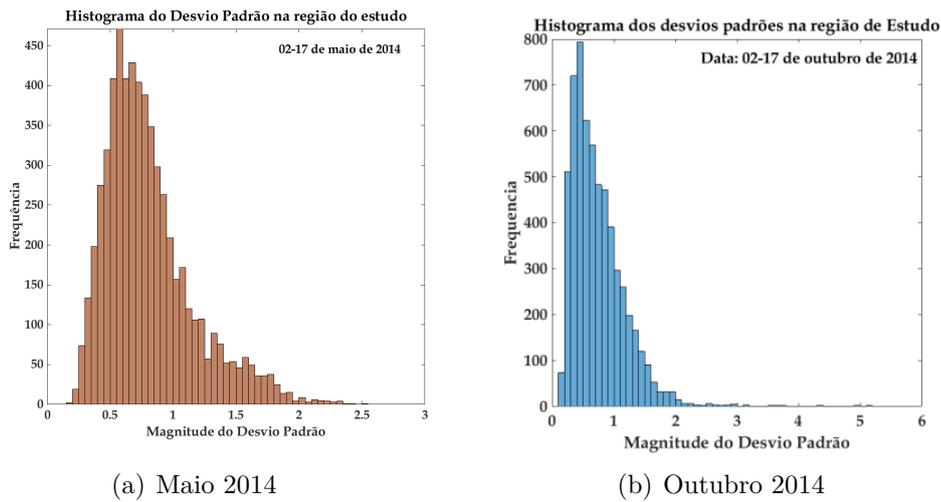
Na grade utilizada pelo CPTEC para o modelo global, a região de estudo ocupa $N_x = 10$ pontos de na horizontal (longitude) e $N_y = 10$ pontos na vertical (latitude), ou seja, um conjunto de 10×10 pontos. Como a grade do modelo é de aproximadamente de 100 km, a área é de aproximadamente 10.000 km^2 de cobertura. Em cada ponto de grade, tem-se informação de temperatura, umidade, ventos meridional e zonal, pressão, além de outras variáveis meteorológicas derivadas fornecidas pelo modelo.

Nos primeiros testes foram usados valores de (N_x, N_y) maiores na definição da região de estudo. Por exemplo, no primeiro teste foram usados 100×100 pontos. No entanto, a quantidade de dados tornava praticamente inviável o uso dos pacotes de software. O domínio de estudo foi então reduzido até tornar o problema tratável. Além disso, os resultados não foram satisfatórios, como mostra a seção C.0.1.

A previsão por conjuntos, assim como o processo de *Breeding*, compreende um período de 15 dias, com saídas intermediárias a cada 6 horas. Portanto, em todo o processo tem-se um total de 60 saídas (4 saídas por dia em 15 dias de simulação). De cada saída se extraem $N_x \times N_y=100$ pontos de informação da região de estudo, e portanto em todo o processo têm-se $60 \times 100 = 6.000$ pontos de informação.

A Figura 6.2 mostra os histogramas das magnitudes do desvio padrão (calculados da previsão por *Ensemble*) na mesma região de estudo para distintas datas. Pode-se verificar na figura como a distribuição de frequência dos valores varia ao longo do tempo. É possível notar também a baixa frequência de grandes valores de desvio padrão, associados a regiões com pouca confiabilidade da previsão.

Figura 6.2 - Histogramas da magnitude do desvio padrão da previsão de temperatura superficial do sistema de previsão por conjuntos, na região de estudo na América do Sul, correspondente à latitude=[11'S - 20'S] e à longitude=[42'W - 50'W].



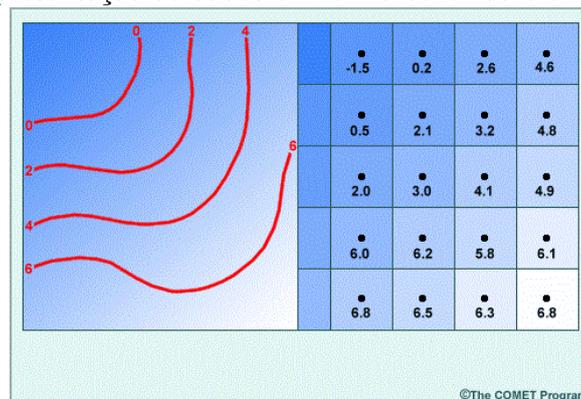
(c) Comparação entre Maio e Outubro

6.1.1 Conjuntos de dados de treinamento e teste

Um dos objetivos deste trabalho é gerar classificadores de previsibilidade utilizando o processo de *Breeding*. Para isso foram testados vários algoritmos de aprendizado de máquina supervisionados, para os quais é preciso fornecer um conjunto de exemplos de pares (entrada, saída desejada).

O modelo AGCM-CPTEC possui uma representação discreta de três dimensões espaciais, com tamanho de passo espacial de aproximadamente 100 km (para a configuração T0126) e 28 camadas verticais. As saídas do modelo são apresentadas como mapas contínuos das distintas variáveis meteorológicas, mas as informações estão disponíveis em pontos de grade (vide Figura 6.3). Mesmo se optando pela a utilização de uma região de estudo menor, ainda é preciso reduzir o conjunto de dados. Por esta razão, neste trabalho somente foram utilizados dados do primeiro nível (superfície) para gerar os conjuntos de dados de treinamento e teste.

Figura 6.3 - Representação de dados em forma contínua e em pontos de grade.



Fonte: *The Comet Program*, imagem adaptada de http://www.goes-r.gov/users/comet/tropical/textbook_2nd_edition/media/graphics/gdspctmp.gif

Neste trabalho, os dados de entrada correspondem aos resultados do processo de *Breeding* em relação a algumas variáveis, como temperatura. Os dados de saída (classes de previsibilidade) foram gerados a partir da variância da previsão por conjunto (*Ensemble*), utilizando-se o desvio padrão (*spread*) como saídas desejadas para o classificador.

A Figura 6.4 mostra a primeira e segunda abordagens na montagem da tabela de dados de entrada-saída. A região de estudo está marcada com um quadro laranja de 10×10 pontos. A região marcada pelo quadro vermelho é uma janela móvel (ou deslizante) em torno de um ponto central (i, j) (em amarelo). Numa primeira tentativa,

Com a movimentação da janela móvel por toda a região de estudo, foram coletadas as informações referentes aos pontos centrais (i, j) como também as médias das informações dos vizinhos, durante 15 dias, com 4 valores por dia (60 tomadas de tempo no total). A montagem final da tabela de dados se visualiza na Figura 6.6, com os valores das 60 saídas de previsões de ambos modelos, tanto o *Breeding* para a entrada e o *Ensemble* para a saída.

Figura 6.6 - Esquema final da tabela de dados com 15 dias de simulação e 100 pontos na grade.

	Ponto	Entrada			Saída
T=0 (dia 1, tomada 1)	(1,1)	BV (Temp)	Media (Vizinhos-Esquinas)	Média (Vizinhos laterais)	Classe (Spread -Temp)
	(2,1)	BV (Temp)	Media (Vizinhos-Esquinas)	Média (Vizinhos laterais)	Classe (Spread -Temp)
	(10,10)	BV (Temp)	Media (Vizinhos-Esquinas)	Média (Vizinhos laterais)	Classe (Spread -Temp)
T=59 (dia 15, tomada 4)	(1,1)	BV (Temp)	Media (Vizinhos-Esquinas)	Média (Vizinhos laterais)	Classe (Spread -Temp)
	(2,1)	BV (Temp)	Media (Vizinhos-Esquinas)	Média (Vizinhos laterais)	Classe (Spread -Temp)
	(10,10)	BV (Temp)	Media (Vizinhos-Esquinas)	Média (Vizinhos laterais)	Classe (Spread -Temp)

Neste trabalho, foram utilizados dados correspondentes aos anos 2011 a 2014, iniciando-se em 02 de maio de 2011 às 00 Hs, totalizando 24000 amostras (6000 amostras por ano em 4 anos). Os dados foram particionados em bases de treinamento e teste, com 3/4 (18000) e 1/4 (6000) dos dados, respectivamente. As bases foram particionadas segundo duas abordagens, que serão apresentadas na próxima seção. Na primeira abordagem, chamada de *Cronológica*, os dados dos anos de 2011 a 2013 foram utilizados para treinamento e o de 2014 para teste. A segunda abordagem, chamada de *Alternada*, aloca três amostras consecutivas no conjunto de treinamento e a quarta no conjunto de teste.

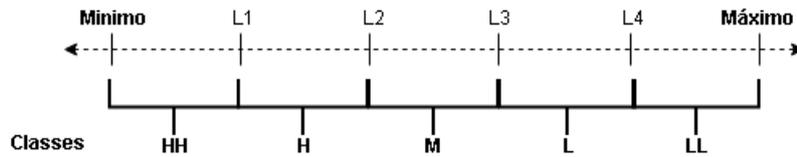
6.1.2 Mapeamento de desvio padrão em classes de previsibilidade

Os valores de desvio padrão são mapeados em 5 classes de previsibilidade, com a seguinte rotulação:

- HH (**H**igh-**H**igh) Classe de muito alta confiabilidade (1)
- H (**H**igh) Classe de alta confiabilidade (2)
- M (**M**edium) Classe de confiabilidade média (3)
- L (**L**ow) Classe de baixa confiabilidade (4)
- LL (**L**ow-**L**ow) Classe de muito baixa confiabilidade (5)

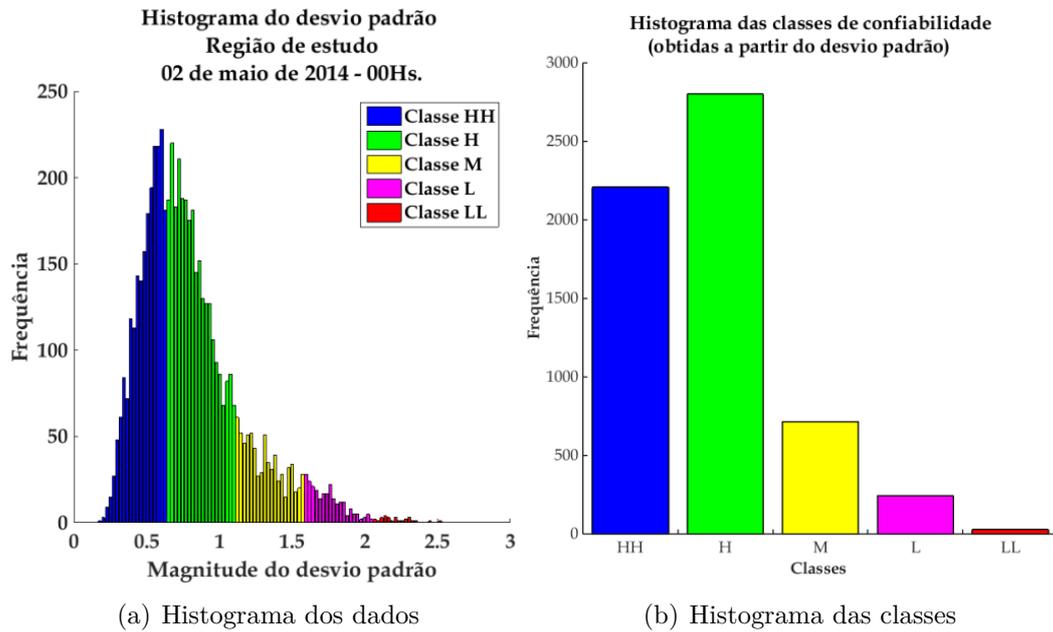
De todas as partições testadas, os melhores resultados foram obtidos particionando-se o domínio do desvio padrão de maneira uniforme, de forma a que cada classe possua um intervalo de mesmo tamanho (vide Figura 6.7): a classe HH toma os valores desde o valor mínimo até o valor L1, a classe H toma os valores desde L1 até L2, e assim sucessivamente.

Figura 6.7 - Esquema de partição para geração de classes.



Na Figura 6.8, visualiza-se o histograma das magnitudes do desvio padrão e das classes correspondentes, segundo o particionamento adotado. A maior parte dos valores estão nas classe HH e H, onde o desvio padrão é pequeno e, portanto, com alta confiabilidade.

Figura 6.8 - Histogramas da magnitude do desvio padrão da previsão de temperatura superficial e das classes correspondentes.



6.2 Classificação com amostragem cronológica

Nesta seção, são apresentadas as classificações realizadas com os dados de simulações na abordagem cronológica, onde os dados dos anos de 2011 a 2013 foram utilizados para treinamento e o de 2014 para teste. Os dados estão distribuídos nas classes como mostra a Tabela 6.1. Abaixo, apresentamos primeiramente as classificações obtidas com redes neuro-difusas e em seguida, aquelas obtidas com redes neurais. Cada amostra têm 16 atributos de entrada e uma classe de saída desejada (os atributos de entrada estão detalhados no apêndice).

Tabela 6.1 - Distribuição de amostras na amostragem cronológica

Classe	1 (HH)	2 (H)	3 (M)	4 (L)	5(LL)	Total
Treinamento	13091	4239	534	113	23	18000
Teste	2209	2802	716	245	28	6000

6.2.1 Classificação com a rede neuro-difusa GUAJE

O uso do sistema neuro-difuso GUAJE gerou um total de 9.612 regras de inferência. Como se observa na matriz de confusão na Tabela 6.2, o GUAJE teve um bom desempenho classificação dos dados de treinamento, obtendo uma acurácia global de 91,1%. Como se verifica na tabela, as classes com menor quantidade de dados apresentaram menor precisão. Por exemplo, a classe 5 (LL) com 23 pares de entrada-saída observadas apresentou uma sensibilidade de 56,5% e uma precisão de 50%. A última coluna da tabela traz o número de amostras que o GUAJE não classificou, seja por erros por ambiguidade ou por falta de ativação de regras. Nota-se na Tabela 6.2 que todas amostras foram classificadas.

Tabela 6.2 - Matriz de confusão do conjunto de treinamento - amostragem cronológica

		Inferido					Sem classe
		Classe 1 (HH)	Classe 2 (H)	Classe 3 (M)	Classe 4 (L)	Classe 5 (LL)	
Observado	Classe 1 (HH)	12438	592	48	5	8	0
	Classe 2 (H)	659	3479	85	13	3	0
	Classe 3 (M)	53	63	403	15	0	0
	Classe 4 (L)	12	11	20	68	2	0
	Classe 5 (LL)	1	6	0	3	13	0

A classificação com o conjunto teste, composta de dados completamente novos de uma simulação correspondente ao ano 2014 na mesma data, o desempenho cai para 41% com uma cobertura de dados de 94,817%, algumas entradas não foram capazes de ativar ao menos uma regra. Visualizando a matriz de confusão na Tabela 6.3, verifica-se que não ocorreram identificações na classe 5 (LL), e que a sensibilidade da classe 4 (L) é bastante baixa (8 acertos em 245 pontos).

Tabela 6.3 - Matriz de confusão do conjunto de teste (2014) - amostragem cronológica

		Inferido					Sem Classe
		Classe 1 (HH)	Classe 2 (H)	Classe 3 (M)	Classe 4 (L)	Classe 5 (LL)	
Observado	Classe 1 (HH)	1698	402	54	17	4	34
	Classe 2 (H)	1839	725	78	13	1	146
	Classe 3 (M)	389	190	29	12	1	95
	Classe 4 (L)	141	53	10	8	0	33
	Classe 5 (LL)	16	5	0	0	0	7

Nota-se que o GUAJE não obteve boa generalização, já que o desempenho dos conjuntos de testes e treinamento foram muito distintos. Um dos possíveis motivos para tanto é que o conjunto de dados de 2011 a 2013 não tem padrões suficientes para se classificar corretamente as amostras de 2014.

Um outro possível motivo seria a combinação de atributos de muitas entradas *fuzzificadas* gerar um valor de pertinência final menor que o limiar mínimo fixado/configurado no GUAJE (*BT-Threshold*), o que levaria o sistema a deixar de classificar muitas amostras. No entanto, nossos resultados foram obtidos com um valor muito baixo para o BT (0,01).

Em particular, pode-se notar que a classe com pior desempenho (baixa taxa de classificação) é a classe 5 (LL), onde das 28 amostras nenhuma foi classificada com sucesso, com um total de 6 falsos positivos, 7 amostras sem classificação e as demais 15 amostras resultam em erro na classificação.

6.2.2 Classificação utilizando redes neurais

A matriz de confusão relativa à classificação dos dados de treinamento da RNA na Tabela 6.4 mostra que RNA obteve uma boa acurácia global (89,8%). Pode-se notar também que a classificação registrou uma precisão de de 93,14% para a classe 1 (HH); 85,45% para a classe 2; 82,93% para a classe 3, 71,22% para a classe 4 e 100% para a classe 5 (LL).

Comparando os dados inferidos, cada classe teve uma sensibilidades de: 95,87% para a classe 1 (HH), um 85,10% para a classe 2 (H), um 76,56% para a classe 3 (M), um

38,67% para a classe 4 (L) e um 13,04% para a classe 5 (LL).

Tabela 6.4 - Matriz de confusão do conjunto de treinamento - amostragem cronológica

		Inferido				
		Classe 1 (HH)	Classe 2 (H)	Classe 3 (M)	Classe 4 (L)	Classe 5 (LL)
Observado	Classe 1 (HH)	10187	421	19	0	0
	Classe 2 (H)	658	4575	141	2	0
	Classe 3 (M)	75	307	1297	16	0
	Classe 4 (L)	10	48	99	99	0
	Classe 5 (LL)	7	3	8	22	6

O conjunto de teste registra uma acurácia global de 75,47%. A matriz de confusão (Tabela 6.5) mostra que dos valores classificados há acerto de 86,21% para a classe 1 (HH), 28,2% para a classe 2, 0,9% para a classe 3 e a rede não consegue identificar as demais classes. Na comparação dos dados classificados pela rede com a quantidade de dados para cada classe correspondente, tem-se uma sensibilidade de 86,2% para a classe 1 (HH), uma taxa de 23,6% para a classe 2 (H), um 5,26% para a classe 3 (M). Para as classes restantes a rede não consegue identificar – o resultado não surpreende, pois ocorreu uma baixa taxa de identificação no treinamento.

Tabela 6.5 - Matriz de confusão do conjunto de teste - amostragem cronológica

		Inferido				
		Classe 1 (HH)	Classe 2 (H)	Classe 3 (M)	Classe 4 (L)	Classe 5 (LL)
Observado	Classe 1 (HH)	4302	556	130	1	0
	Classe 2 (H)	647	224	79	0	0
	Classe 3 (M)	26	10	2	0	0
	Classe 4 (L)	10	2	2	0	0
	Classe 5 (LL)	5	3	1	0	0

6.3 Classificação com amostragem alternada

Os resultados obtidos com dados de treinamento com os anos 2010–2013 e teste com dados de 2014 não foram satisfatórios. Por esta razão, optou-se por uma amostragem chamada de alternada, onde três amostras consecutivas são alocadas no conjunto de treinamento e a quarta no conjunto de teste.

Nesta seção, são apresentadas as classificações realizadas com os dados de simulações na abordagem alternada; os dados nas classes estão distribuídos como mostra a Tabela 6.6. Abaixo, apresentamos primeiramente as classificações obtidas com redes neuro-difusas e em seguida, aquelas obtidas com RNAs, classificador hierárquico e máquina de comitê.

Tabela 6.6 - Distribuição de amostras na amostragem alternada

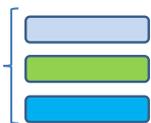
Classe	1 (HH)	2 (H)	3 (M)	4 (L)	5(LL)	Total
Treinamento	11425	5337	928	272	38	18000
Teste	3875	1704	322	86	13	6000

O particionamento do conjunto de dados de entrada-saída está ilustrado na Figura 6.9, com os conjuntos de entrada e saída representado por cores. As n variáveis de entrada ou atributos formam as primeiras n colunas da tabela e última é a saída.

Figura 6.9 - Esquema de geração dos dados de treinamento e teste para misturar os valores dos anos 2011 até 2014.

Var 1	Var 2	Var 3	...	Var n	Saída
0.22	0.24	0.22.	...	0.21	2
0.24	0.25	0.31		0.2	2
0.24	0.021	0.21		0.012	1
0.23	0.21	0.22		0.14	3
0.25	0.11	0.22		0.054	5
0.21	0.21	0.246		0.014	4
0.21	0.245	0.145		.02146	2
0.31	0.12	0.15		0.12	1
.					
.					
.					
0.33	0.122	0.105		0.212	2

Treino=



Teste =



6.3.1 Classificação utilizando uma rede Neuro-Difusa

Utilizando-se a rede neuro-difusa GUAJE, obteve-se um total de 9.629 regras. As Tabelas 6.7 e 6.8 trazem as matrizes de confusão de treinamento e teste, respectivamente.

A acurácia global da classificação do conjunto de treinamento foi de 87,7%, com um total de 2219 erros de classificação, e a do conjunto de teste foi de 76,4%. Nota-se nas tabelas que houve uma cobertura completa de dados no treinamento e uma boa cobertura no conjunto de teste.

Nota-se que as classes com baixa frequência de dados de teste tiveram um bom desempenho; por exemplo somente 12 das 38 amostras foram classificadas erroneamente na classe 5 (LL), o que representa uma sensibilidade de 68,4%. Tem-se um caso de erro por ambiguidade (AC) (ver apêndice B) para uma amostra correspondente a classe 1 (HH) no qual a rede gera valores de pertinência iguais para distintas conclusões de regras, o que a impossibilita a classificação correta dessa amostras. Um resultado importante é que não houveram amostras sem classificação no conjunto de testes, o que indica que os padrões de treinamento também estão presentes no conjunto de teste e que a rede conseguiu identificar corretamente os valores de cada entrada.

Dos resultados mostrados na matriz de confusão do teste, podemos destacar a precisão de cada classe. A classe 5 (HH) obteve um 84,99%, a classe 4 (H) teve uma precisão de 63,27%, a classe 3 (M) teve 50,19%, a classe 4 (L) 47,56% e finalmente a classe 5 (LL) teve uma precisão de 50%. Se destaca o valor da ultima classe, que mesmo com poucas amostras obteve uma inferência boa, mas se confunde muito com a classe 1 (HH).

Tabela 6.7 - Matriz de confusão do conjunto de treinamento - amostragem alternada

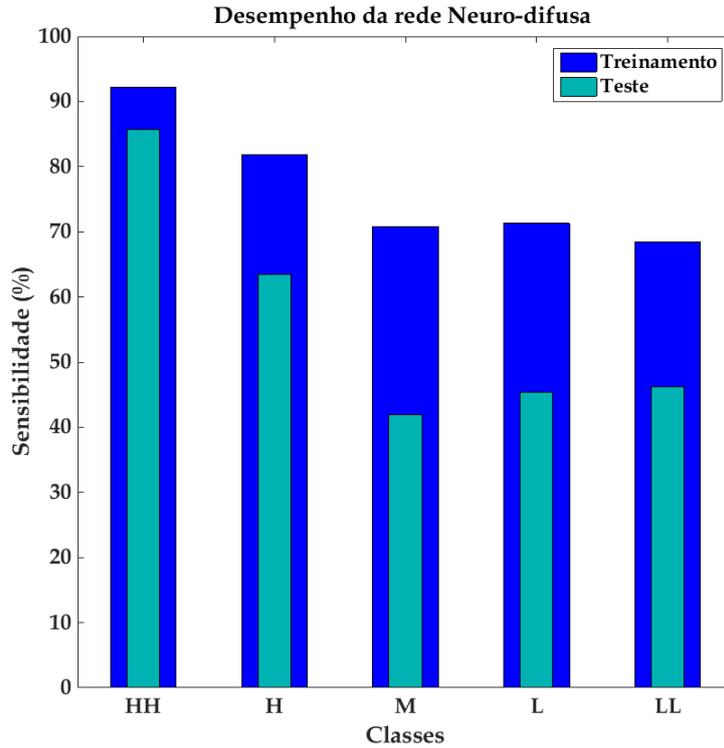
		Inferido					Sem classe
		Classe 1 (HH)	Classe 2 (H)	Classe 3 (M)	Classe 4 (L)	Classe 5 (LL)	
Observado	Classe 1 (HH)	10534	806	72	10	2	1
	Classe 2 (H)	817	4369	122	27	2	0
	Classe 3 (M)	84	147	657	38	2	0
	Classe 4 (L)	24	20	31	194	3	0
	Classe 5 (LL)	2	3	3	4	26	0

Tabela 6.8 - Matriz de confusão do conjunto de teste - amostragem alternada

		Inferido					Sem Classe
		Classe 1 (HH)	Classe 2 (H)	Classe 3 (M)	Classe 4 (L)	Classe 5 (LL)	
Observado	Classe 1 (HH)	3323	506	32	7	4	3
	Classe 2 (H)	523	1082	80	16	1	2
	Classe 3 (M)	63	104	135	20	0	0
	Classe 4 (L)	9	15	22	39	1	0
	Classe 5 (LL)	2	3	0	1	6	1

A Figura 6.10 mostra o desempenho da rede na sensibilidade para os conjuntos de treinamento e teste. Note-se que o algoritmo apresenta um desempenho muito bom com os dados de treinamento mas com os dados de teste, o desempenho cai e não consegue acompanhar o resultado do treinamento, gerando problemas de generalização nas classes com frequências baixas.

Figura 6.10 - Comparação de performance de sensibilidade de classificação como rede Neuro-difusas



6.3.2 Classificação com uma Rede Neural

A classificação com uma RNA obteve uma acurácia global de 88,04% para o conjunto de treinamento e de 83,55% para o conjunto teste (vide Tabelas 6.9 e 6.10).

Tabela 6.9 - Matriz de confusão do conjunto de treinamento - amostragem alternada

		Inferido				
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)
Observado	Classe 1(HH)	10845	548	32	0	0
	Classe 2(H)	772	4481	83	1	0
	Classe 3(M)	67	373	461	27	0
	Classe 4(L)	26	67	108	71	0
	Classe 5(LL)	5	9	11	13	0

A Tabela 6.9 mostra que a precisão para classes de alta frequência de dados de treinamento é alta. Por exemplo, a classe 1 (HH) acertou 10.845 em 11.715 (92,57%), a classe 2 (H) 4.481 em 5.478 (81,80%), a classe 3 (M) acertou 461 de 695 (66,33%), a classe 2 (L) teve uma precisão de 71 em 112 (63,39%) e a classe 1 (LL) não teve nenhuma amostra inferida.

Os resultados do teste se visualizam na matriz 6.10 e são similares aos comentados no parágrafo anterior. O desempenho das classes de alta frequência de dados é muito bom, a precisão das classes tem taxas de acertos de 89,76% para a classe 1 (HH) e de 73,28% para a classe 2 (H), enquanto que para as classes com menos frequência, tem-se uma taxa de 55% para a classe 2 (L) e erro total na última classe.

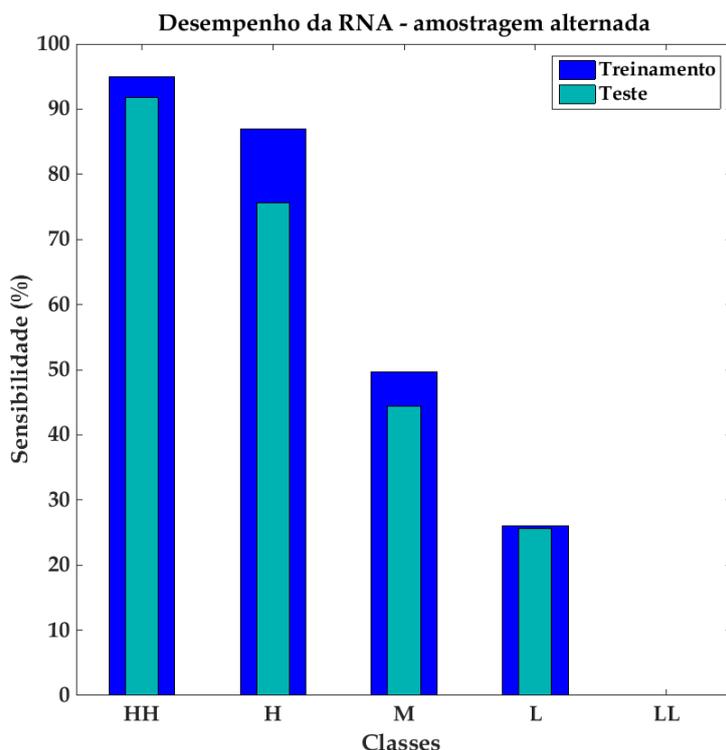
Por outro lado, as taxas de sensibilidade de cada classe mostram um comportamento bom para as classes de alta frequência, como a classe 1 (HH) tem um 91,85% e a classe 2(H) tem um 76,65%, no entanto que a classe de baixa frequência de dados como a classe 2 (L) teve uma taxa de sensibilidade de 25,58%, sendo uma baixa taxa de identificação para essa classe. A classe 1(LL) não teve nenhuma amostra identificada, o que se esperava dado o resultado no treinamento.

Tabela 6.10 - Matriz de confusão do conjunto de teste da amostragem alternada

		Inferido				
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)
Observado	Classe 1(HH)	3559	310	4	2	0
	Classe 2(H)	363	1289	50	2	0
	Classe 3(M)	29	141	143	9	0
	Classe 4(L)	12	15	37	22	0
	Classe 5(LL)	2	4	2	5	0

A Figura 6.11 mostra a performance de sensibilidade, para os conjuntos de treinamento e teste, em que o algoritmo é muito bom tanto no treinamento e teste para as classes com muita frequência de amostras mas devido ao desbalanceamento, as classes com frequências baixas de amostras não conseguem emular o desempenho das primeiras classes.

Figura 6.11 - Comparação de performance de sensibilidade de classificação com RNA



6.3.3 Classificação utilizando uma árvore hierárquica binária

Um classificador neural binário e hierárquico foi implementado (ver seção 6.3.3) utilizando os conjuntos de dados alternados.

6.3.4 Fase 1: Classe HH vs \sim HH

Nesta fase teve-se uma acurácia global de 92,3% no treinamento e 88,9% de acurácia no teste.

Tabela 6.11 - Matriz de Confusão do conjunto de treinamento, fase HH vs \sim HH.

		Inferido		Total
		(\sim HH)	(HH)	
Observado	(\sim HH)	5746	829	6.575
	(HH)	549	10.876	11.425
Total		6.294	11.705	18000

Visualizando os resultados da Tabela 6.11, vemos que tem uma sensibilidade de

95,19% (10876/11425) para distinguir a classe HH do resto das classes e uma precisão de 87,39% para a classe \sim HH. De toda a saída inferida como classe \sim HH (total 6575), um acerto de 87,4% a classe foi registrado.

Tabela 6.12 - Matriz de Confusão do conjunto de teste, fase HH vs \sim HH.

		Inferido		Total
		(\sim HH)	(HH)	
Observado	(\sim HH)	1750	375	2.125
	(HH)	290	3.585	3.875
Total		2.040	3.960	6000

Para a fase de teste (vide Tabela 6.12), tem-se 6.000 amostras e o classificador apresentou bom desempenho: acurácia global de 88,9%, onde o total de acertos como classe HH foi de 92,51% e para a classe \sim HH teve um acerto 82,36%.

6.3.4.1 Fase 2: Classe H vs \sim H

Para a **segunda fase** do classificador hierárquico, o objetivo é discriminar a classe H do restante (M, L e LL). Nesta etapa, são removidas as amostras correspondentes à classe HH dos conjuntos de treinamento e teste. O conjunto de treinamento permanece com 6.575 amostras e o de teste com 2.125 amostras.

Tabela 6.13 - Matriz de Confusão do conjunto de treinamento, fase H vs \sim H.

		Inferido		Total
		(\sim H)	(H)	
Observado	(\sim H)	1014	224	1238
	(H)	117	5.220	5.337
Total		1131	5.444	6575

A rede neural teve uma acurácia global de 94,8% no treinamento e de 90,1% no teste. No treinamento, a rede teve uma precisão ou taxa de acerto de 97,8% na classe H e um 81,9% para a classe \sim H (vide Tabela 6.13).

Tabela 6.14 - Matriz de Confusão do conjunto de teste, fase H vs \sim H.

		Inferido		Total
		(\sim H)	(H)	
Observado	(\sim H)	305	116	421
	(H)	94	1.610	1.704
Total		399	1.704	2.125

O teste é executado com conjunto de 2.125 amostras, com acurácia global de 90,1% e sensibilidade de $1.610/1.704 = 94,48\%$ para a classe H e de 76,45% para a classe \sim H (vide Tabela 6.14).

6.3.4.2 Fase 3: Classe M vs \sim M

O objetivo da **terceira fase** é discriminar a classe M das restantes (L e LL). Amostras correspondentes à classe H e HH são removidas dos conjuntos de treinamento e teste. O conjunto de treinamento fica com 1.238 amostras e o teste com 421 amostras.

Tabela 6.15 - Matriz de Confusão do conjunto de treinamento, fase M vs \sim M

		Inferido		Total
		(\sim M)	(M)	
Observado	(\sim M)	188	122	310
	(M)	28	900	928
Total		216	1.022	1.238

A acurácia global de treinamento é de 87,9% e no teste é de 85,7%. No treinamento a precisão para a classe M é de 88,06% e de 87,0% para a classe \sim M (vide Tabela 6.15).

Tabela 6.16 - Matriz de Confusão do conjunto de teste, fase M vs \sim M.

		Inferido		Total
		(\sim M)	(M)	
Observado	(\sim M)	52	47	99
	(M)	13	309	322
Total		65	356	421

No teste, a precisão da classe M foi de 86,8% e da classe \sim M foi de 80% (vide Tabela 6.16).

6.3.4.3 Fase 4: Classe L vs \sim L

Para a última, foram removidas amostras das classes HH, H e M e o conjunto de treinamento e de teste se reduzem a 310 e 99 amostras, respectivamente. Basicamente, nesta fase temos uma classificação entre as classes L e LL.

Tabela 6.17 - Matriz de Confusão do conjunto de treinamento, fase L vs LL.

		Inferido		Total
		(LL)	(L)	
Observado	(LL)	20	18	38
	(L)	2	270	272
Total		22	272	310

A pesar da pouca quantidade de dados, o treinamento teve uma acurácia global de 93,5% e no teste uma precisão de 87,9%. No treinamento das classes, a classe L teve taxa de acerto de 93,8% e a classe LL acerto de 90,9% (vide Tabela 6.17). Resultados para o conjunto de teste tiveram uma acurácia global de 87,9%, com precisão de 89,4% para a classe L e de 60,0% para a classe LL (vide Tabela 6.18).

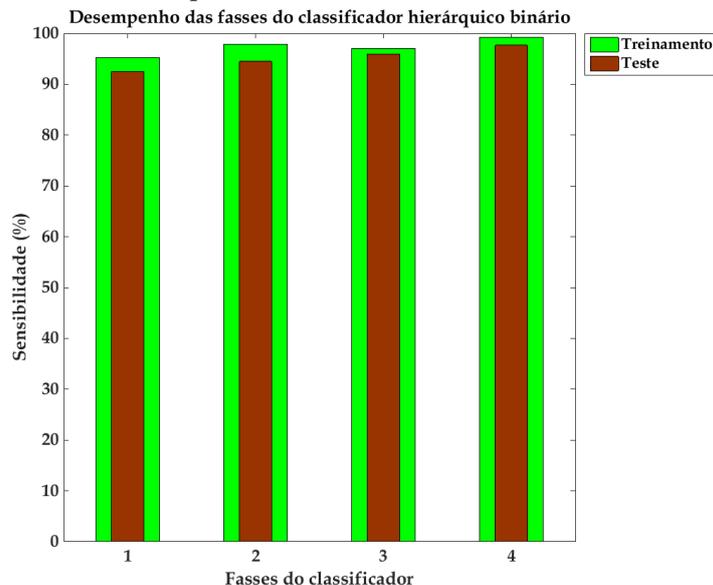
Tabela 6.18 - Matriz de Confusão do conjunto de teste, fase L vs LL.

		Inferido		Total
		(LL)	(L)	
Observado	(LL)	3	10	13
	(L)	2	84	86
Total		5	86	99

6.3.4.4 Desempenho das fases no treinamento e teste - resumo

Cada fase possui seu respectivo conjunto de treinamento e teste. Todas as fases apresentam um desempenho muito bom na sensibilidade da classe, isto é, consegue identificar bem a classe que tenta discriminar na fase correspondente. A Figura 6.12 resume as porcentagens das distintas fases cujo objetivo é discriminar uma classe única.

Figura 6.12 - Desempenho de cada fase do classificador hierárquico



6.3.4.5 Cálculo da taxa global de classificação do algoritmo hierárquico binário

Como cada fase do algoritmo possui um classificador especializado em discriminar uma classe em particular do resto com que foi treinado (*one against all*), o erro se acumula conforme se avança nas fases. Por exemplo, na primeira fase o classificador discrimina a classe HH do restante – classe \sim HH. O conjunto \sim HH é utilizado na sequência para o classificador de segundo nível. Novamente, as amostras que foram classificadas como classe \sim H são utilizadas para terceira fase e assim sucessivamente até a última fase, onde se discrimina entre as classes L e LL. Amostras que não foram classificadas nas classes HH, H, M e L são etiquetadas como classes LL.

Portanto, para calcular a *acurácia final global* do algoritmo são contadas as classificações bem sucedidas, obtendo-se um resultado de classificação de 85,5% no conjunto de treinamento e de de 81,3% no conjunto de teste.

A Tabela 6.19 mostra o resumo da classificação para cada classe, como o algoritmo atua em sequência, cada saída que o algoritmo identifica como classe $\sim C_i$ é utilizado para testar a seguinte fase. No último, o algoritmo etiqueta o resto como classe LL.

Tabela 6.19 - Desempenho da sensibilidade das classes no teste global do classificador hierárquico binário

Classe	1 (HH)	2 (H)	3 (M)	4 (L)	5 (LL)
Treinamento	95,19	82,91	5,49	0	100
Teste	92,51	74,29	4,34	0	100

6.3.5 Comparação dos classificadores

As Figuras 6.13(a,b,c) mostram uma comparação de desempenho de classificação para cada classe e o algoritmo utilizado. Cada barra representa o desempenho em percentagem da sensibilidade de cada classe (a percentagem de identificação correta), onde a barra maior são dados do treinamento e a menor é para o teste.

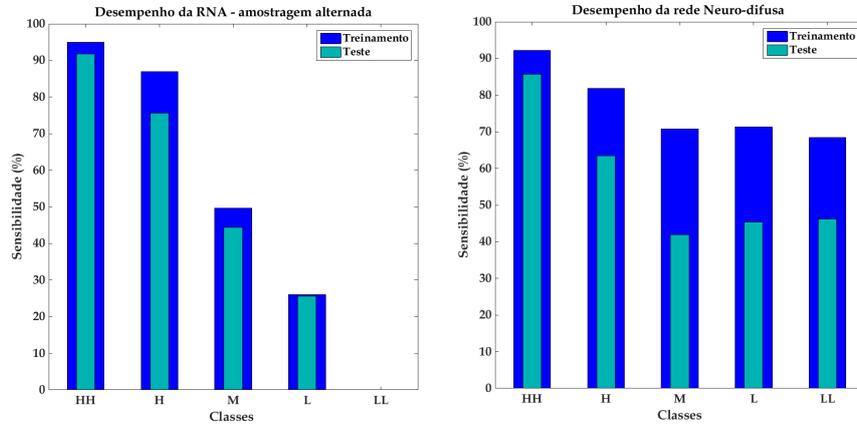
Podemos verificar que o desempenho para as primeiras classes (HH e H) é muito similar na fase de treinamento. Com o conjunto de teste o desempenho das redes já apresentam diferenças, a rede Neuro-difusa tem um menor desempenho que as outras redes nessas classes.

A classe intermédia M no treinamento tem um bom desempenho na rede Neuro-difusa mas nas outras redes não, especialmente o classificador hierárquico (5,49%). Já na fase de teste, a rede neural consegue tem um desempenho similar ao treinamento, a neuro-difusa tem uma queda significativa no treinamento e o classificador hierárquico o desempenho é defeituoso como se esperava.

As últimas classes (L e LL) apresentam um contraste marcado pelo desempenho do classificador hierárquico. A rede neuro-difusa tem um bom desempenho no treinamento mas no teste não se consegue a generalização. Na rede neural o desempenho é péssimo em especial na ultima classe onde o classificador não consegue inferir nenhuma amostra no treinamento e por consequencia também no teste.

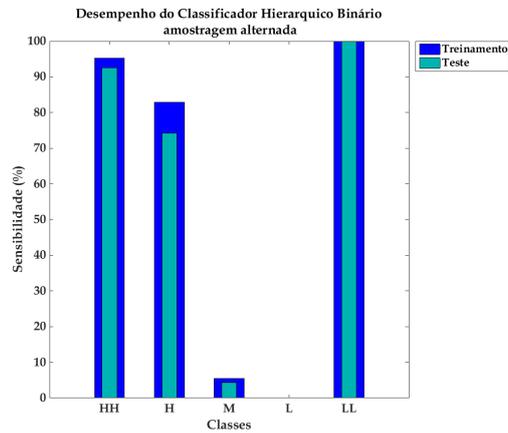
Com o classificador hierárquico a performance arrastada pela fase anterior faz que o classificador não consiga inferir uma amostra para a classe L, o que pela estrutura projetada, classifica todas as amostras restantes como a classe LL tendo assim um excelente desempenho para esta classe.

Figura 6.13 - Comparação de performance de sensibilidade de classificação das redes Neuro-difusas, RNA e o classificador Hierárquico para o conjunto de dados alternados



(a) RNA

(b) Neuro-difuso



(c) Hierárquico

6.3.6 Método de Comitê

Neste projeto, foram implementadas versões do modelo de máquinas estáticas e dinâmicas, no qual cada componente (tanto especialista como integrador) é uma rede neural. Nos primeiros experimentos denominados grupais, as redes especialistas foram treinadas com a estratégia *One-against-all*, onde cada componente tenta identificar uma classe particular em relação às outras quatro. Em outro tipo de experimento, denominado dual, o objetivo foi treinar os especialistas para separar duas classes. Portanto, considerando um conjunto de 5 classes, temos um total de 5 especialistas nos experimentos globais e 10 nos experimentos globais (combinação C_5^2).

Em ambas as fases, a dos especialistas e a do integrador, foram utilizadas redes neurais MLP com 4 camadas intermediárias de [100, 60, 40, 20] neurônios, respectivamente.

No experimento grupal, a entrada de cada uma das 5 redes especialistas (uma para cada classe) é um conjunto de amostras com 25 atributos de entrada. A camada de saída de cada rede neural especializada tem 2 neurônios (um para “pertence à classe” e outro para “não pertence à classe”). Já a rede do integrador tem 5 neurônios na camada de entrada (o resultado de cada rede especialista), seguindo a estratégia *One-hot encoding*.

No experimento dual, a entrada de cada uma das 10 redes especialistas (uma para cada par de classes) é a mesma dos experimentos grupais, e a camada de saída tem 2 neurônios (um para “pertence à primeira classe” e outro para “pertence à segunda classe”). Como o comitê é dinâmico, a rede integradora possui 45 neurônios de entrada (25 correspondentes à entrada e 20 correspondentes aos especialistas), com 5 neurônios na camada de saída.

6.3.6.1 Conjuntos de treinamento e teste para o método de comitê - experimento grupal

No experimento global, cada rede especialista visa detectar uma única classe, e tem seus próprios conjuntos de treinamento e teste, formados por 75% e 25% dos dados, respectivamente. No que segue, denotamos por $\sim i$ a classe complementar de i no conjunto de classes C ($\sim i = C - \{i\}$). Por exemplo, para classe 3 com $C = \{1, 2, 3, 4, 5\}$, a classe ~ 3 denota classe composta, formada pelo subconjunto de classes $\{1, 2, 4, 5\}$.

Para evitar o desbalanceamento nas classes $\sim i$, foi aplicada a estratégia *underbalance*. Por exemplo, para formar o conjunto de treinamento/teste do especialista 3, devemos ter amostras tanto da classes 3 quanto da classe composta ~ 3 . Se a classe 3 possui 10.000 amostras, 7.500 dados são usados para treinamento e 2.500 para teste. A fim de que os conjuntos de treinamento/teste de ~ 3 seja balanceado em relação às classes que o compõem, realiza-se uma amostragem aleatória do conjunto de dados em ~ 3 , escolhendo 7.500 amostras para o treinamento e 2.500 para o teste.

Já para treinamento e teste da rede integradora, realiza-se um novo processo de balanceamento com *undersampling*, utilizando como dados de entrada as saídas inferidas pelas redes especialistas. São utilizadas 75% das amostras da classe com

menor quantidade de dados e o restante é balanceado com a mesma quantidade de dados. Para o teste, utiliza-se o mesmo procedimento, mas com 25% das amostras correspondentes.

Na estratégia *one-hot* utilizada para codificar a saída desejada dos conjuntos de treinamento e teste das redes especialistas, a camada de saída da rede neural especialista está configurada com 2 neurônios (um neurônio para a classe i e o outro para a classe $\sim i$). Em cada um desses neurônios, a saída assume o valor 1 se for considerada como a classe correta, O neurônio com maior valor de saída determina a classe correspondente inferida pela rede. Neste trabalho, foi implementada a função *Argmax* como função decisora, indicando o valor de 1 se o neurônio correspondente à classe i resulta com maior valor de inferência e 0 no caso contrário. No caso de empate, a resposta inferida é de 0 (classe $\sim i$).

As saídas inferidas pelas redes especialistas, após passar por seus respectivos decisores, formam uma quintupla de 0s e 1s, onde a posição dos valores na tupla correspondem a cada rede especialista. Por exemplo, a quintupla [0 1 0 0 1] indica que os especialistas 2 e 5 consideram que a amostra pertence a suas respectivas classes.

Ambos os conjuntos de treinamento e teste da RNA integradora são formados por quintuplas tanto na entrada quanto na saída desejada, sendo que nesta última se utiliza a codificação *one-hot*.

Para esta fase, foram efetuadas mais simulações de processos de *Breeding*, realizadas entre 02 de maio as 00 Hs e 07 de maio às 00 Hs, com 2 processo de *Breeding* por dia (00 Hs e 12 Hs.), por 3 anos (2011-2012-2013).

São gerados 11 processos de *Breeding* por ano, portanto. Como cada processo gera um total de 6.000 amostras para a janela de estudo deste projeto, obtém-se 66.000 amostras por ano. Considerando-se as amostras de 3 anos, chega-se pois a um universo de 198.000 amostras.

Além de gerar mais processos de *Breeding*, foram considerados mais atributos de entrada que nas fases anteriores (redes neurais, neuro-difusa e hierarquico), num total de 25 atributos de entrada (ao invés dos 16 atributos considerados nos testes anteriores). Foram adicionados os *bred vectors* da precipitação, vorticidade e função corrente e valores das suas previsões de controle, além das longitudes e latitudes como informação espacial e o *time-step* da simulação como informação temporal.

A Figura 6.14(a) mostra a distribuição do desvio padrão da temperatura do *Ensemble*, utilizando o histograma de frequências, com um total de 198.000 dados. Foram retirados os dados correspondentes desde o mínimo até o percentil 1 e desde o percentil 99 até o máximo, formando o conjunto de dados da Figura 6.14(b). Neste experimento, para gerar um esquema de classes mais homogêneo, foram considerados para as classes de saída somente os dados desde os percentis 1 até 99. A Figura 6.15(a) mostra as classes correspondentes a este novo conjunto. Com este processo as classes L e LL não ficaram com tão poucas amostras. A Figura 6.15(b) mostra a divisão das classes, considerando-se somente este conjunto de dados (percentis 1 a 99).

Figura 6.14 - Histograma da magnitude do desvio padrão e o conjunto de dados considerado para geração das classes de confiabilidade

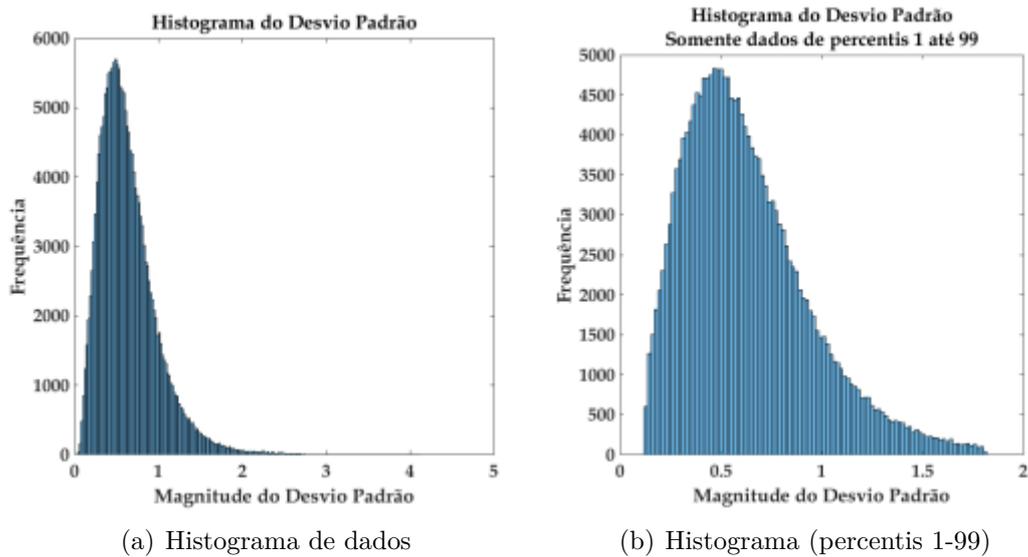
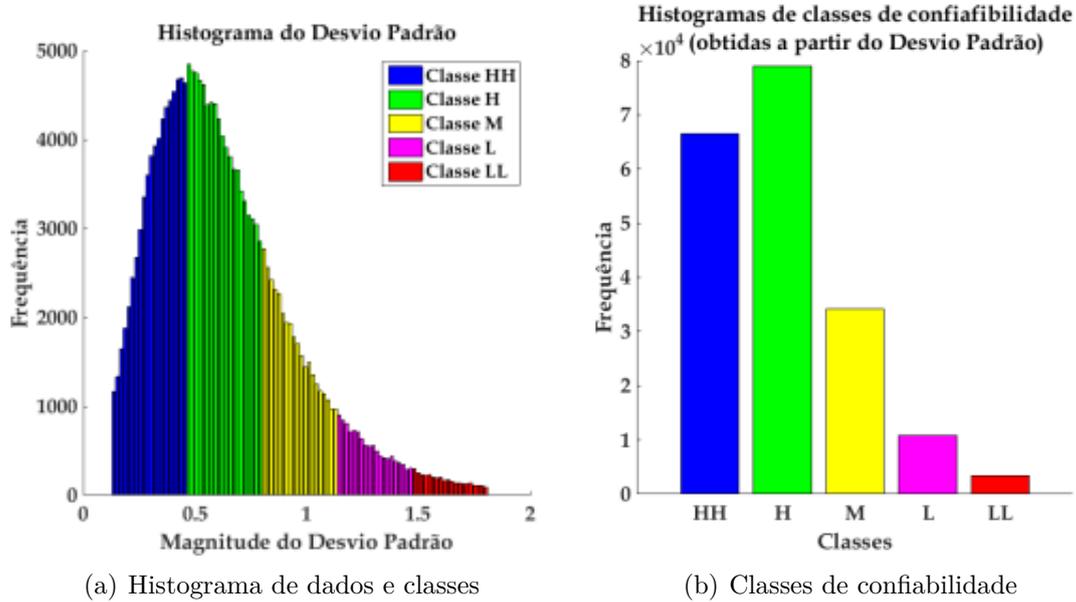


Figura 6.15 - Histograma da magnitude do desvio padrão e as classes de confiabilidade gerada a partir desses dados



Particionando o domínio do desvio padrão e gerando-se as classes, são obtidas 66.550 amostras para a classe 1 (HH), 79.053 amostras para a classe 2 (H), 34.149 amostras para a classe 3 (M), 10.884 amostras para a classe 4 (L) e 3.404 amostras para a classe 5 (LL). Deste grupo de dados, são escolhidos os conjuntos de treinamento e teste como se mostra na Tabela 6.20. Por exemplo, para formar o conjunto de treinamento para a rede especialista 1 (HH), se tomaram 49.912 (75%) amostras da classe 1(HH) e 49.912 amostras do resto (classe ~ 1) totalizando assim 99.824 amostras.

Tabela 6.20 - Distribuição de amostras para os conjuntos de treinamento-teste para as redes especialistas

Especialista	1 (HH)	2 (H)	3 (M)	4 (L)	5(LL)	Total
Treinamento	99.824	118.580	51.224	16.326	5.106	291.060
Teste	33.276	39.526	17.074	5.442	1.702	97.020

6.3.6.2 Matrizes de confusão das distintas redes especialistas do método de comitê - Experimento grupal

As matrizes seguintes mostram o desempenho do treinamento-teste de cada rede especialista, que visa distinguir uma classe do resto.

Tabela 6.21 - Matriz de confusão treinamento da rede especialista Classe 1 (HH)

		Obs.		(%)	
		($\sim HH$)	(HH)		
Inf.	($\sim HH$)	42.043	7.472	84,9	
	(HH)	7.869	42.440	84,4	
		(%)	84,4	85	84,6

Tabela 6.22 - Matriz de confusão teste - rede especialista classe 1 (HH)

		Obs.		(%)	
		($\sim HH$)	(HH)		
Inf.	($\sim HH$)	13.132	3.377	79,5	
	(HH)	3.506	13.261	79,1	
		(%)	78,9	79,1	79,3

Tabela 6.23 - Matriz de confusão treinamento da rede especialista Classe 2 (H)

		Obs.		(%)	
		($\sim H$)	(H)		
Inf.	($\sim HH$)	44.497	10.283	81,3	
	(HH)	14.493	49.007	77,2	
		(%)	75,6	82,7	79,1

Tabela 6.24 - Matriz confusão de teste da rede especialista Classe 2 (H)

		Obs.		(%)	
		($\sim H$)	(H)		
Inf.	($\sim H$)	13.890	4.345	76,2	
	(H)	5.873	15.418	72,4	
		(%)	70,3	78	74,1

Tabela 6.25 - Matriz de confusão treinamento da rede especialista Classe 3 (M)

		Obs.		(%)	
		($\sim M$)	(M)		
Inf.	($\sim M$)	24.459	1.800	93,1	
	(M)	1.153	23.812	95,4	
		(%)	95,5	93	94,2

Tabela 6.26 - Matriz de confusão teste da rede especialista Classe 3 (M)

		Obs.		(%)	
		($\sim M$)	(M)		
Inf.	($\sim M$)	8.109	667	92,4	
	(M)	428	7.870	94,8	
		(%)	95	92,2	93,6

Tabela 6.27 - Matriz de confusão treinamento da rede especialista Classe 4 (L)

		Obs.		(%)	
		($\sim L$)	(L)		
Inf.	($\sim L$)	8.125	94	98,4	
	(L)	38	8.069	99,5	
		(%)	99,5	98,8	99,2

Tabela 6.28 - Matriz de confusão teste da rede especialista Classe 4 (L)

		Obs.		(%)	
		($\sim L$)	(L)		
Inf.	($\sim L$)	2.691	62	97,7	
	(L)	30	2.659	98,9	
		(%)	98,9	97,7	98,3

Tabela 6.29 - Matriz de confusão treinamento da rede especialista Classe 5 (LL)

		Obs.		(%)
		($\sim LL$)	(LL)	
Inf.	($\sim LL$)	2.549	10	99,6
	(LL)	4	2.543	99,8
(%)		99,8	99,6	99,7

Tabela 6.30 - Matriz confusão teste da rede especialista Classe 5 (LL)

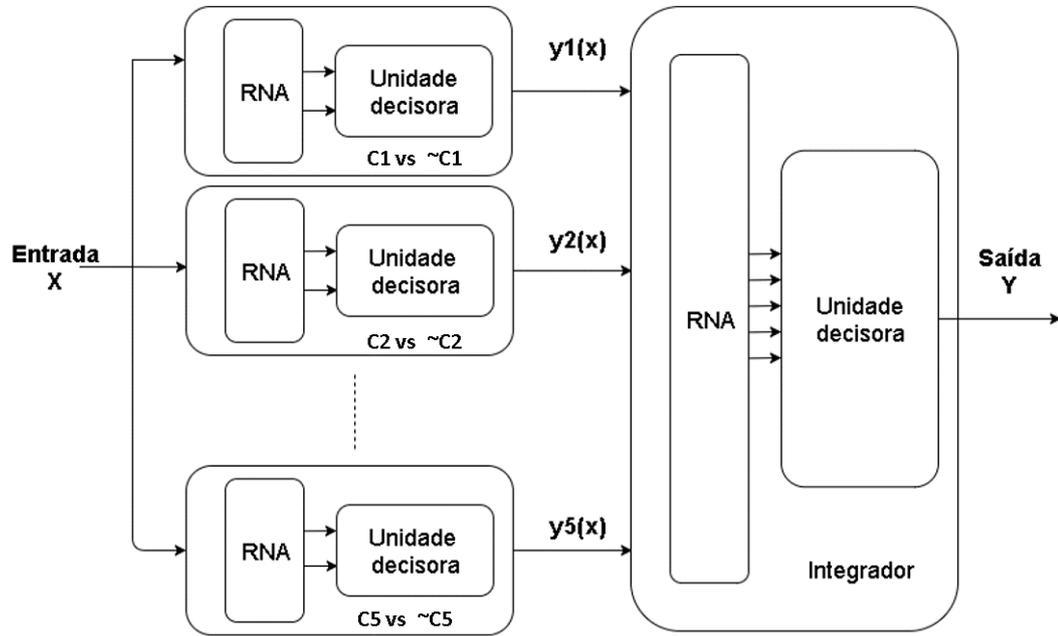
		Obs.		(%)
		($\sim LL$)	(LL)	
Inf.	($\sim LL$)	849	21	97,6
	(LL)	2	830	99,8
(%)		99,8	97,5	98,6

6.3.6.3 Estratégias de treinamento-teste da rede integradora - experimento grupal

Foram implementadas várias estratégias para a rede integradora, visando obter melhores resultados de classificação. Primeiro, foi testado uma rede integradora estática com unidade decisora (vide Figura 6.16). Esta unidade consiste na função *Argmax*, que utiliza como entrada as saídas da rede especialista e codifica como 1 se o neurônio correspondente à classe i da rede i tem valor máximo ou 0 no caso contrário. Em caso de empate, a função *Argmax* escolhe pela classe $\sim i$. As Tabelas 6.31 e 6.32 trazem os resultados da classificação.

Para treinar e testar a rede integradora, foi criado novos conjuntos de treinamento e teste, tomando-se um sub-conjunto aleatório de cada classe, e balanceando as classes com *undersampling*, tendo como referência a classe com menos amostras. Foram tomadas 2553 (75% da classe 5) amostras de cada classe para o conjunto treinamento e 851 (25% da classe 5) amostras de cada classe para o conjunto de teste.

Figura 6.16 - Comitê estático implementado, com unidade decisora



Fonte: Produção do autor.

Tabela 6.31 - Matriz de confusão do conjunto de treinamento da RNA integradora

		Inferido					%
		Classe 1 (HH)	Classe 2 (H)	Classe 3 (M)	Classe 4 (L)	Classe 5 (LL)	
Observado	Classe 1(HH)	1774	593	72	36	78	69,5
	Classe 2(H)	199	1773	104	7	470	69,4
	Classe 3(M)	62	959	244	31	1257	9,6
	Classe 4(L)	35	253	88	62	2115	2,4
	Classe 5(LL)	41	127	57	49	2279	89,3
(%)		84	43,2	33,5	36,8	36,8	48,0

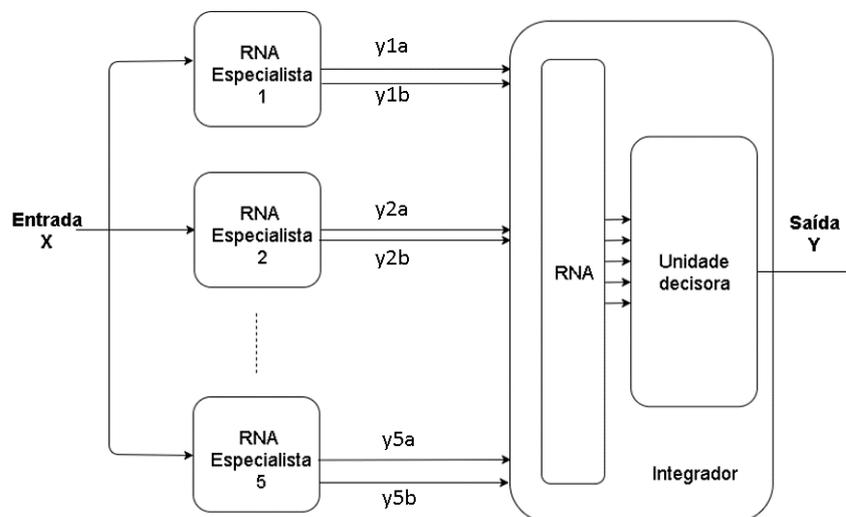
Tabela 6.32 - Matriz de confusão do conjunto de teste da RNA integradora

		Inferido					(%)
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)	
Observado	Classe 1(HH)	590	210	16	13	22	69,3
	Classe 2(H)	65	574	33	6	173	67,5
	Classe 3(M)	18	334	70	9	423	8,2
	Classe 4(L)	15	77	22	23	714	2,7
	Classe 5(LL)	15	36	18	20	762	89,5
(%)		84,3	46,6	44	32,4	36,4	47,5

As matrizes dos sistemas especialistas (Tabelas 6.21 até 6.30) mostram que o desempenho destas redes é muito boa, conseguindo distinguir uma classe das demais. Contudo, quando essa informação é utilizada em conjunto através da quintupla, a mistura dos padrões confunde a RNA integradora (vide matrizes 6.31 e 6.32). Os padrões da classe 1 (HH) tem classificação mais efetiva do que as das outras classes, mesmo tendo um conjunto balanceado de treinamento e teste.

No segundo caso, a fase decisora foi retirada, deixando essa tarefa à rede integradora. Portanto, a rede tem 10 neurônios na camada de entrada com 2 neurônios (classe i e classe $\sim i$) por cada rede especialista - vide Figura 6.17. Os resultados da classificação se encontram nas Tabelas 6.33 e 6.34.

Figura 6.17 - Comitê estático implementado, sem unidade decisora



Fonte: Produção do autor.

Tabela 6.33 - Matriz de confusão do conjunto de treinamento da RNA integradora sem unidade decisora

		Inferido					(%)
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)	
Observado	Classe 1(HH)	2061	312	72	16	92	80,7
	Classe 2(H)	304	1205	407	145	432	47,2
	Classe 3(M)	141	478	889	181	864	34,8
	Classe 4(L)	132	287	786	200	1148	7,8
	Classe 5(LL)	137	190	584	174	1468	57,5
(%)		74,3	48,7	31,8	27,9	36,7	45,6

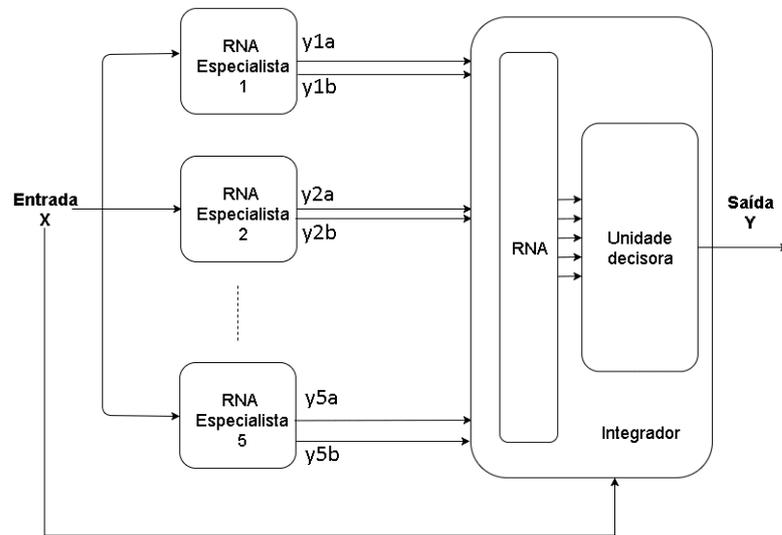
Tabela 6.34 - Matriz de confusão do conjunto de teste da RNA integradora sem unidade decisora

		Inferido					(%)
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)	
Observado	Classe 1(HH)	676	108	24	7	36	79,4
	Classe 2(H)	109	381	175	42	144	44,8
	Classe 3(M)	39	172	296	55	289	34,8
	Classe 4(L)	46	89	233	79	404	9,3
	Classe 5(LL)	55	69	209	56	460	54,1
(%)		73,1	46,5	31,6	32,8	34,5	44,5

Os resultados da classificação evidenciam que o uso de unidade decisora favorece à classificação mas os resultados não se alteram significativamente e continuam sendo insatisfatórios.

Um terceiro teste foi realizado utilizando uma máquina de comitê do tipo dinâmico, no qual para a entrada da rede integradora são consideradas não somente as informações fornecidas pelas redes especialistas, mas também as próprias entradas destas redes. A implementação pode-se visualizar na Figura 6.18 e os resultados da classificação se encontram nas Tabelas 6.35 e 6.36.

Figura 6.18 - Comitê dinâmico implementado, sem unidade decisora na fase especialista



Fonte: Produção do autor.

Tabela 6.35 - Matriz de confusão do conjunto de treinamento da RNA integradora dinâmica

		Inferido					(%)
		Classe 1 (HH)	Classe 2 (H)	Classe 3 (M)	Classe 4 (L)	Classe 5 (LL)	
Observado	Classe 1(HH)	2093	276	87	36	61	82
	Classe 2(H)	315	1327	443	224	244	52
	Classe 3(M)	116	495	930	549	463	36,4
	Classe 4(L)	59	226	477	836	955	32,7
	Classe 5(LL)	44	100	194	437	1778	69,6
(%)		79,7	54,7	43,6	40,2	50,8	54,6

Tabela 6.36 - Matriz de confusão do conjunto de teste da RNA integradora dinâmica

		Inferido					(%)
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)	
Observado	Classe 1(HH)	679	112	27	9	24	79,8
	Classe 2(H)	109	424	158	78	82	49,8
	Classe 3(M)	35	184	264	212	156	31
	Classe 4(L)	31	82	151	251	336	29,5
	Classe 5(LL)	20	33	69	154	575	67,6
(%)		77,7	50,8	39,4	35,7	49	51,5

O uso de uma máquina de comitê dinâmica levou a uma melhora no treinamento e teste, com uma acurácia global no treinamento passando de 48% com a máquina estática para 54,6% com a máquina dinâmica. Também houve melhora na sensibilidade das classes 1 e 5, com um 82% de identificação correta das amostras da classe 1 (HH) no treinamento e um desempenho similar (79,8%) no teste. A classe 5 (LL) que é a classe com menor número de amostras, também obteve um desempenho bastante satisfatório de identificação correta de amostras com um 69,6% no treinamento e um 67.5% no teste.

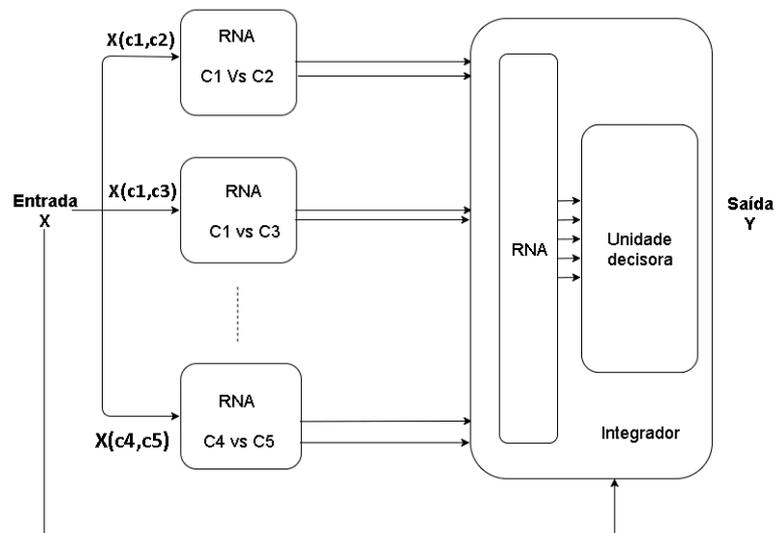
6.3.6.4 Máquinas de comitê dinâmicas com contraste entre duas classes - Experimento dual

Neste teste, o intuito é verificar como é o desempenho de uma máquina de comitê onde os especialistas contrastam uma classe de outra. Para tanto, na fase inicial as redes somente visam identificar uma classe de outra em particular.

A Figura 6.19 ilustra a implementação da máquina de comitê do tipo dinâmico com 10 redes especialistas, cada rede tem um sub-conjunto de treinamento-teste específico, com balanceamento (*undersampling*) de dados para cada conjunto de treinamento. Por exemplo, a rede especialista 1 que visa identificar a classe 1 (HH) da classe 2 (H), então esta rede somente é treinada e testada com amostras correspondentes a essas classes. Como nas implementações anteriores, aqui também se utiliza a codificação *hot-encoding* nas saídas. A rede integradora recebe como entrada as 20 saídas de cada especialista e a entrada inicial de 25 atributos, totalizando assim 45 atributos.

Cada rede neural foi treinada com 20.000 épocas e uma taxa de aprendizagem adaptativa de 0,01. Os resultados do desempenho de cada rede especialista e o resultados do desempenho da redes integradoras se encontram nas Tabelas 6.37 até 6.58. Pode-se verificar nas tabelas que as redes especialistas tiveram um bom desempenho no treinamento com as classes balanceadas. No teste, foram utilizadas as amostras restantes de cada classe e foi obtido um desempenho muito bom para a classe de maior quantidade de dados, isso depende do especialista e da dualidade de classes. Por exemplo, no caso do especialista 1 que tenta identificar amostras da classe HH e H, no teste tem-se maior quantidade de amostras da classe H que de HH. Agora para o especialista 8, que tenta identificar amostras das classes L e M, no teste tem-se maior número de amostras da classe M que de L.

Figura 6.19 - Comitê dinâmico implementado, com um contraste entre classes



Fonte: Produção do autor.

Tabela 6.37 - Matriz de confusão treinamento da rede especialista 1 - Classe 1 (HH) vs Classe 2 (H)

		Obs.		(%)
		(H)	(HH)	
Inf.	(H)	41.756	9.406	81,6
	(HH)	8.156	40.506	83,2
	(%)	83,7	81,2	82,4

Tabela 6.38 - Matriz de confusão teste - rede especialista Classe 1 (HH) vs Classe 2 (H)

		Obs.		(%)
		(H)	(HH)	
Inf.	(H)	23.176	3.802	85,9
	(HH)	5.965	12.836	68,3
	(%)	79,5	77,1	78,7

Tabela 6.39 - Matriz de confusão treinamento da rede especialista 2 - Classe 1 (HH) vs Classe 3 (M)

		Obs.		(%)
		(M)	(HH)	
Inf.	(M)	24.794	1.116	95,7
	(HH)	818	24.496	96,8
	(%)	96,8	95,6	96,2

Tabela 6.40 - Matriz de confusão teste - rede especialista 2 - Classe 1 (HH) vs Classe 3 (M)

		Obs.		(%)
		(M)	(HH)	
Inf.	(M)	8.022	3.453	69,9
	(HH)	515	37.485	98,6
	(%)	94	91,6	92

Tabela 6.41 - Matriz de confusão treinamento da rede especialista 3 - Classe 1 (HH) vs Classe 4 (L)

		Obs.		(%)
		(L)	(HH)	
Inf.	(L)	8.032	226	97,3
	(HH)	131	7.937	98,4
	(%)	98,4	97,2	97,8

Tabela 6.42 - Matriz de confusão teste - rede especialista 3 - Classe 1 (HH) vs Classe 4 (L)

		Obs.		(%)
		(L)	(HH)	
Inf.	(L)	2.596	3.528	42,4
	(HH)	125	54.859	99,8
	(%)	95,4	94	94

Tabela 6.43 - Matriz de confusão treinamento da rede especialista 4 - Classe 1 (HH) vs Classe 5 (LL)

		Obs.		(%)
		(LL)	(HH)	
Inf.	(LL)	2.518	61	97,6
	(HH)	35	2.492	98,6
	(%)	98,6	97,6	98,1

Tabela 6.44 - Matriz de confusão teste - rede especialista 4 - Classe 1 (HH) vs Classe 5 (LL)

		Obs.		(%)
		(LL)	(HH)	
Inf.	(LL)	806	3.806	17,5
	(HH)	45	60.191	99,9
	(%)	94,7	94,1	94,1

Tabela 6.45 - Matriz de confusão treinamento da rede especialista 5 - Classe 2 (H) vs Classe 3 (M)

		Obs.		(%)
		(M)	(H)	
Inf.	(M)	19.929	5.760	77,6
	(H)	5.683	19.852	77,7
	(%)	77,8	77,5	77,7

Tabela 6.46 - Matriz de confusão teste - rede especialista 5 - Classe 2 (H) vs Classe 3 (M)

		Obs.		(%)
		(M)	(H)	
Inf.	(M)	6.297	14.582	30,2
	(H)	2.240	38.859	94,5
		(%) 73,8	72,7	72,9

Tabela 6.47 - Matriz de confusão treinamento da rede especialista 6 - Classe 2 (H) vs Classe 4 (L)

		Obs.		(%)
		(L)	(H)	
Inf.	(L)	7.710	667	92
	(H)	453	7.496	94,3
		(%) 94,5	91,8	93,1

Tabela 6.48 - Matriz de confusão teste - rede especialista 6 - Classe 2 (H) vs Classe 4 (L)

		Obs.		(%)
		(L)	(H)	
Inf.	(L)	2.391	12.078	16,5
	(H)	330	58.812	99,4
		(%) 87,9	83	83,1

Tabela 6.49 - Matriz de confusão treinamento da rede especialista 7 - Classe 2 (H) vs Classe 5 (LL)

		Obs.		(%)
		(LL)	(H)	
Inf.	(LL)	2.486	112	95,7
	(H)	67	2.441	97,3
		(%) 97,4	95,6	96,5

Tabela 6.50 - Matriz de confusão teste - rede especialista 7 - Classe 2 (H) vs Classe 5 (LL)

		Obs.		(%)	
		(LL)	(H)		
Inf.	(LL)	780	10.221	7,1	
	(H)	71	66.279	99,9	
		(%)	91,7	86,6	86,7

Tabela 6.51 - Matriz de confusão treinamento da rede especialista 8 - Classe 3 (M) vs Classe 4 (L)

		Obs.		(%)	
		(L)	(M)		
Inf.	(L)	6.448	2.131	75,2	
	(M)	1.715	6.032	77,9	
		(%)	79	73,9	76,4

Tabela 6.52 - Matriz de confusão teste - rede especialista 8 - Classe 3 (M) vs Classe 4 (L)

		Obs.		(%)	
		(L)	(M)		
Inf.	(L)	1.974	8.720	18,14	
	(M)	747	17.266	95,9	
		(%)	72,5	66,4	67

Tabela 6.53 - Matriz de confusão treinamento da rede especialista 9 - Classe 3 (M) vs Classe 5 (LL)

		Obs.		(%)	
		(LL)	(M)		
Inf.	(LL)	2.456	137	94,7	
	(M)	97	2.416	96,1	
		(%)	96,2	94,6	95,4

Tabela 6.54 - Matriz de confusão teste - rede especialista 9 - Classe 3 (M) vs Classe 5 (LL)

		Obs.		(%)
		(LL)	(M)	
Inf.	(LL)	759	5.672	11,8
	(M)	92	25.924	99,6
	(%)	89,2	82	82,2

Tabela 6.55 - Matriz de confusão treinamento da rede especialista 10 - Classe 4 (L) vs Classe 5 (LL)

		Obs.		(%)
		(LL)	(L)	
Inf.	(LL)	1.889	621	75,3
	(L)	664	1.932	74,4
	(%)	74	75,7	74,8

Tabela 6.56 - Matriz de confusão teste - rede especialista 10 - Classe 4 (L) vs Classe 5 (LL)

		Obs.		(%)
		(LL)	(L)	
Inf.	(LL)	568	2.618	17,8
	(L)	283	5.713	95,3
	(%)	66,7	68,6	68,4

Tabela 6.57 - Matriz de confusão do conjunto de treinamento da RNA integradora dinâmica classe vs classe

		Inferido					(%)
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)	
Observado	Classe 1(HH)	2.063	401	59	20	10	80,8
	Classe 2(H)	389	1.476	439	194	55	57,8
	Classe 3(M)	45	393	1.130	780	205	44,3
	Classe 4(L)	16	64	261	1.466	746	57,4
	Classe 5(LL)	2	24	32	313	2.182	85,5
(%)		82	62,6	58,8	52,9	68,2	65,2

Tabela 6.58 - Matriz de confusão do conjunto de teste da RNA integradora dinâmica classe vs classe

		Inferido					(%)
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)	
Observado	Classe 1(HH)	680	136	24	7	4	79,9
	Classe 2(H)	126	469	166	67	23	55,1
	Classe 3(M)	25	112	389	260	65	45,7
	Classe 4(L)	4	28	87	490	242	57,6
	Classe 5(LL)	2	10	11	102	726	85,3
(%)		81,2	62,1	57,5	52,9	68,5	64,7

Com a utilização de uma máquina de comitê dinâmica com especialistas que contrastem uma classe de outra temos um ganho significativo no desempenho da classificação, tendo um desempenho global de 65,2% no treinamento e um 64,7% no teste. Além disso, as sensibilidades são altas para as classes 1(HH) e 5(LL), mesmo com pouca frequência de dados. Outra questão importante é a alta taxa de inferência também para essas classes, com desempenhos de 82% para a classe 1(HH) e 68,2% para a classe 5(LL) no treinamento e valores similares no teste.

Isto indica que os especialistas conseguem identificar melhor os padrões para as distintas classes, quando utilizam dados de duas classes para treinar e testar as redes.

6.4 Resumo do capítulo

Este capítulo inicia com uma descrição da metodologia utilizada para a obtenção dos dados a serem utilizados nos variados algoritmos de classificação testados.

Primeiro foi testado uma rede neuro-difusa, utilizando o software GUAJE. A vantagem de utilizar uma rede neuro-difusa reside na possibilidade de dar interpretabilidade ao processo de classificação com a geração de regras de inferência. Os testes evidenciam que a medida que melhoramos a acurácia da classificação, adicionando atributos de entrada, também aumentava a quantidade de regras geradas, o que diminui a interpretabilidade do sistema.

Depois, usando redes neurais e treinando com resultados do processo de *Breeding* correspondentes a um conjunto de três anos distintos, não foi possível generalizar a rede para fazer a classificação com dados do ano seguinte aos usados no treinamento. Quando se utilizam os dados de *Breeding* dos quatro anos para gerar os conjuntos de treinamento e teste, resultados aceitáveis na classificação são obtidos, o que dá um indicativo que se mais dados e padrões são adicionados uma melhor classificação pode ser factível.

Verificou-se que algoritmos de classificação de árvores binárias hierárquicas e métodos de comitê melhoram o desempenho da classificação, com resultados interessantes, desde o ponto de vista operacional para o primeiro caso, e na classificação de classes com menor frequência de dados com o segundo método.

7 CONCLUSÕES E TRABALHOS FUTUROS

Este projeto de mestrado teve como primeira atividade a migração do código do modelo de circulação global atmosférica do CPTEC-INPE para o *cluster* Lacibrido. Esse código é executado no supercomputador TUPÃ com compilador Portland e está adaptado para a arquitetura do supercomputador massivamente paralelo. Após um trabalho de modificações/adaptações e *debugging*, foi instalado com sucesso, usando um compilador GNU. O modelo está disponível para futuros trabalhos de pesquisa no *cluster* Lacibrido.

Durante o processo de migração, também foi utilizado outro modelo de menor complexidade, denominado SPEEDY. Os resultados dos testes de *breeding* realizados com o modelo SPEEDY foram apresentados na Conferência de Ciências não lineares e Complexidade NSC-2016 (vide apêndice A).

Realizando o processo de *breeding* com o modelo de circulação global da atmosfera do CPTEC-INPE, os campos apresentados pelos *bred vectors* foram comparados com o desvio padrão gerado pelo modelo de previsão por conjuntos (*Ensemble*) do CPTEC-INPE. Foram observados comportamentos qualitativamente semelhantes em ambas as técnicas, onde uma magnitude grande dos *bred vectors* coincidem regionalmente com grandes magnitudes do desvio padrão do *ensemble*. Ou seja, as grandes magnitudes de *bred vector* coincidem com a discrepância observada através de gráficos espaguete gerados com os membros do *Ensemble*. Os resultados de previsibilidade por *bred vector* e por *ensemble* foram apresentados no Congresso Internacional de Ciências Computacionais Interdisciplinares (CCIS'16)¹.

Na última fase do projeto, foram testados vários algoritmos de classificação baseados em aprendizado de máquina, a fim de mapear a previsibilidade do modelo global em classes, com o emprego da técnica de *breeding*.

Escolhendo como região de estudo uma área da América do Sul correspondente às coordenadas latitude=[11'S-20'S] e longitude=[42'W-50'W], foi gerado um conjunto grande de dados de entrada do processo de *breeding* aplicado ao modelo global. No processo de classificação, a entrada do classificador corresponde a magnitude do *bred vector*. Para a configuração do classificador, a saída desejada foi calibrada com os dados do desvio padrão do *ensemble*. De fato, o CPTEC-INPE adota a variância do *ensemble* como uma medida da confiabilidade da previsão, associando áreas de boa confiabilidade com regiões onde o desvio padrão ou espalhamento é pequeno e

¹CCIS'16: <http://www.epacis.net/ccis2016/en/>.

vice-versa.

Iniciando os testes de classificação com redes neuro-difusas, utilizando o software GUAJE, com o objetivo de estabelecer uma melhor interpretabilidade para o sistema de previsão, o processo gerou muitas regras de inferência. A medida que se adicionam atributos de entrada, que melhoram o desempenho do algoritmo de classificação, novas regras de inferência são geradas.

Redes neurais de múltiplas camadas, com distintas configurações e distintos parâmetros, para realizar uma classificação com dados de simulações do processo de *breeding* não tiveram resultados satisfatórios com a estratégia de conjunto de amostragem cronológica. A conclusão é que os padrões apresentados para a rede no treinamento não são suficientes para classificar categorias de previsibilidade de um novo ano. Embora a quantidade de dados seja grande, não se obtém os padrões suficientes que possam generalizar o processo de classificação.

Porém, quando se misturam os dados dos anos de 2010 a 2014 e formam-se os conjuntos da fase de treinamento e de teste, a rede neural MLP consegue melhor desempenho na identificação das classes de previsibilidade. Além disso, uma rede neuro-difusa logrou um comportamento mais homogêneo na classificação das distintas classes com estes dados, mesmo tendo um grande desbalanceamento do número de elementos das classes tanto no treinamento como no teste. Este resultado sugere que a previsibilidade do modelo global pode ser estabelecida via análise de *breed vectors*.

Os testes com os demais algoritmos de aprendizado de máquina, árvore hierárquica e os métodos de comitê, mostraram um bom desempenho. No caso da classificação com árvore hierárquica, a estratégia de classificação *one-against-all* é bastante efetiva para atacar problemas de dados desbalanceados. Foi verificado que o balanceamento no conjunto de treinamento e teste no método de máquinas de comitê ajuda muito no desempenho das redes-especialistas. Também verificou-se que a utilização de estratégias de treinamento dual com uma máquina dinâmica leva a resultados de classificação bastante satisfatórios.

Esta dissertação tem foco na área de previsibilidade com uma metodologia que obteve sucesso em problemas anteriores, como os modelos de Lorenz 3D e o modelo de 3 ondas de plasma acopladas da física solar. Embora a complexidade e dimensão destes modelos sejam extremamente mais simples em comparação a um modelo atmosférico operacional, foi mostrado que a previsibilidade pode ser quantificada

utilizando uma estratégia de baixo custo computacional como é o *breeding*.

7.1 Trabalhos Futuros

A previsibilidade de um modelo meteorológico é uma tarefa desafiadora, pois o modelo é desenvolvido para emular processos da dinâmica caótica da atmosfera com processos que, em muitos casos, não são completamente entendidos e/ou modelados.

Todavia, a quantidade de dados gerados no processo de previsão numérica de tempo é enorme e a extração de padrões por técnicas de inteligência artificial exige criar um banco de dados gigante para a geração dos conjuntos de treinamento e de teste para os distintos algoritmos de classificação. Há um desafio no gerenciamento e no processamento desta grande massa de dados. O método de *breeding* tem custo computacional menor do que a metodologia por *ensemble*, mas também requer o uso de computação intensiva.

Portanto, uma primeira etapa futura é gerar mais processos de *breeding* para vários meses de estudo a fim de enriquecer os conjuntos de treinamentos com mais padrões atmosféricos.

Diante da grande quantidade de dados, o problema se tipifica para ser identificado na área de *Data Science*. Técnicas de *deep learning* (DENG; YU, 2014) podem ser as ferramentas mais adequadas para lidar com o problema. Além disso, computação paralela é um componente essencial para lidar com grande quantidade de processamento, assim como o uso de co-processadores (computação heterogênea): GPU (Graphical Processing Unit), MIC (Multi-Integrated Core) e FPGA (Field Programmable Gate Array), que são os dispositivos disponíveis atualmente.

Avaliação de novas técnicas de previsibilidade merecem também uma investigação comparativa em termos de desempenho de acurácia e de custo computacional. Além das técnicas tratadas nesta tese, uma técnica de interesse aplicada em computação gráfica é a decomposição de Helmholtz-Hodge (RIBEIRO et al., 2016).

Outras investigações mais imediatas dentro da própria metodologia apresentada nesta dissertação também merecem ser analisadas. Por exemplo, avaliar os resultados com outra estratégia para a Máquina de Comitê, já utilizando outros especialistas que não sejam necessariamente redes neurais.

REFERÊNCIAS BIBLIOGRÁFICAS

- ANOCHI, J. A. **Previsão climática de precipitação por redes neurais autoconfiguradas**. 159 p. Tese (Doutorado) — Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2015-11-03 2015. Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m21b/2015/09.16.22.02>>. Acesso em: 03 jul. 2017. 29, 30, 31, 32, 33
- BONATTI, J. P. Modelo de circulação geral atmosférico do cptec. **Climanálise - Boletim de Monitoramento e Análise Climática**, x, p. 198–202, out. 1996. ISSN 0103-0019. Acesso em: 19 jul. 2017. 11, 12, 13, 14
- BULLINARIA, J. A. **Committee machines. Lecture notes in introduction of neural networks**. 2004. Disponível em: <<http://www.cs.bham.ac.uk/~jxb/NN/115.pdf>>. Acesso em: 29 de abril, 2017. 44, 45
- BUSTAMANTE, J. F. **Previsão por conjunto**. 2010. Disponível em: <http://ftp.cptec.inpe.br/etamd/Workshop/WorkEtaIII/Anais/JosianeBustamantes_PrevisaoPorConjunto.pdf>. Acesso em: 20 de março, 2017. 15, 17, 18
- CINTRA, R. S.; VELHO, H. F. d. C. Global data assimilation using artificial neural networks in speedy model. In: INTERNATIONAL SYMPOSIUM ON UNCERTAINTY QUANTIFICATION AND STOCHASTIC MODELING, 1., Feb. 26th to Mar. 2nd, 2012, São Sebastião, SP. **Proceedings...** [S.l.], 2012. p. 648–654. ISSN 2238-1007. Acesso em: 03 jul. 2017. 115
- CINTRA, R. S. C. **Assimilação de dados com redes neurais artificiais em modelo de circulação geral da atmosfera**. 200 p. Tese (Doutorado) — Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2010-09-27 2010. Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m19/2010/09.20.14.46>>. Acesso em: 03 jul. 2017. 3
- CINTRA, R. S. C.; VELHO, H. F. d. C. Breeding and predictability in chaotic dynamics. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL (CNMAC)., 8 - 11 set., Belém, PA. **Proceedings...** [S.l.], 2008. Acesso em: 03 jul. 2017. 19, 20, 23, 24

COIFFIER, J. **Fundamentals of numerical weather prediction**. Cambridge University Press, 2011. ISBN 9781139502702. Disponível em:

<https://books.google.com.py/books?id=vY29h_Bvqk0C>. 1

DENG, L.; YU, D. Deep learning: methods and applications. **Foundations and Trends in Signal Processing**, v. 7, n. 3-4, p. 1–99, 2014. 107

DETLEF, N. **What are Neuro-Fuzzy systems?** 1997. Disponível em:

<<http://fuzzy.cs.ovgu.de/nfdef.html>>. Acesso em: 20 de março, 2017. 39

ECMWF. **Twenty-five years of ensemble forecasting**. 2017. Disponível em:

<<http://www.ecmwf.int/en/about/media-centre/news/2017/twenty-five-years-ensemble-forecasting>>. Acesso em: 30 de abril, 2016. 14

ENSEMBLE-CPTEC. **Modelo Ensemble**. 2013. Disponível em:

<http://previsaonumerica.cptec.inpe.br/mod_ens.shtml>. Acesso em: 02 de outubro, 2016. 14

EVANS, E. et al. Rise undergraduates find that regime changes in Lorenz model are predictable. **Bull. Amer. Meteor. Soc**, v. 15, p. 521–524, 2004. 20, 21, 22, 23

FIGUEROA, S. N.; BONATTI, J. P.; KUBOTA, P. Y.; GRELL, G. A.; MORRISON, H.; BARROS, S. R. M.; FERNANDEZ, J. P. R.; GUTIERREZ, E. M. A. R.; SIQUEIRA, L.; COSTA, G. L. d.; SILVA, J. d.; SILVA, J. R. d.; PENDHARKARA, J.; CAPISTRANO, V. B.; ALVIM, D. S.; ENORÉ, D. P.; DINIZ, F. L. R.; SATYAMURTY, P.; CAVALCANTI, I. F. d. A.; NOBRE, P.; BARBOSA, H. M. J.; MENDES, C. L.; PANETTA, J. The Brazilian global atmospheric model (bam): Performance for tropical rainfall forecasting and sensitivity to convective scheme and horizontal resolution. **Weather and Forecasting**, v. 31, n. 5, p. 1547–1572, 2016. ISSN 0882-8156. Acesso em: 03 jul. 2017. 13, 14

FISPRO. **Linguistic variable and fuzzy inference system**. 2003. Disponível em: <<https://www7.inra.fr/mia/M/fispro/fisprodocen/inline-help/node39.html>>.

Acesso em: 28 de março, 2017. 34, 124

GLOBAL-CPTEC. **Modelo Global**. 2013. Disponível em:

<http://previsaonumerica.cptec.inpe.br/mod_glb..shtml>. Acesso em: 02 de outubro, 2016. 11

HAYKIN, S. **Neural networks and learning machines**. [S.l.]: Prentice Hall, 2009. ISBN 9780131471399. 27, 29, 30, 31, 33

HÄRTER, F. P.; VELHO, H. F. de C. New approach to applying neural network in nonlinear dynamic model. **Applied Mathematical Modelling**, v. 32, n. 12, p. 2621 – 2633, 2008. ISSN 0307-904X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0307904X07002296>>. 3

INPE. **CPTEC/INPE inicia 2016 com novo modelo global de previsão do tempo**. 2015. Disponível em: <http://www.inpe.br/noticias/noticia.php?Cod_Noticia=4084>. Acesso em: 26 de setembro, 2016. 4, 14

KALNAY, E. **Atmospheric modeling, data assimilation and predictability**. Cambridge University Press, 2003. ISBN 9780521796293. Disponível em: <<https://books.google.com.py/books?id=Uqc7zC7NULMC>>. 2, 3, 7, 8, 9, 10, 20

KALNAY, E.; YANG, S.; PEÑA, M. **Bred vectors: a simple tool to understand complex dynamics**. 2007. Disponível em: <<https://www.atmos.umd.edu/~ekalnay/pubs/ArakawaSymposiumKalnay.ppt.pdf>>. Acesso em: 15 de abril, 2016. 17, 19

KNAPP, B. **Lectures Notes on fuzzy sets and pattern recognition**. 2004. Disponível em: <<http://www.cs.princeton.edu/courses/archive/fall07/cos436/HIDDEN/Knapp/fuzzy.htm>>. Acesso em: 28 de março, 2017. 38, 41, 42

KUBOTA, P. Y. **Variabilidade da energia armazenada na superfície e o seu impacto na definição do padrão de precipitação na América do Sul**. 309 p. Tese (Doutorado) — Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2012-08-06 2012. Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m19/2012/08.02.02.42>>. Acesso em: 03 jul. 2017. 10, 12, 13, 14

LEVER, J.; KRZYWINSKI, M.; ALTMAN, N. Points of significance: Classification evaluation. **Nat Meth**, Nature Publishing Group, a division of Macmillan Publishers Limited. All Rights Reserved., v. 13, n. 8, p. 603–604, Aug 2016. ISSN 1548-7091. This Month. Disponível em: <<http://dx.doi.org/10.1038/nmeth.3945>>. 46, 47

LORENZ, E. **Estimates of atmospheric predictability at medium range**. 1984. Disponível em: <http://eaps4.mit.edu/research/Lorenz/Estimates_of_Atmospheric_Predictability_1984.pdf>. Acesso em: 20 de abril, 2017. 2

LORENZ, E. N. Deterministic nonperiodic flow. **Journal of the Atmospheric Sciences**, v. 20, n. 2, p. 130–141, 1963. Disponível em: <[https://doi.org/10.1175/1520-0469\(1963\)020<0130:DNF>2.0.CO;2](https://doi.org/10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2)>. 2

LYNCH, P. **The emergence of numerical weather prediction: Richardson's dream**. Cambridge University Press, 2014. ISBN 9781107414839. Disponível em: <<https://books.google.com.py/books?id=PpGooAEACAAJ>>. 1, 2

MATHWORKS. 2011. Disponível em: <<https://www.mathworks.com/help/fuzzy/fuzzy-inference-process.html>>. Acesso em: 22 de março de 2017. 37

MENDONÇA, A. M.; BONATTI, J. P. O sistema de previsão de tempo global por ensemble do cptec. In: CONGRESSO BRASILEIRO DE METEOROLOGIA, 12., 4-9 ago. 2002, Foz de Iguaçu (PR). **Anais...** [S.l.]: SBMET, 2002. p. 3341–3351. Acesso em: 21 jul. 2017. 15

MIT-NEWS. **Edward Lorenz, father of chaos theory and butterfly effect**. 2008. Disponível em: <<http://news.mit.edu//2008/techtalk52-24.pdf>>. Acesso em: 15 de março, 2017. 3

MOLTENI, F.; KUCHARSKI, F. **Description of the ICTP AGCM (SPEEDY) - Version 40**. 2009. Disponível em: <http://users.ictp.it/~kucharsk/speedy_description/km_ver40_appendixA.pdf>. Acesso em: 20 de abril, 2016. 115

MOURA, A. D. Von Neumann e a previsão numérica de tempo e clima. **Estudos Avançados**, scielo, v. 10, p. 227 – 236, 04 1996. ISSN 0103-4014. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-40141996000100021&nrm=iso>. 2

NCEP. **The history of numerical weather prediction**. jun 2007. Disponível em: <http://celebrating200years.noaa.gov/foundations/numerical_wx_pred/welcome.html>. Acesso em: 26 de setembro, 2016. 2

PALMER, T.; HAGEDORN, R. **Predictability of weather and climate**. Cambridge University Press, 2006. ISBN 9780511225017. Disponível em: <<https://books.google.com.au/books?id=vobtjwEACAAJ>>. 3

PARRA, N. H. **Smart hierarchical WiFi localization system for indoors**. Tese (Doutorado) — Universidad de Alcalá, 2014. 40

RIBEIRO, P. C.; VELHO, H. F. d. C.; LOPES, H. Helmholtz-hodge decomposition and the analysis of 2d vector field ensembles. **Computers and Graphics (Pergamon)**, v. 55, p. 80–96, Apr. 2016. ISSN 0097-8493. Acesso em: 03 jul. 2017. 107

ROMERO, L. F. S.; SANDRI, S. A.; VELHO, H. F. d. C. Analysis of the breeding technique applied to the cptec-agcm model. In: CONFERENCE OF COMPUTATIONAL INTERDISCIPLINARY SCIENCES, 4. (CCIS), 7-10 nov., São José dos Campos, SP. **Proceedings...** [S.l.], 2016. Acesso em: 03 jul. 2017. 59

ROMERO, L. F. S.; SANDRI, S. A.; VELHO, H. F. d. C.; CINTRA, R. S. C.; FREITAS, S. R. d. Bred vector applied to the atmospheric dynamics. In: INTERNATIONAL CONFERENCE ON NONLINEAR SCIENCE AND COMPLEXITY, 6., 16-20 May, São José dos Campos, SP. **Proceedings...** [S.l.], 2016. Acesso em: 03 jul. 2017. 49

SANCHEZ, E. C. M. **Controle por aprendizado acelerado e Neuro-Fuzzy de sistemas servo-hidraulico de alta frequência.** Dissertação (Mestrado) — PUC-RIO, 2009. Disponível em: <http://www.dbd.puc-rio.br/pergamum/tesesabertas/0721406_09_pretextual.pdf>. 40

SANDRI, S.; CORREA, C. **Lógica nebulosa. V Escola de redes neurais.** 1999. Disponível em: <http://www.gta.ufrj.br/ensino/cpe717-2011/curso_ERN99_fuzzy.pdf>. Acesso em: 02 de february, 2017. 33, 35, 36, 37

SANTOS, P. L. B. d. **Previsibilidade em sistemas caóticos utilizando sistemas neuro-difusos.** 114 p. Dissertação (Mestrado) — Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2014-04-15 2014. Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m21b/2014/06.12.01.41>>. Acesso em: 03 jul. 2017. 19, 23, 25

SILLA, C. N.; FREITAS, A. A. A survey of hierarchical classification across different application domains. **Data Mining and Knowledge Discovery**, v. 22, n. 1, p. 31–72, Jan 2011. ISSN 1573-756X. Disponível em: <<http://dx.doi.org/10.1007/s10618-010-0175-9>>. 42, 44

STERGIOU, C.; SIGANOS, D. **Neural Networks.** 1996. Disponível em: <https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html>. Acesso em: 15 de abril, 2017. 27

TANSCHKEIT, R. **Sistemas Fuzzy**. 2004. Disponível em: <<http://www.dainf.cefetpr.br/~myriam/PastaWeb/Fuzzy/ApostilaFuzzy.pdf>>.

Acesso em: 30 de setembro, 2016. 34, 35, 37, 38

TOTH, Z.; KALNAY, E. Ensemble forecasting at ncep and the breeding method.

Monthly Weather Review, v. 125, p. 3292–3302, 1997. 19, 20

VILLANUEVA, W. J. P. **Comitê de máquinas em predição de séries temporais**. Dissertação (Mestrado) — Universidade Estadual de Campinas, 2006.

44

WANG, R. **Lecture notes on hierarchical lassifiers**. 2016. Disponível em:

<<http://fourier.eng.hmc.edu/e161/lectures/classification/node13.html>>.

Acesso em: 21 de março, 2017. 42

WARNER, T. T. **Numerical weather and climate prediction**. [S.l.]:

Cambridge University Press, 2011. 10

ZADEH, L. Fuzzy sets. **Information and Control**, p. 338–353, 1965. 33

APÊNDICE A - Método de *Breeding* aplicado ao modelo global SPEEDY

Antes de implementar o processo de *Breeding* ao modelo global do CPTEC foi primeiro utilizado um modelo de menor complexidade chamado de SPEEDY ("Simplified Parameterizations, primitive-Equation Dynamics"), um modelo de caráter de circulação global, baseado no núcleo dinâmico espectral desenvolvido no laboratório GFDL (*Geophysical Fluid Dynamics*) dos Estados Unidos (MOLTENI; KUCHARSKI, 2009) ¹.

Esta fase inicial de implementação da metodologia de breeding ao modelo SPEEDY ajudou na capacitação, implementação do processo e entendimento do mesmo. A razão é que o processo de *debugging* do modelo global do CPTEC atrasou a implementação do *Breeding* ao mesmo.

A.1 Descrição do modelo

Este modelo, desenvolvido no ICTP (*International Centre for Theoretical Physics*) localizado em Trieste-Italia, é um modelo de circulação global da atmosfera simplificado mas válido para pesquisas e o ambiente acadêmico.

O modelo dinâmico é integrado mediante métodos espectrais na horizontal e cada camada vertical e além disso possui um esquema de parametrizações similares aos modelos de maior complexidade (CINTRA; VELHO, 2012).

Como se menciona no manual de descrição do modelo em Molteni e Kucharski (2009), é um modelo hidrostático, espectral com tratamento semi-implícito nas ondas de gravidade. As variáveis prognósticas são a vorticidade (Vor), divergência (Div), temperatura absoluta (T) e o logaritmo da pressão superficial (log(ps)).

O modelo também calcula outras variáveis adicionais que são geradas pelo núcleo dinâmico. Para o tratamento temporal utiliza um esquema de leapfrog, com uma malha horizontal que corresponde a um truncamento espectral triangular de 30 números de ondas (configuração T30)².

Na configuração vertical, SPEEDY possui 8 camadas ³, com as saídas que são processadas a níveis de pressão de 30, 100, 200, 300, 500, 700, 850 e 925 hPa.

¹O modelo CPTEC-COLA também foi baseado no núcleo dinâmico espectral do laboratório GFDL

²No modelo do CPTEC, se utiliza T126 para o *Ensemble* e por isso foi adotada essa configuração para realizar o *breeding* com o modelo global.

³O modelo AGCM do CPTEC possui 28 camadas verticais

A.2 Parametrizações Físicas

Como o modelo é simplificado, se assumem critérios visando que as parametrizações sejam adequadas com a resolução do modelo, que é maior que as resoluções utilizadas em modelos numéricos operacionais como a que utiliza o CPTEC.

Processos como convecção, condensação de escala-longa e nuvens possuem módulos específicos que são ativados conforme os parâmetros atingem certos limiares. Outras parametrizações como radiação de campo curto e longo, fluxos superficiais de momento e energia assim como a difusão vertical são compostos por equações que dependem da dinâmica e da topografia (como o caso do fluxo superficial).

A.3 Condições de borda

Como qualquer modelo numérico que é baseado em equações diferenciais parciais, é preciso fornecer condições de borda a fim de que o sistema possa ser resolvido.

Condições como a temperatura superficial do mar (SST, pelas siglas em inglês), fração de gelo-mar, temperatura superficial da terra (perto de 1 metro), umidade, fração de vegetação, entre outros são parâmetros importantes para a correta determinação de variáveis como fluxo de momento, calor, umidade, fluxo de radiação solar entrante, etc.

A.4 Trabalho apresentado na Conferência de Ciências não lineares e Complexidade 2016

A Conferência de ciências não lineares e complexidade (NSC pelas siglas em inglês), realizada em maio de 2016 no INPE, foi uma boa oportunidade para mostrar os primeiros testes feitos utilizando breeding aplicados ao modelo global SPEEDY. O trabalho, como estava na fase inicial, foi apresentado na categoria de pôster.

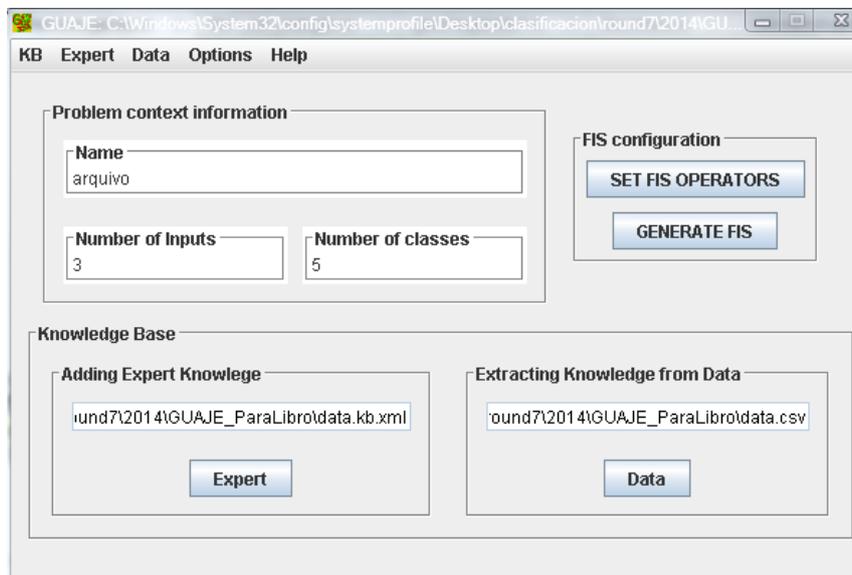
APÊNDICE B - Breve tutorial de uso do software GUAJE

O GUAJE é um *software* que implementa um sistema de extração de conhecimento a través de uma combinação de distintas ferramentas e *softwares*

Na Figura B.1 se visualiza a janela principal do software, onde pode-se ver 3 divisões de configuração:

- *Problem Context Configuration* onde se configura o nome do arquivo, o numero de entradas a considerar e o numero de classes de saída,
- configuração do arquivo FIS (Fuzzy Inference System), se configuram os processos de conjunção, disjunção e defuzzyficação, assim como a geração do arquivo FIS para outras plataformas como Matlab, etc.
- *Knowledge Base*, no qual se configura a base de conhecimento. A base de conhecimento se sub-divide novamente em duas areas: a *Adding Expert Knowledge* na esquerda e o *Extracting Knowledge from Data* na direita.

Figura B.1 - Janela Principal do software GUAJE



Fonte: Produção do Autor

Carregando o arquivo de entrada através de **KB»New**, **Data»Open** uma questão importante é que o arquivo de entrada deve ser uma tabela de amostras de

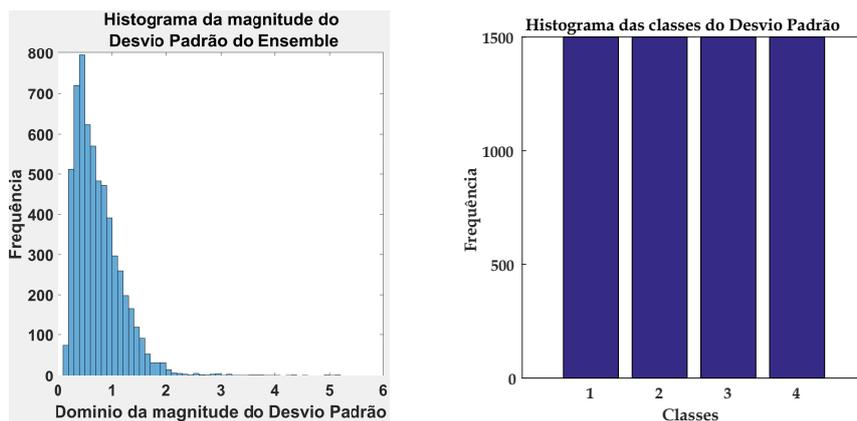
entradas-saída no formato CSV (separado por vírgulas). Cada linha da tabela é um conjunto de amostras de entrada-saída, onde as entradas e saídas estão organizadas em colunas.

Para este exemplo, considera-se um conjunto de 6.000 amostras de dados, onde as entradas correspondem à simulação do processo de *Breeding* e a saída corresponde ao desvio padrão do *ensemble*, para a simulação com início no dia 02 de outubro de 2014 as 00 Hs na mesma região de estudo indicada na seção 6.1.

O processo de formação da tabela de extração de conhecimento se mostra na seção 6.1.1, para esta análise foi particionado o histograma do desvio padrão em 4 classes utilizando os percentis (25, 50, 75) para que as classes tenham a mesma quantidade de elementos assim temos as classes balanceadas (vide Figura B.2). Por exemplo,

- classe 1 (HH): desde o mínimo do desvio padrão até percentil 25,
- classe 2 (H): desde percentil 25 até percentil 50,
- classe 3 (M): desde percentil 50 até percentil 75,
- classe 4 (L): desde percentil 75 até o valor máximo do desvio padrão.

Figura B.2 - Histogramas da magnitude do desvio padrão do *Ensemble* e das classes geradas



(a) Magnitude do desvio padrão

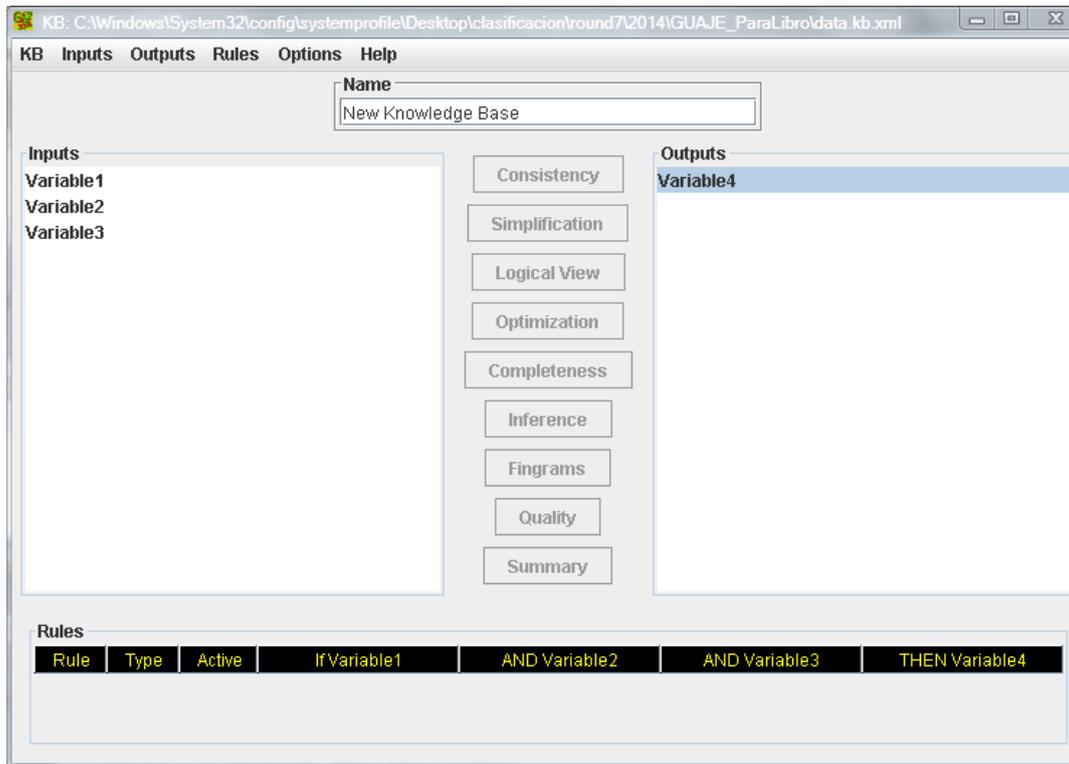
(b) Histograma das classes

Fonte: Produção do autor

Clicando no botão **Expert** da área de **Adding Expert Knowledge**, podemos ver as configurações dos conjuntos de entrada e saída. Se abre uma nova janela como se

mostra na Figura B.3 no qual se visualizam os atributos de entrada como a saída, também são mostrados vários processos que ficam habilitados uma vez realizado o processo de geração de regras difusas.

Figura B.3 - Janela **Expert**, de configuração dos conjuntos de entrada-saída



Fonte: Produção do Autor

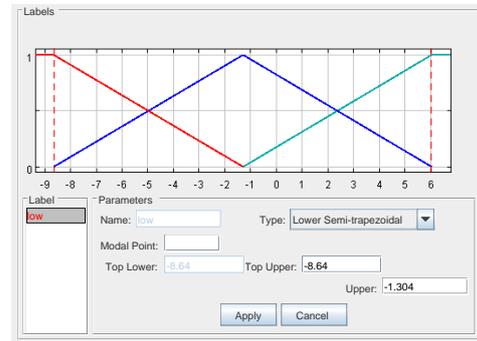
Clicando em uma variável, se abra uma nova janela como a Figura B.4, se mostram as janelas de configurações das variáveis de entrada como pode ser também da saída e as funções de membresia associadas às variáveis.

Nessa área, podem-se configurar os nomes das variáveis, renomear os parâmetros das variáveis como foi feito para a saída, no qual inicialmente a codificação da classe era numérica (1, 2, 3, 4) passando a ser etiquetas alfabéticas (HH, H, M, L).

Nessa área, pode-se definir o tipo da variável no qual tem-se três opções (binário, numérico e categórico), na categoria numérica podem-se definir os conjuntos difusos clicando na opção **Advance Options** como foi feita para a variável de entrada, agora para a variável de saída o tipo é categórico o qual permite definir as classes de saída no domínio discreto.

Figura B.4 - Configuração dos conjuntos de entrada e saída

(a) Variável de entrada



(b) Conjuntos difusos da entrada 1

(c) Variável de saída

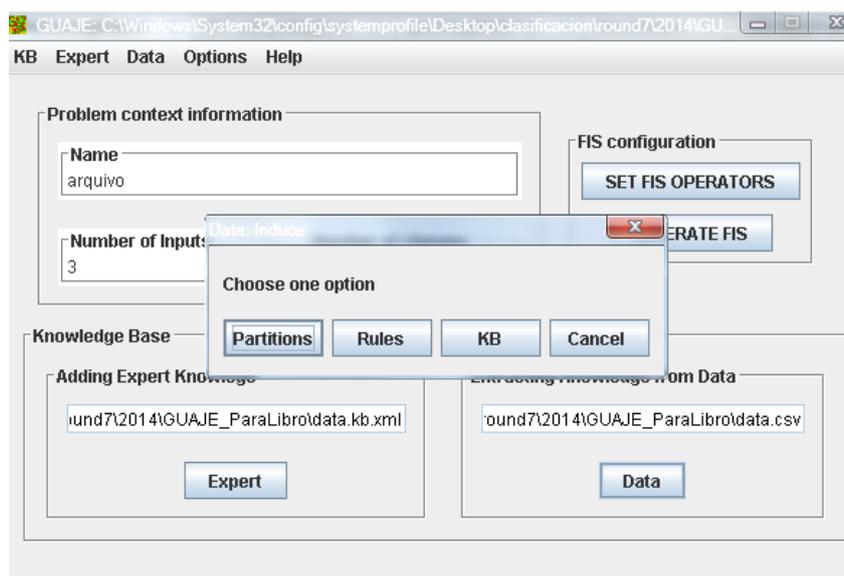
Dentro da janela de configuração dos conjuntos de entrada e saída, pode-se mudar o nome da variável, assim como visualizar o domínio do dado, etc.

Fonte: Produção do autor

Depois de serem feitas as configurações correspondentes para cada variável de entrada e saída no caso que for necessário, se fecham todas as janelas de configuração e se volta para a janela principal a fim de iniciar o processo de inferência.

O processo de inferência de conhecimento inicia com a definição da partição dos conjuntos difusos de entrada. Clicando na área **Extracting Knowledge from Data** da janela principal é aberta uma nova janela com as opções **Partitions**, **Rules**, **KB** e **Cancel** como se visualiza na Figura B.5.

Figura B.5 - Janela de Processos de inferência



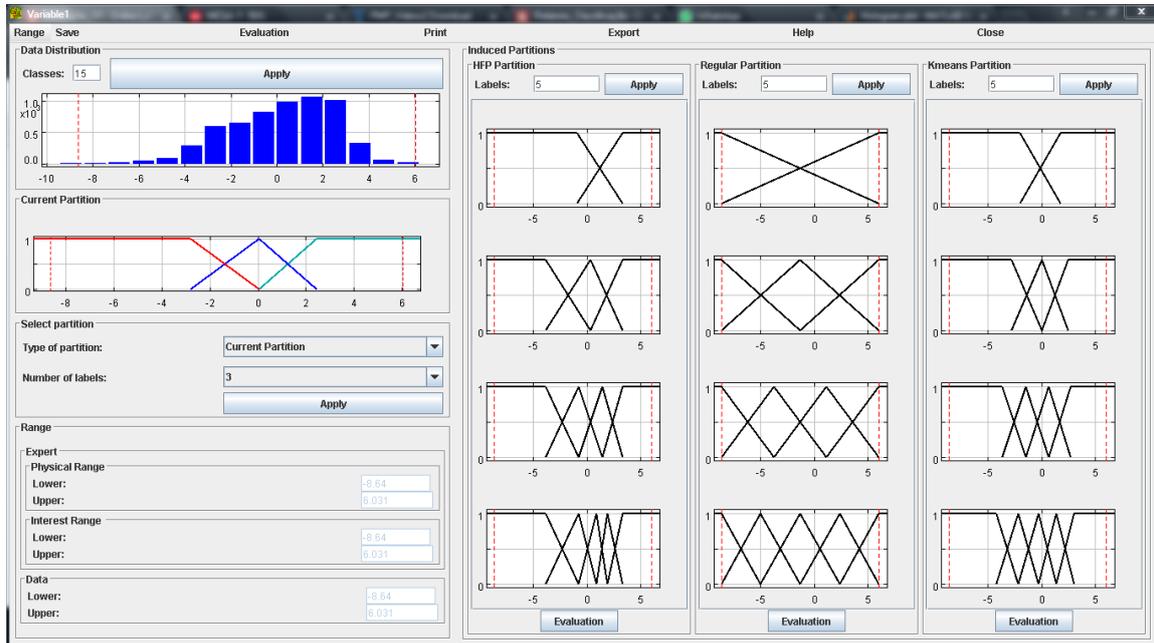
Fonte: Produção do Autor

Logo de iniciar o processo de partição, novas janelas para cada variável são abertas, semelhantes à Figura B.6 onde se visualiza o histograma de frequência da variável, o domínio da variável e o tipo de algoritmo de partição difusa utilizada para montar as funções de pertinência para à variável. No inicio na área de **current partition** se mostra a partição inicial da variável e na área de **select partition**, o software sugere o algoritmo que melhor particiona a variável na área **Type of Partition**.

Têm-se três opções de algoritmos de partição que podem ser escolhidos na área **Type of Partition**: o Regular, o k-means e o HFP (*High Fuzzy Partition*). Na parte direita se visualizam como ficariam estas partições com os novos conjuntos difusos definidos pelos três algoritmos.

A janela também possui uma opção chamada **Evaluation**, onde são apresentados três índices de avaliação da partição. As métricas de avaliação são: PC (Partition Coefficient), PE (Partition Entropy) e CH (Chen Index). Utilizando um código de cores, o software mostra quais são as partições (considera vários conjuntos difusos e algoritmos de partição possíveis) que apresentam melhores índices. Em azul mostra qual é o melhor (que geralmente já é indicado no **Type of Partition**) e em verde o que esta definido inicialmente. Na área de **HELP** são apresentados melhores explicações e referencias do funcionamento do algoritmo.

Figura B.6 - Janela de processo de escolha do algoritmo de partição difusa



Fonte: Produção do Autor

Logo depois de ter configurado as partições e definidos os conjuntos difusos das variáveis de entrada. Pode-ser feito a inferência de regras a partir dos dados de entrada-saída.

Na Figura B.5 se escolhe a opção **Rules**, e o software fornece 4 algoritmos de inferência de regras: *Fuzzy Decision Trees*, *Wang e Mendel (WM)*, *Fast Prototyping Algorithm (FPA)* e *Prototyping Rules*.

Os dois algoritmos escolhidos para este trabalho foram a de Wang-Mendel e o *Fast Prototyping Algorithm*. WM segundo Fispro (2003), inicia gerando uma regra para cada par de amostra no treinamento, a amostra i é escrita como

$$SE x_1 \text{ é } A_1^i \text{ e } x_2 \text{ é } A_2^i \dots \text{ e } x_p \text{ é } A_p^i \text{ ENTÃO } y \text{ é } C^i$$

Os conjuntos difusos A_j^i são aqueles nos quais o grau de pertinência da entrada x_j^i é máxima para cada atributo j da amostra. O conjunto difuso C_i é o qual o grau de pertinência da saída desejada y_i , é máxima.

Para um conjunto crisp (conjunto com um elemento único), uma conversão interna é feita a fim de criar funções de pertinências centradas no valor de saída, depois é definido um número igual ao centro dessa função para a conclusão da regra.

FPA pelo contrario gera regras mais gerais que WM, mas ao mesmo tempo é mais específico que *Fuzzy Decision Tree*. Inicia criando um malhando com todas as combinações de regras correspondentes às entradas, depois são removidas as repetições e inconsistências. Perde eficiência conforme ao tamanho de entradas e amostras.

Na Figura B.7 se visualiza as regras geradas neste caso utilizando o algoritmo WM. As regras do tipo SE-ENTÃO depende fortemente da quantidade de atributos de entrada que se levam em consideração, tem pouca diferença na quantidade de regras geradas com o algoritmo FPA para este projeto.

Figura B.7 - Regras geradas com o algoritmo de Wang-Mendel

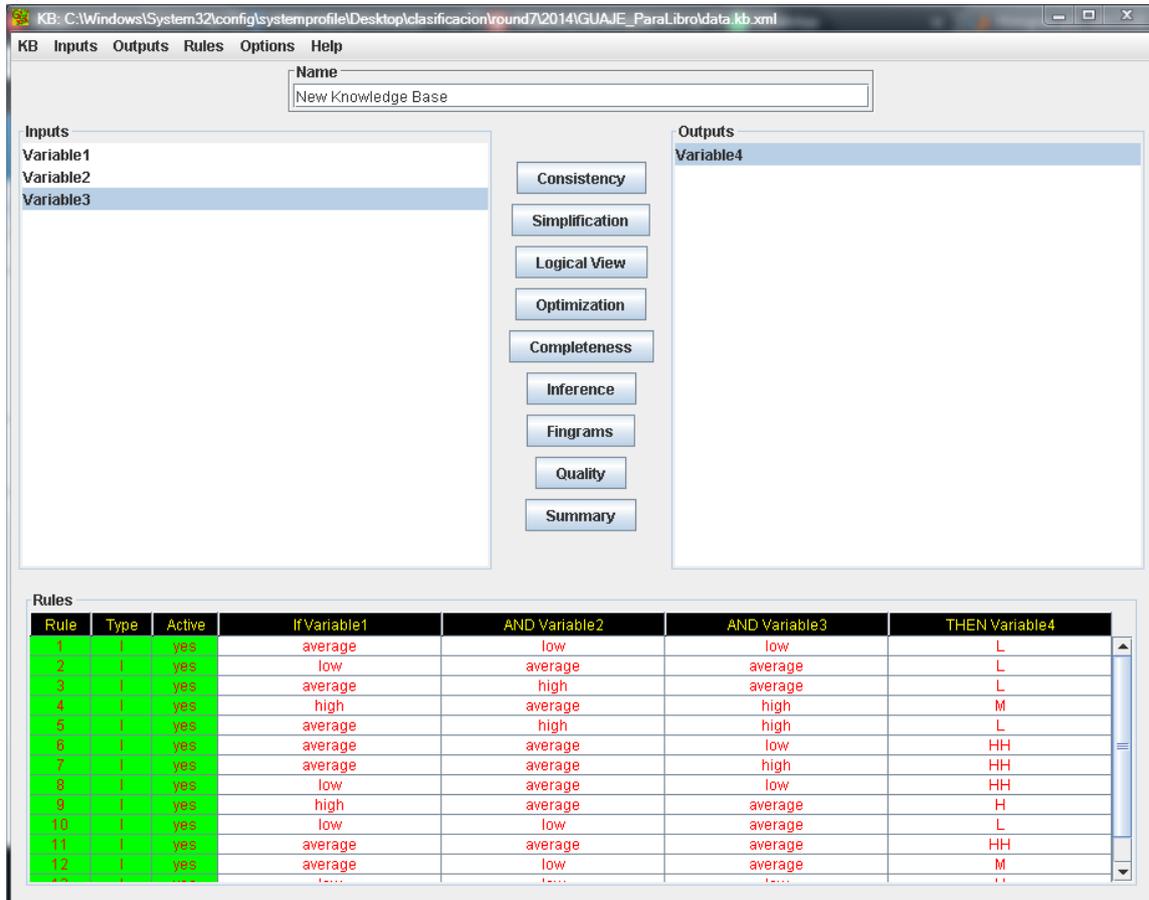
Rule	Type	If Variable1	AND Variab..	AND Variab..	THEN Varia..
1	I	average	low	low	4.0
2	I	low	average	average	4.0
3	I	average	high	average	4.0
4	I	high	average	high	3.0
5	I	average	high	high	4.0
6	I	average	average	low	1.0
7	I	average	average	high	1.0
8	I	low	average	low	1.0
9	I	high	average	average	2.0
10	I	low	low	average	4.0
11	I	average	average	average	1.0
12	I	average	low	average	3.0
13	I	low	low	low	2.0
14	I	high	high	high	4.0

As regras do tipo SE-ENTÃO depende dos atributos de entrada e da quantidade de amostras

Fonte: Produção do Autor

Uma vez gerado o conjunto de regras é preciso que seja salvo para ser adicionado à base de conhecimento. Feito isto, os demais processos que se visualizam na Figura B.8 são habilitados.

Figura B.8 - Regras geradas adicionadas à base de conhecimentos



Quando as regras geradas são salvas, estas são adicionadas à base de conhecimento

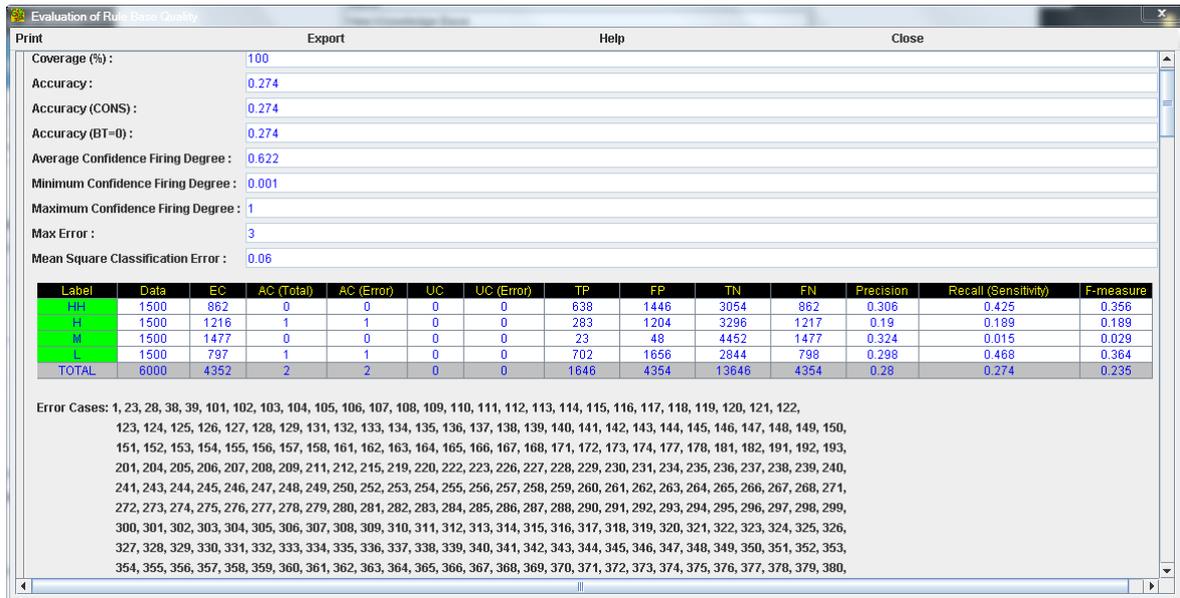
Fonte: Produção do Autor

Entre as operações habilitadas as mais utilizadas neste projeto foi **Simplification**, **Inference** e **Quality**. Com **simplification** é possível diminuir a quantidade de regras gerando um novo conjunto menor com a mesma equivalência, **inference** permite inferir uma saída utilizando um novo conjunto de dados de entrada, com **quality** podem ser feitos teste de acurácia e interpretabilidade. Realizando o teste de acurácia, têm-se a possibilidade de gerar uma matriz de confusão, que é automaticamente gerada se a variável de saída é uma variável do tipo *categorical* ou binária.

Clicando no botão **Quality** e indicando os arquivos de testes, temos a janela de resultados da classificação como se visualiza na Figura B.9. Entre os resultados mais importantes podemos citar a cobertura dos dados utilizados (**Coverage(%)**) que indicam a quantidade de dados que accionam pelo menos uma regra de inferência gerada nas fases anteriores, a acurácia da classificação (**Accuracy**), a tabela principal

de resultados com a quantidade de dados (**Data**) para cada classe (**Label**) assim como os casos de erro (**EC**), os casos de True Positive (TP), sensibilidade da classe (*Sensibility*), entre outros.

Figura B.9 - Janela de resultados de classificação



Fonte: Produção do Autor

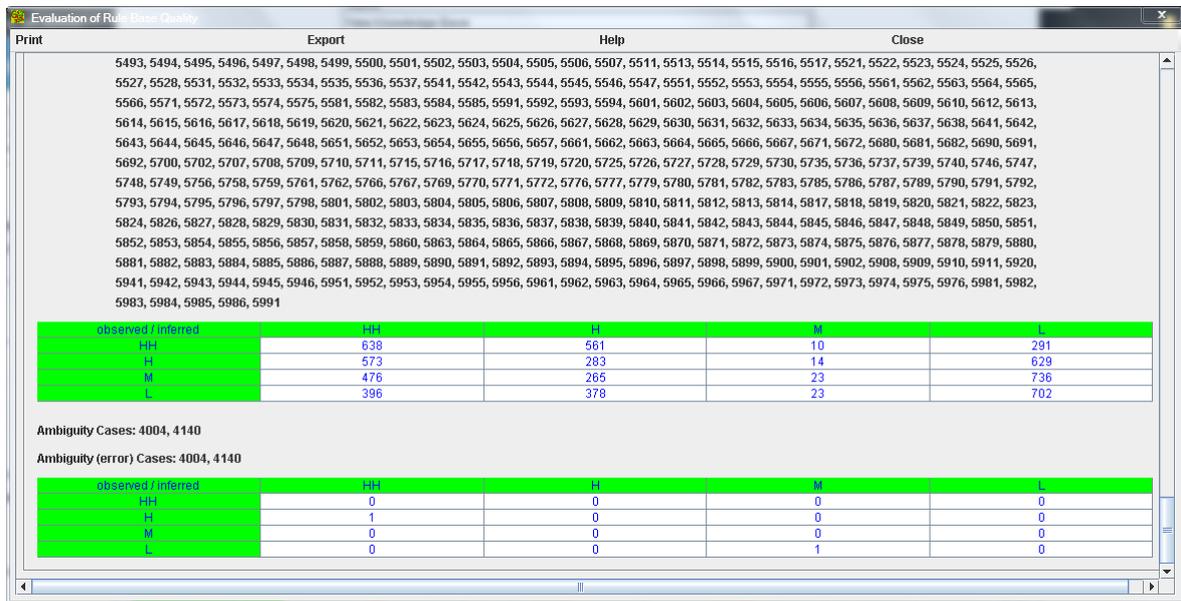
Dois limiares (**Blank Threshold**, **Ambiguity Threshold**) são definidos antes de iniciar a classificação. Estes limiares definem a partir de que valor uma entrada é considerada pelo conjunto de regras e os casos de ambiguidade. Cada erro de classificação é detalhado por meio da linha de posição na tabela de dados de entrada.

Blank Threshold (BT) define o valor mínimo de pertinência gerada por uma regra que deve ser considerado para que esse conjunto de valores de entrada seja classificado. Por exemplo, se a regra R_j gera um valor de pertinência de 0,2 na conclusão para classe C_i e o valor mínimo de $BT=0,1$ então essa entrada é classificada para essa classe C_i . Se nenhuma regra gera um valor de pertinência maior que BT então essa entrada não é classificada e a cobertura de dados (**Coverage (%)**) diminui. Essa entrada é detalhada como UC (**UnClassified**) na tabela principal.

Ambiguidade se dá quando duas regras distintas geram conclusões distintas e a diferença entre valores de pertinência é menor que o limiar definido pelo **Ambiguity**

Threshold (AT). Se o método de defuzificação é o **MaxCrisp** (defuzifica para a classe correspondente ao máximo valor de pertinência) e duas regras estão defuzificando para classes distintas com valores de pertinência levemente diferentes, o classificador não sabe concluir para que classe finalmente corresponde essa entrada. Para evitar este tipo problemas foi definido um limiar igual a zero, com isso diminuiram os casos de ambiguidade embora tenham-se casos em que distintas regras geram igual valor de pertinência para classes distintas.

Figura B.10 - Janela de resultados da classificação - Matrizes de confusão



Fonte: Produção do Autor

Finalmente, na Figura B.10 são mostrados os resultados da classificação utilizando matrizes de confusão para os dados devidamente classificados. Assim como os casos de ambiguidade encontrados.

APÊNDICE C - Primeiros testes de classificação realizados

C.0.1 Primeiros testes: Região de 100x100

O primeiros testes foram utilizando dados de América do Sul, com regiões de estudo de $Nx = Ny = 100$ e $Nx = Ny = 40$. Eram áreas maiores que a região final adotada já que o objetivo era dar uma cobertura para todo o continente americano, com a finalidade de capturar as dinâmicas da região. As simulações tinham como data de início o 02 de outubro as 00 Hs.

Neste exemplo se mostram os resultados da classificação no continente sul-americano, utilizando uma região de 100x100 pontos de grade. Com isso, temos em total 10.000 amostras de entrada-saída por previsão, como têm-se 60 previsões finalmente totalizam 600.000 amostras de entrada-saída para a região.

Como foram feitas 3 simulações com a mesma data de início para 3 anos diferentes (2012-2013-2014), em total têm-se 1.800.000 amostras. O primeiro algoritmo testado foi o classificador neuro-difuso através do software GUAJE, como a quantidade de dados gerava problemas para utilizar o software teve que ser feita uma amostragem aleatória de 500.000 dados de entrada.

Com este grupo menor de dados, foi projetado um classificador com rede neuro-difusa do tipo sugeno, com 5 classes de saída do tipo *singleton*.¹ A particularidade de utilizar Sugeno com saída *singleton* é que o próprio software reconhece o problema como de classificação e permite avaliar a rede gerando a matriz de confusão e as taxas de acertos e erros de cada classe.

Utilizando o algoritmo de geração de regras de inferência Wang-Mendel foram geradas as regras do tipo SE-ENTÃO utilizados para a tarefa de classificação

¹um *singleton* é um conjunto que possui um único elemento no seu universo de discurso com grau de compatibilidade igual a 1

Figura C.1 - Regras de inferência geradas

Rule	Type	Active	If Variable1	AND Variab..	AND Variab..	THEN Varia..
1	I	true	average	high	high	H
2	I	true	average	average	average	HH
3	I	true	low	low	low	M
4	I	true	high	high	average	M
5	I	true	high	average	average	H
6	I	true	average	high	average	H
7	I	true	average	low	low	M
8	I	true	low	average	average	M
9	I	true	average	average	high	H
10	I	true	low	average	low	H
11	I	true	low	low	average	M
12	I	true	high	high	high	M
13	I	true	high	average	high	H
14	I	true	average	average	low	HH
15	I	true	average	low	average	HH

Fonte: Produção do autor, utilizando a saída do software GUAJE.

Na Figura C.2, se visualiza o resultado do processo de classificação utilizando o conjunto de treinamento. A rede possui 15 regras de inferência com os 100% de cobertura de dados (cada amostras ativa pelo menos uma regra de inferência) e note que para as classes com menos frequência de dados (classe (LL) e classe (L)) possuem uma alta taxa de erro de 100% embora tenha-se uma acurácia global de 86,8%. O motivo principal pela taxa de erro das classes de menos frequência é que não se têm regra com conseqüente apontando para essas classes (vide Figura C.1).

Figura C.2 - Classificação utilizado a o conjunto de treinamento

Label	Data	EC	AC (Total)	AC (Error)	UC	UC (Error)	TP	FP	TN
LL	56	56	0	0	0	0	0	0	499944
L	1503	1503	0	0	0	0	0	0	498497
M	7616	6937	3	2	0	0	677	7184	485200
H	53341	52976	19	12	0	0	353	445	446214
HH	437484	4531	25	19	0	0	432934	58407	4109
TOTAL	500000	66003	47	33	0	0	433964	66036	1933964

Fonte: Produção do autor, utilizando a saída do software GUAJE.

Os indicadores mais importantes da Figura C.2 são as colunas EC e TP ², onde o primeiro detalha os erros de cada classe e o segundo a quantidade de acerto de cada classe. Com esses valores podemos inferir as taxas de acerto e de inferência de cada classe.

C.0.2 Aumentando a quantidade de atributos

Com o objetivo de aumentar a acurácia de classificação foram feitos vários testes aumentando os atributos de entrada até chegar aos mencionados na seção 6.1.1.

Com os resultados dos primeiros testes foram obtidos resultados aceitáveis de acurácia global, mas essa acurácia era devido a boas classificações nas classes com maior representatividade. Procurando melhores classificações já atingindo as classes com menor representatividade, ao conjunto inicial de dados de entrada foi adicionado de forma progresiva, atributos de entrada a fim de fornecer maior informação referente ao ponto $\{i, j\}$ em questão.

Informações do *bred vector* do vento, pressão, umidade entre outras variáveis e também o valor da previsão de controle do processo de *breeding* para esse ponto $\{i, j\}$ foi adicionado ao conjunto de dados de entrada.

Todos os teste desta seção foram feitos utilizando a rede neuro-difusa a través do software GUAJE e a data correspondente à simulação desde 02 até 17 de outubro de 2014.

²EC: *error case*, TP: *true positive*

Figura C.3 - Resultados da classificação utilizando o software GUAJE com 3 atributos de entrada.

Variable4											
Coverage (%) :	100										
Accuracy :	0.587										
Accuracy (CONS) :	0.587										
Accuracy (BT=0) :	0.587										
Average Confidence Firing Degree :	0.609										
Minimum Confidence Firing Degree :	0										
Maximum Confidence Firing Degree :	1										
Max Error :	3										
Mean Square Classification Error :	0.027										
Label	Data	EC	AC (Total)	AC (Error)	UC	UC (Error)	TP	FP	TN	FN	
HH	4989	1883	1	1	0	0	3106	536	475	1883	
H	950	530	1	1	0	0	419	1939	3111	531	
M	38	38	0	0	0	0	0	0	5962	38	
L	14	14	0	0	0	0	0	0	5986	14	
LL	9	9	0	0	0	0	0	0	5991	9	
TOTAL	6000	2474	2	2	0	0	3525	2475	21525	2475	

Fonte: Produção do autor, utilizando a saída do software GUAJE

Na Figura C.3 se visualizam os resultados da classificação utilizando uma rede neuro-difusa com 3 atributos de entrada (assim como foi explicado na Figura 6.5). Com esse esquema obtém-se uma acurácia global de 58.7%, mas somente obtendo com boas taxas de acertos ³ para as classes HH e H, o resto das classes não consegue identificar.

Esta falta de distinção para as classes com menor representatividade é devido a que, com poucos atributos de entrada, somente foram geradas 14 regras de inferência, todas com conclusões para as classes HH e H.

Por exemplo, a regra 1 se interpreta da seguinte forma: se uma entrada tem o valor da variável 1 dentro do domínio do conjunto difuso *average* e o valor da variável 2 dentro do domínio do conjunto difuso *low* e o valor da variável 3 dentro do conjunto difuso *low*, então essa entrada corresponde a uma classe H.

³coluna TP (*true positive*) da Figura C.3

Figura C.4 - Regras de inferência geradas utilizando o software GUAJE

Rule	Type	Active	If Variable1	AND Variab..	AND Variab..	THEN Varia..
1	I	true	average	low	low	H
2	I	true	low	average	average	H
3	I	true	average	high	average	H
4	I	true	high	average	high	HH
5	I	true	average	high	high	H
6	I	true	average	average	low	HH
7	I	true	average	average	high	HH
8	I	true	low	average	low	HH
9	I	true	high	average	average	HH
10	I	true	low	low	average	H
11	I	true	average	average	average	HH
12	I	true	average	low	average	HH
13	I	true	low	low	low	HH
14	I	true	high	high	high	H

Fonte: Produção do autor.

Aumentando a quantidade de atributos de entrada permitiu uma evolução positiva da acurácia global classificação, este ganho incrementou também a produção de regras de inferência aparecendo mais variedades de conclusões, que no início não se tinha, para as distintas classes.

Figura C.5 - Resultados da classificação utilizando o software GUAJE com 16 atributos de entrada

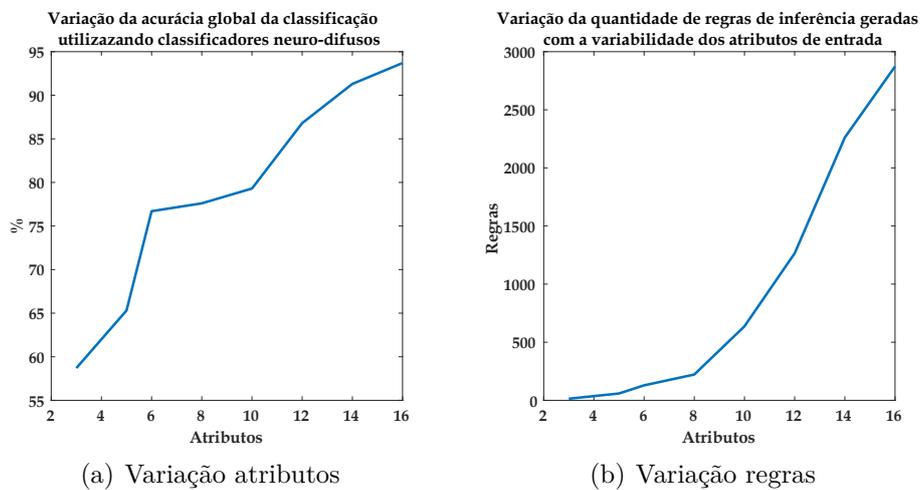
Label	Data	EC	AC (Total)	AC (Error)	UC	UC (Error)	TP	FP	TN	FN
1.0	4989	203	0	0	0	0	4786	148	863	203
2.0	950	156	0	0	0	0	794	206	4844	156
3.0	38	16	0	0	0	0	22	16	5946	16
4.0	14	5	0	0	0	0	9	3	5983	5
5.0	9	0	0	0	0	0	9	7	5984	0
TOTAL	6000	380	0	0	0	0	5620	380	23620	380

Fonte: Produção do autor, utilizando a saída do software GUAJE

Na Figura C.5 pode-se visualizar os resultados de uma classificação utilizando uma rede neuro-difusa onde se obteve uma acurácia global de 93,7% e uma taxa de acerto de 56,25% para a última classe LL com somente 9 amostras.

Este ganho na acurácia global teve como consequência uma perda da interpretabilidade do sistema devido ao aumento de regras de inferência, as conclusões para as distintas classes são mais distribuídas o que pode ser visto na matriz de confusão C.1.

Figura C.6 - Variação da percentagem da acurácia global da classificação e o aumento das regras de inferência conforme o aumento de atributos



(a) Variação atributos

(b) Variação regras

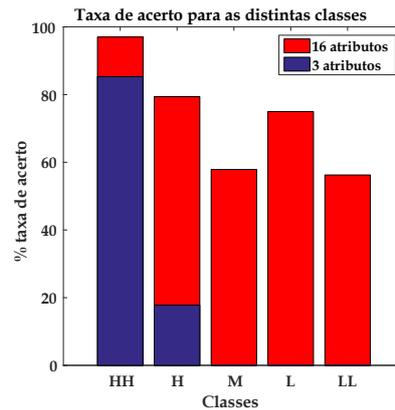
Fonte: Produção do autor

Tabela C.1 - Matriz de Confusão do conjunto de treinamento com 16 atributos de entrada

		Inferido				
		Classe 1(HH)	Classe 2(H)	Classe 3(M)	Classe 4(L)	Classe 5(LL)
Observado	Classe 1(HH)	4786	196	7	0	0
	Classe 2(H)	144	794	9	2	1
	Classe 3(M)	4	8	22	1	3
	Classe 4(L)	6	2	0	9	3
	Classe 5(LL)	0	0	0	0	9

A taxa de acerto das classes L e LL aumentaram conforme foram adicionados os atributos de entrada, por exemplo como se visualiza na Figura C.7 de um conjunto de 3 atributos somente se tinha inferência para as primeiras classes mas logo se melhoraram ditas taxas e além disso, também a inferência para as outras classes. É por este motivo que todas as análises foram feitas com 16 atributos de entrada.

Figura C.7 - Variação da taxa de acerto para 3 e 16 atributos de entrada



A taxa de acerto é calculada a partir da matriz de confusão sendo a taxa entre o que acertou dividido a somatória total de inferência para a classe correspondente.

Fonte: Produção do autor

Com um conjunto de 3 atributos se tinham taxas de acerto de 85,3% para a classe HH e 17,7% para a classe H mas com um conjunto de 16 atributos se tem taxas de 97 % para a classe HH, 79,4% para a classe H, 57,9% para a classe M, 75% para a classe L e 56,25% para a classe LL.

PUBLICAÇÕES TÉCNICO-CIENTÍFICAS EDITADAS PELO INPE

Teses e Dissertações (TDI)

Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

Manuais Técnicos (MAN)

São publicações de caráter técnico que incluem normas, procedimentos, instruções e orientações.

Notas Técnico-Científicas (NTC)

Incluem resultados preliminares de pesquisa, descrição de equipamentos, descrição e ou documentação de programas de computador, descrição de sistemas e experimentos, apresentação de testes, dados, atlas, e documentação de projetos de engenharia.

Relatórios de Pesquisa (RPQ)

Reportam resultados ou progressos de pesquisas tanto de natureza técnica quanto científica, cujo nível seja compatível com o de uma publicação em periódico nacional ou internacional.

Propostas e Relatórios de Projetos (PRP)

São propostas de projetos técnico-científicos e relatórios de acompanhamento de projetos, atividades e convênios.

Publicações Didáticas (PUD)

Incluem apostilas, notas de aula e manuais didáticos.

Publicações Seriadas

São os seriados técnico-científicos: boletins, periódicos, anuários e anais de eventos (simpósios e congressos). Contam destas publicações o Internacional Standard Serial Number (ISSN), que é um código único e definitivo para identificação de títulos de seriados.

Programas de Computador (PDC)

São a seqüência de instruções ou códigos, expressos em uma linguagem de programação compilada ou interpretada, a ser executada por um computador para alcançar um determinado objetivo. Aceitam-se tanto programas fonte quanto os executáveis.

Pré-publicações (PRE)

Todos os artigos publicados em periódicos, anais e como capítulos de livros.