Pós-Graduação em Ciência da Computação

ANDERSON TENÓRIO SERGIO

SELEÇÃO DINÂMICA DE COMBINADORES DE PREVISÃO DE SÉRIES TEMPORAIS

Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE
2017

Anderson Tenório Sergio

Seleção Dinâmica de Combinadores de Previsão de Séries Temporais

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Doutor em Ciência da Computação.

ORIENTADORA: **Profa. Teresa Bernarda Ludermir**

RECIFE
2017

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

S484s Sergio, Anderson Tenório
Seleção dinâmica de combinadores de previsão de séries temporais / Anderson Tenório Sergio. – 2017.
128 f.: il., fig., tab.

Orientadora: Teresa Bernarda Ludermir.
Tese (Doutorado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2017.
Inclui referências e apêndice.

1. Inteligência artificial. 2. Previsão de séries temporais. I. Ludermir, Teresa Bernarda (orientadora). II. Título.

006.3 CDD (23. ed.) UFPE- MEI 2017-128

Anderson Tenório Sergio

Seleção Dinâmica de Combinadores de Previsão de Séries Temporais

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutora em Ciência da Computação.

Aprovado em: 17/03/2017.

_____

Orientadora: Profa. Dra. Teresa Bernarda Ludermir

BANCA EXAMINADORA

_____

Prof. Dr. Adriano Lorena Inácio Oliveira
Centro de Informática / UFPE

_____

Prof. Dr. Leandro Maciel Almeida
Centro de Informática / UFPE

_____

Profa. Dra. Aida Araújo Ferreira
Instituto Federal de Pernambuco/IFPE

_____

Profa. Dra. Anne Magaly de Paula Canuto
Departamento de Informática e Matemática Aplicada/UFRN

_____

Profa. Dra. Tatijana Stosic
Departamento de Estatística e Informática/UFRPE

Aos meus pais, que desde sempre acreditaram em mim.

Agradecimentos

Faz pouco mais de 25 anos que entrei no colégio. Hoje, ao final de mais um ciclo, sinto que devo agradecimentos a todos que passaram por minha vida acadêmica nesse período. Foi uma longa e satisfatória caminhada (às vezes agoniada e cansativa!), desde o jardim de infância à escrita de artigos científicos.

Agradeço a minha amada e querida esposa, que sempre me oferece um porto seguro de compreensão e incentivo. Aunia, você é a razão de tudo que eu faço, hoje e sempre. Minhas realizações são suas também!

Agradeço a minha família primordial. Meus pais, Edite e Edson, que sempre me incentivaram nos estudos e que me fizeram acreditar que os meus sonhos seriam conquistados. Minha irmã, Andreza, que compartilhou comigo inesquecíveis e divertidos momentos da minha vida. Meus avós, Ló (*in memorian*) e Edvaldo, Edilene e Nelson (*in memorian*), que também são parte do que sou hoje.

Agradeço aos parceiros, colegas e amigos que fiz na escola (Instituto Santa Tereza, Liceu de Artes e Ofícios e Colégio Nóbrega) e na faculdade (Escola Politécnica de Pernambuco e Centro de Informática). Seu companheirismo e apoio foram diretamente responsáveis pelos meus êxitos.

Agradeço aos meu mestres, que me introduziram nos mais diversos saberes e no desejo de aprender, sempre. Em especial a minha orientadora Teresa Bernarda Ludermir, com toda a sua sabedoria e dedicação aos seus alunos e a seu papel como profissional acadêmica.

Obrigado a todos! Entretanto, este não é o fim. “A princesa está em outro castelo”.

Try not. Do, or do not. There is no try.
(Master Yoda)

Resumo

A previsão de séries temporais é um importante campo de estudo em aprendizado de máquina. Já que a literatura mostra diversas técnicas para a solução desse problema, combinar saídas de diferentes modelos é uma estratégia simples e robusta. Entretanto, mesmo quando se usam tais combinadores, o experimentador pode encarar o seguinte dilema: qual técnica deve ser usada para combinar os preditores individuais? Este trabalho apresenta um arcabouço para seleção dinâmica de combinadores de previsão de séries temporais. O processo de seleção dinâmica pode ser resumido em três fases. A primeira delas é responsável pela geração do conjunto de especialistas base, sendo que esse conjunto pode ser formado por modelos de mesma natureza ou heterogêneos. A diversidade dos especialistas é importante em ambas as situações. A segunda fase, de seleção, é realizada através da estimação da competência dos modelos disponíveis no conjunto gerado na primeira fase, em respeito a regiões locais do espaço de características. No caso da seleção dinâmica, a escolha dos modelos é realizada para cada padrão de teste, ao invés de utilizar a mesma seleção para todos eles (seleção estática). A terceira fase é a integração dos modelos selecionados. No método proposto, foram utilizados como preditores individuais modelos estatísticos (lineares e não-lineares) e de aprendizado de máquina. Em relação aos combinadores, foram utilizadas algumas técnicas que usam uma base de dados independente para determinação dos pesos da combinação linear e outros métodos que não possuem essa necessidade. Foram propostos dois algoritmos de seleção dinâmica, baseados em acurácia e comportamento. Para cada um deles, foram implementadas variações no que diz respeito ao uso de todos ou dos melhores preditores e combinadores do comitê. Para testar o método proposto, dez séries temporais caóticas foram utilizadas: Mackey-Glass, Lorenz, Rossler, Henon, Periodic, Quasi-Periodic, Laser e três séries produzidas a partir de exames de eletroencefalograma. A previsão de séries caóticas tem importância para várias áreas de atuação humana como astronomia e processamento de sinais, sendo que algumas das séries que foram testadas também funcionam como *benchmark* em diversas pesquisas. As melhores variações dos algoritmos de seleção dinâmica propostos alcançaram resultados satisfatórios em todas as bases de dados. Após a realização de testes estatísticos, comprovou-se que os métodos foram superiores aos melhores combinadores e preditores base na maioria dos cenários, para previsão de curto e longo alcance.

Palavras-chave: Previsão de séries temporais. Seleção dinâmica. Séries temporais caóticas. Comitês de previsão de séries temporais. Combinação de previsão de séries temporais.

Abstract

Time series forecasting is an important research field in machine learning. Since the literature shows several techniques for the solution of this problem, combining outputs of different models is a simple and robust strategy. However, even when using combiners, the experimenter may face the following dilemma: which technique should one use to combine the individual predictors? This work presents a framework for dynamic selection of forecast combiners. The dynamic selection process can be summarized in three steps. The first one is responsible for the generation of the base experts set, and this set can be formed by models of the same kind or heterogeneous ones. The diversity of the experts is important in both cases. The second phase (selection) is carried out by estimating the competence of the available models in the set generated in the first phase, with respect to local regions of the feature space. In the case of dynamic selection, the model selection is performed for each test pattern, instead of using the same selection for all of them (static selection). The third phase is the integration of the selected models. In the proposed method, predictors from statistics (linear and nonlinear) and machine learning were used. As combiners, we chose techniques that use extra data and some others that do not require an independent dataset for determining the weights of the linear combination. Two dynamic selection algorithms were proposed, based on accuracy and behavior. For each of them, variations were implemented with respect to the use of all or the best predictors and combiners of the pool. To test the proposed method, ten chaotic time series were used: Mackey- Glass, Lorenz, Rossler, Henon, Periodic, Quasi-Periodic, Laser and three time series produced from electroencephalogram exams. The prediction of chaotic series is important for many areas of human activity such as astronomy and signal processing, and those that were tested also are used as benchmark in several works. The best variations of the proposed dynamic selection algorithms have achieved satisfactory results in all databases. After performing statistical tests, it was verified that the methods were superior to the best combiners and predictors based on most scenarios, for short and long term forecasting.

Keywords: Time series forecasting. Dynamic selection. Chaotic time series. Time series forecasting ensemble. Time series forecasting combining.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AR	*Autoregressive*
ARCH	*Autoregressive Conditional Heteroscedasticity*
ARIMA	*Autoregressive Integrated Moving Average*
ARMA	*Autoregressive Moving Average*
DBN	*Deep Belief Network*
DL	*Deep Learning*
DSFC-A	*Dynamic Selection of Forecast Combiners - Acuracy*
DSFC-B	*Dynamic Selection of Forecast Combiners - Behavior*
ELM	*Extreme Learning Machines*
ESN	*Echo State Network*
FANN	*Feedforward Artificial Neural Network*
GARCH	*Generalized Autoregressive Conditional Heteroscedasticity*
GRNN	*Generalized Regression Neural Network*
LCA	*Local Class Accuracy*
LSM	*Liquid State Machines*
MA	*Moving Average*
MCB	*Multiple Classifier Behavior*
OLA	*Overall Local Accuracy*
PSO	*Particle Swarm Optimization*
PST	Previsão de Séries Temporais
RBLC	*Rank-Based Linear Combination*
RNA	Redes Neurais Artificiais
RNR	Redes Neurais Recorrentes
SDAE	*Stacked Denoising Autoencoders*

SVM	*Support Vector Machines*
SVR	*Support Vector Regression*

Sumário

1 INTRODUÇÃO

Este capítulo apresenta a introdução do trabalho, contendo principalmente o contexto e a motivação da utilização da seleção dinâmica no problema de previsão de séries temporais. Em seguida, são apresentados os objetivos da pesquisa, bem como as contribuições da tese. Ao final, a estrutura do documento é mostrada.

1.1 Contexto e Motivação

A previsão de séries temporais (PST) é um dos problemas tradicionais em estatística e aprendizado de máquina. Nesse tipo de problema, valores passados de uma determinada medição são coletados e posteriormente utilizados na previsão de valores futuros. Ao longo dos anos, diversas técnicas de aprendizado de máquinaa vêm sendo utilizadas para esse fim, por exemplo: Redes Neurais Artificiais (RNN) (LI; HAN; WANG, 2012) (YAN, 2012), Máquinas de Vetor de Suporte (SVM - *Support Vector Machines*) (BAO; XIONG; HU, 2014) (ZHANG et al., 2013), técnicas de lógica difusa (EGRIOGLU; ALADAG; YOLCU, 2013) (MELIN et al., 2012) e sistemas híbridos de algumas dessas técnicas com computação evolucionária (DONATE et al., 2013) (FERREIRA; LUDERMIR; AQUINO, 2013) ou inteligência de enxames (ZHIQIANG; HUAIQING; QUAN, 2013) (SERGIO; LUDERMIR, 2014). Embora mais antigos, modelos estatísticos são ainda estudados e utilizados com resultados satisfatórios. Entre esses preditores, podemos encontrar modelos lineares como AR (*Autoregressive*), MA (*Moving Average*), ARMA (*Autoregressive Moving Average*) e ARIMA (*Autoregressive Integrated Moving Average*) (ROJAS et al., 2008) e não-lineares, como ARCH (*Autoregressive Conditional Heteroscedasticity*) e GARCH (*Generalized ARCH*) (FIRMINO; NETO; FERREIRA, 2015). A literatura mostra diversas aplicações de PST em problemas do mundo real, nas mais diversas áreas de atuação humana: energia (LIU et al., 2013), mercado financeiro (ARAÚJO; OLIVEIRA; MEIRA, 2015), agricultura (ZHAI et al., 2012), meteorologia (VOYANT et al., 2015), epidemiologia (SHAMAN; YANG; KANDULA, 2014), controle de tráfego (MORETTI et al., 2015), atividades sísmicas (TIAMPO; SHCHERBAKOV, 2012) etc.

Uma importante categoria de séries temporais são as chamadas séries caóticas. A teoria do caos é o campo de pesquisa da matemática que estuda o comportamento de sistemas dinâmicos (KELLERT, 1994) (LORENZ, 1963). Sistemas dinâmicos são altamente sensíveis às condições iniciais, propriedade popularmente conhecida como efeito borboleta. Séries temporais com comportamento dinâmico constituem uma importante classe de base de dados para *benchmark* de modelos e aplicações práticas como processamento de sinais (KENNEL; ISABELLE, 1992), hidrologia (YANG et al., 2011), astronomia (CHANDRA; ZHANG, 2012) e biomedicina (SAMANTA, 2011).

Dada a grande variedade de modelos para a previsão de séries temporais, a dúvida sobre qual deles utilizar naturalmente emerge nos estágios iniciais da tentativa de resolução do problema. Adicionalmente, levando-se em consideração o teorema da inexistência do "almoço grátis" (WOLPERT, 1996), não há qualquer garantia de que um determinado modelo tenha um desempenho aceitável para todas ou mesmo mais de uma categoria de base de dados. Uma possível solução para esse cenário é a combinação de previsões. A combinação de previsões vêm sendo utilizada de maneira efetiva desde trabalhos seminais como o de Bates e Granger (BATES; GRANGER, 1969). Um comitê utiliza múltiplos especialiatas para melhorar o desempenho de um modelo individual (DIETTERICH, 2002). Medidas estatísticas simples podem ser utilizadas para combinar previsões geradas por um comitê de modelos distintos, como média, mediana ou média aparada (ELLIOTT; TIMMERMANN, 2013) (JOSE; WINKLER, 2008). Modelos mais sofisticados podem ser usados para combinar as saídas do comitê, já que combinadores estatísticos não consideram o desempenho relativo dos modelos individuais e são mais apropriados quando os especialistas base alcançam resultados semelhantes (ADHIKARI, 2015). Por outro lado, a combinação também pode ser realizada de maneira não-linear como visto nos trabalhos de Adhikari e Agrawal (ADHIKARI; AGRAWAL, 2012) e de Gheyas e Smith (GHEYAS; SMITH, 2011). A justificativa para o uso de combinadores não-lineares é o fato de que combinadores lineares somente consideram as contribuições individuais de cada preditor, mas não o relacionamento entre eles.

Apesar de normalmente a utilização de combinadores melhorar substancialmente o desempenho de preditores base, a grande quantidade de métodos disponíveis faz surgir uma questão fundamental: qual combinador utilizar? Não há qualquer indicação formal sobre quais métodos são melhores em determinadas situações. A resposta para essa questão pode estar na seleção dinâmica. A seleção dinâmica é uma área emergente da aprendizado de máquina, com muitos trabalhos publicados nos últimos anos (um resumo das principais contribuições pode ser visto em (BRITTO; SABOURIN; OLIVEIRA, 2014)). Na maioria dos casos, o cenário de seu uso é na solução de problemas de classificação e reconhecimento de padrões (LIN et al., 2014) (GALAR et al., 2013) (KAPP; SABOURIN; MAUPIN, 2012). De maneira geral, o processo de seleção dinâmica pode ser abreviado em três fases: geração, seleção e integração. Na geração, um conjunto de especialistas é gerado. Na seleção, um subconjunto desses especialistas é selecionado dinamicamente (i.e., para cada padrão de teste) de acordo com algum critério. Finalmente, na fase de integração, uma decisão final é tomada a respeito de qual ou quais especialistas selecionados devem ser utilizados para a classificação ou previsão de um determinado padrão de entrada dos dados. Assim, a seleção dinâmica poderia representar uma maneira efetiva de, em um sistema de previsão de séries temporais, apresentar uma saída adequada levando-se em consideração um comitê de preditores e suas possíveis combinações.

Utilizar seleção dinâmica no contexto de PST provoca uma série de questões:

quais e quantos preditores base devem ser utilizados? Quais e quantos combinadores devem ser utilizados? Na seleção dinâmica, devem ser levados em consideração somente os especialistas com melhor desempenho de validação ou todos os modelos gerados? Qual estratégia de seleção dinâmica deve ser utilizada? Devem ser usados na seleção dinâmica todos os combinadores do modelo ou apenas aqueles com melhor desempenho de validação? Qual o comportamento da seleção dinâmica quando se varia o alcance das previsões? A tese apresentada neste trabalho tenta responder essas e outras questões de acordo com os objetivos apresentados a seguir.

1.2 **Objetivo Geral**

Este trabalho tem como objetivo principal propor e implementar um arcabouço de seleção dinâmica de combinadores de previsão de séries temporais. Preditores base são treinados e seus resultados são combinados de acordo com algumas técnicas. Já que pode haver variação a respeito da estratégia de seleção dinâmica utilizada, foram propostas duas abordagens independentes de seleção dinâmica para previsão de séries temporais, baseados em abordagens aplicadas a problemas de classificação. Cada uma dessas abordagens é responsável por dinamicamente selecionar qual o melhor combinador para cada padrão apresentado ao sistema. Na implementação do modelo, algumas variações nos métodos de seleção foram testadas, e seus resultados serão apresentados e discutidos. Como exemplo de alguma dessas variações, podemos citar se a seleção dinâmica deve utilizar as combinações de todos os preditores do modelo ou somente daqueles que obtiverem melhor desempenho em uma base de dados de validação.

Os preditores base que necessitam de treinamento (i.e., as técnicas de aprendizado de máquina) são gerados segundo o algoritmo de Otimização por Enxame de Partículas (PSO, do inglês *Particle Swarm Optimization*). Nessa fase inicial, são definidos diversos parâmetros para geração de cada um dos modelos. Assim, o PSO é responsável pela otimização da busca do conjunto de parâmetros mais adequado para determinada base de dados.

Pela sua importância na academia e nas atividades humanas, séries temporais caóticas foram utilizadas como base de dados nos experimentos. Os testes foram realizados levando-se em consideração a previsão de curto e longo alcance, verificando assim a eficácia do modelo em contextos de uso distintos.

O uso de seleção dinâmica tem-se mostrado bem-sucedido em problemas de classificação e reconhecimento de padrões. De maneira geral, a seleção dinâmica é realizada através do cálculo de certas medidas de competência dos modelos que compõem o comitê. Essas medidas de competência podem atuar a partir de diferentes características do comitê, como por exemplo medidas de diversidade ou de probabilidade. O trabalho proposto buscou alcançar a diversidade dos preditores base a partir do uso de um comitê heterogêneo (foram

utilizados preditores tanto estatísticos como de aprendizado de máquina) e na preparação das bases de dados de treinamento.

A originalidade do trabalho vem do uso de técnicas de seleção dinâmica no contexto de PST. Outro ponto importante é que o método independe dos modelos base e dos combinadores selecionados na implementação, sendo assim um arcabouço para seleção dinâmica no problema de previsão de séries temporais.

1.3 Objetivos Específicos

O objetivo geral desse trabalho pode ser decomposto nos seguintes objetivos específicos:

- Implementação de métodos de previsão de séries temporais competitivos com resultados recentes alcançados na literatura. São eles: uma rede neural *feedforward* com uma camada escondida, uma rede neural *feedforward* com duas camadas escondidas, uma rede neural *deep learning* DBN (*Deep Belief Network*), uma rede neural *deep learning* SDAE (*Stacked Denoising Autoencoders*), um modelo SVR (*Support Vector Regression*) e modelos estatísticos (AR, MA, ARMA, ARIMA e GARCH).
- Implementação de métodos de combinação de preditores. São eles: média, média aparada, média winsorizada, mediana, RBLC (*Rank-Based Linear Combination*) e o método conhecido como softmax.
- Proposição e implementação de dois métodos de seleção dinâmica no problema de previsão de séries temporais.
- Proposição e implementação de um arcabouço para geração de preditores e seleção dinâmica de combinadores. Como explicitado anteriormente, as abordagens propostas nesta tese não se restringem à utilização dos modelos implementados e testados.
- Aplicação do novo método desenvolvido em séries temporais caóticas, para previsão de curto e longo alcance.
- Análise dos resultados e comparação do desempenho do novo método com resultados recentes alcançados na literatura.
- Análise do comportamento dos métodos implementados a partir da variação de alguns dos seus parâmetros.

1.4 Constribuições da Tese

Os tópicos discutidos nesta tese geraram as seguintes publicações científicas:

- SERGIO, A. T.; DE LIMA, T. P.F. ; LUDERMIR, Teresa B. . Dynamic Selection of Forecast Combiners. Neurocomputing (Amsterdam), v. 218, p. 37-50, 2016. (SERGIO; LIMA; LUDERMIR, 2016)
- SERGIO, A. T.; LUDERMIR, T. B.. Deep Learning for Wind Speed Forecasting in Northeastern Region of Brasil. In: Brazilian Conference on Intelligent Systems (BRACIS), 2015, Natal, RN. (SERGIO; LUDERMIR, 2015)
- LIMA, Tiago P. F.; SERGIO, A. T.; LUDERMIR, T. B.. Improving Classifiers and Regions of Competence in Dynamic Classifier Ensemble Selection. In: Joint Conference on Robotics and Intelligent Systems, 2014, São Carlos. Brazilian Conference on Intelligent Systems (BRACIS), 2014. (LIMA; SERGIO; LUDERMIR, 2014)
- SERGIO, A. T.; LUDERMIR, T. B. Reservoir computing optimization with a hybrid method. In: IEEE. 2014 International Joint Conference on Neural Networks (IJCNN), 2014. (SERGIO; LUDERMIR, 2014)

1.5 Estrutura do Documento

Esta tese encontra-se estruturada da seguinte forma:

1. **Capítulo 1: Introdução**
   Este capítulo apresentou o contexto e a motivação da tese, bem como os objetivos e as contribuições do trabalho. Por fim, a estrutura do documento está sendo apresentada.
2. **Capítulo 2: Previsão de Séries Temporais e Combinação de Preditores**
   O capítulo 2 abordará o tema da Previsão de Séries Temporais e da possível combinação de diversos preditores. Após o problema PST ser formalmente descrito, algumas de suas soluções possíveis serão apresentadas. Tais soluções englobam tanto modelos clássicos como modelos mais atuais. Será dada uma maior ênfase aos modelos utilizados no trabalho. A motivação da combinação de preditores também será discutida, seguida da apresentação de combinadores tanto tradicionais quanto mais recentes.
3. **Capítulo 3: Seleção Dinâmica**
   O capítulo 3 apresenta os conceitos que envolvem a seleção dinâmica. De maneira geral, a seleção dinâmica pode apresentar variações em algumas de suas fases (geração, seleção e integração). Algumas dessas variações serão discutidas. Nesse ponto, também será realizada uma revisão da literatura dos modelos mais recentes.
4. **Capítulo 4: Método Proposto**
   O capítulo 4 apresenta o método proposto. O método proposto é composto pela descrição das técnicas de seleção dinâmica apresentadas e pelo detalhamento e configuração dos experimentos realizados. Na seção de experimentos, também serão

apresentadas as séries temporais que foram utilizadas para testar o arcabouço desenvolvido.

5. **Capítulo 5: Resultados e Discussão**
   O objetivo do capítulo 5 é apresentar os resultados do método proposto e posteriormente discuti-los. Os resultados serão apresentados levando-se em consideração principalmente as medidas de erro dos modelos implementados, além de detalhes do processo de geração dos preditores base e análise de custo computacional. Os resultados alcançados serão comparados com trabalhos recentes da literatura.

6. **Capítulo 6: Conclusão e Trabalhos Futuros**
   O capítulo final discute as conclusões que podem ser levantadas a partir dos resultados alcançados. Adicionalmente, são listados alguns dos trabalhos que podem ser derivados a partir desta tese.

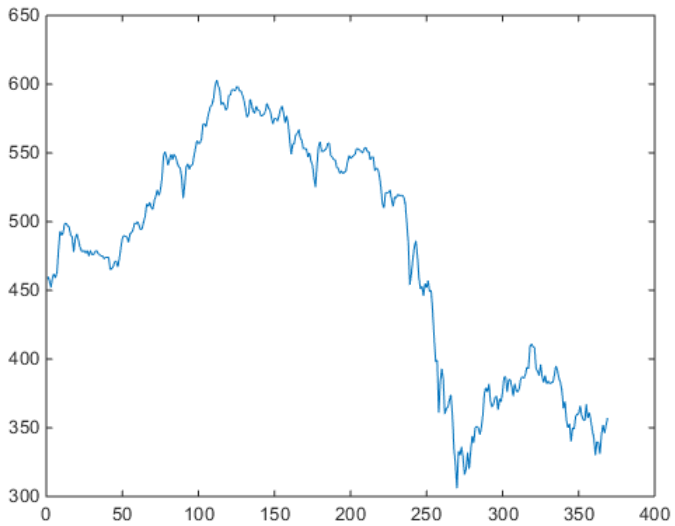
2 PREVISÃO DE SÉRIES TEMPORAIS E COMBINAÇÃO DE PREDITORES

O início deste capítulo define formalmente a previsão de séries temporais. Em seguida, são apresentados e discutidos alguns dos modelos que podem ser utilizados para resolver esse tipo de problema, bem como uma revisão da literatura relacionada. Finalmente, são apresentados os conceitos que envolvem a combinação de preditores.

2.1 Previsão de Séries Temporais

Séries temporais podem ser definidas como um conjunto de observações coletadas ao longo do tempo, em uma determinada ordem. Do ponto de vista estatístico, uma série de dados históricos pode ser tratada como uma sequência de variáveis aleatórias. Uma série temporal pode então ser referida como um processo estocástico discreto ao longo do tempo. Cada uma das observações podem ser consideradas como pontos em um gráfico de duas dimensões, onde o eixo das ordenadas determinam as medições dos dados e o eixo das abscissas delimitam em que momento discreto do tempo tais medições foram aferidas. A figura 1 mostra um exemplo de uma série temporal. Essa série mostra 363 medições do preço de uma ação da corporação IBM, entre 1961 e 1962 (DONATE et al., 2013).

Figura 1 – Série temporal IBM



Fonte: do autor.

Séries temporais são aferidas e analisadas nas mais diversas áreas de interesse humano, tais como finanças, economia, demografia, epidemiologia etc. A análise de séries temporais também envolve observações artificiais, onde os dados são gerados a partir de funções matemáticas. Tais medições auxiliam no estudo de diferentes características das

séries temporais. Algumas das mais importantes características de uma série temporal são as variações de tendência e sazonalidade. A tendência capta elementos de longo prazo relacionados com a série e a sazonalidade capta seus padrões regulares ao longo do tempo. Outra característica de interesse na análise de séries temporais é a presença ou ausência de estacionariedade. Uma série estacionária, ou convergente, tem seus pontos flutuando em torno de uma mesma média ao longo do tempo, em oposição a uma série não-estacionária (ou divergente). Outras características das séries temporais podem ser vistas em (COWPERTWAIT; METCALFE, 2009).

A análise de séries temporais pode abranger diversos propósitos, como por exemplo análise exploratória (ANDRIENKO; ANDRIENKO, 2006), ajuste de curva (BRADLEY et al., 2007), classificação (XI et al., 2006), análise de entropia (STOSIC et al., 2016), extração de características (CAVALCANTE; MINKU; OLIVEIRA, 2016) e segmentação (KEOGH et al., 2004). Esta tese trata da previsão de séries temporais. Seja uma série temporal univariada (objeto de estudo da tese) formalizada como uma sequência de observações escalares aleatórias $Y = \{y_1, y_2, ..., y_N\}$. A janela da série é dada pelo atraso utilizado para formar os padrões de treinamento e teste. Prever uma série temporal implica em descobrir um valor futuro da sequência, dado por $\hat{Y}_{i+1} = F[y_i, y_{i-k}, ..., y_{i-(d-1)k}]$. d é o atraso, k é o passo do atraso e F o modelo utilizado.

A previsão compreende extrapolar valores futuros baseados no passado e presente dos dados. Dessa forma, a previsão de séries temporais envolve risco e incerteza, visto que o comportamento da série não necessariamente segue o padrão histórico. Além disso, as observações aferidas podem conter *outliers* ou mesmo medições equivocadas. O modelo F deve, então, ser capaz de incorporar em sua generalização tais oscilações e imprecisões. Naturalmente, a literatura mostra numerosas formas de se construir o modelo de previsão. As técnicas mais tradicionais são aquelas originadas da estatística, lineares (como AR, MA, ARMA e ARIMA) ou não-lineares (como ARCH e GARCH). Mais recentemente, diversos modelos e algoritmos de aprendizado de máquina tem sido propostos para PST, como redes neurais artificiais e máquinas de vetor de suporte.

Uma das vantagens da utilização de modelos de aprendizado de máquina frente a técnicas estatísticas é o fato daqueles serem considerados "caixas-preta" e orientados a dados. Isso quer dizer que, a princípio, o experimentador não necessita conhecer de antemão a natureza dos dados que serão analisados, bastando que os parâmetros do modelo sejam ajustados adequadamente. O sucesso do uso de aprendizado de máquina na PST pode ser observado em diversas competições de previsão (NN3, 2016) (NN5, 2016). Por outro lado, a aplicação de técnicas estatísticas normalmente requerem menos esforço computacional. Para dados essencialmente lineares, técnicas mais simples podem ser suficientes.

Um aspecto importante da PST é o chamado horizonte de previsão. Na previsão de curto alcance (com poucos passos, comumente apenas um), o modelo deve ser capaz de

prever o valor imediatamente posterior da série temporal. Já na previsão de longo alcance (ou de n passos), os valores preditos são mais distantes dos dados utilizados para construir o modelo. De maneira geral, a previsão de longo alcance envolve complicações adicionais como acumulação de erros, acurácia reduzida e aumento de incerteza (SORJAMAA; LENDASSE, 2006).

Formalmente, a previsão de longo alcance consiste em prever h valores $\{y_{n+1}, ..., y_{n+h}\}$ de uma série temporal $\{y_1, ..., y_n\}$ composta de n observações, onde $h > 1$ denota o horizonte de previsão. As duas abordagens mais importantes na aprendizado de máquina para resolver esse problema são conhecidas como estratégia recursiva e estratégia direta (SORJAMAA et al., 2007).

Na estratégia recursiva, o modelo F é treinado normalmente para previsão de curto alcance. Em sua fase de utilização, com uma base de testes, são realizadas sucessivas previsões, utilizando as saídas prévias do modelo como entrada para previsão de horizontes maiores. A estratégia recursiva é conhecidamente mais sensível aos erros de estimação, já que valores futuros são utilizados como entrada para previsões mais longas.

Por outro lado, na estratégia direta, F é construído de maneira a incorporar o horizonte em sua construção. Em modelos de aprendizado de máquina, isso é feito modificando a base de dados no sentido de incluir como saída desejada o horizonte de previsão. Apesar de normalmente ser menos sensível a erros de estimação do que a estratégia recursiva, a estratégia direta frequentemente apresenta complexidade funcional mais alta e não é capaz de absorver as dependências estatísticas no conjunto de dados. Além disso, o número de horizontes de previsão desejados implica no número de modelos F que devem ser construídos.

2.1.1 Modelos de Previsão

A seguir, serão apresentados alguns modelos de previsão de séries temporais. A literatura mostra numerosas formas de se resolver esse problema. Serão levados em consideração os modelos que são diretamente relacionados com este trabalho.

2.1.1.1 Modelos Estatísticos

Os modelos estatísticos normalmente consideram a previsão de séries temporais como um problema de regressão. Nesse caso, é preciso estimar os parâmetros de uma função que represente a natureza do conjunto de dados. Inicialmente os modelos de previsão advindos da estatística eram em sua maioria lineares. Entretanto, a partir do final da década de 1970 e início da década de 1980, surgiram também modelos não-lineares. A seguir, são apresentadas as principais ideias de alguns desses modelos mais tradicionais.

De maneira geral, a regressão linear trata de ajustar retas ao conjunto de dados.

A partir de uma equação linear, a variável de interesse (também chamada de variável dependente) é prevista a partir de outras variáveis chamadas independentes. Se y denota a variável dependente e $x_1, ..., x_k$ são as variáveis independentes, então o valor de y no tempo (ou registro) t é determinado pela equação linear 2.1.

$$y_t = \beta_0 + \beta_1 x_t + \beta_2 x_{2t} + ... + \beta_k x_{kt} + \epsilon_t \tag{2.1}$$

onde os parâmetros β são as constantes do modelo que devem ser estimados (β_0 é chamado intercepto) e ϵ é uma distribuição normal com média 0, representando assim o ruído. A partir da estimação desses parâmetros, tem-se o modelo que representa a série. Dessa forma, valores futuros podem ser preditos.

Muitos conjuntos de dados exibem autocorrelação serial (i.e., uma associação linear atrasada). É nesse sentido que uma série temporal pode ser modelada levando-se em consideração observações passadas. O modelo autoregressivo (AR) (BOX et al., 2015) segue esse princípio, especificando que a variável de saída depende linearmente dos valores passados segundo um termo estocástico. A notação $AR(p)$ indica um modelo autoregressivo de ordem p, de acordo com a equação 2.2.

$$x_t = c + \sum_{i=i}^{p} \phi_i x_{t-1} + \epsilon_t \tag{2.2}$$

onde $\phi_1, ..., \phi_p$ são os parâmetros do modelo, c é uma constante e ϵ_t o ruído branco.

Seguindo o mesmo preceito, o modelo de médias móveis (MA) também utiliza informações passadas para predição de valores futuros. A notação $MA(q)$ refere-se ao modelo de médias móveis de ordem q, de acordo com a equação 2.3.

$$x_t = \mu + \sum_{i=i}^{q} \theta_i \epsilon_{t-1} + \epsilon_t \tag{2.3}$$

onde μ é a média da série, $\theta_1, ..., \theta_p$ são os parâmetros do modelo e $\epsilon_t, ..., \epsilon_{t-q}$ os termos de erro em ruído branco. O modelo AR indica que a variável de interesse pode ser obtida a partir de seus próprios atrasos, enquanto que o modelo MA indica que a predição pode ser obtida a partir de uma combinação linear dos termos de erro dos valores passados.

É possível combinar os dois polinômios do modelo AR e MA. Essa é justamente a definição do chamado modelo autorregressivo-médias móveis (ARMA) (BOX et al., 2015). A notação $ARMA(p, q)$ refere-se ao modelo ARMA com p termos autorregressivos e q termos de médias móveis, dado pela equação 2.4.

$$x_t = c + \epsilon_t + \sum_{i=i}^{p} \phi_i x_{t-1} + \sum_{i=i}^{q} \theta_i \epsilon_{t-1} \tag{2.4}$$

O modelo ARMA (assim como o MA) é sempre estacionário, o que quer dizer que as distribuições finito-dimensionais permanecem as mesmas sob translações no tempo. Entretanto, muitas séries temporais reais não são estacionárias devido a efeitos de tendência ou sazonalidade. Nessa perspectiva, o modelo autorregressivo integrado (ARIMA) apresenta-se como uma generalização do modelo ARMA, capaz de incluir termos que modelem comportamento não estacionário. No modelo ARIMA, o termo "integrado" indica a inclusão da diferença entre os valores atuais e os valores passados, sendo que tal diferenciação pode ser realizada mais de uma vez.

Na representação $ARIMA(p, d, q)$, p indica o número de termos autorregressivos (AR), d o número de diferenças (termo integrado) e q o número de termos de médias móveis (MA). $ARIMA(p, 0, q)$ implica em $ARMA(p, q)$. $AR(p)$ pode ser obtido com $ARIMA(p, 0, 0)$, enquanto que $ARIMA(0, 0, q)$ indica $MA(q)$.

O modelo $ARIMA(p, d, q)$ pode ser generalizado pela equação 2.5:

$$(1 - \sum_{i=1}^{p} \phi_i L^i)(1 - L)^d x_t = (1 + \sum_{i=1}^{q} \theta_i L^i)\epsilon_i \tag{2.5}$$

onde d é um inteiro positivo que determina o número de diferenças e L é o operador de defasagem.

AR, MA, ARMA e ARIMA possuem diversas variações. Algumas dessas variações são discutidas no trabalho de Box e Jenkins, pesquisadores que popularizaram o uso de tais modelos (BOX et al., 2015).

Os modelos autorregressivos condicionais heterocedásticos (ARCH) foram desenvolvidos no contexto do estudo de séries temporais econométricas não-estacionárias, como preço de ações. De maneira geral, a volatilidade variante no tempo produz séries com períodos de oscilações intercalados por períodos de estacionariedade.

No sentido de descrever o modelo ARCH, seja ϵ_t os termos de erros residuais em relação a uma média, de maneira similar ao modelo ARMA. Cada termo ϵ_t é dividido em um fator estocástico ζ_t e um desvio-padrão em relação ao tempo σ_t, resultando em $\epsilon_t = \sigma_t \zeta_t$. A notação $ARCH(q)$ implica em um modelo ARCH com variância dada pela equação 2.6:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i \epsilon_{t-1}^2 \tag{2.6}$$

onde os termos α são os parâmetros do modelo.

O chamado modelo autorregressivo condicional heterocedástico generalizado (GARCH) utiliza a variância como variável independente. A notação $GARCH(p, q)$ indica que há p

termos de variância e q termos de erros residuais, dada pela equação 2.7:

$$\sigma_t^2 = \omega + \sum_{i=1}^{q} \alpha_i \epsilon_{t-1}^2 + \sum_{i=1}^{p} \beta_i \sigma_{t-1}^2 \tag{2.7}$$

onde os termos α, β e ω são os parâmetros do modelo.

O modelo ARCH foi incialmente proposto por Engle, em 1982 (ENGLE, 1982). As diversas variações do modelo original podem ser vistas em (HENTSCHEL, 1995).

2.1.1.2 Modelos de Aprendizado de Máquina

O aprendizado de máquina explora o estudo e a construção de algoritmos que podem aprender a partir de dados e fazer predições. De acordo com Mitchell (MITCHELL, 1997), através do chamado aprendizado indutivo, um modelo aprende a partir da experiência E em relação a uma classe de tarefas T, com medida de desempenho P, se seu desempenho em T, medido por P, melhora com E.

Modelos de aprendizado de máquina são orientados a dados, o que significa dizer que, a princípio, não há necessidade de entender o conjunto de dados no qual o algoritmo será treinado. Por exemplo, dada uma série temporal, não é preciso decidir de antemão a respeito da linearidade ou não do modelo: os algoritmos são responsáveis pelo ajuste automático. Essa característica é frequentemente citada como uma vantagem da aprendizado de máquina frente a modelos estatísticos. O inconveniente na aprendizado de máquina advém da necessidade de treinamento dos algoritmos, onde dados são apresentados ao modelo de maneira a realizar a aproximação da função desejada. A aprendizado de máquina pode ser utilizada para resolver problemas como classificação, regressão, agrupamento e redução de dimensionalidade (MICHALSKI; CARBONELL; MITCHELL, 2013).

Desde a popularização da aprendizado de máquina, em meados de 1980, diversos modelos e algoritmos de treinamentos surgiram. A seguir, as linhas gerais de modelos que podem ser utilizados para resolver o problema da previsão de séries temporais.

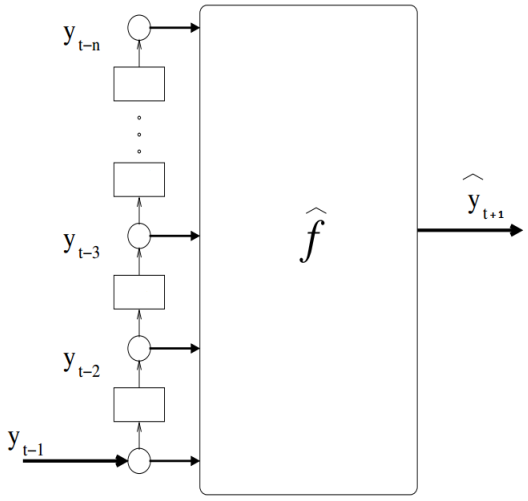
2.1.1.2.1 Redes Neurais *Feedforward*

Redes neurais artificiais *feedforward* (FANN, do inglês *Feedforward Artificial Neural Networks*) são sistemas paralelos distribuídos compostos por unidades de processamento simples que calculam funções matemáticas (HAYKIN, 2004). As unidades (chamadas nodos ou neurônios) são dispostas em camadas e interligadas por conexões. As conexões são associadas a determinados pesos, que são atualizadas de acordo com algum algoritmo de treinamento. Algumas das vantagens de seu uso são sua flexibilidade, boa capacidade de generalização e o fato de serem aproximadores universais. No contexto de previsão de séries temporais, uma rede neural com uma camada escondida pode ser representada como

um vetor p x h x 1, isto é, p neurônios na camada de entrada, h neurônios na camada escondida e um neurônio na camada de saída. O número de neurônios na camada de entrada é dado pela janela de atraso da série temporal utilizada para formar os dados de treinamento e teste. Com algum algoritmo de treinamento, como retropropagação ou Levenberg-Marquardt (HAYKIN, 2004), o erro de previsão tende a ser minimizado.

A figura 2 ilustra um modelo de rede neural para previsão de um passo. O aproximador $\hat{f}$ retorna a previsão do valor da série temporal no tempo $t + 1$, levando em consideração os n valores prévios da série.

Figura 2 – FANN para PST



Fonte: (BONTEMPI; TAIEB; BORGNE, 2013).

As chamadas redes neurais recorrentes (RNR) são uma classe de redes neurais com conexões entre neurônios que formam ciclos diretos, criando um estado interno capaz de exibir comportamento dinâmico. Essa característica faz com que as redes recorrentes sejam propícias ao problema de previsão de séries temporais. Uma das arquiteturas bastante difundidas que utilizam esse princípio são as redes de Elman (ELMAN, 1990).

O treinamento das redes recorrentes é em geral mais computacionalmente custoso do que em redes *feedforward*. Nesse sentido, mais recentemente surgiram modelos recorrentes com conexão esparsa entre as camadas e alguns pesos que podem ser calculados aleatoriamente, sem a necessidade de algum algoritmo de treinamento. Esses modelos são conhecidos como *Reservoir Computing*, sendo as redes com estado de eco (ESN, do inglês *Echo State Network*) (JAEGER, 2001) e as máquinas de estado líquido (LSM, do inglês *Liquid State Machines*) (MAASS; NATSCHLÄGER; MARKRAM, 2002) as arquiteturas proeminentes.

2.1.1.2.2 *Deep Learning*

Treinar redes neurais com mais de uma camada escondida tem sido de grande interesse para a comunidade de aprendizado de máquina após o surgimento da área conhecida como *Deep Learning* (BENGIO; COURVILLE; VINCENT, 2013) (DL). A profundidade de uma arquitetura diz respeito ao número de níveis de composição de operações não-lineares na função aprendida. Numa rede neural, por exemplo, a profundidade é dada pelo número de camadas. Atualmente, os algoritmos de aprendizagem mais comuns possuem arquitetura "rasa" (um, dois ou três níveis). Em contrapartida, o cérebro dos mamíferos é organizado em uma arquitetura profunda (SERRE et al., 2007). Funções que podem ser representadas por uma arquitetura de profundidade k podem requerer um número exponencial de elementos computacionais para que sejam representadas por uma arquitetura de profundidade $k-1$ (BENGIO, 2009). Isso faz com que algumas funções (e consequentemente alguns problemas no mundo real) não possam ser eficientemente representadas por arquiteturas que são muito rasas. Já que cada elemento computacional deve ser treinado, utilizando exemplos, é possível concluir que a profundidade pode ser muito importante para o desempenho das arquiteturas, sendo necessárias pesquisas de como treinar arquiteturas profundas de forma eficiente.

Por décadas, diversos pesquisadores tentaram, sem sucesso, treinar redes neurais de múltiplas camadas profundas (UTGOFF; STRACUZZI, 2002). Sendo que inicializadas com pesos aleatórios, as redes geralmente ficavam presas em mínimos locais. À medida que a profundidade aumentava, tornava-se ainda mais difícil uma boa generalização. Experimentos mostraram que os resultados de redes neurais profundas partindo de pesos inicializados aleatoriamente obtinham resultados piores que redes neurais com uma ou duas camadas escondidas (LAROCHELLE et al., 2009). Em 2006, Hinton et al. descobriram que os resultados de uma rede neural profunda poderiam ser sensivelmente melhorados quando pré-treinadas com um algoritmo de aprendizagem não-supervisionado, uma camada após outra e partindo da primeira camada (HINTON; OSINDERO; TEH, 2006). Esse trabalho inicializou a área hoje conhecida como *Deep Learning*.

Grande parte dos modelos de redes neurais conhecidas como *Deep Learning* compartilham as seguintes características: aprendizagem não-supervisionada das representações dos dados para pré-treinar cada uma das camadas; treinamento não-supervisionado cada camada por vez, sendo a representação aprendida em cada nível a entrada para a próxima camada; utilizar treinamento supervisionado para ajuste fino de todas as camadas pré-treinadas, bem como uma ou mais camadas adicionais dedicadas à produção de predições. Essas características também podem gerar diversidade no comitê, visto que um modelo *Deep Learning* possui uma forma particular de inicializar os pesos da rede, em contrapartida à utilização de pesos aleatórios em arquiteturas convencionais. Destacam-se entre os modelos de DL as redes neurais conhecidas como *Deep Belief Networks* (DBN) e *Stacked*

Denoising Autoencoders (SDAE).

Deep Belief Networks são redes neurais que seguem um modelo generativo probabilístico, inicialmente introduzidas em (HINTON; OSINDERO; TEH, 2006). Modelos generativos oferecem uma distribuição de probabilidades acerca de dados e rótulos, provendo a estimação de $P(Observação|Rótulo)$ e $P(Rótulo|Observação)$. Modelos discriminativos, como a maioria das redes neurais convencionais, somente disponibilizam $P(Rótulo|Observação)$. DBN têm o objetivo de mitigar alguns dos problemas do treinamento das redes neurais convencionais: necessidade de muitos dados rotulados no conjunto de treinamento; aprendizagem lenta e técnicas inadequadas de seleção de parâmetros que levam a ótimos locais.

DBN são baseadas em *Sigmoid Belief Networks*, rede neural de múltiplas camadas generativas propostas antes do surgimento de *Deep Learning*, treinadas com aproximações variacionais (DAYAN et al., 1995). Um exemplo de uma *Sigmoid Belief Network* pode ser vista na figura 3 (página 31). A rede é representada como um modelo gráfico direcionado, sendo cada variável aleatória representada por um nodo e arcos direcionados indicando dependência direta. O dado observado é x e os fatores escondidos no nível k são os elementos do vetor h^k. A parametrização das distribuições condicionais (em sentido para baixo) é similar à ativação de neurônios de redes convencionais, dada pela equação 2.8:

$$P(h_i^k = 1|h^{k+1}) = sigm(b_i^k + \sum_j W_{i,j}^{k+1} h_j^{k+1}) \tag{2.8}$$

onde h_i^k é a ativação binária do nó escondido i na camada k, h^k é o vetor $(h_1^k, h_2^k, \dots)$, sendo $x = h^0$.

A camada de baixo gera um vetor x no espaço de entrada. Considerando níveis múltiplos, o modelo generativo de uma *Sigmoid Belief Network* é dado pela equação 2.9.

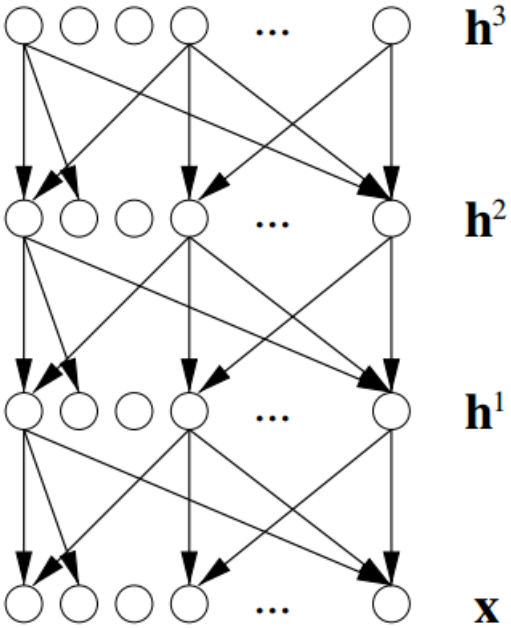
$$P(x, h^1, ..., h^l) = P(h^l)(\prod_{k=1}^{l-1} P(h^k|h^{k+1}))P(x|h^1) \tag{2.9}$$

Nesse caso, $P(x)$ é intratável na prática, exceto em modelos muito pequenos. A camada superior é então fatorizada, dada por $P(h^l) = \prod_i P(h_i^l)$.

Deep Belief Networks são bastante similares a *Sigmoid Belief Networks*. A principal diferença reside nas duas camadas superiores e pode ser vista na figura 4. $P(h^2, h^3)$ é dado por uma Máquina de Boltzmann Restrita (RBM, do inglês *Restricted Boltzmann Machines*).

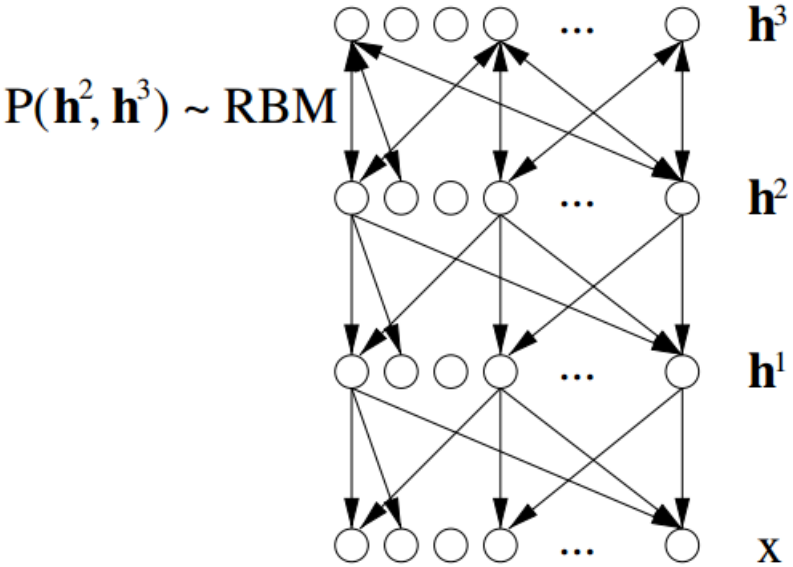
RBM, representadas pela figura 5 (página 32), são modelos gráficos não direcionados sem conexões entre nodos da mesma camada. São compostas por unidades de entrada (ou visíveis) x_j e unidades escondidas h_i, permitindo o cálculo de $P(h|x)$ e $P(x|h)$. A utilização

Figura 3 – *Sigmoid Belief Network*



Fonte: (BENGIO, 2009).

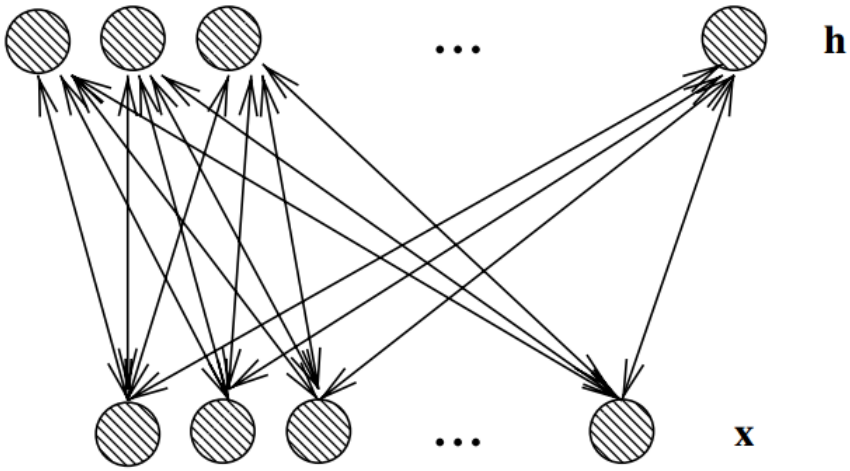
Figura 4 – *Deep Belief Network*



Fonte: (BENGIO, 2009).

de RBM é acompanhada de um algoritmo de aprendizagem que treina de forma gulosa uma camada de cada vez, construindo representações mais abstratas dos dados originais através de $P(h^l|x)$. Hinton chamou esse algoritmo de *Contrastive Divergence* (HINTON, 2002).

Figura 5 – Máquina de Boltzmann Restrita



Fonte: (BENGIO, 2009).

Em comparação com outros modelos de uma camada, RBM são mais atrativas como *building blocks* devido a sua facilidade de treinamento. De acordo com o algoritmo *Contrastive Divergence*, um vetor v é apresentado às unidades visíveis durante a fase de treinamento. Esses valores são propagados até as unidades escondidas. De maneira reversa, as entradas são estocasticamente calculadas visando a reconstrução dos dados originais, através de um processo conhecido como *Gibbs Sampling*. A correção dos pesos é dada pela diferença na correlação das ativações escondidas e entradas visíveis. O tempo de treinamento é sensivelmente reduzido e cada camada adicionada aumenta a probabilidade de obter os dados de treinamento.

Após esse pré-treinamento, uma DBN pode admitir um ajuste fino para um melhor desempenho discriminativo. O ajuste é realizado através da utilização de dados rotulados e outro algoritmo de treinamento, como por exemplo, a retropropagação de erros. Nesse caso, um conjunto de rótulos é anexado às camadas superiores da rede.

Autoencoders são um tipo de rede neural também conhecidas como redes Diabolo (BOURLARD; KAMP, 1988). De maneira geral, treinar um *autoencoder* é mais simples do que treinar uma RBM. *Autoencoders* têm sido utilizadas como *building blocks* em arquiteturas *Deep Learning*, onde cada nível é treinado separadamente (BENGIO et al., 2007) (POULTNEY et al., 2006).

Um *autoencoder* é treinado para codificar uma entrada x em alguma representação $c(x)$ de forma que ela possa ser reconstruída a partir dessa representação. A função de decodificação, $f(c(x))$ produz a reconstrução da rede, sendo geralmente um vetor de números obtidos através de uma função sigmoide. Espera-se que $c(x)$ seja uma representação distribuída dos dados que capture os fatores principais de suas variações.

De acordo com (BENGIO, 2009), o procedimento para treinar *autoencoders* empilhados, em um arquitetura profunda, é similar ao processo realizado em *Deep Belief Networks*:

1. Treinar a primeira camada como um auto-encoder de forma a minimizar o erro de reconstrução da entrada original, de modo puramente não-supervisionado.
2. Utilizar as saídas das unidades escondidas como entradas para outra camada, também treinadas como um auto-encoder. Esses dois passos não necessitam de dados rotulados.
3. Iterar o passo dois até inicializar o número desejado de camadas adicionais.
4. Utilizar a saída da última camada escondida como entrada para uma camada supervisionada e inicializar seus parâmetros aleatoriamente ou de forma supervisionada.
5. Realizar ajuste fino de todos os parâmetros dessa arquitetura profunda através de um critério supervisionado. De forma alternativa, desdobre todos os auto-encoders em um auto-encoder muito profundo e realize ajuste fino no erro de reconstrução global, como em (HINTON; SALAKHUTDINOV, 2006).

A ideia é que *autoencoders* devem ter baixo erro de reconstrução nos exemplos de treinamento, mas alto erro de reconstrução na maioria das outras configurações da entrada. *Autoencoders* podem ser regularizados para evitar a simples aprendizagem da função identidade. Um exemplo disso são os chamados *Denoising Auto-Encoders* (VINCENT et al., 2008), que utilizam versões com ruídos dos dados de entrada.

Experimentos comparativos mostram que *Deep Belief Networks* possuem desempenho levemente superior a *Stacked Autoencoders* (BENGIO et al., 2007) (POULTNEY et al., 2006). Entretanto, essa desvantagem desaparece quando auto-encoders simples são substituídos por *autoencoders* estocásticos (VINCENT et al., 2008). Uma vantagem dos *autoencoders* quando comparados a RBM é o fato de que eles permitem quase qualquer parametrização das camadas, além do que o critério de treinamento pode ser contínuo. Uma desvantagem é que *Stacked Auto-Encoders* não são modelos generativos. Isto é, eles não são capazes de gerar amostras que podem ter seu desempenho avaliado qualitativamente.

2.1.1.2.3 *Support Vector Regression*

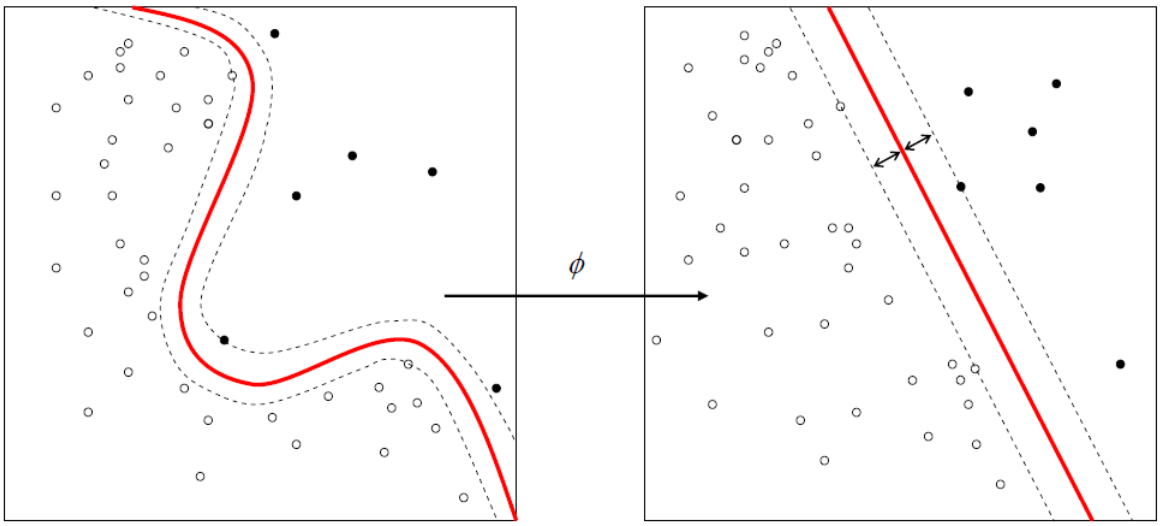
Máquinas de vetor de suporte (SVM, do inglês *Support Vector Machines*) são modelos de aprendizado de máquina baseados na Teoria da Aprendizagem Estatística, propostas por Vapnik e Chervonenkis (VAPNIK, 2013). O objetivo das SVM é minimizar o chamado risco estrutural através de um subconjunto de padrões de dados de treinamento chamados vetores de suporte. A minimização do risco estrutural é uma tentativa de tratamento para

o problema da escolha de uma dimensão VC (Vapnik-Chervonenkis) apropriada, sendo esta uma medida para a complexidade das hipóteses (funções) consideradas por um algoritmo de busca por soluções. De forma geral, podem ser citados como vantagens do uso de SVM sua alta capacidade de generalização, robustez para categorização de dados com dimensões altas e teoria bem estabelecida nas áreas de matemática e estatística.

Um SVM constrói um hiperplano em um espaço n-dimensional. O modelo original foi proposto no intuito de ser aplicado em dados linearmente separáveis nesse espaço, para o problema de classificação. Como podem existir vários hiperplanos separando os dados corretamente, o hiperplano ótimo é definido como aquele que possui a margem de separação mais alto e que minimiza os erros de treinamento e teste. A margem de separação de um classificador é definida como a menor distância entre exemplos do conjunto de treinamento e o hiperplano utilizado na separação desses dados em classes.

Para os dados que não são linearmente separáveis, a máquina de vetor de suporte pode ser construída através de um mapeamento do espaço de características original para um espaço com dimensão superior. Essa tarefa é realizada pela função de *kernel* $\phi(x, y)$. Diversas funções de *kernel* podem ser utilizadas, como funções polinomiais ou de tangente hiperbólica (BOSER; GUYON; VAPNIK, 1992). A figura 6 ilustra essa ideia.

Figura 6 – Função de *kernel* em um espaço de características



Fonte: do autor.

Uma versão do SVM para o problema de regressão foi proposta em 1997, denominado SVR (do inglês *Support Vector Regression*) (SMOLA; VAPNIK, 1997). Treinar o modelo SVR original implica em minimizar $1/2\|\omega\|^2$ de acordo com a equação 2.10.

$$\langle \omega, x_i \rangle + b - y_i \leq \epsilon \tag{2.10}$$

onde x_i é um exemplo de treinamento com y_i como valor desejado, $\langle \omega, x_i \rangle + b - y_i$ é a predição para esse exemplo e ϵ um parâmetro livre que atua como limiar.

2.2 Combinação de Previsão de Séries Temporais

Uma das principais motivações de combinar preditores vem da dificuldade em identificar um único melhor modelo e do desempenho orientado ao contexto no qual um preditor individual é aplicado. Ademais, a instabilidade de um determinado preditor pode ser mitigada quando um comitê é utilizado para gerar a previsão final, já que erros cometidos podem ser suavizados (TIMMERMANN, 2006). Uma justificativa teórica da combinação de previsões a partir de um modelo bayesiano pode ser vista em (HOETING et al., 1999). A utilização de comitês é bastante ampla em aprendizado de máquina, não somente em previsão de séries temporais (ALMEIDA; GALVÃO, 2016) (CANUTO; FAIRHURST; PINTRO, 2014).

É possível realizar a combinação de preditores com ou sem uma base de dados de validação (treináveis e não treináveis). Sem uma base de validação, a combinação é normalmente obtida através de cálculos estatísticos simples, como média ou mediana. Em contrapartida, alguns combinadores necessitam de parte do conjunto de dados para ajustar os pesos que são aplicados a cada um dos preditores base. A seguir, a apresentação de alguns desses combinadores, devidamente categorizados.

2.2.1 Combinadores Não Treináveis

A forma mais simples de combinar previsões é a combinação linear dos preditores. Seja $Y = \{y_1, y_2, ..., y_N\}$ a série temporal e $\hat{Y}^{(i)} = \{\hat{y}_1^{(i)}, \hat{y}_2^{(i)}, ..., \hat{y}_N^{(i)}\}$ as previsões obtidas a partir do i-ésimo método. A série obtida através da combinação linear dessas n séries previstas é dada pela equação 2.11:

$$\begin{cases} \hat{Y}^{(c)} = \{\hat{y}_1^{(c)}, \hat{y}_2^{(c)}, ..., \hat{y}_N^{(c)}\} \\ \hat{y}_k^{(c)} = w_1\hat{y}_k^{(1)} + w_2\hat{y}_k^{(2)} + ... + w_n\hat{y}_k^{(n)} = \sum_{i=1}^{N} w_i\hat{y}_k^{(i)} \\ \forall k = 1, 2, ..., N \end{cases} \tag{2.11}$$

onde w_i é o peso associado ao i-ésimo método de previsão. Os pesos normalmente têm soma igual a um, para evitar enviesamento.

De maneira geral, os métodos de combinação linear existentes na literatura variam na forma como esses pesos são calculados. Algumas dessas combinações são realizadas com um simples cálculo aritmético sobre todas ou algumas das previsões individuais. É o caso da média simples, onde são atribuídos pesos iguais a todos os modelos, isto é, $w_i = 1/n (i = 1, 2, ..., n)$. Apesar de simples, tal combinação tem se provado um método bastante robusto, por vezes utilizado como modelo de referência para comparações com

combinadores mais sofisticados (LEMKE; GABRYS, 2010). De maneira semelhante, a combinação pode ser feita com outras medidas estatísticas, como mediana, valor máximo ou valor mínimo.

A média aparada diferencia-se da média simples pelo fato de a média aritmética ser calculada excluindo $k\%$ dos modelos com pior desempenho. De acordo com (JOSE; WINKLER, 2008), o valor recomendado de k varia de 10% a 30%. A chamada média winsorizada é uma medida de tendência central que envolve o cálculo da média depois de descartar os valores mais extremos do conjunto de dados a partir de uma distribuição de probabilidades, frequentemente também de 10% a 30%. Teoricamente, a média aparada e a média winsorizada são estimadores mais robustos do que a média comum porque, ao mesmo tempo que fornecem uma medida central adequada, são menos sensíveis a dados discrepantes ou ruídos.

Todos esses combinadores estatísticos não necessitam de um conjunto de dados extra para cálculo dos pesos. Isso significa que, em um modelo de combinação de previsão, todo o conjunto de dados pode ser utilizado para geração dos preditores. A combinação, então, é obtida através de um cálculo aritmético simples, sem a necessidade de definição dos pesos por algum método mais sofisticado.

2.2.2 Combinadores Treináveis

Certos combinadores necessitam do desempenho dos modelos base em alguma base de dados de validação. Tal ideia é seguida nos chamados métodos baseados em erro, onde os pesos são escolhidos de maneira inversamente proporcional ao desempenho passado (ARMSTRONG, 2001a) e *outperformance*, onde os pesos são calculados de acordo como o número de vezes que determinado método foi melhor no passado (BUNN, 1975). Outra importante combinação nesse sentido é o softmax, calculada de acordo com as equações 2.12 e 2.13:

$$f_i^{''} = \frac{f_i^{'} - min(f^{'})}{max(f^{'}) - min(f^{'})} \tag{2.12}$$

$$w_{smi} = \frac{e^{(f_i^{''})}}{\sum_{k=1}^{N} e^{(f_k^{''})}} \tag{2.13}$$

onde $f^{'}$ é um inverso do erro de previsão em uma base de validação, $f_i^{'} = 1/f_i$ e $min(f^{'})$ e $max(f^{'})$ são os valores mínimo e máximo de todos os valores $f^{'}$. É possível observar através das equações 2.12 e 2.13 que os pesos da combinação não poderiam ser estipulados sem determinar o desempenho dos preditores em um conjunto de dados. O erro de previsão pode ser definido por uma simples subtração entre o valor desejado e o valor esperado.

A combinação linear baseada em *ranking* (RBLC, do inglês *Rank-Based Linear Combination*) também estipula os pesos através do desempenho dos preditores em um conjunto de dados (YAO; LIU, 1998). Em RBLC, os pesos são dados pela equação 2.14:

$$w_i = \frac{exp(\beta(N+1-i))}{\sum_{k=1}^{N} exp(\beta_j)} \tag{2.14}$$

onde β é um parâmetro que influencia o peso dos preditores com melhor desempenho.

O uso de um conjunto de dados para determinação dos pesos da combinação implica em um maior esforço computacional. Combinações como softmax e RBLC devem levar esse fato em consideração, visto que os combinadores mais simples, com média ou mediana, podem alcançar resultados similares através de um cálculo trivial.

2.3 **Revisão da Literatura**

2.3.1 Previsão de Séries Temporais

Os modelos apresentados na seção anterior são amplamente utilizados em previsão de séries temporais. A seguir, alguns trabalhos recentes em PST que utilizam os preditores discutidos previamente.

Modelos lineares, apesar de mais antigos, ainda são usados com sucesso em problemas atuais. Trabalhos recentes utilizaram o modelo ARIMA na análise e previsão de consumo de energia na China (YUAN; LIU; FANG, 2016) e uso de aerossol em Nova Déli, Índia (TANEJA et al., 2016). Dada sua importância entre os preditores mais utilizados, o modelo ARIMA foi testado em uma abordagem para modelagem de erro no intuito de aumentar a qualidade das previsões, em séries temporais financeiras (FIRMINO; NETO; FERREIRA, 2015). Modelos regressivos também foram recentemente utilizados para analisar séries temporais de eletroencefalograma (ATYABI; SHIC; NAPLES, 2016).

O modelo heterocedástico generalizado (GARCH) foi utilizado para prever preços de petróleo (KRISTJANPOLLER; MINUTOLO, 2016) e séries temporais de eletrocardiograma (MIHANDOOST; AMIRANI, 2017). No primeiro caso, as previsões foram obtidas através de uma hibridização entre o modelo GARCH e redes neurais artificiais. A hibridização de modelos estatísticos lineares e não-lineares pode ser observada na previsão de séries do mercado financeiro indiano (MIHANDOOST; AMIRANI, 2017) e chinês (SHI et al., 2012).

No caso de preditores advindos da aprendizado de máquina, o clássico artigo de Zhang et al. (ZHANG; PATUWO; HU, 1998) apresentou um *survey* das principais orientações no tocante à utilização de redes neurais artificiais para PST, incluindo a criação da base de dados de treinamento a partir de uma janela de atrasos de tempo. Desde então, redes neurais têm sido aplicados na previsão de séries tanto em suas arquiteturas originais

quanto em hibridizações com computação evolucionária (DONATE et al., 2013) e técnicas de lógica difusa (EGRIOGLU; ALADAG; YOLCU, 2013).

Arquiteturas específicas de RNA, como os modelos classificados como *Reservoir Computing*, também foram utilizados para prever séries temporais. As redes de estado de eco foram usadas para prever séries temporais caóticas em (LI; HAN; WANG, 2012) e séries de carga elétrica em (DEIHIMI; ORANG; SHOWKATI, 2013). As mesmas redes de estado de eco podem ter seus pesos treinados utilizando-se algoritmos genéticos (FERREIRA; LUDERMIR; AQUINO, 2013) ou recozimento simulado (SERGIO; LUDERMIR, 2014). Esses dois últimos trabalhos foram aplicados na previsão de séries de velocidade de ventos, no intuito de otimizar os recursos de usinas de energia eólica.

Os modelos *Deep Learning* foram originalmente utilizados em problemas de classificação e reconhecimento de padrões. Desde então, tais modelos passaram a ser aplicados nas mais diversas tarefas de aprendizado de máquina, dentre elas, a previsão de séries temporais. Apesar de a literatura possuir alguns trabalhos de DL com séries (alguns deles revisados a seguir), ainda não há resultados conclusivos sobre o papel do pré-treinamento no modelo e a relação entre a arquitetura e a dimensionalidade dos dados, informação essa provida geralmente pelos atrasos da série temporal.

Romeu et al. (ROMEU et al., 2013) utilizaram redes neurais profundas para prever uma série de temperaturas. Os autores utilizaram *Stacked Denoising Autoencoders* como pré-treinamento. O experimento levou em consideração o ajuste fino realizado somente na última camada ou em todas as camadas da rede. O desempenho foi melhorado quando comparado a redes sem pré-treinamento, mas não tanto quanto em outros tipos de aplicações. A justificativa seria a característica da série real utilizada e a baixa dimensionalidade dos dados.

Kuremoto et al. (KUREMOTO et al., 2014) utilizaram uma *Deep Belief Network* de apenas uma camada escondida para prever uma série temporal de competição. Segundo os experimentos dos autores, o desempenho da DBN foi melhor do que modelos como uma rede neural *feedforward*, aprendizado bayesiano e ARIMA. Na DBN, o desempenho foi melhorado com a utilização de dados diferenciados. Os hiperparâmetros do modelo (atraso da série temporal, número de neurônios na camada escondida e taxas de aprendizagem) foram otimizados por PSO.

Chen et al. (CHEN; JIN; CHAO, 2012) também empregaram uma *Deep Belief Network* para prever uma série contendo dados de índices de seca em uma bacia hidrográfica asiática. Como componente da DBN, os autores usaram uma RBM para dados contínuos. Diferentemente do modelo anterior, a rede foi construída com duas camadas escondidas. Os resultados mostram-se melhores do que uma rede neural comum.

Chao et al. (CHAO; SHEN; ZHAO, 2011) usaram DBN para prever uma série

temporal de taxa de câmbio. Nesse caso, máquinas de Boltzmann contínuas também foram utilizados como *building blocks* do modelo. Os resultados mostraram-se melhores do que as arquiteturas *feedforward*.

Sergio e Ludermir (SERGIO; LUDERMIR, 2015) utilizaram DBN para prever séries temporais de velocidade de ventos. Através dos experimentos realizados e com a análise dos resultados obtidos, ficou constatado que a metodologia utilizada produziu modelos de redes neurais com desempenho satisfatório em todas as séries estudadas, em algumas delas inclusive apresentando os melhores resultados encontrados na literatura.

O modelo de regressão das máquinas de vetor de suporte (SVR) também tem sido utilizado como preditor de séries temporais desde que foi proposto. Fan et al., por exemplo, usaram SVR na previsão de séries de carga elétrica (FAN et al., 2016). Assim como no uso de redes neurais, é possível treinar o preditor através de algoritmos de otimização (LIU; WANG, 2016) ou mesmo usar variações do modelo original (BAO; XIONG; HU, 2014).

2.3.2 Combinação de Previsão de Séries Temporais

A combinação de previsão de séries temporais foi introduzida nos trabalhos de Bates e Granger (BATES; GRANGER, 1969) e de Reid (REID, 1969). Desde então, várias publicações têm utilizado essa ideia, passando pelo uso de modelos de aprendizado de máquina como as redes neurais artificiais (HASHEM; SCHMEISER, 1995).

Apesar de mais simples, a combinação de previsão a partir de medidas de tendência central como média e mediana ainda são bastante utilizadas na literatura. Andrawis et al. utilizaram a média para combinar as previsões de um comitê constituído por redes neurais, modelos de regressão Gaussiana e linear, em uma série temporal de competição (ANDRAWIS; ATIYA; EL-SHISHINY, 2011). Lian et al. combinaram as saídas de um ensemble de ELM (do inglês *Extreme Learning Machines*) também com uma média aritmética, na tentativa de prever um índice de deslizamento de terra (LIAN et al., 2014). O uso da mediana e da média winsorizada pode ser visto em (ADHIKARI; AGRAWAL, 2012).

O uso de métodos que utilizam parte do conjunto de dados para gerar os pesos da combinação também pode ser observado. Softmax foi utilizado como combinador em um comitê de redes neurais gerado a partir de validação cruzada (DONATE et al., 2013). Já o RBLC foi o combinador com melhor desempenho em uma previsão de longo alcance calculada por um comitê de redes neurais (LANDASSURI-MORENO; BULLINARIA, 2009).

Apesar dessas combinações com relativamente pouco esforço computacional, a literatura mostra alguns outros métodos construídos com mais sofisticação. Adhikari, por exemplo, propôs uma nova combinação linear que determina os pesos pela análise de

padrões dos dados nas previsões sucessivas de uma base de dados de validação (ADHIKARI, 2015).

Combinações não-lineares, apesar de mais incomuns, também podem ser encontradas. No trabalho de Gheyas e Smith, foi construído um comitê de modelos híbridos de redes neurais e regressão linear, chamados GRNN (do inglês *Generalized Regression Neural Network*) (GHEYAS; SMITH, 2011). A saída de diversos GRNN para cada subcaracterística da série temporal é apresentada para treinamento em um segundo GRNN. O trabalho de Adhikari e Agrawal (ADHIKARI; AGRAWAL, 2012) estende o modelo de combinação linear de Freitas e Rodrigues (FREITAS; RODRIGUES, 2006) para calcular os pesos de maneira não-linear.

Nóbrega e Oliveira (NÓBREGA; OLIVEIRA, 2014) combinaram ELM e modelos SVR com filtros de Kalman. O modelo resultante foi aplicado a séries temporais financeiras, tendo superado os preditores individuais.

Um comitê tende a apresentar melhor desempenho quando os modelos que o compõe possui um bom grau de diversidade, aumentando a possibilidade de previsões mais robustas (ANDRAWIS; ATIYA; EL-SHISHINY, 2011). Existem diversas formas de gerar essa diversidade, como utilizar modelos diferentes, diferentes especificações de um mesmo modelo, diferentes tipos de pré-processamento dos dados, diferentes variáveis de entrada etc. Os dados de entrada, por exemplo, podem passar por uma validação cruzada ou *bootstraping* e *bagging*, como visto em (OLIVEIRA; TORGO, 2014). Zhang adicionou ruído aos dados de entrada e formou conjuntos de treinamento distintos (ZHANG, 2007). Andrawis et al. pré-processaram a série, removendo a tendência (ANDRAWIS; ATIYA; EL-SHISHINY, 2011). Naturalmente, uma combinação dessas técnicas é possível, caso do trabalho apresentado nesta tese.

Outro fator que deve ser discutido na implementação de um modelo de combinação de séries temporais é o número de preditores base que serão utilizados. Esse número pode impactar significativamente o combinador. Por exemplo, com poucos preditores, a média aparada tem comportamento muito similar à média tradicional, necessitando de pelo menos cinco deles para ser efetiva (ARMSTRONG, 2001b). Por outro lado, há uma indicação na literatura de utilizar até no máximo cinco modelos para redução dos erros de previsão, sendo que a utilização de mais preditores pode diminuir drasticamente a acurácia da combinação (MAKRIDAKIS; WINKLER, 1983). Adhikari (ADHIKARI, 2015) (ARMSTRONG, 2001b), por exemplo, alcançou bons resultados de combinação com a utilização de apenas quatro preditores base.

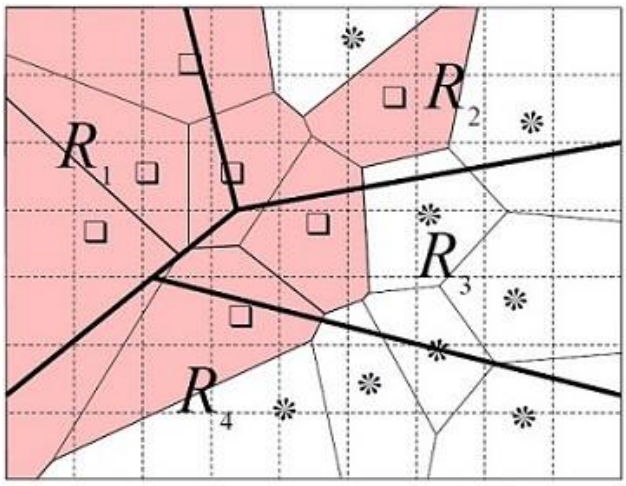
2.4 Considerações do Capítulo

A previsão de séries temporais é uma importante área em aprendizado de máquina e estatística. Diversos modelos têm sido propostos para resolução desse problema, utilizando as mais diferentes técnicas e modelos. Mais recentemente, a combinação de preditores tomou uma maior atenção dos pesquisadores, já que essa abordagem pode se beneficar das vantagens de vários dos métodos de previsão desenvolvidos. O número também elevado de combinadores leva à necessidade de algum método automático de seleção desses modelos. Dessa forma, o experimentador pode não precisar conhecer profundamente as vantagens e desvantagens dos preditores e combinadores disponíveis.

3 SELEÇÃO DINÂMICA

Considere um problema de classificação ou regressão. Seja $C = \{h_1, h_2, ..., h_L\}$ um conjunto de L especialistas e $E = \{e_1, e_2, ..., e_M\}$ um conjunto de M comitês formados a partir de C. A seleção dinâmica pode ser vista como uma divisão do espaço de características em $K > 1$ regiões de competência, denotadas por $R_1, R_2, ..., R_K$. Então, para cada região R_j, $j = 1, 2, ..., K$, é designado o comitê mais competente em E. A figura 7 apresenta uma ilustração da divisão do espaço de características em quatro regiões de competência.

Figura 7 – Particionamento do espaço de características em regiões de competência



Fonte: (KUNCHEVA, 2000).

Seja $e^* \in E$ o comitê com a maior acurácia em todo o espaço de características. Denote por $p(e_i|R_j)$ a probabilidade de correta classificação de e_i em R_j. Considere $e_{i(j)}$ como sendo o comitê designado para R_j. A probabilidade de correta classificação p_c é descrita na equação 3.1, onde $p(R_j)$ é a probabilidade de um padrão $\vec{x}$ pertencer a R_j. Para maximizar p_c, deve-se atribuir $e_{i(j)}$ conforme a equação 3.2.

$$p_c = \sum_{j=1}^{K} p(R_j) p_c(R_j) = \sum_{j=1}^{K} p(R_j) P(e_{i(j)}|R_j) \tag{3.1}$$

$$p(e_{i(j)}|R_j) \geq p(e_t|R_j), t = 1, 2, ..., M \tag{3.2}$$

$$\sum_{j=1}^{K} p(R_j) P(e_{i(j)}|R_j) \geq \sum_{j=1}^{K} p(R_j) p(e^*|R_j) \tag{3.3}$$

A partir das equações 3.1 e 3.2, tem-se na equação 3.3 que ao selecionar dinamicamente $e_{i(j)}$, sendo este o mais competente comitê na região, pode-se alcançar uma

probabilidade de correta classificação superior ou igual ao do comitê e^*, independente da maneira na qual o espaço de características tenha sido particionado.

Como explicitado anteriormente, o processo de seleção dinâmica pode ser resumido em três fases. A primeira delas é responsável pela geração do conjunto de especialistas base, sendo que esse conjunto pode ser formado por modelos de mesma natureza ou heterogêneos. A diversidade dos especialistas é importante em ambas as situações.

A segunda fase, de seleção, é realizada através da estimação da competência dos modelos disponíveis no conjunto gerado na primeira fase, em respeito a regiões locais do espaço de características. No caso da seleção dinâmica, objeto de estudo desta tese, a escolha dos modelos é realizada para cada padrão de teste, ao invés de utilizar a mesma seleção para todos eles (seleção estática). É comum, por exemplo, a utilização de algum esquema baseado em regras de vizinhança para definir região próxima de um padrão desconhecido na fase de teste. Britto Jr et al. (BRITTO; SABOURIN; OLIVEIRA, 2014) propõem uma taxonomia para as diversas medidas de competência encontradas na literatura, como pode ser observado na figura 8 (página 44). De acordo com essa taxonomia, a seleção dos modelos divide-se entre a utilização de medidas baseadas no indivíduo e medidas baseadas no conjunto, em que a acurácia dos especialistas base são combinadas com alguma informação relacionada à interação entre eles. No primeiro caso, as medidas podem ser baseadas em *ranking*, na acurácia, na probabilidade, no comportamento ou no oráculo. Em relação às medidas baseadas no conjunto, elas podem ter como referência a diversidade ou a ambiguidade.

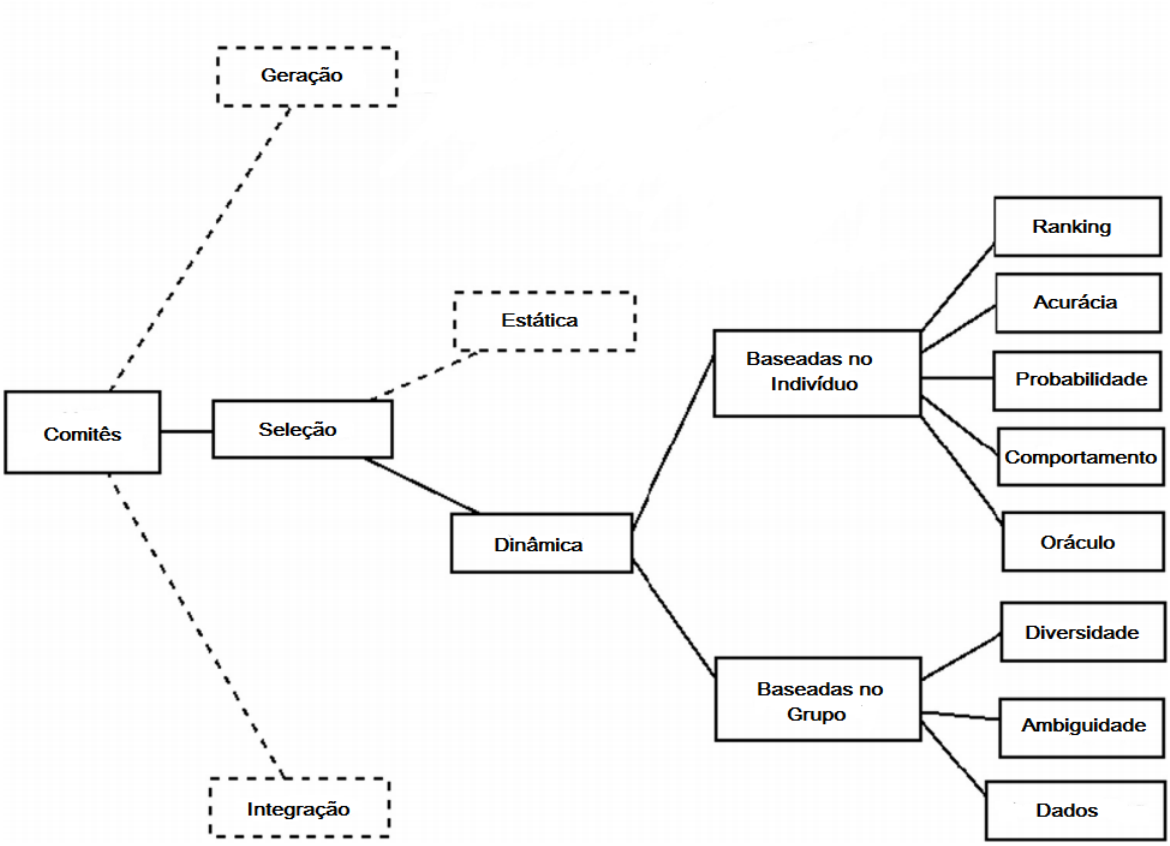
A terceira fase é a integração dos modelos selecionados. Naturalmente, a literatura também mostra diversas formas de realizar essa etapa. Uma taxonomia proposta pode ser vista em (KITTLER et al., 1998).

A seguir, serão apresentados mais detalhes a respeito de cada uma das medidas de competência citadas, de acordo com a taxonomia de Britto Jr et al. (BRITTO; SABOURIN; OLIVEIRA, 2014). Nos algoritmos que serão apresentados, as seguintes notações são utilizadas: $\Omega = \{\omega_1, \omega_2, ...\omega_C\}$ é o conjunto de classes do problema de reconhecimento de padrões; T_r, V_a e T_e representam respectivamente os conjuntos de treinamento, validação e teste; $C = \{c_1, c_2, ..., c_M\}$ é o conjunto constituído de M especialistas; $CC = \{CC_1, CC_2, ..., CC_N\}$ é o conjunto de N comitês formados a partir dos classificadores; t é um padrão de teste desconhecido; Ψ é a região do espaço de características utilizado para calcular a competência dos classificadores individuais.

3.1 Medidas Baseadas no Indivíduo

Nas medidas de desempenho baseadas no indivíduo, os especialistas são selecionados de acordo com alguma medida individual em uma região local da base de treinamento ou

Figura 8 – Taxonomia da seleção dinâmica em comitês de especialistas



Fonte: (BRITTO; SABOURIN; OLIVEIRA, 2014).

de validação. Essa região local pode ser calculada a partir de um particionamento durante a fase de treinamento ou da aplicação de alguma regra de vizinhança sobre um padrão desconhecido, na fase de testes. Abaixo, algumas dessas medidas.

3.1.1 *Ranking*

Os métodos pertencentes a essa subcategoria utilizam algum tipo de *ranking* para os especialistas do comitê. Sabourin et al. propuseram um algoritmo pioneiro em 1993 apresentado como *DSC-Rank* (SABOURIN et al., 1993), aplicado ao problema de classificação. No *DSC-Rank*, três parâmetros são calculados para cada classificador a partir da base de treinamento.

Seja X um conjunto de parâmetros dos classificadores, constituído pela distância para o classificador vencedor (aquele com melhor desempenho), a distância para o primeiro classificador não-vencedor e a razão da distância do primeiro não-vencedor para o vencedor. Adicionalmente, seja S uma variável de sucesso para o classificador, definida como $S = \delta(t, o)$ (t é o rótulo correto de um determinado exemplo de treinamento e o é a saída do classificador). A informação mútua entre S e X, dada por $I(S, X)$, é estimada a partir de uma função de entropia entre esses dois conjuntos. Após determinação dos parâmetros,

o chamado espaço de meta-padrões é determinado (MP), seguido da criação de um *ranking* de classificadores. Por fim, o classificador com melhor classificação nesse *ranking* é selecionado para determinar o rótulo do padrão desconhecido. O algoritmo 1 descreve o *DSC-Rank*.

Pseudocodigo 1 Método DSC-Rank

INPUT: o conjunto de classificadores C; o conjunto de parâmetros dos classificadores X; as bases de dados de treinamento T_r e de teste T_e

OUTPUT: c_t^*, o classificador mais promissor para cada padrão t em T_e

1: Compute $S = (t, o)$ como sendo a variável de sucesso dos classificadores utilizando os exemplos de treinamento em T_r
2: Compute $I(S, X)$ como sendo a informação mútua entre X e S utilizando os exemplos de treinamento em T_r
3: Determine $X^{'}$ como sendo os parâmetros mais informativos dos classificadores baseados em $I(S, X)$
4: Crie o espaço de meta-padrões MP como um subconjunto de T_r, a partir dos valores correspondentes dos parâmetros em $X^{'}$
5: **for** cada padrão de teste $t \in T_e$ **do**
6: Aplique uma regra de vizinhança para encontrar Ψ, como sendo o vizinho mais próximo do padrão desconhecido t em MP
7: Calcule o *rank* dos classificadores baseados nos valores dos parâmetros associados a Ψ
8: Selecione c_t^* como sendo o classificador na melhor posição do *ranking*
9: Use c_t^* para classificar t
10: **end for**

Uma simplificação do *DSC-Rank* foi proposta por Woods et al. (WOODS; BOWYER; JR, 1996). Nesse algoritmo, chamado *DS-MR*, o processo de particionamento para definir a região local é realizado na fase de testes.

3.1.2 Acurácia

Nessa subcategoria, a seleção dos especialistas é realizada a partir de sua acurácia em uma base de validação. No caso de problemas de classificação, essa estimativa pode ser calculada de acordo com uma porcentagem simples de classificações corretas em uma região local ou na base de dados completa. O trabalho de Woods et al. apresenta dois algoritmos que implementam essa ideia (WOODS; BOWYER; JR, 1996). O primeiro deles é o *DS-LA* baseado em OLA (Acurácia Local Geral, do inglês *Overall Local Accuracy*). Nessa abordagem, o método calcula o OLA de cada um dos classificadores base na região mais próxima de determinado padrão desconhecido na base de treinamento. O OLA é calculado como sendo a porcentagem de classificações corretas dos exemplos da região local. O outro algoritmo é o *DS-LA* baseado em LCA (Acurácia de Classe Local, do inglês *Local Class Accuracy*). No *DS-LA-LCA*, a medida de desempenho é estimada como sendo a porcentagem de classificações corretas dentro de uma região local, mas apenas considerando

os exemplos em que o classificador tem a mesma classe do padrão desconhecido. Em ambos os algoritmos, o particionamento é realizado através da regra dos k vizinhos mais próximos do padrão desconhecido de teste na base de treinamento. O *DS-LA-OLA* e o *DS-LA-LCA* são mostrados nos algoritmos 2 e 3.

Pseudocodigo 2 DS-LA-OLA

INPUT: o conjunto de classificadores C; as bases de dados de treinamento T_r e de teste T_e; o tamanho da vizinhança K
OUTPUT: c_t^*, o classificador mais promissor para cada padrão t em T_e

for cada padrão de teste $t \in T_e$ **do**
    Submeta t a todos os classificadores em C
    if todos os classificadores concordam quanto ao rótulo do padrão t **then**
        Retorne o rótulo de t
    else
        Encontre Ψ como sendo os K vizinhos mais próximos do padrão t em T_r
        for cada classificador c_i em C **do**
            Calcule OLA_i como sendo a porcentagem de classificações corretas de c_i em Ψ
        end for
        Selecione o melhor combinador $c_t^* = argmax_i\{OLA_i\}$
        Use c_t^* para classificar t
    end if
end for

Pseudocodigo 3 Método DS-LA-LCA

INPUT: o conjunto de classificadores C; as bases de dados de treinamento T_r e de teste T_e; o tamanho da vizinhança K
OUTPUT: c_t^*, o classificador mais promissor para cada padrão t em T_e

for cada padrão de teste $t \in T_e$ **do**
    Submeta t a todos os classificadores em C
    if todos os classificadores concordam quanto ao rótulo do padrão t **then**
        Retorne o rótulo de t
    else
        for cada classificador c_i em C **do**
            Calcule $\omega_j = c_i(t)$ como sendo a saída de c_i para o padrão t
            Encontre Ψ como sendo os K vizinhos mais próximos do padrão t em T_r que pertence à classe ω_j
            Calcule $LCA_{(i,j)}$ como sendo a porcentagem de padrões corretamente rotulados da classe ω_j pelo classificador c_i em Ψ
        end for
        Selecione o melhor combinador $c_t^* = argmax_i\{LCA_{(i,j)}\}$
        Use c_t^* para classificar t
    end if
end for

3.1.3 Probabilidade

Os métodos dessa subcategoria utilizam representações probabilísticas dos modelos. Giacinto et al. propuseram em (GIACINTO; ROLI, 1999) duas abordagens chamadas seleção *A Priori* e *A Posteriori*. No método *A Priori*, a seleção de um classificador é baseada em sua acurácia na região local, sem considerar a classe do padrão desconhecido. A medida de acurácia é calculada como sendo a probabilidade do classificador c_j pertencer a determinada classe na vizinhança Ψ para o padrão desconhecido t, dada pela equação 3.4:

$$\hat{p}(correta_j) = \frac{\sum_{i=1}^{K} \hat{P}_j(\omega_l|\Psi_i \in \omega_l) * \delta_i}{\sum_{i=1}^{K} \delta_i} \tag{3.4}$$

onde ω_l é a classe e δ_i é o peso atribuído, representando a distância euclideana entre Ψ_i e o padrão desconhecido.

No caso da abordagem *A Posteriori*, a medida leva em consideração a classe ω_i atribuída pelo classificador c_j ao padrão desconhecido t. Nesse sentido, a probabilidade é dada pela equação 3.5:

$$\hat{p}(correta_j|c_j(t) = \omega_l) = \frac{\sum_{\Psi_i \in \omega_l}^{K} \hat{P}_j(\omega_l|\Psi_i) * \delta_i}{\sum_{i=1}^{K} \hat{P}_j(\omega_l|\Psi_i) * \delta_i} \tag{3.5}$$

onde ω_l é o inverso da distância euclideana entre Ψ_i e t. O algoritmo 4 mostra a seleção dinâmica por probabilidade *a priori* e *a posteriori*.

3.1.4 Comportamento

Nessa subcategoria, os métodos analisam o comportamento dos modelos individuais levando em consideração as suas predições e informações adjacentes. Nessa perspectiva, Giacinto et al. propuseram um método baseado no MCB (do inglês *Multiple Classifier Behavior*), denominado DS-MCB (GIACINTO; ROLI, 2001). O MCB é calculado a partir de uma função de similaridade entre os padrões de saída e os classificadores bases em uma região local do espaço de características. Ao final do processo, o rótulo dos padrões é calculado a partir de uma acurácia local, de maneira similar ao DS-LA-OLA. O DS-MCB pode ser visto no algoritmo 5 (página 48).

Outros exemplos de seleção dinâmica por comportamento podem ser vistos em (NABIHA; NADIR, 2012) e (CAVALIN; SABOURIN; SUEN, 2013). No primeiro caso, os autores estendem as ideias apresentadas até o momento para selecionar um subconjunto de comitês ao invés de um comitê único. O segundo trabalho calcula a confiabilidade de cada classificador base em uma base de dados de validação.

Pseudocodigo 4 Método *a priori/posteriori*

INPUT: o conjunto de classificadores C; as bases de dados de treinamento T_r e de teste T_e; o tamanho da vizinhança K

OUTPUT: c_t^*, o classificador mais promissor para cada padrão t em T_e

1: **for** cada padrão de teste $t \in T_e$ **do**
2: Encontre Ψ como sendo os K vizinhos mais próximos do padrão t em T_r
3: **for** cada classificador c_j em C **do**
4: Compute $\hat{p}(correta_j)$ em Ψ usando equações 3.4 ou 3.5
5: **if** $\hat{p}(correta_j) \geq 0.5$ **then**
6: $CS = CS \cup c_j$
7: **end if**
8: **end for**
9: $\hat{p}(correta_m) = max_j(\hat{p}(correta_j))$
10: $c_m = argmax_j(\hat{p}(correta_j))$
11: $selecionado = TRUE$
12: **for** cada classificador c_j em CS **do**
13: $d = \hat{p}(correta_m) - \hat{p}(correta_j)$
14: **if** $(j \neq m)$ **and** $(d < Threshold)$ **then**
15: $selecionado = FALSE$
16: **end if**
17: **end for**
18: **if** $selecionado == TRUE$ **then**
19: $c_t^* = c_m$
20: **else**
21: c_t^* = um classificador aleatoriamente selecionado a partir de CS, com $(d < Threshold)$
22: **end if**
23: Use o classificador c_t^* para classificar t
24: **end for**

3.1.5 Oráculo

Os métodos dessa subcategoria utilizam o conselho dos chamados oráculos. Tais oráculos podem ser calculados a partir de formas distintas. No trabalho de Kuncheva et al. (KUNCHEVA; RODRIGUEZ et al., 2007), por exemplo, o oráculo é uma função linear aleatória responsável por decidir, dado um conjunto de classificadores previamente selecionados, qual deles utilizar para classificar um padrão desconhecido. Outro algoritmo dessa subcategoria que também pode ser citado é o KNORA (*k-nearest-oracles*), proposto em (SHIN; SOHN, 2003). Nesse caso, o oráculo é dado pelos k vizinhos mais próximos do padrão desconhecido no conjunto de validação, sendo que os classificadores com resposta correta para cada exemplo são previamente conhecidos. Esse conjunto de validação é referido como meta-espaço. Uma das abordagens do KNORA, conhecido como KNORA-Eliminate (KNE), pode ser visto no algoritmo 6.

Pseudocodigo 5 Método DS-MCB

INPUT: o conjunto de rótulos Ω; o conjunto de combinadores C; as bases de dados de treinamento T_r e de teste T_e; o tamanho da vizinhança K;
OUTPUT: c_t^*, o classificador mais promissor para cada padrão t em T_e;

for cada padrão de teste $t \in T_e$ **do**
  Compute o vetor MCB_t como sendo os rótulos associados a t por todos os classificadores em C;
  Encontre Ψ como sendo o conjunto dos K vizinhos mais próximos do padrão de teste t em T_r;
  for cada padrão $\psi_j \in \Psi$ **do**
    Compute MCB_{ψ_j} como sendo o rótulo associado a ψ_j por todos os classificadores em C;
    Compute Sim como sendo a similaridade entre MCB_t e MCB_{ψ_j};
    if $(Sim > SimilarityThreshold)$ **then**
      $\Psi' = \Psi' \cup \psi_j$;
    end if
  end for
  for cada classificador $c_i \in C$ **do**
    Calcule OLA_i a acurácia local do classificador c_i em Ψ';
  end for
  Selecione o melhor classificador $c_t^* = argmax_i\{OLA_i\}$;
  if c_t^* é significativamente melhor que outros classificadores em Ψ' **then**
    Use o classificador c_t^* para classificar t
  else
    Use o voto majoritário de todos os classificadores para classificar t
  end if
end for

3.2 Medidas Baseadas no Conjunto

As medidas de desempenho desta categoria são dadas principalmente pela interação entre os especialistas que compõem o comitê. Abaixo, algumas delas.

3.2.1 Diversidade

Shin et al. (SHIN; SOHN, 2003) modificaram o DL-SA (proposto em (WOODS; BOWYER; JR, 1996)) de forma a considerar a seleção dos especialistas não somente pela acurácia, mas também pela diversidade do erro. Essa ideia é revisitada no trabalho de Santana et al. (SANTANA et al., 2006). Nesse caso, duas variações foram propostas, sendo que ambas ordenam os classificadores em ordem decrescente de acurácia e em ordem crescente de diversidade. Na primeira abordagem, chamada DS-KNN, as medidas de acurácia e diversidade são calculadas pela definição dos k vizinhos mais próximos do padrão desconhecido na base de validação. Já na outra variação, chamada DS-Cluster, um processo de agrupamento é realizado para dividir o conjunto de validação no qual grupos com classificadores mais promissores serão associados. O DS-KNN pode ser visto

Pseudocodigo 6 Método KNE

INPUT: o conjunto de combinadores C; meta-espaço sVA onde para cada padrão são associados os classificadores que melhor o reconhecem; a base de dados de teste T_e; o tamanho da vizinhança K;

OUTPUT: CC_t^*, o comitê de classificadores mais promissor para cada padrão t em T_e;

```
1: for cada padrão de teste t ∈ T_e do
2:     k = K
3:     while k > 0 do
4:         Encontre Ψ como sendo os k vizinhos mais próximos do padrão de teste t em sVA
5:         for cada classificador c_i ∈ C do
6:             if c_i reconhece corretamente todos os padrões em Ψ then
7:                 CC_t^* = CC_t^* ∪ c_i
8:             end if
9:         end for
10:        if CC_t^* == ∅ then
11:            k = k − 1
12:        else
13:            break
14:        end if
15:    end while
16:    if CC_t^* == ∅ then
17:        Encontre o classificador c_i que reconhece corretamente mais padrões em Ψ
18:        Selecione os classificadores capazes de reconhecer a mesma quantidade de padrões de c_i para compor o comitê CC_t^*
19:    end if
20:    Use o comitê CC_t^* para classificar t
21: end for
```

no algoritmo 7 (página 51).

3.2.2 Ambiguidade

Ao invés da diversidade, os métodos pertencentes a esta subcategoria utilizam o consenso dos classificadores base. De maneira geral, tais métodos selecionam os especialistas de um comitê com bom desempenho e com a menor ambiguidade entre seus membros. Com o DSA, método proposto em (SANTOS; SABOURIN; MAUPIN, 2008), por exemplo, os autores perceberam um aumento na generalização já que a confiança nos classificadores passou a aumentar. Trabalhos como (HO; HULL; SRIHARI, 1994) e (SANTOS; SABOURIN; MAUPIN, 2007) seguem caminhos semelhantes. O DSA é mostrado no algoritmo 8.

Pseudocodigo 7 Método DS-KNN

INPUT: o conjunto de combinadores C; as bases de dados de validação V_a e de teste T_e; o tamanho da vizinhança K; o número de classificadores que serão selecionados N' e $N"$

OUTPUT: CC_t^*, o comitê de classificadores mais promissor para cada padrão t em T_e;

for cada padrão de teste $t \in T_e$ **do**
    Encontre Ψ como sendo os k vizinhos mais próximos do padrão de teste t em sVA
    for cada classificador $c_i \in C$ **do**
        Compute A_i como sendo a acurácia de c_i em Ψ
    end for
    for cada classificador $c_i \in C$ **do**
        for cada classificador $c_j \in C$ **do**
            if $i \neq j$ **then**
                Compute D_{ij} como sendo a diversidade entre c_i e c_j em Ψ
            end if
        end for
    end for
    Crie R_1 como sendo o *rank* de classificadores em C em ordem decrescente da acurácia A
    Crie R_2 como sendo o *rank* de classificadores em C em ordem crescente da diversidade D
    Baseado em R_1, selecione os N' mais precisos classificadores em C para compor o comitê CC
    Baseado em R_2, selecione os $N"$ mais diversos classificadores em CC para compor CC_t^*
    Use o comitê CC_t^* para classificar t
end for

Pseudocodigo 8 Método DSA

INPUT: o conjunto de rótulos Ω; o conjunto de combinadores C; as bases de dados de validação V_a e de teste T_e; o tamanho da vizinhança K

OUTPUT: CC_t^*, um comitê de classificadores para cada padrão t em T_e;

$CC' = ProcessoOtimização(C, V_a, \Omega)$
for cada padrão de teste $t \in T_e$ **do**
    if todos os N comitês em CC' concordam a respeito do rótulo de t **then**
        Classifique t
    else
        for cada $CC'_i \in CC'$ **do**
            Compute A_i como sendo a ambiguidade do comitê CC'_i
        end for
    end if
    Selecione o melhor comitê para t sendo que $CC_t^* = argmin_i\{A_i\}$
    Use o comitê CC_t^* para classificar t
end for

3.3 Considerações do Capítulo

A literatura apresentada nesta seção é predominantemente aplicada a problemas de classificação e reconhecimento de padrões. Apesar da falta de trabalhos sobre seleção dinâmica em previsão de séries temporais, algumas pesquisas com viés semelhante podem ser encontradas na literatura. Algumas delas são as que tratam de seleção de modelos. Seleção de modelos diz respeito a selecionar, a partir dos dados, um modelo específico para conclusão da tarefa. Entretanto, em problemas de previsão de séries temporais, esse processo é normalmente realizado de maneira estática (HURVICH; TSAI, 1989) (QI; ZHANG, 2001).

4 SELEÇÃO DINÂMICA DE COMBINADORES DE PREVISÃO

4.1 Método Proposto: Um Arcabouço para Seleção Dinâmica de Combinadores de Previsão

O principal objetivo do trabalho descrito nesta tese é a proposição de um arcabouço para seleção dinâmica de combinadores de previsão de séries temporais. Como mostrado nos capítulos anteriores, diversos modelos foram propostos na literatura com o objetivo de prever séries temporais, incluindo modelos estatísticos e aprendizado de máquina. Devido ao fato de que um único modelo não é capaz de ser aplicado adequadamente em todos os tipos de problemas, combinar a saída de diferentes preditores pode produzir resultados que alcancem menores erros de previsão. Entretanto, o número de combinadores de previsão que podem ser utilizados também é abundante, fazendo-se necessário uma metodologia de seleção automática dos combinadores a partir da base de dados. A seleção automática discutida neste trabalho é dinâmica. Dessa forma, para cada padrão de teste apresentado ao modelo, um combinador é selecionado com o intuito de produzir a saída a partir de um dado critério.

O arcabouço para seleção dinâmica de combinadores de previsão consiste em gerar preditores base, combiná-los e selecionar qual combinação é mais promissora para cada um dos padrões de teste. Na geração dos preditores base, duas medidas são tomadas no arcabouço para garantir um bom grau de diversidade. A primeira delas é a utilização de modelos de natureza heterogênea, como visto em (ANDRAWIS; ATIYA; EL-SHISHINY, 2011). Além disso, a diversidade também deve ser buscada com o uso de validação cruzada nos dados de treinamento. Os preditores base são gerados, então, com conjuntos de treinamento distintos. Ideia similar foi seguida, por exemplo, em (OLIVEIRA; TORGO, 2014). O arcabouço também determina que os preditores advindos da aprendizado de máquina e que, portanto, precisam de treinamento, devem ser obtidos através de um algoritmo de otimização como o PSO. A utilização de algoritmos de otimização surgidos da computação evolucionária ou inteligência de enxames é ampla na literatura (FERREIRA; LUDERMIR; AQUINO, 2013) (SERGIO; LUDERMIR, 2014). Nesses casos, o objetivo é aplicar o algoritmo para otimizar os hiper-parâmetros dos modelos, visto que esta é uma tarefa empírica e altamente dependente do problema e do conjunto de dados utilizado.

Em relação à escolha automática dos combinadores, foram desenvolvidos dois algoritmos de seleção dinâmica para o contexto de previsão de séries temporais: DSFC-A (do inglês *Dynamic Selection of Forecast Combiners - Accuracy*) e DSFC-B (do inglês *Dynamic Selection of Forecast Combiners - Behavior*). O DSFC-A e o DSFC-B são respectivamente inspirados pelo DS-LA-OLA (WOODS; BOWYER; JR, 1996) e pelo DS-

MCB (GIACINTO; ROLI, 2001)). Foi mostrado que os métodos de seleção dinâmica para reconhecimento de padrões e classificação podem ser categorizados entre os que utilizam medidas baseadas no indivíduo e os que utilizam medidas baseados no conjunto. Foram selecionados como inspiração dois métodos que utilizam medidas baseadas no indivíduo devido a natureza do arcabouço desenvolvido. Na metodologia proposta, a seleção dinâmica é realizada a partir dos combinadores, que já são integrações dos especialistas base e poderiam não gerar informações significativas para o modelo de seleção. Por outro lado, as interações entre os preditores base poderiam não ser suficientemente importantes para os combinadores. Dentre as medidas baseadas em indíviduos, a acurácia e o comportamento foram selecionados para a construção dos métodos propostos pois utilizam grandezas independentes de tipo de problema, podendo ser utilizadas tanto em classificação como em regressão. Métodos como os de probabilidade, por exemplo, são naturalmente mais propícios em problemas de reconhecimento de padrões.

Como visto em (BRITTO; SABOURIN; OLIVEIRA, 2014), a seleção dinâmica pode ser observada como sendo composta por três fases claramente definidas: geração, seleção e integração. A fase de integração é opcional, aplicada quando a fase de seleção estipula mais de um modelo promissor para cada padrão de teste. Na seleção dinâmica apresentada nesta tese, a fase de seleção define modelos que por natureza já são integrações dos especialistas individuais, os combinadores. Assim, a fase de integração acaba não sendo necessária.

A figura 9 ilustra o arcabouço proposto para seleção dinâmica de combinadores de previsão. Após a definição de três diferentes conjuntos de dados obtidos a partir da série temporal (treino I, treino II e teste), o arcabouço pode ser determinado por três passos, a saber:

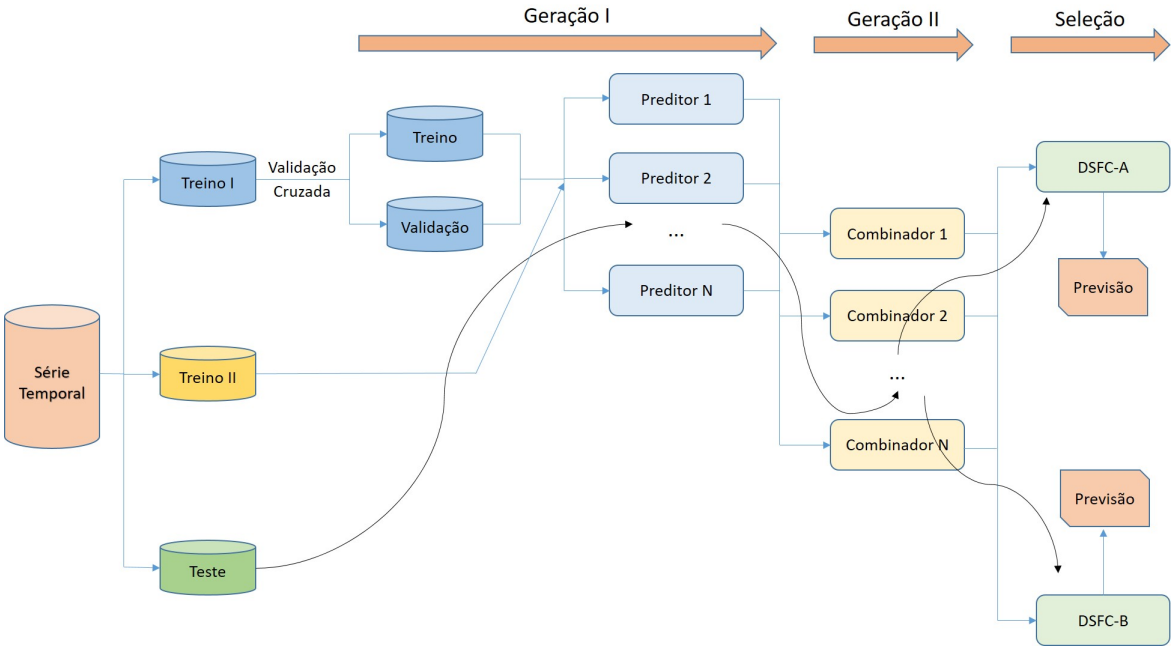
1. Geração I: na primeira parte da geração, os preditores base são criados. Os preditores são construídos a partir do primeiro conjunto de treino (treino I). O conjunto de treino é submetido a uma validação cruzada, no intuito de aumentar a variabilidade dos modelos de previsão. O número de *folds* da validação cruzada é dado pelo número de preditores, no intuito de gerar bases independentes para os especialistas individuais. A melhor configuração dos hiper-parâmetros dos preditores de aprendizado de máquina é obtida através de um algoritmo de otimização como o PSO ou Algoritmos Genéticos.

2. Geração II: a segunda parte da geração abrange a definição dos combinadores a partir da segunda base de treinamento (treino II) e dos preditores base. Nem todos os combinadores precisam passar por essa fase, visto que apenas alguns deles necessitam de uma base de dados de treinamento para serem gerados.

3. Seleção: na fase de seleção, os padrões de testes são submetidos aos preditores, aos combinadores e finalmente aos métodos de seleção dinâmica desenvolvidos nesta tese

(DSFC-A e DSFC-B). Para cada padrão de teste, os métodos de seleção dinâmica alcançam a previsão a partir do combinador mais promissor. O DSFC-A e o DSFC-B são detalhados na próxima seção.

Figura 9 – Arcabouço para seleção dinâmica de combinadores de previsão



Fonte: do autor.

4.1.1 DSFC-A

O algoritmo 9 descreve o DSFC-A. A entrada do algoritmo de seleção é o comitê formado pelos combinadores, de acordo com o arcabouço proposto (figura 9). Para cada padrão de teste, são calculadas as suas respectivas saídas em todos os combinadores pertencentes ao comitê. Após o cálculo das saídas, são encontrados os K vizinhos mais próximos do padrão de teste em uma base de dados de treinamento (treino II), formando assim o conjunto Ψ. Então, o algoritmo calcula a acurácia local geral (OLA) de cada combinador na sub-região de características Ψ. No método desenvolvido e implementado, OLA é dado pelo erro médio de previsão. Finalmente, seleciona-se o combinador com o menor erro de previsão na sub-região Ψ para calcular a saída do padrão de teste. O DSFC-A tem como saída o combinador mais promissor para cada padrão de teste. A previsão do método é obtida a partir da seleção dinâmica dos combinadores pertencentes ao comitê definido em fase anterior do arcabouço.

4.1.2 DSFC-B

O algoritmo 10 (página 57) descreve o DSFC-B. Assim como ocorre no DSFC-A, a entrada do DSFC-B é o comitê formado pelos combinadores, de acordo com o

Pseudocodigo 9 DSFC-A

INPUT: o conjunto de combinadores C; as bases de dados de treinamento T_r e de teste T_e; o tamanho da vizinhança K

OUTPUT: Previsão realizada por c_t^*, o combinador mais promissor para cada padrão t em T_e

1: **for** cada padrão de teste $t \in T_e$ **do**
2: Submeta t a todos os preditores em C
3: Submeta a saída dos preditores a todos os combinadores em C
4: Encontre Ψ como sendo os K vizinhos mais próximos do padrão t em T_r
5: **for** cada combinador c_i em C **do**
6: Calcule OLA_i como sendo o erro médio de previsão de c_i em Ψ
7: **end for**
8: Selecione o melhor combinador $c_t^* = argmin_i\{OLA_i\}$
9: Use c_t^* para produzir a saída a partir de t
10: **end for**

arcabouço proposto (figura 9). Para cada padrão de teste é calculado um vetor denominado $PRED_t$. O vetor $PRED_t$ contém a previsão de cada um dos combinadores pertencentes ao comitê para o padrão de teste. Após o cálculo de $PRED_t$, são encontrados os K vizinhos mais próximos do padrão de teste em uma base de dados de treinamento (treino I), formando assim o conjunto Ψ. Para cada padrão do conjunto Ψ, o vetor $PRED_\psi$ também é computado como sendo composto pelas previsões dos combinadores pertencentes ao comitê. Uma nova sub-região de características, denominada Ψ', é obtida a partir de Ψ de acordo com uma medida de similaridade entre $PRED_t$ e $PRED_\psi$, denominada Sim. No método desenvolvido e implementado, a medida de similaridade Sim é dada pela distância euclidiana entre $PRED_t$ e $PRED_\psi$. Os padrões de treino são adicionados à sub-região Ψ' se a medida de similaridade for maior que um dado limiar, definido como entrada do algoritmo. Após a definição de Ψ', computa-se a saída de cada combinador nessa sub-região, bem como a média dos erros de previsão, denominada $ERROR$. O combinador com menor erro de previsão em Ψ' é então selecionado para produzir a saída do padrão de teste.

Assim como no DSFC-A, Ψ é obtida através da regra de vizinhos mais próximos. Entretanto, o DSFC-B não utiliza somente o desempenho dos combinadores para realizar a seleçao dinâmica. Nesse caso, o cálculo de Ψ também leva em consideração o padrão de saída dos combinadores, representando assim o seu comportamento. Esse cenário implica, portanto, em uma maior complexidade computacional do DSFC-B. O processamento do DSFC-A é proporcional à multiplicação do número de padrões de teste pelo númedo de combinadores (dois laços de repetição aninhados). O DSFC-B possui um processamento extra, dado pelo laço de repetição que itera todos os padrões de treino da sub-região de características Ψ (linha 4 do algoritmo 10).

Ambos o DSFC-A e o DSFC-B são inspirados em técnicas de seleção dinâmica para problemas de classificação: DS-LA-OLA e DS-MCB, respectivamente. As principais

Pseudocodigo 10 DSFC-B

INPUT: o conjunto de combinadores C; as bases de dados de treinamento T_r e de teste T_e; o tamanho da vizinhança K; limiar $SimilarityThreshold$

OUTPUT: Previsão realizada por c_t^*, o combinador mais promissor para cada padrão t em T_e;

1: **for** cada padrão de teste $t \in T_e$ **do**
2: Submeta t a todos os preditores em C
3: Compute o vetor $PRED_t$ como sendo a previsão de t por todos os combinadores em C;
4: Encontre Ψ como sendo o conjunto dos K vizinhos mais próximos do padrão de teste t em T_r;
5: **for** cada padrão $\psi_j \in \Psi$ **do**
6: Compute $PRED_{\psi_j}$ como sendo a previsão de ψ_j por todos os combinadores em C;
7: Compute Sim como sendo a similaridade entre $PRED_t$ e $PRED_{\psi_j}$;
8: **if** $(Sim > SimilarityThreshold)$ **then**
9: $\Psi' = \Psi' \cup \psi_j$;
10: **end if**
11: **end for**
12: **for** cada combinador $c_i \in C$ **do**
13: Calculate $PRED_i$ como previsão de c_i em Ψ' e $ERROR_i$ como sendo a média do erro de previsão;
14: **end for**
15: Selecione o melhor combinador $c_t^* = argmin_i\{ERROR_i\}$;
16: Use c_t^* para produzir a saída a partir de t
17: **end for**

diferenças advêm da aplicação (o DSFC-A e o DSFC-B são métodos para combinadores de previsão de séries temporais) e das medidas utilizadas para realizar a seleção dinâmica. Tanto a medida de acurácia quanto a de comportamento são específicas para o contexto de problemas de regressão. Adicionalmente, é importante salientar que os métodos de seleção dinâmica desenvolvidos são apenas uma das fases do arcabouço proposto e descrito nesta seção.

4.2 Método Experimental

Esta seção descreve o método experimental que foi desenvolvido para gerar os resultados do arcabouço proposto para seleção dinâmica de combinadores de previsão. Serão descritas individualmente as três fases do processo: Geração I, onde os preditores base são gerados; Geração II, onde são mostrados os combinadores utilizados para combinar as saídas dos preditores base; Seleção, onde são aplicados os métodos propostos de seleção dinâmica, DSFC-A e DSFC-B. Em seguida, detalhes da configuração dos exeperimentos são apresentados, bem como a descrição das séries temporais utilizadas como bases de dados. De maneira geral, o método experimental especifica uma implementação do arcabouço

proposto, com o objetivo de validá-lo através da obtenção de resultados e posterior análise.

4.2.1 Geração I - Preditores Base

A etapa denominada Geração I é responsável por gerar os preditores base que serão posteriormente combinados. Como explicitado anteriormente, a fase de geração deve fornecer especialistas com níveis de diversidade adequados. Nesse sentido, a etapa de Geração I do arcabouço proposto utiliza validação cruzada para geração das bases de treinamento. Outro fator que pode aumentar o nível de diversidade é a natureza dos especialistas. O método experimental desenvolvido utiliza dez diferentes modelos de previsão. Cinco desses modelos são advindos da aprendizado de máquina: uma rede neural *feedforward* com uma camada escondida (FANN1, do inglês *Feedforward Artificial Neural Network*); uma rede neural *feedforward* com duas camadas escondidas (FANN2); o modelo *deep learning* conhecido como *Deed Belief Network* (DBN); o modelo *deep learning* conhecido como *Stacked Denoising Autoencoders* (SDAE); uma máquina de vetor de suporte para regressão (SVR). Os cinco preditores restantes são modelos estatísticos, quatro deles lineares (AR, MA, ARMA e ARIMA) e um não-linear (GARCH).

Redes neurais artificiais foram utilizadas como especialistas devido a seu uso extenso na tarefa de previsão de séries temporais. Além de sua ampla utilização na literatura, redes neurais estão entre os modelos selecionados também por conta de sua facilidade em prover diversidade entre os especialistas. Nas redes neurais, a diversidade pode ser alcançada modificando-se os parâmetros do modelo ou o método de inicialização dos pesos. No caso do método experimental desenvolvido, os quatro modelos de redes neurais (incluindo os modelos *deep learning*) alcançam diversidade ora por mudanças na arquitetura da rede ora por diferentes métodos de inicialização dos pesos. Enquanto que em FANN1 e FANN2 os pesos são iniciados aleatoriamente, DBN e SDAE realizam um pré-treinamento através de técnicas não-supervisionadas. O modelo SVR foi utilizado como o quinto modelo advindo da aprendizado de máquina por possuir estrutura similar às redes neurais, por vezes alcançando resultados com menor custo computacional. Para complementar o comitê, foram utilizados na etapa Geração I os modelos estatísticos lineares mais tradicionais (AR, MA, ARMA e ARIMA). Para gerar ainda mais diversidade, foi incluído o GARCH, modelo não-linear.

Determinar os parâmetros de treinamento de modelos de aprendizado de máquina como redes neurais e máquinas de vetor de suporte é difícil e normalmente orientado a problema. Sob essa perspectiva, o arcabouço proposto determina que seja utilizado algum algoritmo de otimização. O método experimental descrito nessa seção utilizou o PSO (Apêndice A) para este fim, para cada modelo. O PSO é bastante utilizado para otimizar a seleção dos parâmetros de treinamento de modelos de aprendizado de máquina. Quando comparado aos Algoritmos Genéticos o PSO possui a vantagem de ter implementação mais

simples e, em alguns casos, convergência relativamente mais rápida e custo computacional menor (HASSAN et al., 2005). Um exemplo da utilização de PSO para otimizar uma rede neural para previsão de séries temporais pode ser visto em (SERGIO; LUDERMIR, 2014).

Visto que o PSO foi utilizado como algoritmo de otimização, foi preciso definir o esquema de codificação da solução para cada um dos preditores base advindos da aprendizado de máquina. A tabela 1 mostra os parâmetros e conjunto de valores utilizados para construir as soluções candidatas. Os modelos estatísticos foram definidos a partir do *Statistics and Machine Learning Toolbox* do Matlab (MATLAB, 2016), ferramenta utilizada no método experimental.

Tabela 1 – Esquema de codificação do PSO

Modelo	Parâmetro	Conjunto de valores
FANN1	Número de unidades na camada escondida	[5 25]
	Épocas de treinamento	[100 5000]
	mu inicial (algoritmo Levenberg–Marquardt)	[0.0001 0.1]
FANN2	Número de unidades na primeira camada escondida	[5 25]
	Número de unidades na segunda camada escondida	[5 25]
	Épocas de treinamento	[100 5000]
	mu inicial (algoritmo Levenberg–Marquardt)	[0.0001 0.1]
DBN	Número de unidades na primeira camada	[5 25]
	Número de unidades na primeira camada	[5 25]
	Épocas de treinamento	[100 5000]
	mu inicial (algoritmo Levenberg–Marquardt)	[0.0001 0.1]
	Taxa de aprendizagem do pré-treinamento	[0.01 1]
	Épocas de pré-treinamento	[50 500]
SDAE	Número de unidades na primeira camada	[5 25]
	Número de unidades na primeira camada	[5 25]
	Épocas de treinamento	[100 5000]
	mu inicial (algoritmo Levenberg–Marquardt)	[0.0001 0.1]
	Taxa de aprendizagem do pré-treinamento	[0.01 1]
	Épocas de pré-treinamento	[50 500]
SVR	Tipo do SVR (SVM, 2016)	$Epsilon - SVR$, $nu - SVR$
	Tipo da função de *kernel* (SVM, 2016)	Linear, polinomial, base radial, sigmóide
	Custo (SVM, 2016)	[0.1 100]
	Nu (SVM, 2016)	[0.1 1]
	Epsilon (SVM, 2016)	[0.1 1]
	Shrinking (SVM, 2016)	[0 1]

4.2.2 Geração II - Combinadores

No contexto de previsão de séries temporais, combinar preditores pode dizer respeito a calcular uma média ponderada das saídas dos especialistas base. Os pesos dessa média ponderada podem ser determinados através da utilização ou não de uma base de dados de treinamento. Tendo em vista experimentar diversas configurações nas simulações, o método experimental utilizou seis combinadores. Quatro desses combinadores são medidas estatísticas e, portanto, não necessitam de base de dados de treinamento: média, média aparada, média winsorizada e mediana. Os outros dois combinadores utilizados no método experimental necessitam de uma base de dados para cálculo dos pesos: RBLC (*Rank-Based*

Linear Combination) e *softmax*. Esses combinadores foram selecionados para o método experimental devido a sua facilidade de uso, pouco consumo computacional e uso bem sucedido na literatura.

Combinadores não-lineares incluem um fator de complexidade a mais no modelo, além de seu uso ser menos generalizado do que os métodos lineares. O menor apelo dos combinadores não-lineares, inclusive, pode ser observado como falta de evidência de sucesso em problemas menos específicos, como visto em (ELLIOTT; TIMMERMANN, 2013).

No método experimental desenvolvido, foram gerados para cada base de dados dez preditores base. Apesar de que em problemas de classificação um número elevado de especialistas base seja comum, quando se trata de previsão de séries temporais há indicações na literatura de se utilizar não mais do que cinco modelos (MAKRIDAKIS; WINKLER, 1983) (ADHIKARI, 2015). No sentido de verificar a eficácia dos combinadores com mais ou menos especialistas base, o método experimental inclui os resultados dos combinadores utilizando todos os preditores base e utilizando os melhores preditores de acordo com o seu desempenho em uma base de dados de validação. Assim, foi possível analisar se seria mais adequado utilizar todos os especialistas disponíveis ou um conjunto menor que contivesse os melhores.

4.2.3 Seleção

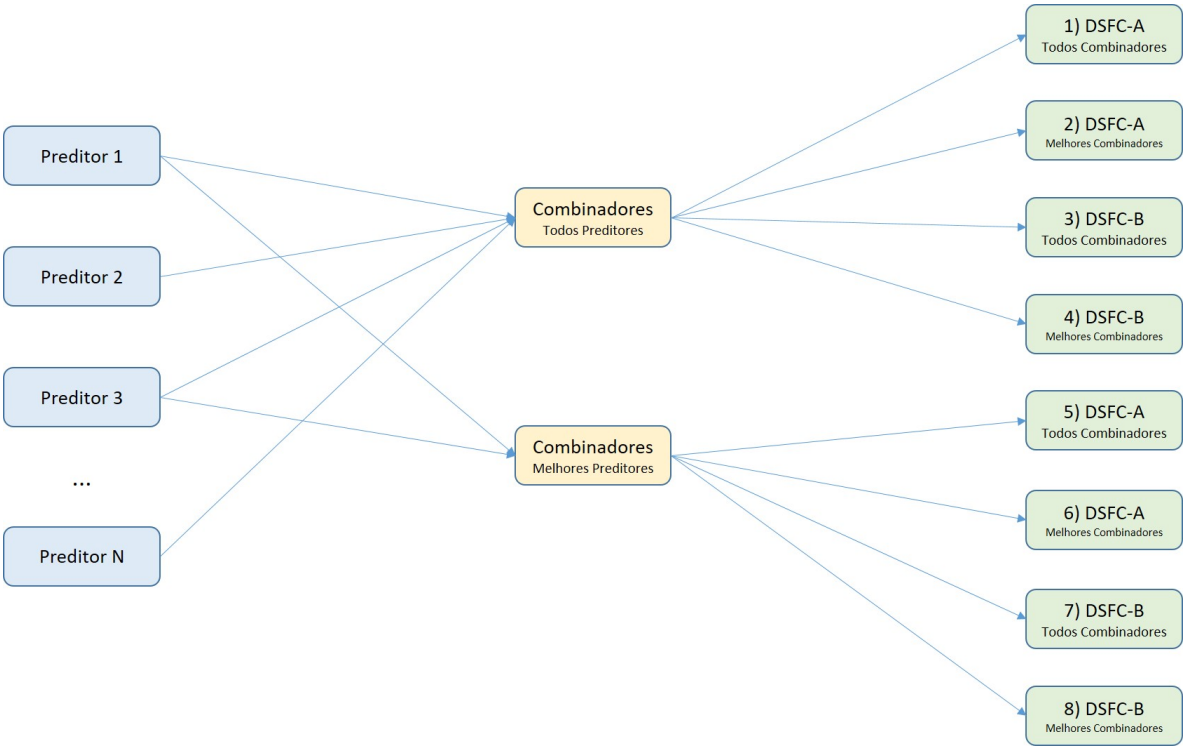
A partir da geração dos preditores base e sua posterior combinação, o método experimental produz a saída do arcabouço proposto utilizando os dois métodos de seleção desenvolvidos: DSFC-A e DSFC-B. Assim como no caso da combinação de preditores, a questão de quantos e quais combinadores utilizar na seleção dinâmica também é um tema importante. Já que a seleção dinâmica de combinadores é uma tarefa original, o método experimental buscou testar os métodos de seleção dinâmica utilizando separadamente todos os combinadores e apenas parte deles. Assim, poderia-se observar se um número maior ou menor de combinadores na seleção dinâmica seria capaz de aumentar ou diminuir o desempenho do método como um todo.

A figura 10 (página 61) mostra como as simulações numéricas foram segmentadas. Primeiro, os preditores base são gerados. Em seguida todos os combinadores são testados utilizando-se todos os preditores ou somente os melhores. Para os combinadores que utilizam as saídas de todos os preditores, são aplicados os dois métodos de seleção dinâmica, DSFC-A e DSFC-B, utilizando todos ou apenas os melhores combinadores. O mesmo raciocínio ocorre para os combinadores que utilizam as saídas dos melhores preditores em uma base de dados de validação. Por exemplo, na figura 10, a simulação numérica 7 indica o uso do DSFC-B que seleciona dinamicamente todos os combinadores que produziram sua saída a partir dos melhores preditores. Em contrapartida, a simulação numérica 2 indica o uso do DSFC-A que seleciona dinamicamente apenas os melhores combinadores que produziram

sua saída a partir de todos os preditores. Para determinar os melhores preditores e melhores combinadores, foram utilizadas bases de dados de validação.

O principal objetivo da realização dessas oito simulações numéricas é verificar e discutir o desempenho do arcabouço quando se variam algumas das características da combinação e da seleção dinâmica, respondendo algumas das questões lançadas na introdução desse trabalho. A seguir, a configuração e os parâmetros do método experimental.

Figura 10 – Simulações numéricas no método experimental



Fonte: do autor.

4.2.4 Configuração e Parâmetros dos Experimentos

O Algoritmo 11 (página 62) mostra a metodologia utilizada neste trabalho para obter os resultados, sendo estes apresentados e discutidos no próximo capítulo. A tabela 2 mostra a descrição e valores dos parâmetros utilizados. Os parâmetros foram definidos empiricamente, após realização de testes iniciais exaustivos sem a base de dados de teste.

De acordo com a figura 10, o algoritmo 11 é executado com oito diferentes variações, no que diz respeito ao uso de todos ou melhores preditores na combinação e ao uso de todos ou melhores combinadores na seleção dinâmica, além do uso do DSFC-A ou do DSFC-B. Nos resultados apresentados na próxima seção, o número dos melhores preditores e dos melhores combinadores foi dado por metade dos preditores e combinadores implementados: cinco e três, respectivamente.

Pseudocodigo 11 Seleção Dinâmica de Combinadores de Previsão de Séries Temporais

INPUT: Série temporal
OUTPUT: Média e desvio-padrão dos erros de previsão do método

1: **for** cada rodada $r \in runsNumber$ **do**
2: Construa a base de dados a partir da série temporal com janela de atraso n;
3: Baseado em $trainPercent$, $valPercent$ e $testPercent$, divida a base de dados em conjunto de treinamento (DB_{trI}), de validação (DB_{trII}) e de teste (DB_{te});
4: Faça validação cruzada *10-fold* de DB_{tr}, gerando bases de treinamento diferentes para cada um dos preditores base;
5: De acordo com $iterMax$, $swarmSize$ e $fitnessFunction$, rode o PSO padrão (KENNEDY; EBERHART, 1995) para obter os melhores preditores base dos modelos advindos da aprendizado de máquina;
6: Obtenha a melhor configuração dos modelos estatísticos
7: Utilizando DB_{trII}, calcule os pesos das combinações que necessitam de base de dados de treinamento.
8: Utilizando DB_{te}, calcule as previsões dos modelos base e dos combinadores;
9: Utilizando DB_{te}, obtenha a previsão do arcabouço proposto a partir da seleção dinâmica de combinadores;
10: Calcule os erros de previsão dos modelos base, dos combinadores e da seleção dinâmica;
11: **end for**

Tabela 2 – Parâmetros do método proposto

Nome	Descrição	Valor
n	Janela de tempo para geração da base de dados	5
runsNumber	Número de execuções do método	30
trainPercent	Porcentagem do conjunto de treinamento I	70%
valPercent	Porcentagem do conjunto de treinamento II	20%
testPercent	Porcentagem do conjunto de teste	10%
iterMax	Número máximo de iterações do PSO	10
swarmSize	Tamanho da população do PSO	20
fitnessFunction	Função de aptidão do PSO	MSE (Equação 4.3)
SimilarityThreshold (DSFC-B)	Limiar do algoritmo de seleção dinâmica	0.15

Outro fator que implica uma variabilidade nos resultados alcançados é o horizonte de previsão. Nas simulações numéricas de todas as séries temporais utilizadas, foram calculadas as saídas e erro de previsão de curto alcance (horizonte de previsão 1) e de longo alcance (horizonte de previsão 10).

Diversas medidas de erro de previsão foram calculadas, para facilitar a análise dos resultados e a comparação com trabalhos na literatura. São elas: NMSE (*Normalised Mean Square Error*, equação 4.1), NRMSE (*Normalised Root Mean Square Error*, equação 4.2), MSE (*Mean Square Error*, equação 4.3) e RMSE (*Root Mean Square Error*). Nas medidas de erro de previsão, P é o número total de padrões no conjunto, T_{ij} e L_{ij} são respectivamente os valores reais e os valores calculados pelo modelo e $var(t)$ é a variância

dos valores no conjunto de saídas desejadas.

$$\text{NMSE} = \frac{\sum_{i=1}^{P} \frac{(T_i - L_i)^2}{var(t)}}{P} \tag{4.1}$$

$$\text{NRMSE} = \frac{\sum_{i=1}^{P} sqrt(\frac{(T_i - L_i)^2}{var(t)})}{P} \tag{4.2}$$

$$\text{MSE} = \frac{\sum_{i=1}^{P} (T_i - L_i)^2}{P} \tag{4.3}$$

$$\text{RMSE} = sqrt(\frac{\sum_{i=1}^{P} (T_i - L_i)^2}{P}) \tag{4.4}$$

4.2.5 Séries Temporais

A seleção dinâmica proposta neste trabalho foi aplicada ao problema de previsão de séries temporais. No método experimental realizado, foram utilizadas séries temporais com comportamento caótico. Como mostrado anteriormente, a importância do estudo de séries caóticas passa por áreas como processamento de sinais e astronomia, sendo também um importante *benchmark* para modelos de previsão. Dez séries temporais caóticas foram utilizadas, apresentadas nas próximas subseções. Todas as séries foram apresentadas ao arcabouço de seleção dinâmica com 1000 pontos.

4.2.5.1 Mackey-Glass

A série Mackey-Glass (figura 11), contínua, unidimensional e *benchmark* padrão para teste de previsão de séries temporais é formada pela equação 4.5:

$$\frac{dx}{dt} = \beta \frac{x_\tau}{1 + x_\tau^n} - \gamma x, \gamma, \beta, n > 0 \tag{4.5}$$

onde β, τ, γ e n são números reais e x_τ representa o valor da variável x no tempo $(t - \tau)$. A dinâmica caótica aparece quando $\tau > 16.8$. Os seguintes parâmetros foram utilizados: $\tau = 17$, $\beta = 2$, $\gamma = 0.1$ e $n = 10$.

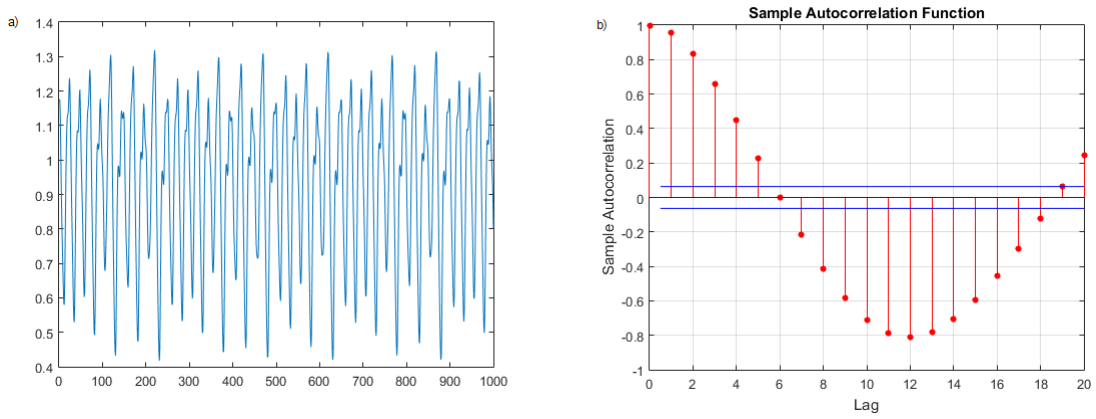


Figura 11 – Série temporal Mackey-Glass: pontos (a) e autocorrelações (b)

4.2.5.2 Lorenz

A série temporal de Lorenz (figura 12), introduzida pelo pesquisador de mesmo nome em (LORENZ, 1963), é dada pela equação 4.6:

$$\begin{aligned} \frac{dx}{dt} &= \sigma[y - x] \\ \frac{dy}{dt} &= rx - y - xz \\ \frac{dz}{dt} &= xy - bz \end{aligned} \tag{4.6}$$

onde os seguintes parâmetros foram utilizados nos experimentos: $\sigma = 10$, $r = 28$ e $b = 8/3$.

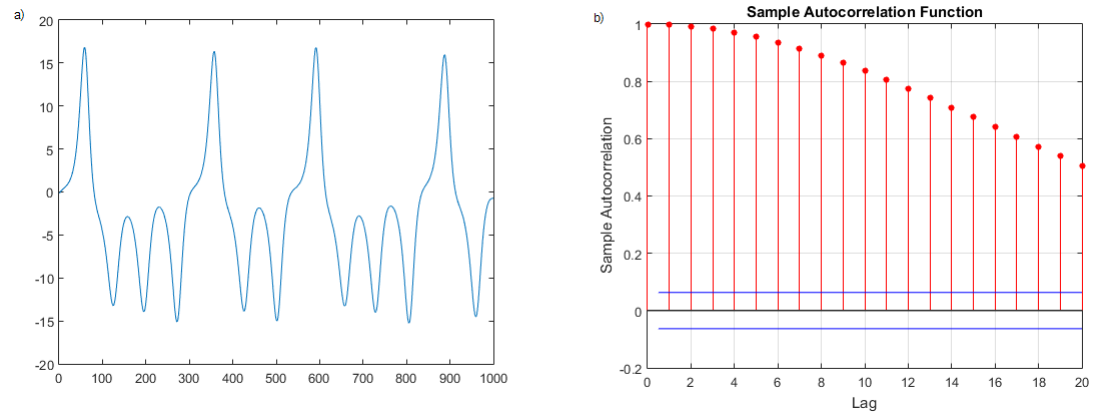


Figura 12 – Série temporal Lorenz: pontos (a) e autocorrelações (b)

4.2.5.3 Henon

O mapa de Henon (figura 13) é dado pela equação 4.7 (FLAKE, 1998):

$$\begin{aligned} x_{n+1} &= y_n + 1 - \alpha x_n^2 \\ y_{n+1} &= \beta x_n \end{aligned} \tag{4.7}$$

onde os seguintes parâmetros foram utilizados nos experimentos: $\alpha = 1.4$ e $\beta = 0.3$.

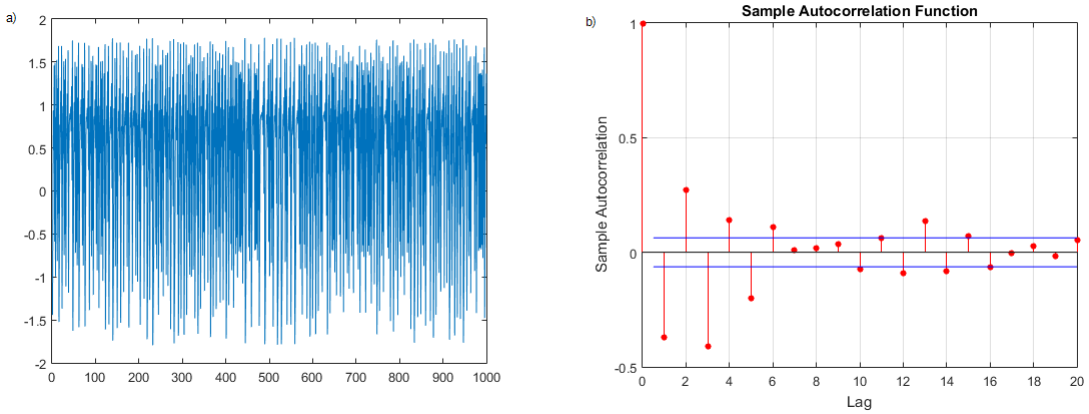


Figura 13 – Série temporal Henon: pontos (a) e autocorrelações (b)

4.2.5.4 Rossler

A série temporal de Rossler (figura 14), introduzida pelo pesquisador de mesmo nome em (RÖSSLER, 1976), é dada pela equação 4.8:

$$\begin{aligned} \frac{dx}{dt} &= -z - y \\ \frac{dy}{dt} &= x + ay \\ \frac{dz}{dt} &= b + z(x - c) \end{aligned} \tag{4.8}$$

onde os seguintes parâmetros foram utilizados nos experimentos: $a = 0.15$, $b = 0.2$ e $c = 10$.

4.2.5.5 *Periodic* e *Quasi-Periodic*

Estas séries temporais de velocidade foram obtidas através de experimentos em filmes fotográficos. Detalhe podem ser vistos em (CHAOTIC, 2016). As figuras 15 e 16 mostram respectivamente as séries chamadas *Periodic* e *Quasi-Periodic*.

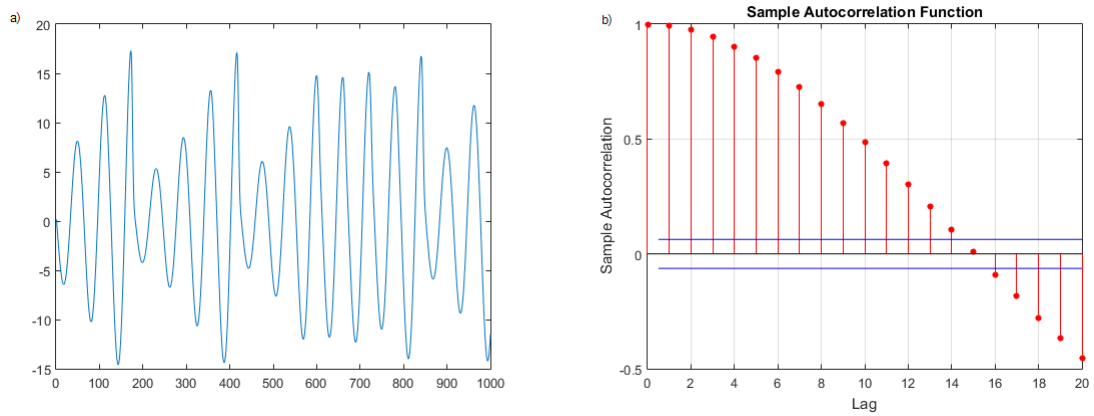


Figura 14 – Série temporal Rossler: pontos (a) e autocorrelações (b)

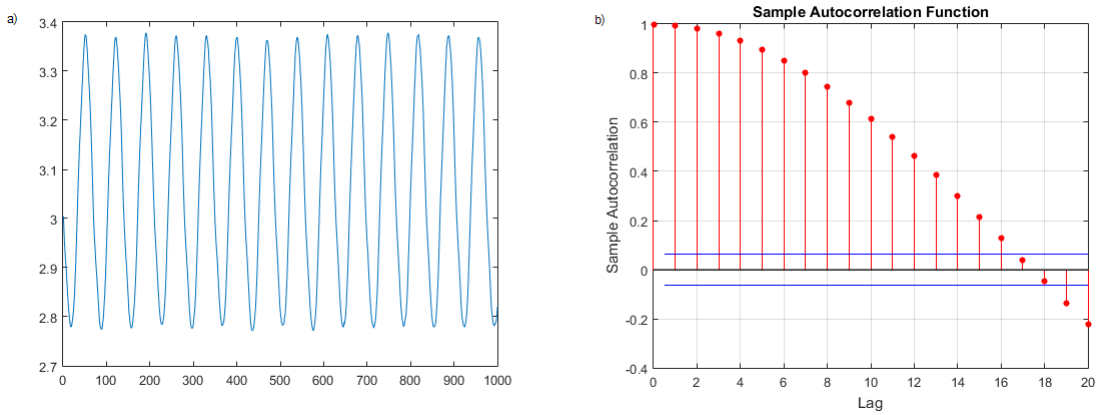


Figura 15 – Série temporal *Periodic*: pontos (a) e autocorrelações (b)

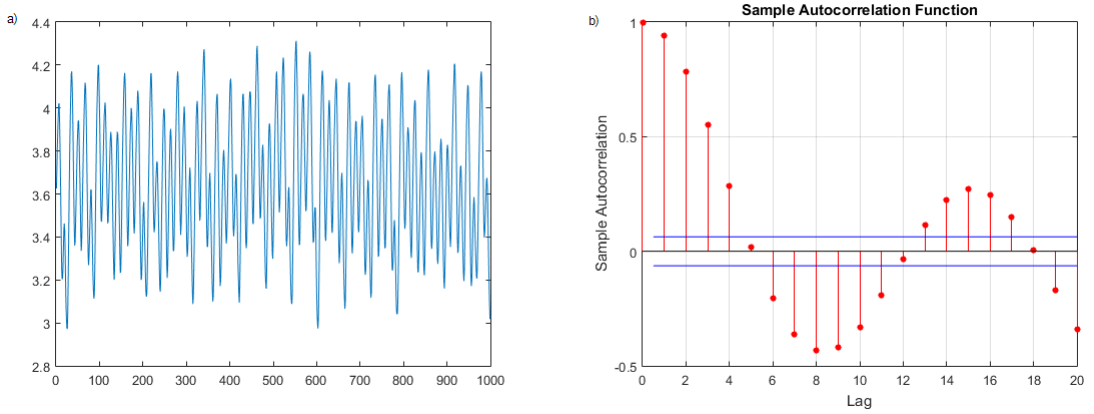


Figura 16 – Série temporal *Quasi-Periodic*: pontos (a) e autocorrelações (b)

4.2.5.6 Laser

Laser é uma série temporal univariada obtida a partir de medidas coletadas em um laboratório de física. Os dados são um corte transversal periódico da intensidade de um laser, sendo que as pulsações geradas seguem um padrão semelhante ao modelo teórico de Lorenz. A série é utilizada como *benchmark* de previsão de séries temporais devido a sua simplicidade e padrões bem documentados e inteligíveis. Os dados foram obtidos em

(LASER, 2016) e são mostrados na figura 17.

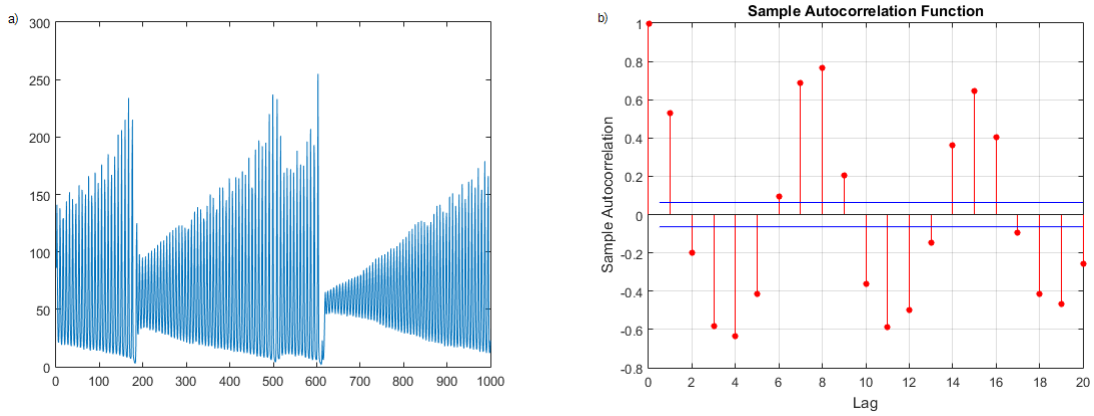


Figura 17 – Série temporal Laser: pontos (a) e autocorrelações (b)

4.2.5.7 Eletroencefalograma

O eletroencefalograma, conhecido como EEG, é um exame médico que analisa a atividade cerebral instantânea dos indivíduos, normalmente captada através de eletrodos. A relação entre séries temporais de eletroencefalograma com sistemas dinâmicos podem ser vistos em diversos trabalhos, como em (SAMANTA, 2011) e (WANG et al., 2010).

A primeira série temporal de eletroencefalograma usada neste trabalho, EEG1, foi obtida através de exames em um paciente humano (ANDRZEJAK et al., 2001). Já as outras duas, EEG2 e EEG3, foram obtidas através de exames em ratos de laboratório (EEG, 2016). As figuras 18, 19 e 20 mostram os dados de EEG1, EEG2 e EEG3, respectivamente.

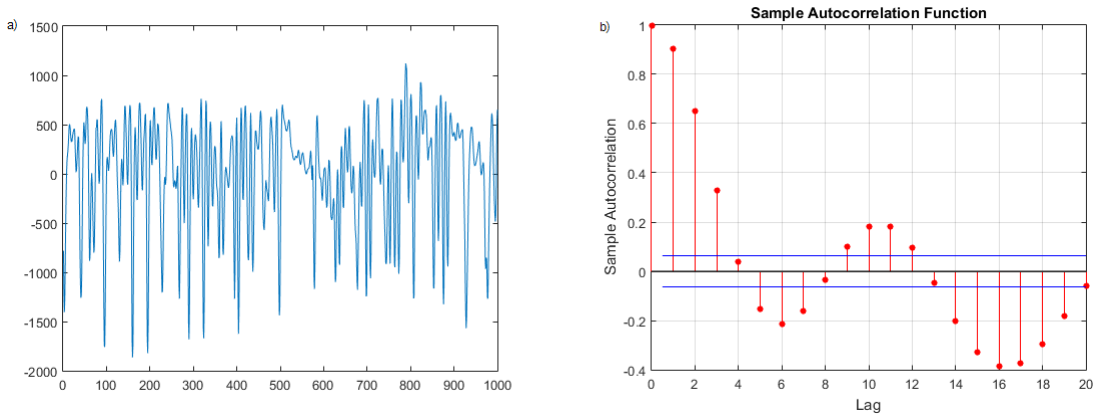


Figura 18 – Série temporal EEG1

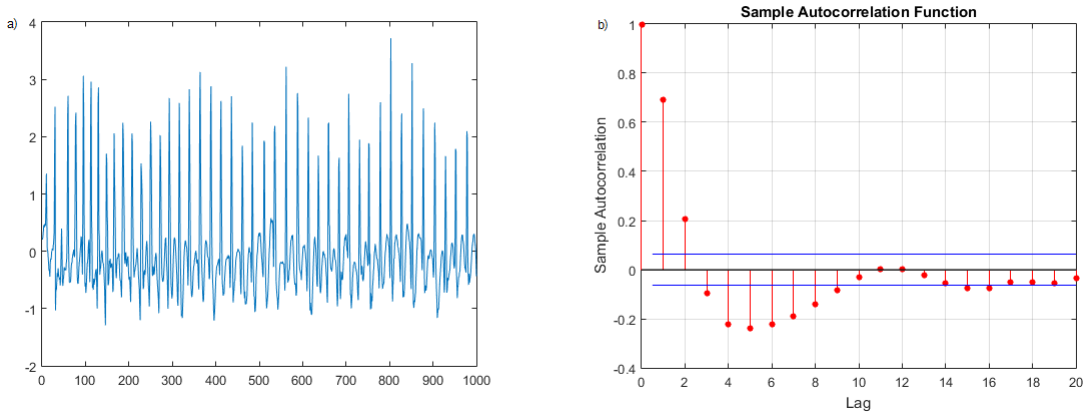


Figura 19 – Série temporal EEG2

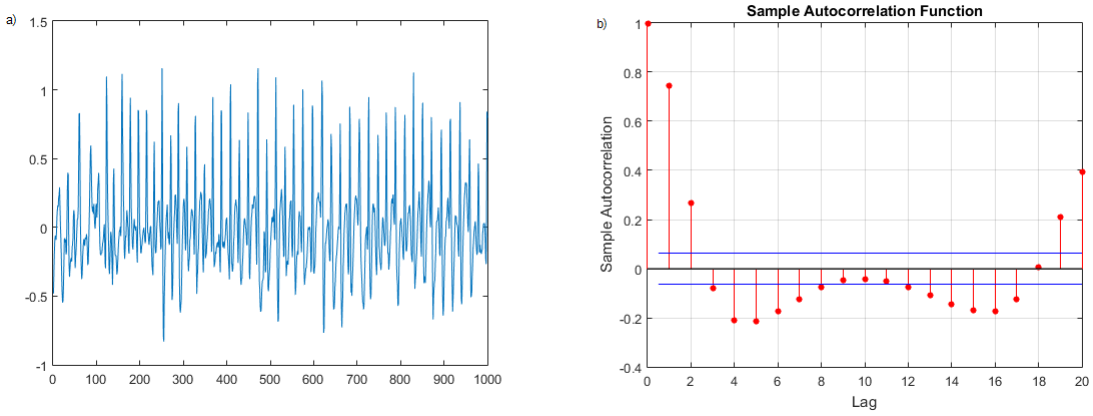


Figura 20 – Série temporal EEG3

5 RESULTADOS E DISCUSSÃO

Este capítulo apresenta e discute os resultados alcançados através da execução do método experimental desenvolvido. O capítulo está dividido entre a análise dos resultados da previsão de curto alcance e da previsão de longo alcance.

5.1 Previsão de Curto Alcance

A seguir, a apresentação dos resultados alcançados na previsão de curto alcance. A análise foi dividida de acordo com as fases do arcabouço proposto: preditores bases, combinadores e seleção dinâmica. A seção também inclui uma argumentação a respeito da complexidade computacional das abordagens propostas em relação ao tempo de processamento.

5.1.1 Preditores Base

A tabela 3 mostra a média e o desvio-padrão do MSE dos preditores de aprendizado de máquina na base de testes, após 30 execuções do método. A tabela 4 (página 70) faz o mesmo para os preditores estatísticos. Em negrito, os melhores desempenhos para cada base de dados, não considerando se há diferença estatística.

Tabela 3 – MSE dos preditores de aprendizado de máquina (30 execuções, teste)

Base de Dados	FANN-1	FANN-2	DBN	SDAE	SVR
Mackey-Glass	1.35e-06	1.11e-06	**1.03e-06**	1.12e-06	2.21e-05
	(2.48e-07)	(3.22e-07)	**(2.05e-07)**	(2.37e-07)	(9.66e-07)
Lorenz	8.48e-09	5.07e-09	**1.95e-09**	5.56e-09	1.51e-05
	(8.34e-09)	(7.69e-09)	**(2.55e-09)**	(8.52e-09)	(5.12e-06)
Henon	1.54e-10	1.28e-10	**4.62e-11**	1.25e-09	2.46e-05
	(1.40e-10)	(2.84e-10)	**(1.54e-11)**	(2.73e-09)	(2.59e-06)
Rossler	3.69e-08	5.43e-08	**1.21e-08**	3.40e-08	1.33e-04
	(2.79e-08)	(8.67e-08)	**(1.35e-08)**	(5.25e-08)	(6.47e-05)
Periodic	6.85e-06	6.80e-06	**6.55e-06**	7.20e-06	7.82e-06
	(2.89e-07)	(4.14e-07)	**(3.89e-07)**	(5.55e-07)	(1.62e-07)
Quasi-Periodic	**3.65e-04**	3.84e-04	3.69e-04	3.99e-04	4.15e-04
	(2.12e-05)	(3.36e-05)	(2.32e-05)	(3.18e-05)	(1.16e-05)
Laser	7.40e+00	**4.48e+00**	5.87e+00	6.09e+00	2.17e+01
	(3.83e+00)	**(1.93e+00)**	(2.32e+00)	(2.70e+00)	(4.84e+00)
EEG1	5.00e+03	5.58e+03	4.98e+03	7.43e+03	**4.89e+03**
	(4.98e+02)	(1.08e+03)	(1.21e+03)	(2.19e+03)	**(2.94e+02)**
EEG2	7.04e-02	**6.09e-02**	6.44e-02	6.42e-02	8.95e-02
	(8.77e-03)	**(9.61e-03)**	(8.25e-03)	(1.27e-02)	(5.06e-03)
EEG3	9.41e-03	**8.94e-03**	8.97e-03	9.61e-03	1.31e-02
	(6.46e-04)	**(1.01e-03)**	(8.68e-04)	(1.14e-03)	(7.92e-04)

É possível observar que, dentre os preditores base, o modelo DBN obteve melhor desempenho em metade das bases de dados: Mackey-Glass, Lorenz, Henon, Rossler e

Tabela 4 – MSE dos preditores estatísticos (30 execuções, teste)

Base de Dados	AR	MA	ARMA	ARIMA	GARCH
Mackey-Glass	**6.80e-05**	1.59e-03	1.12e-04	1.92e-04	3.19e-02
	(2.76e-20)	(1.10e-18)	(2.76e-20)	(1.10e-19)	(2.12e-17)
Lorenz	**5.16e-08**	1.43e+00	5.37e-08	8.87e-08	4.35e+01
	(2.69e-23)	(9.03e-16)	(2.69e-23)	(4.04e-23)	(2.17e-14)
Henon	1.08e+00	**7.36e-01**	2.24e+00	1.37e+00	1.53e+00
	(4.52e-16)	**(2.26e-16)**	(1.81e-15)	(0.00e+00)	(2.26e-16)
Rossler	7.36e-05	9.97e+01	**6.06e-06**	7.67e-04	3.54e+01
	(0.00e+00)	(5.78e-14)	**(1.72e-21)**	(3.31e-19)	(1.45e-14)
Periodic	**1.10e-05**	2.27e-03	1.79e-05	1.50e-05	8.50e-03
	(6.89e-21)	(8.82e-19)	(3.45e-21)	(6.89e-21)	(3.53e-18)
Quasi-Periodic	**5.43e-04**	1.39e-01	9.43e-04	1.74e-03	7.77e-02
	(3.31e-19)	(5.65e-17)	(6.62e-19)	(4.41e-19)	(2.82e-17)
Laser	1.13e+03	**6.47e+02**	3.05e+03	2.77e+03	3.07e+03
	(4.63e-13)	**(1.16e-13)**	(9.25e-13)	(2.31e-12)	(2.31e-12)
EEG1	4.79e+03	1.39e+04	8.61e+03	**4.22e+03**	3.91e+05
	(0.00e+00)	(7.40e-12)	(5.55e-12)	**(2.78e-12)**	(5.92e-11)
EEG2	1.85e-01	**1.64e-01**	1.24e+00	3.97e-01	9.01e-01
	(0.00e+00)	**(2.82e-17)**	(0.00e+00)	(1.13e-16)	(3.39e-16)
EEG3	2.18e-02	**1.89e-02**	7.61e-02	4.10e-02	2.30e-01
	(0.00e+00)	**(3.53e-18)**	(0.00e+00)	(1.41e-17)	(1.13e-16)

Periodic. Outro modelo que se destacou foi a rede neural *feedforward* com duas camadas escondidas, sem qualquer espécie de pré-treinamento, obtendo o melhor resultado em Laser, EEG2 e EEG3. Ambos os modelos podem ser considerados *Deep Learning*, fazendo dessa categoria de rede neural uma importante solução para o problema de previsão de séries temporais caóticas. Uma camada a mais nos modelos de redes neurais pode ter sido a responsável por captar mais convenientemente as relações entre os pontos das séries temporais testadas. Em algumas bases, a diferença de desempenho da DBN em relação ao SVR chega a ser de ordens de grandeza distintas. É o que pode ser visto na série temporal Lorenz, onde o MSE varia de e-05 a e-09. O SVR, entretanto, destacou-se na base de dados EEG1.

De maneira geral, os preditores estatísticos obtiveram resultados sensivelmente piores do que os melhores modelos de aprendizado de máquina. Porém, na base EEG1, o melhor modelo individual foi o ARIMA. Esse cenário corrobora a ideia de que um comitê com o maior grau de diversidade possível é mais aconselhável. Por mais que os modelos estatísticos não tenham obtido desempenho satisfatório quando comparados aos preditores de aprendizado de máquina, um modelo estatístico linear alcançou a menor taxa de erro em uma das bases de dados testadas. A confiança em apenas uma categoria de preditor pode deixar escapar desempenhos apropriados de modelos com resultados em geral piores. Além disso, por mais que os preditores estatísticos tenham apresentados taxas de erro maiores, eles contribuem para a diversidade do comitê, impactando positivamente a combinação dos modelos individuais e a posterior seleção dinâmica.

Como mostrado anteriormente, os preditores originários da aprendizado de máquina

tiveram seus parâmetros de treinamento otimizados pelo PSO. As figuras 21, 22, 23, 24, 25, 26, 27, 28, 29 e 30 mostram a evolução da curva de aptidão do melhor indivíduo de cada preditor em uma rodada de execução de cada uma das séries temporais testadas. Nas curvas, o eixo das ordenadas indica o número de iterações do PSO, enquanto que o eixo das abscissas indica a função de aptidão. No método experimental realizado, a função de aptidão é dada pelo MSE do modelo em uma base de dados de validação.

É possível perceber que, na maioria dos cenários testados, a curva de aptidão apresentou comportamento similar ao esperado quando se usa a otimização por enxame de partículas. A aptidão normalmente tem uma melhora íngreme no início das iterações (representada pela queda da taxa de erro) e uma evolução mais modesta nas iterações posteriores, equivalente a uma curva exponencial negativa. Tais estados são conhecidos como exploração e explotação. O estado de exploração corresponde à busca inicial dos indivíduos, compreendendo as primeiras iterações. No estado de explotação, algumas partículas estão já agrupadas, possivelmente em ótimos locais. Obviamente distorções acabam acontecendo, como no modelo FANN-2 na base de dados Quasi-Periodic (figura 26).

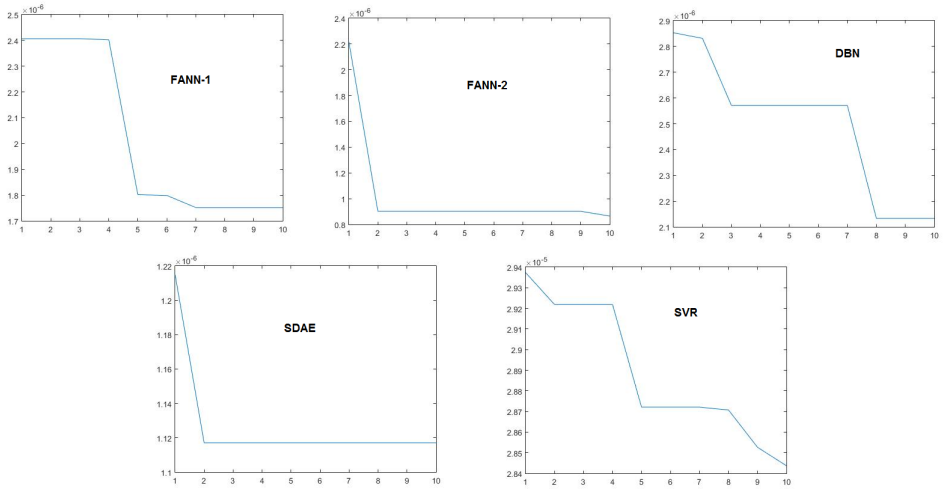


Figura 21 – Mackey-Glass - Curva de aptidão do PSO

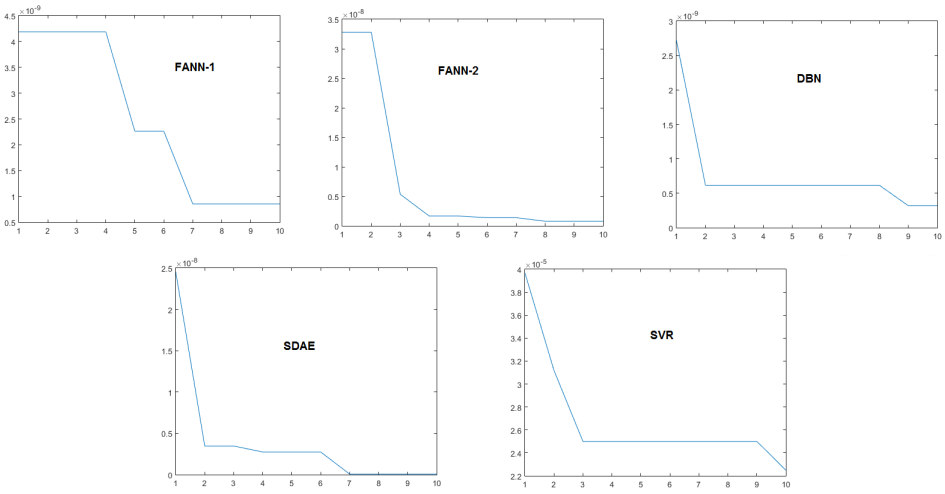


Figura 22 – Lorenz - Curva de aptidão do PSO

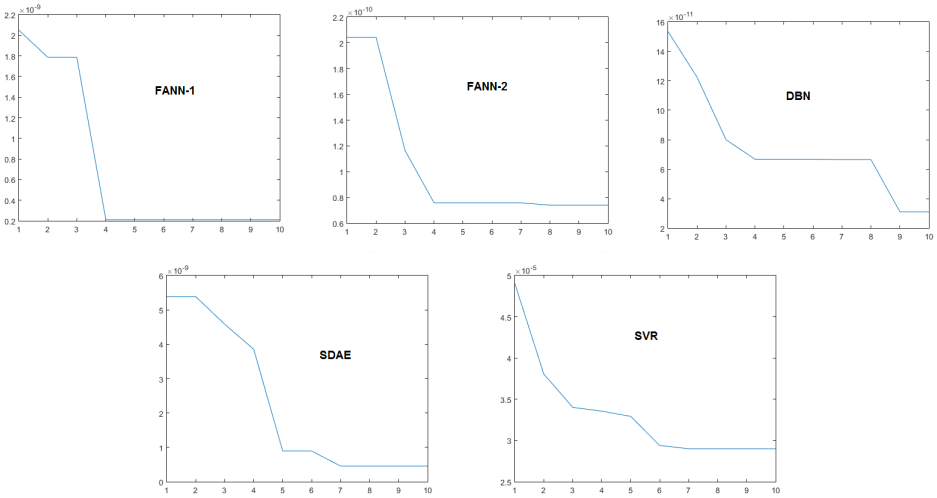


Figura 23 – Henon - Curva de aptidão do PSO

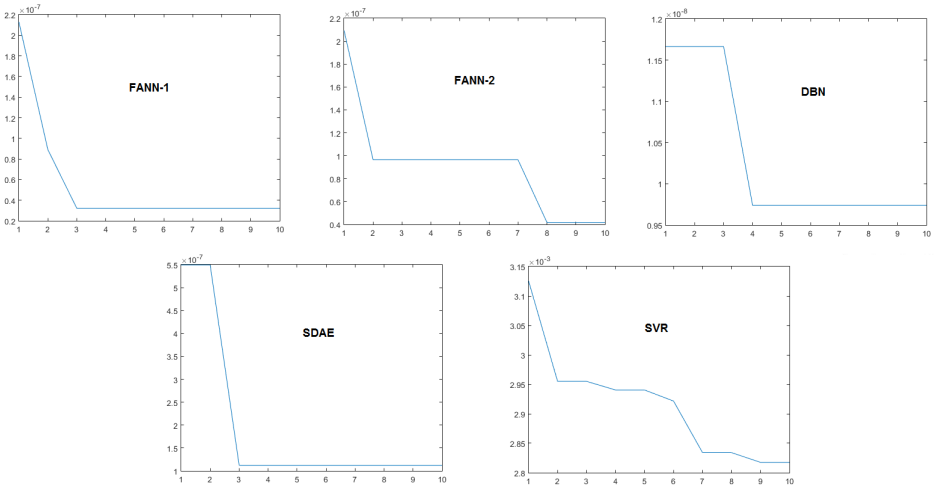


Figura 24 – Rossler - Curva de aptidão do PSO

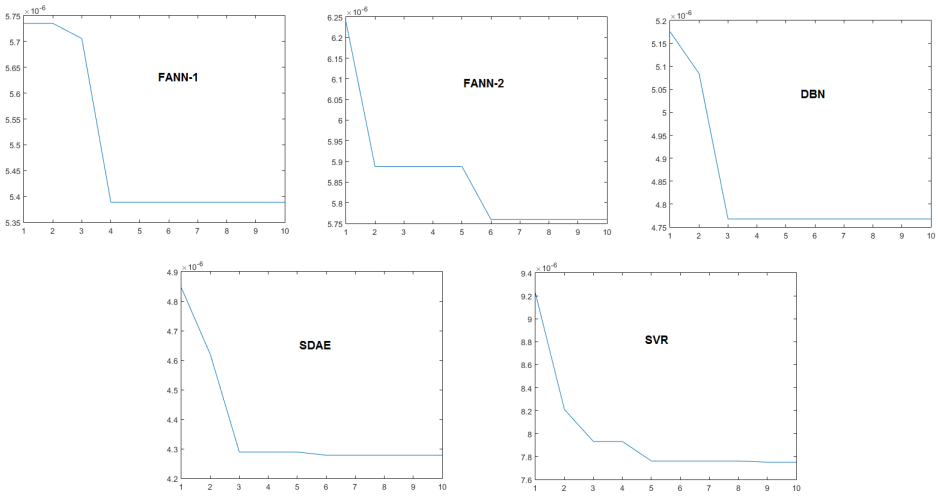


Figura 25 – Periodic - Curva de aptidão do PSO

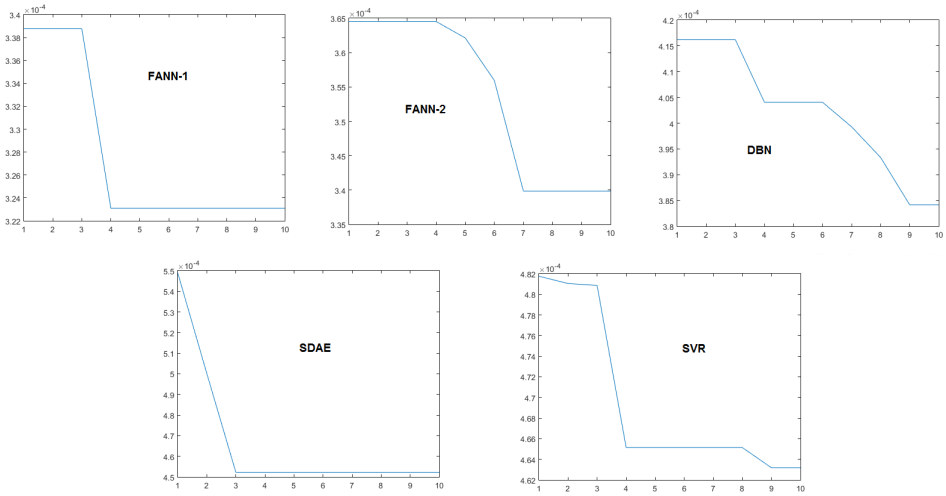


Figura 26 – Quasi-Periodic - Curva de aptidão do PSO

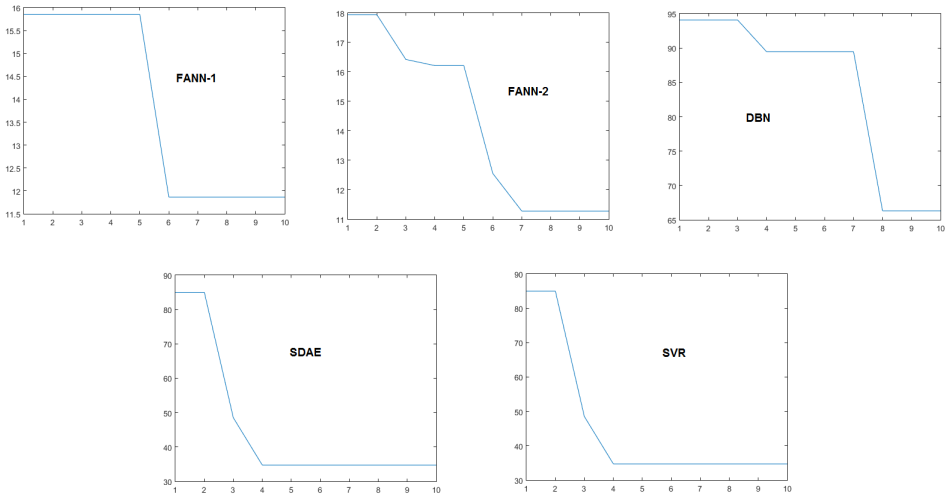


Figura 27 – Laser - Curva de aptidão do PSO

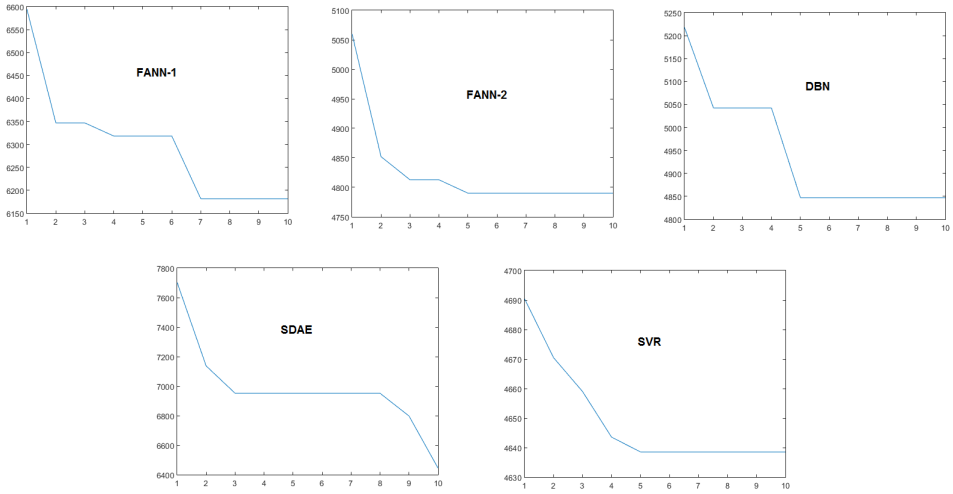


Figura 28 – EEG1 - Curva de aptidão do PSO

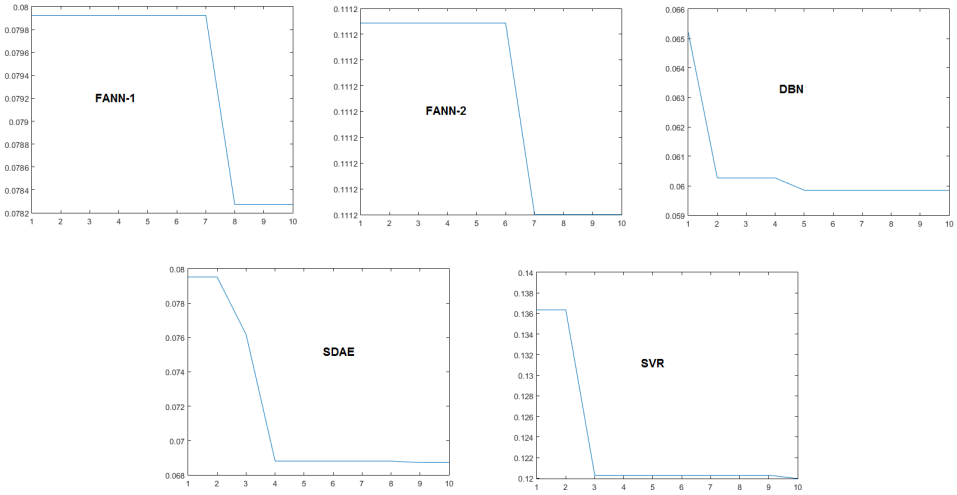


Figura 29 – EEG2 - Curva de aptidão do PSO

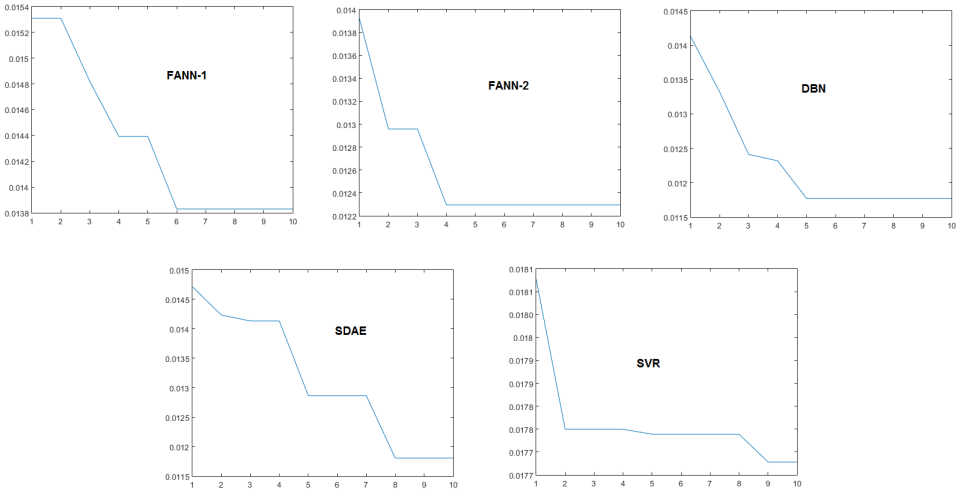


Figura 30 – EEG3 - Curva de aptidão do PSO

5.1.2 Combinadores

A tabela 5 mostra a média e o desvio-padrão do MSE dos combinadores que tiveram como entrada todos os dez preditores disponíveis na base de dados de teste, após 30 execuções. Já a tabela 6 mostra os resultados para os combinadores que tiveram como entrada os cinco melhores preditores, definidos a partir de seu desempenho em uma base de validação. Como explicado no capítulo anterior, os experimentos consideraram essas duas diferentes configurações no sentido de verificar se a combinação deve ter como entrada um número maior ou menor de preditores. Em trabalhos de classificação e reconhecimento de padrões, o comitê tende a ter um número de especialistas maior do que a quantidade de preditores nestas simulações numéricas. Por outro lado, quando o problema a ser resolvido é a combinação de previsão de séries temporais, alguns autores recomendam um número reduzido de especialistas base (MAKRIDAKIS; WINKLER, 1983).

Tabela 5 – MSE dos combinadores para todos os preditores (30 execuções, teste)

Base de Dados	Média	Média Aparada	Média Winsorizada	Mediana	RBLC	Softmax
Mackey-Glass	3.11e-04	2.19e-05	2.09e-05	5.75e-06	6.92e-03	**9.79e-07**
	(5.23e-07)	(1.87e-07)	(1.76e-07)	(2.09e-07)	(8.10e-06)	**(2.44e-07)**
Lorenz	4.59e-01	1.39e-02	8.89e-03	1.92e-08	2.80e+00	**1.20e-09**
	(1.15e-04)	(1.28e-05)	(1.60e-05)	(4.93e-09)	(3.37e-04)	**(1.90e-09)**
Henon	2.45e-01	1.73e-01	1.83e-01	7.60e-02	1.25e-01	**4.36e-11**
	(2.69e-05)	(2.00e-05)	(1.65e-05)	(4.58e-05)	(2.11e-05)	**(1.23e-11)**
Rossler	1.26e+00	1.64e-01	1.06e-01	3.04e-06	1.32e+00	**8.24e-09**
	(5.86e-04)	(9.72e-05)	(7.37e-05)	(1.35e-06)	(4.52e-02)	**(7.20e-09)**
Periodic	1.12e-04	1.35e-05	1.15e-05	7.16e-06	4.09e-03	**6.52e-06**
	(5.65e-07)	(1.81e-07)	(1.67e-07)	(1.91e-07)	(5.15e-06)	**(3.11e-07)**
Quasi-Periodic	3.35e-03	1.25e-03	1.02e-03	3.62e-04	6.58e-03	**3.39e-04**
	(1.98e-05)	(1.35e-05)	(1.23e-05)	(1.09e-05)	(3.58e-05)	**(1.51e-05)**
Laser	2.88e+02	1.82e+02	1.94e+02	8.02e+01	2.67e+02	**3.86e+00**
	(3.58e+00)	(2.79e+00)	(2.70e+00)	(4.79e+00)	(6.22e+00)	**(7.77e-01)**
EEG1	7.97e+03	4.21e+03	**4.18e+03**	4.20e+03	4.34e+03	4.32e+03
	(2.39e+02)	(1.36e+02)	**(1.34e+02)**	(1.40e+02)	(1.71e+02)	(1.54e+02)
EEG2	1.30e-01	9.71e-02	9.50e-02	9.64e-02	9.94e-02	**6.29e-02**
	(3.65e-03)	(3.14e-03)	(3.22e-03)	(4.13e-03)	(4.09e-03)	**(4.49e-03)**
EEG3	1.45e-02	1.11e-02	1.12e-02	9.69e-03	1.79e-02	**8.76e-03**
	(3.83e-04)	(3.28e-04)	(3.22e-04)	(4.48e-04)	(5.13e-04)	**(4.43e-04)**

Embora a média seja uma combinação largamente utilizada na literatura, e que em geral apresenta um melhor desempenho em relação aos modelos individuais, tal situação não ocorreu quando se levaram em consideração todos os preditores. Uma explicação para esse comportamento é a natureza da média, medida estatística muito suscetível a discrepâncias. Foi justamente o que aconteceu nos experimentos realizados: na maioria das bases de dados testadas o desempenho do modelo SVR destoou consideravelmente das redes neurais, puxando o desempenho da média para baixo. A média aparada e a média winsorizada têm justamente o objetivo de suavizar os valores discrepantes, explicando assim seu desempenho melhor em relação a média simples. De acordo com os resultados, a mediana conseguiu superar esse cenário, fazendo-se útil em situações que tenham essa

Tabela 6 – MSE dos combinadores para os melhores preditores (30 execuções, teste)

Base de Dados	Média	Média Aparada	Média Winsorizada	Mediana	RBLC	Softmax
Mackey-Glass	7.64e-07	7.64e-07	7.48e-07	**7.31e-07**	5.21e-03	9.79e-07
	(1.18e-07)	(1.18e-07)	(1.12e-07)	**(1.05e-07)**	(1.49e-05)	(2.44e-07)
Lorenz	1.56e-09	1.56e-09	1.35e-09	1.13e-09	2.50e+00	**1.20e-09**
	(1.26e-09)	(1.26e-09)	(1.16e-09)	(1.07e-09)	(6.96e-05)	**(1.90e-09)**
Henon	1.21e-10	1.21e-10	8.52e-11	4.46e-11	1.03e-01	**4.36e-11**
	(1.82e-10)	(1.82e-10)	(9.92e-11)	(1.53e-11)	(3.69e-06)	**(1.23e-11)**
Rossler	1.12e-08	1.12e-08	9.97e-09	9.03e-09	4.38e+00	**8.24e-09**
	(8.83e-09)	(8.83e-09)	(7.21e-09)	(6.58e-09)	(4.94e-04)	**(7.20e-09)**
Periodic	**6.45e-06**	**6.45e-06**	**6.45e-06**	6.47e-06	1.47e-03	6.52e-06
	(2.29e-07)	**(2.29e-07)**	**(2.24e-07)**	(2.25e-07)	(6.33e-06)	(3.11e-07)
Quasi-Periodic	**3.39e-04**	**3.39e-04**	**3.39e-04**	3.42e-04	3.32e-03	3.42e-04
	(1.54e-05)	**(1.54e-05)**	**(1.46e-05)**	(1.31e-05)	(8.28e-05)	(1.59e-05)
Laser	4.24e+00	4.24e+00	4.04e+00	**3.80e+00**	1.87e+01	3.87e+00
	(1.05e+00)	(1.05e+00)	(9.63e-01)	**(9.18e-01)**	(3.16e+00)	(7.80e-01)
EEG1	**4.34e+03**	**4.34e+03**	4.35e+03	4.39e+03	7.04e+03	4.40e+03
	(2.27e+02)	**(2.27e+02)**	(2.24e+02)	(2.32e+02)	(6.76e+02)	(2.04e+02)
EEG2	5.60e-02	5.60e-02	5.53e-02	5.45e-02	**5.34e-02**	5.56e-02
	(4.67e-03)	(4.67e-03)	(4.50e-03)	(4.49e-03)	**(4.44e-03)**	(5.10e-03)
EEG3	8.05e-03	8.05e-03	7.99e-03	**7.93e-03**	8.23e-03	7.98e-03
	(5.04e-04)	(5.04e-04)	(4.50e-04)	**(4.09e-04)**	(4.64e-04)	(4.98e-04)

característica, a possibilidade de um dos modelos individuais apresentar desempenho consideravelmente inferior. O desempenho da mediana foi melhor do que o melhor preditor em metade das séries temporais.

Em termos absolutos, o softmax obteve o melhor desempenho dentre os combinadores que tiveram como entrada todos os preditores. A taxa de erro desse combinador, inclusive, foi menor do que o melhor preditor em nove das dez bases de dados testadas. A explicação para esse resultado origina-se da forma com que o softmax calcula os pesos da combinação linear, levando em conta o desempenho dos preditores individuais em uma base de dados de validação. Nesse sentido, os preditores com desempenho inferior acabam tendo uma influência menor no resultado da combinação. Entretanto, utilizar um combinador que utiliza base de validação para o cálculo dos pesos não é garantia de sucesso. O RBLC acabou tendo desempenho inferior na maioria das séries temporais testadas.

De maneira geral, os resultados dos combinadores que tiveram como entrada os cinco melhores preditores alcançaram resultados melhores. Tanto a média simples como a média winsorizada melhoraram seu desempenho, devido ao fato de que os preditores com maiores taxas de erros foram excluídos da combinação linear. A média aparada obteve os mesmos resultados da média simples, pois não houve a necessidade de se excluir os valores discrepantes. A mediana também teve seu desempenho melhorado com a exclusão dos preditores com taxas de erro mais altas. Esses resultados corroboram os autores que recomendam o uso de até cinco modelos para a combinação linear (MAKRIDAKIS; WINKLER, 1983). Principalmente em relação aos combinadores estatísticos, o uso de

mais preditores base acaba sendo nocivo à predição final. Quando da presença de muitos modelos, os combinadores mais simples tendem a levar em consideração mesmo aqueles preditores com desempenho inferior. Por mais que um preditor tenha uma alta taxa de erro, ele vai acabar influenciando a predição do combinador de alguma forma, por vezes de maneira excessiva.

Na maioria das bases de dados, o softmax teve o mesmo desempenho tanto com o uso de todos os preditores quanto com o uso dos cinco melhores. Essa característica faz desse combinador uma boa alternativa para quando o experimentador não sabe ao certo a quantidade de preditores base que deve ser utilizada. O softmax foi capaz de excluir automaticamente os preditores com menor desempenho. O custo para isso é o uso da base de dados de validação, diferentemente dos combinadores estatísticos. Porém, como no cenário de combinação com todos os dez modelos individuais, o RBLC também não foi capaz de superar os combinadores mais simples na maioria das bases de dados testadas.

Em termos absolutos, apesar de o softmax ter se destacado na maioria das séries temporais, os melhores combinadores para cada base de dados foram variáveis. Mesmo o RBLC, com o pior desempenho geral, foi o melhor combinador na base de dados EEG2. Importante enfatizar que os melhores resultados foram obtidos sempre com as combinações que utilizaram os cinco melhores modelos de acordo com uma base de validação. Os resultados dos combinadores atestam a necessidade de um método de seleção dinâmica, pois não foi possível prever qual o combinador teria o melhor desempenho. Confiar na média, devido a seu extensivo uso, seria um erro no caso da previsão dessas séries temporais caóticas. Apesar do softmax ter sido o melhor combinador, há de se salientar que ele precisa de uma base de dados de validação. Gerar os pesos da soma ponderada, como o softmax o faz, acarreta um pequeno adicional de custo computacional ao método.

5.1.3 Seleção Dinâmica

A tabela 7 (página 78) mostra a média e o desvio-padrão do MSE para as abordagens do algoritmo de seleção dinâmica DSFC-A na base de dados de testes, após 30 execuções. A tabela 8 (página 79) faz o mesmo para o DSFC-B. Nessas tabelas, o primeiro termo ALL ou BEST refere-se à estratégia utilizada nos combinadores: o uso de todos os preditores ou apenas os melhores. O segundo termo ALL ou BEST indica se foram utilizados todos os combinadores ou somente os melhores. Assim, as oito abordagens que surgem como consequência dessa variação podem ser descritas da seguinte maneira:

- DSFC-A ALL-ALL: algoritmo proposto DSFC-A utilizando na seleção dinâmica todos os seis combinadores do modelo que possuem como entrada todos os dez preditores base.
- DSFC-A ALL-BEST: algoritmo proposto DSFC-A utilizando na seleção dinâmica

os três melhores combinadores do modelo que possuem como entrada todos os dez preditores base.

- DSFC-A BEST-ALL: algoritmo proposto DSFC-A utilizando na seleção dinâmica todos os seis combinadores do modelo que possuem como entrada os cinco melhores preditores base.
- DSFC-A BEST-BEST: algoritmo proposto DSFC-A utilizando na seleção dinâmica os três melhores combinadores do modelo que possuem como entrada os cinco melhores preditores base.
- DSFC-B ALL-ALL: algoritmo proposto DSFC-B utilizando na seleção dinâmica todos os seis combinadores do modelo que possuem como entrada todos os dez preditores base.
- DSFC-B ALL-BEST: algoritmo proposto DSFC-B utilizando na seleção dinâmica os três melhores combinadores do modelo que possuem como entrada todos os dez preditores base.
- DSFC-B BEST-ALL: algoritmo proposto DSFC-B utilizando na seleção dinâmica todos os seis combinadores do modelo que possuem como entrada os cinco melhores preditores base.
- DSFC-B BEST-BEST: algoritmo proposto DSFC-B utilizando na seleção dinâmica os três melhores combinadores do modelo que possuem como entrada os cinco melhores preditores base.

As abordagens testadas buscam responder algumas das questões levantadas na introdução deste trabalho: qual estratégia de seleção dinâmica deve ser utilizada? Na seleção dinâmica, devem ser levados em consideração somente os especialistas com melhor desempenho de validação ou todos os preditores gerados? Devem ser usados na seleção dinâmica todos os combinadores do modelo ou apenas aqueles com melhor desempenho de validação?

Técnicas estatísticas de comparação de conjuntos de medidas devem ser usadas para determinar se existem diferenças significativas entre os resultados de métodos diferentes. O teste de Wilcoxon baseado nos postos é um teste de hipóteses estatístico não-paramétrico utilizado para comparar duas amostras pareadas a partir da mesma população, sendo cada par independente e aleatoriamente selecionado. A eficácia do teste de Wilcoxon frente a outros testes na comparação de modelos de aprendizado de máquina é discutida em (DEMŠAR, 2006). Nos experimentos realizados, seu uso foi motivado por um ponto de vista estatístico: o teste de Wilcoxon é mais seguro, dado que não assume que as distribuições

Tabela 7 – MSE dos DSFC-A (30 execuções, teste)

Base de Dados	ALL-ALL	ALL-BEST	BEST-ALL	BEST-BEST
Mackey-Glass	9.46e-07	9.46e-07	**6.91e-07**	**6.91e-07**
	(2.26e-07)	(2.26e-07)	**(1.18e-07)**	**(1.19e-07)**
Lorenz	1.30e-09	1.30e-09	**6.67e-10**	6.71e-10
	(1.93e-09)	(1.93e-09)	**(7.38e-10)**	(7.39e-10)
Henon	4.36e-11	4.36e-11	**3.46e-11**	**3.46e-11**
	(1.22e-11)	(1.22e-11)	**(6.43e-12)**	**(6.43e-12)**
Rossler	8.26e-09	8.26e-09	6.01e-09	**6.00e-09**
	(7.17e-09)	(7.17e-09)	(4.06e-09)	**(4.07e-09)**
Periodic	6.52e-06	6.55e-06	**6.41e-06**	**6.41e-06**
	(2.81e-07)	(2.88e-07)	**(2.29e-07)**	**(2.29e-07)**
Quasi-Periodic	**3.34e-04**	3.35e-04	3.38e-04	3.39e-04
	(1.32e-05)	(1.34e-05)	(1.41e-05)	(1.40e-05)
Laser	3.85e+00	3.85e+00	**3.49e+00**	3.52e+00
	(7.39e-01)	(7.37e-01)	**(6.85e-01)**	(7.32e-01)
EEG1	4.39e+03	4.40e+03	4.40e+03	**4.18e+03**
	(1.54e+02)	(1.56e+02)	(2.11e+02)	**(2.22e+02)**
EEG2	6.21e-02	6.23e-02	5.64e-02	**5.32e-02**
	(4.38e-03)	(4.42e-03)	(5.29e-03)	**(4.30e-03)**
EEG3	8.86e-03	9.06e-03	7.97e-03	**7.85e-03**
	(3.87e-04)	(4.62e-04)	(4.67e-04)	**(4.02e-04)**

Tabela 8 – MSE dos DSFC-B (30 execuções, teste)

Base de Dados	ALL-ALL	ALL-BEST	BEST-ALL	BEST-BEST
Mackey-Glass	6.92e-03	9.82e-07	**6.64e-07**	6.79e-07
	(8.10e-06)	(2.43e-07)	**(1.09e-07)**	(1.21e-07)
Lorenz	2.80e+00	1.29e-09	**7.38e-10**	7.54e-10
	(3.37e-04)	(1.93e-09)	**(8.62e-10)**	(9.25e-10)
Henon	1.25e-01	4.39e-11	3.56e-11	**3.52e-11**
	(2.11e-05)	(1.18e-11)	(7.25e-12)	**(7.13e-12)**
Rossler	1.32e+00	8.30e-09	**5.85e-09**	6.02e-09
	(4.52e-02)	(7.19e-09)	**(3.78e-09)**	(3.93e-09)
Periodic	4.09e-03	6.55e-06	**6.41e-06**	**6.41e-06**
	(5.15e-06)	(2.88e-07)	**(2.27e-07)**	**(2.29e-07)**
Quasi-Periodic	6.58e-03	**3.35e-04**	3.40e-04	3.38e-04
	(3.58e-05)	**(1.37e-05)**	(1.34e-05)	(1.24e-05)
Laser	2.67e+02	3.89e+00	**3.37e+00**	3.44e+00
	(6.22e+00)	(7.74e-01)	**(5.84e-01)**	(7.33e-01)
EEG1	4.34e+03	4.30e+03	4.47e+03	**4.19e+03**
	(1.71e+02)	(1.57e+02)	(4.07e+02)	**(2.32e+02)**
EEG2	9.94e-02	6.21e-02	5.51e-02	**5.32-02**
	(4.09e-03)	(4.35e-03)	(4.96e-03)	**(4.29e-03)**
EEG3	1.79e-02	8.70e-03	7.82e-03	**7.81e-03**
	(5.13e-04)	(5.02e-04)	(5.00e-04)	**(4.30e-04)**

sejam normais. Assim, em caso das premissas não serem cumpridas, seu desempenho é mais confiável do que o teste t de *Student*.

As tabelas 9, 10, 11, 12, 13, 14, 15, 16, 17 e 18 mostram a comparação do MSE do método proposto com o melhor preditor individual e com o melhor combinador para cada base de dados testada. No caso do melhor combinador da base de dados, a alcunha ALL e BEST referem-se respectivamente ao uso de todos ou dos melhores preditores base. O MSE

foi selecionado para comparação porque é sensível à escala da série temporal, incorporando tanto a variância do preditor como também um possível enviesamento. Ademais, dado que as previsões das séries em grande parte dos modelos ficaram bem próximas dos valores esperados, o MSE é uma medida interessante de se analisar porque penaliza erros maiores.

A função de Wilcoxon testa a hipótese nula de que os dados vêm de uma distribuição cuja mediana é zero a 5% de nível de confiança, retornando a probabilidade *p-value*. Se o *p-value* é suficientemente baixo, então se pode assumir que a hipótese nula é falsa (a diferença entre as distribuições é significativa). Nas tabelas do teste de Wilcoxon, o sinal = indica que a hipótese nula não foi rejeitada (a diferença entre as médias dos erros não é estatisticamente relevante) e os modelos apresentam o mesmo desempenho. O sinal > indica que a hipótese nula foi rejeitada e que o método proposto tem desempenho superior em relação ao método utilizado para comparação. Por sua vez, o sinal < indica que a hipótese nula não foi rejeitada, e portanto o método proposto tem desempenho inferior em relação ao método utilizado para comparação. Os valores entre parêntesis são os *p-value* dos testes de hipótese.

Tabela 9 – Mackey-Glass - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	> (2.56e-02)	> (1.73e-06)
DSFC-A ALL-BEST	> (2.56e-02)	> (1.73e-06)
DSFC-A BEST-ALL	> (1.73e-06)	> (8.22e-03)
DSFC-A BEST-BEST	> (1.73e-06)	> (8.73e-03)
DSFC-B ALL-ALL	> (1.73e-06)	> (1.73e-06)
DSFC-B ALL-BEST	= (2.92e-01)	< (1.73e-06)
DSFC-B BEST-ALL	> (1.73e-06)	> (4.45e-05)
DSFC-B BEST-BEST	> (1.73e-06)	> (2.05e-04)

Tabela 10 – Lorenz - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	= (3.99e-01)	= (5.17e-01)
DSFC-A ALL-BEST	= (3.99e-01)	= (5.17e-01)
DSFC-A BEST-ALL	> (2.96e-03)	> (8.73e-03)
DSFC-A BEST-BEST	> (2.96e-03)	> (8.73e-03)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (3.74e-01)	= (5.17e-01)
DSFC-B BEST-ALL	> (8.73e-03)	> (1.48e-02)
DSFC-B BEST-BEST	> (8.73e-03)	> (1.40e-02)

A tabela 19 mostra o resumo das diferenças de desempenho entre as abordagens testadas e os melhores preditores e combinadores em cada uma das bases de dados. O símbolo > indica o número de vezes que o método foi estatisticamente superior ao melhor preditor ou combinador e o símbolo < indica o número de vezes em que o método foi estatisticamente inferior. O símbolo = aponta o número de vezes em que não houve diferença significativa.

Tabela 11 – Henon - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Softmax BEST
DSFC-A ALL-ALL	> (4.80e-06)	= (5.00e-01)
DSFC-A ALL-BEST	> (4.80e-06)	= (5.00e-01)
DSFC-A BEST-ALL	> (1.73e-06)	> (5.22e-06)
DSFC-A BEST-BEST	> (1.73e-06)	> (6.98e-06)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	> (3.88e-06)	= (7.21e-01)
DSFC-B BEST-ALL	> (1.73e-06)	> (2.05e-04)
DSFC-B BEST-BEST	> (1.92e-06)	> (1.15e-04)

Tabela 12 – Rossler - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Softmax BEST
DSFC-A ALL-ALL	= (1.17e-01)	= (7.00e-01)
DSFC-A ALL-BEST	= (1.17e-01)	= (7.00e-01)
DSFC-A BEST-ALL	> (1.29e-03)	> (1.17e-02)
DSFC-A BEST-BEST	> (1.71e-03)	> (1.57e-02)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (1.17e-01)	= (7.70e-01)
DSFC-B BEST-ALL	> (1.48e-03)	> (9.27e-03)
DSFC-B BEST-BEST	> (1.96e-03)	> (2.56e-02)

Tabela 13 – Periodic - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Média BEST
DSFC-A ALL-ALL	= (8.61e-01)	= (4.78e-01)
DSFC-A ALL-BEST	= (7.50e-01)	< (4.07e-02)
DSFC-A BEST-ALL	> (4.95e-02)	> (6.56e-02)
DSFC-A BEST-BEST	> (5.45e-02)	> (6.56e-02)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (7.50e-01)	< (4.07e-02)
DSFC-B BEST-ALL	> (6.27e-02)	> (7.52e-02)
DSFC-B BEST-BEST	> (5.45e-02)	> (6.56e-02)

Tabela 14 – Quasi-Periodic - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-1	Melhor Combinador - Mediana/Softmax BEST
DSFC-A ALL-ALL	> (5.75e-06)	> (4.20e-04)
DSFC-A ALL-BEST	> (5.75e-06)	> (1.66e-02)
DSFC-A BEST-ALL	> (1.02e-05)	= (1.85e-01)
DSFC-A BEST-BEST	> (1.36e-05)	= (9.10e-01)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	> (5.22e-06)	> (7.27e-03)
DSFC-B BEST-ALL	> (1.97e-05)	= (9.92e-01)
DSFC-B BEST-BEST	> (6.98e-06)	= (2.99e-01)

Em relação à comparação com os melhores preditores base, sete das oito abordagens testadas foram competitivas. DSFC-A ALL-BEST, DSFC-A BEST-ALL, DSFC-B ALL-BEST e DSFC-B BEST-ALL foram estatisticamente inferiores ao melhor preditor em apenas uma base de dados. O melhor desempenho geral nesse aspecto ocorreu com as abordagens que utilizaram os combinadores com os melhores preditores (BEST-ALL e BEST-BEST). Nesses cenários, as abordagens foram superiores aos melhores preditores em nove das dez séries temporais. Essa situação era esperada, visto que o melhor desempenho

Tabela 15 – Laser - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-2	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	> (3.00e-02)	= (6.29e-01)
DSFC-A ALL-BEST	> (3.00e-02)	= (6.44e-01)
DSFC-A BEST-ALL	> (3.06e-04)	> (2.96e-03)
DSFC-A BEST-BEST	> (5.71e-04)	> (4.68e-03)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (7.52e-02)	= (3.93e-01)
DSFC-B BEST-ALL	> (4.86e-05)	> (8.92e-05)
DSFC-B BEST-BEST	> (1.15e-04)	> (9.71e-05)

Tabela 16 – EEG1 - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - ARIMA	Melhor Combinador - Média Winsorizada ALL
DSFC-A ALL-ALL	< (2.37e-05)	< (6.34e-06)
DSFC-A ALL-BEST	< (2.16e-05)	< (5.75e-06)
DSFC-A BEST-ALL	< (2.83e-04)	< (6.89e-05)
DSFC-A BEST-BEST	= (4.20e-01)	= (3.06e-01)
DSFC-B ALL-ALL	< (8.94e-04)	< (1.13e-05)
DSFC-B ALL-BEST	< (2.07e-02)	< (1.48e-04)
DSFC-B BEST-ALL	< (6.84e-03)	< (1.83e-03)
DSFC-B BEST-BEST	= (1.29e-01)	= (2.63e-01)

Tabela 17 – EEG2 - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-2	Melhor Combinador - RBLC BEST
DSFC-A ALL-ALL	= (2.89e-01)	< (1.73e-06)
DSFC-A ALL-BEST	= (2.80e-01)	< (1.73e-06)
DSFC-A BEST-ALL	> (1.04e-02)	< (1.36e-04)
DSFC-A BEST-BEST	> (1.74e-04)	> (3.18e-01)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (3.18e-01)	< (1.73e-06)
DSFC-B BEST-ALL	> (1.04e-03)	< (2.18e-02)
DSFC-B BEST-BEST	> (1.74e-04)	> (4.41e-02)

Tabela 18 – EEG3 - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-2	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	= (8.77e-01)	< (1.73e-06)
DSFC-A ALL-BEST	= (3.09e-01)	< (1.73e-06)
DSFC-A BEST-ALL	> (3.41e-05)	= (7.66e-01)
DSFC-A BEST-BEST	> (8.47e-06)	> (3.00e-02)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (4.41e-01)	< (1.73e-06)
DSFC-B BEST-ALL	> (1.64e-05)	> (3.33e-02)
DSFC-B BEST-BEST	> (8.47e-06)	> (1.48e-03)

dos combinadores dos melhores preditores já havia sido verificado. Quando se leva em consideração a utilização dos combinadores dos melhores preditores, não houve diferenças significativas entre utilizar o algoritmo DSFC-A ou o DSFC-B. Nessas abordagens, também não houve diferença entre utilizar na seleção dinâmica todos ou os melhores combinadores.

Foi notório o desempenho inferior da variação DSFC-B ALL-ALL, tendo sido estatisticamente pior do que o melhor preditor em nove das dez base de dados. A abordagem

Tabela 19 – Resultados no teste de Wilcoxon

Modelo	Preditores			Combinadores		
	>	=	<	>	=	<
DSFC-A ALL-ALL	4	5	1	2	5	3
DSFC-A ALL-BEST	4	5	1	2	4	4
DSFC-A BEST-ALL	9	0	1	6	1	3
DSFC-A BEST-BEST	9	1	0	7	3	0
DSFC-B ALL-ALL	1	0	9	1	2	7
DSFC-B ALL-BEST	2	7	1	1	4	5
DSFC-B BEST-ALL	9	0	1	7	1	2
DSFC-B BEST-BEST	9	1	0	7	3	0

utiliza os combinadores com as maiores taxas de erro, aqueles resultantes a partir do uso de todos os preditores base. Porém, mesmo com essa característica, as variações DSFC-A ALL-ALL e DSFC-A ALL-BEST chegaram a superar estatisticamente os melhores preditores em quatro das dez séries temporais. O método DSFC-B utiliza o comportamento da saída do padrão de teste nos combinadores com o comportamento de uma sub-região do espaço de características. É possível argumentar que o algoritmo não seja capaz de encontrar o combinador mais promissor somente pelo seu comportamento de saída quando há um número maior de opções não muito efetivas, caso da abordagem DSFC-B ALL-ALL. O desempenho melhora um pouco quando a variedade de opções diminui, caso da abordagem DSFC-B ALL-BEST.

No que diz respeito à comparação com os combinadores de menor taxa de erro em cada base de dados, as melhores abordagens decorreram da utilização dos melhores preditores: DSFC-A BEST-ALL, DSFC-A BEST-BEST, DSFC-B BEST-ALL e DSFC-B BEST-BEST. Dentre estas, o destaque fica para DSFC-A BEST-BEST e DSFC-B BEST-BEST. Ambas as abordagens foram estatisticamente superiores ao melhor combinador em sete das dez séries temporais testadas, tendo obtido desempenho estatisticamente semelhante em três delas. Pode-se então alegar que, considerando-se o problema da seleção dinâmica de combinadores de previsão, a melhor estratégia seja utilizar o método de seleção a partir de um número reduzido de combinadores que produzem suas saídas a partir de uma quantidade também restrita de preditores base. No caso do número de preditores individuais, já havia indicações na literatura atentando para esse fato, que pôde ser comprovado experimentalmente neste trabalho. Já em relação ao número reduzido de combinadores, uma argumentação possível para esse comportamento é a essência do problema tratado. Para que as previsões sejam apropriadas e que o tempo de processamento do método seja praticável, a dimensionalidade dos dados (determinado pelo atraso de tempo) não deve ultrapassar um limite razoável. No método experimental realizado, o atraso de tempo utilizado pode ter prejudicado o algoritmo quando este possuía as seis opções de combinadores para realizar a seleção dinâmica. O desempenho melhorou quando esse número foi reduzido pela metade.

Os resultados comprovam a hipótese inicial de que, a partir do momento que não se sabe qual combinador produzirá as melhores predições, faz-se necessário um método para dinamicamente selecionar a melhor combinação a partir de cada padrão de teste. Por exemplo, a melhor combinação para a base Mackey-Glass foi a mediana, enquanto que para Henon o softmax foi o melhor combinador. As melhores abordagens de seleção dinâmica, DSFC-A BEST-BEST e DSFC-B BEST-BEST, obtiveram resultados estatisticamente superiores independente de qual tenha sido o melhor combinador em sete das dez séries, tendo alcançando um desempenho mínimo em outras três e não tendo sido superadas em nenhuma série temporal.

As figuras 31, 32, 33, 34, 35, 36, 37, 38, 39 e 40 (páginas 84 a 87) mostram as previsões de uma rodada de execução da melhor abordagem dos dois algoritmos propostos de seleção dinâmica, para cada série temporal. Na previsão, a linha azul com pontos é a saída desejada e a linha vermelha a saída produzida. Em cada figura, o eixo vertical do primeiro gráfico é dado pelos valores desejados e predições. O eixo horizontal é dado pelos pontos do conjunto de teste. É importante notar que as previsões das melhores abordagens dos métodos DSFC-A e DSFC-B ficaram muito próximas uma das outras, por vezes sendo difícil identificar as diferenças nas ilustrações.

Em todas as bases de dados, as predições ficaram muito próximas da saída desejada. Presumivelmente, as maiores discrepâncias ocorreram nas séries temporais naturais de exames de eletroencefalograma. Pelo fato de serem naturais, os modelos de predição têm mais dificuldade para absorver o comportamento dos dados, já que são mais suscetíveis a ruídos e discrepâncias. As predições poderiam ser um pouco melhoradas com o tratamento prévio das séries temporais, como por exemplo uma remoção de sazonalidade. Mas este trabalho optou por construir os modelos com o máximo de automatização, sem a necessidade de o experimentador conhecer as características dos dados.

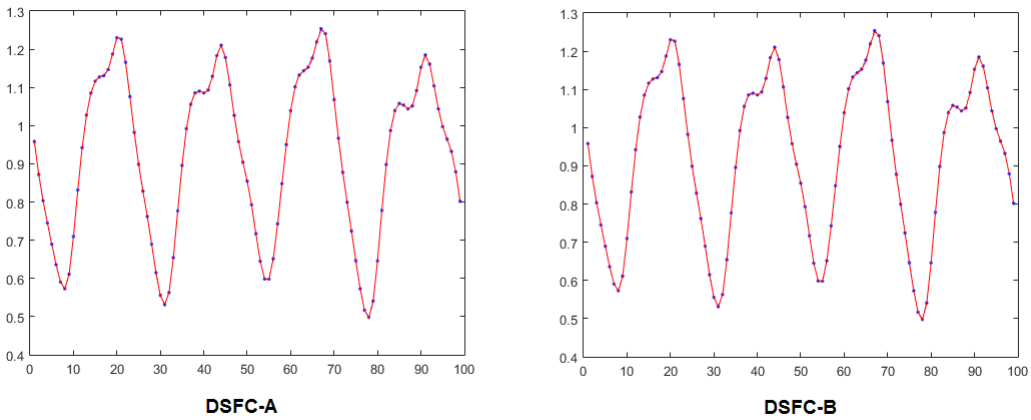


Figura 31 – Mackey-Glass - Previsão das melhores abordagens para DSFC-A e DSFC-B

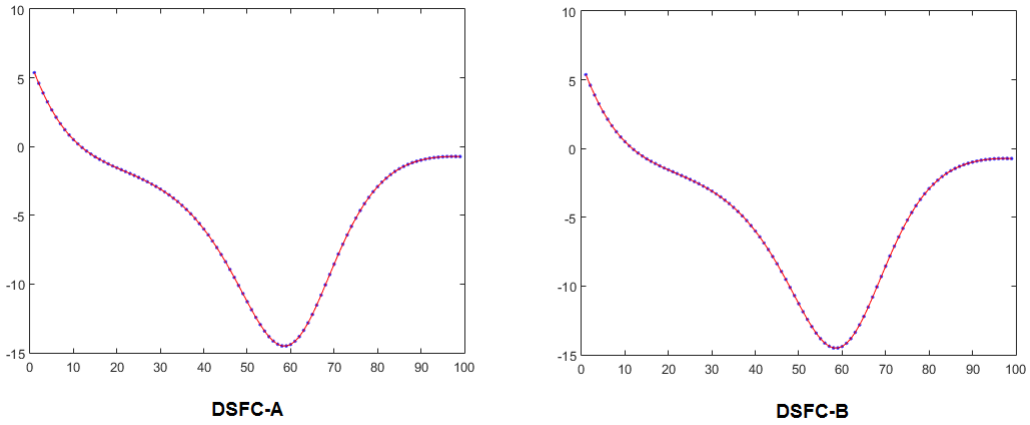


Figura 32 – Lorenz - Previsão das melhores abordagens para DSFC-A e DSFC-B

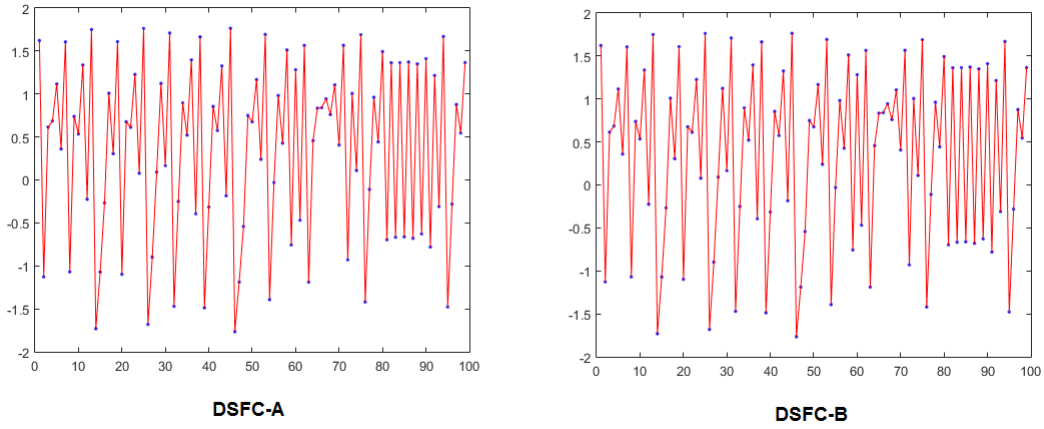


Figura 33 – Henon - Previsão das melhores abordagens para DSFC-A e DSFC-B

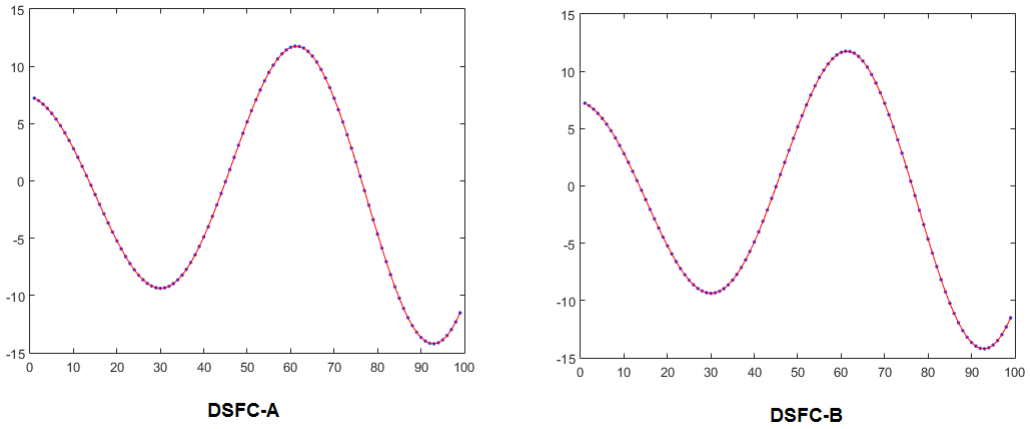


Figura 34 – Rossler - Previsão das melhores abordagens para DSFC-A e DSFC-B

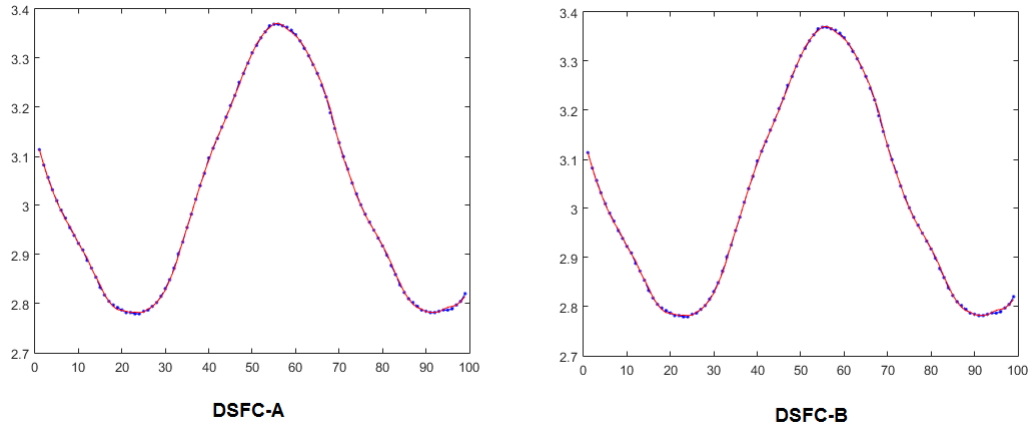


Figura 35 – Periodic - Previsão das melhores abordagens para DSFC-A e DSFC-B

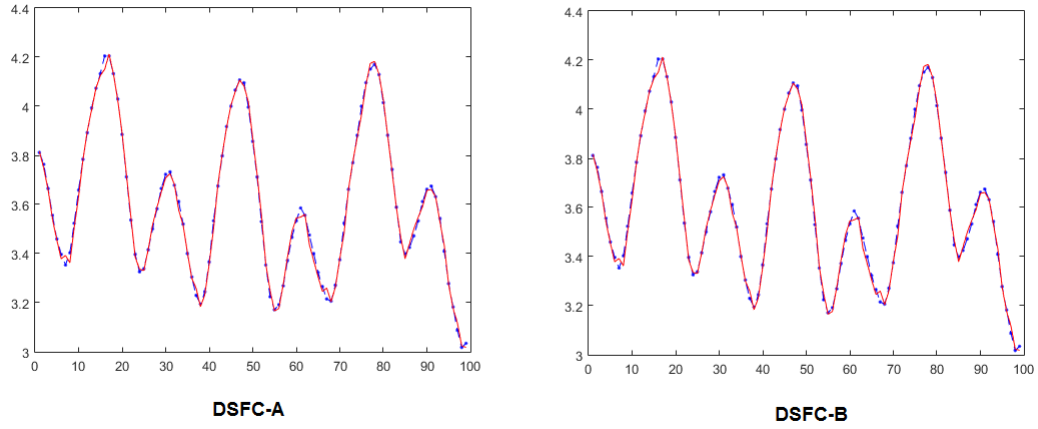


Figura 36 – Quasi-Periodic - Previsão das melhores abordagens para DSFC-A e DSFC-B

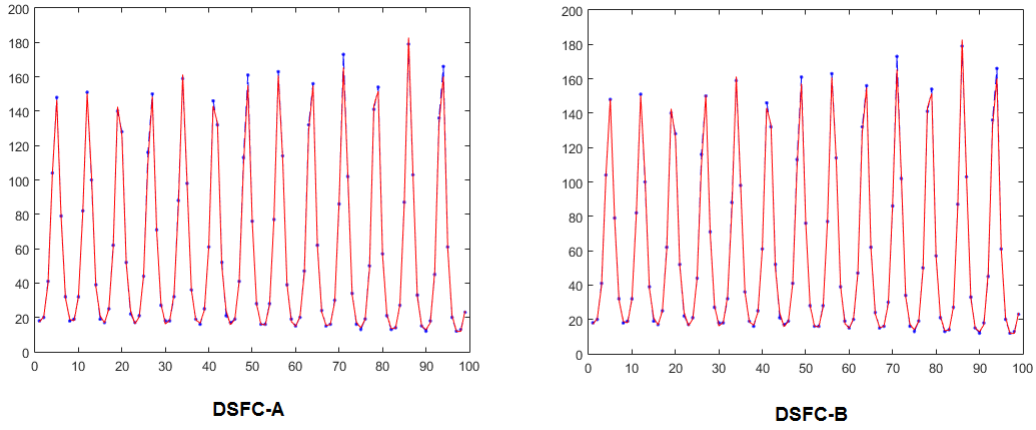


Figura 37 – Laser - Previsão das melhores abordagens para DSFC-A e DSFC-B

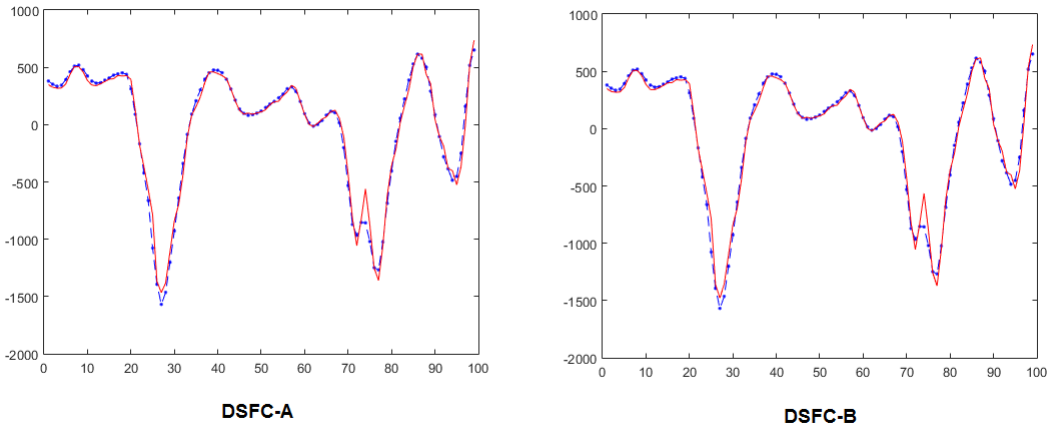


Figura 38 – EEG1 - Previsão das melhores abordagens para DSFC-A e DSFC-B

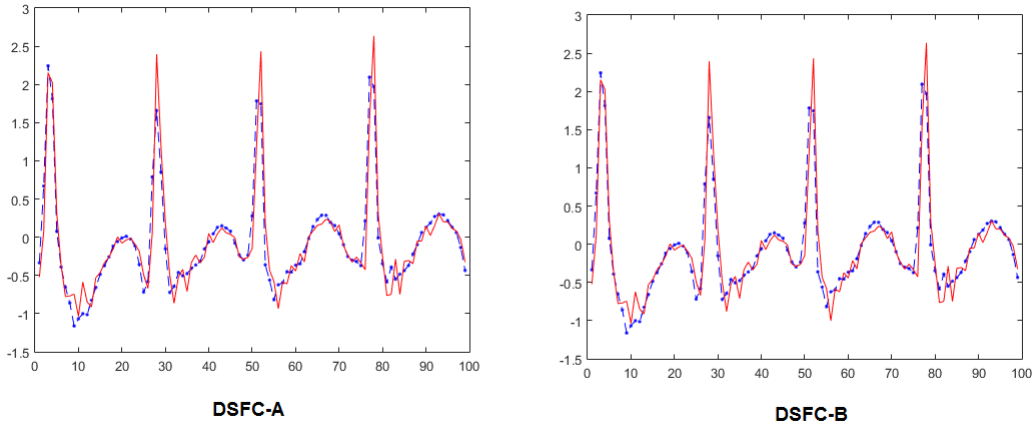


Figura 39 – EEG2 - Previsão das melhores abordagens para DSFC-A e DSFC-B

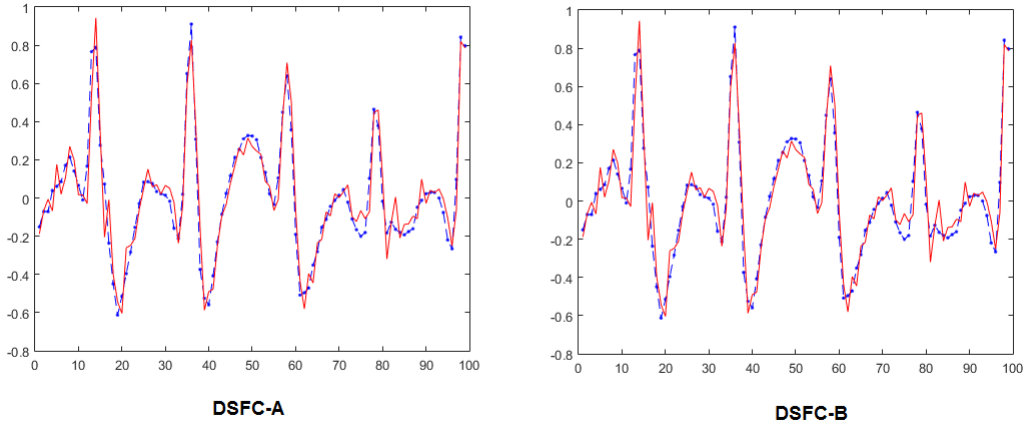


Figura 40 – EEG3 - Previsão das melhores abordagens para DSFC-A e DSFC-B

5.1.4 Tempo de Processamento

As tabelas 20 e 21 mostram a média e o desvio-padrão do tempo de procesamento do modelo em segundos para cada base de dados, após 30 execuções. A primeira linha da tabela aponta o tempo decorrido das fases Geração I e Geração II do arcabouço proposto, compreendendo a construção dos preditores base com o PSO e a definição dos combinadores. As demais linhas indicam o tempo gasto com cada uma das abordagens propostas. Os experimentos foram realizados em uma máquina com sistema operacional Windows 7 64 bits, processador Intel Core i7-4770 e memória RAM de 16GB.

Tabela 20 – Tempo de processamento, parte I

Fase	Mackey-Glass	Lorenz	Henon	Rossler	Periodic
Geração	3294.2 (195.3)	6649.5 (321.6)	7737.0 (601.7)	6045.5 (491.5)	1261.4 (72.8)
DSFC-A ALL-ALL	205.0 (0.6)	205.2 (0.6)	206.7 (1.5)	205.1 (0.3)	212.3 (1.2)
DSFC-A ALL-BEST	204.8 (1.4)	204.6 (1.5)	206.3 (1.5)	204.4 (0.5)	211.6 (1.2)
DSFC-A BEST-ALL	205.3 (1.5)	205.3 (1.5)	206.7 (1.5)	205.1 (0.5)	212.4 (1.1)
DSFC-A BEST-BEST	204.8 (1.6)	204.6 (1.7)	206.1 (1.4)	204.7 (0.7)	211.8 (1.1)
DSFC-B ALL-ALL	335.5 (0.8)	401.5 (0.5)	323.7 (2.4)	375.7 (1.0)	422.2 (2.0)
DSFC-B ALL-BEST	367.8 (0.6)	403.8 (0.6)	338.7 (2.4)	402.5 (2.4)	422.5 (0.6)
DSFC-B BEST-ALL	334.9 (0.7)	401.1 (0.4)	297.3 (2.1)	366.6 (2.2)	422.5 (1.0)
DSFC-B BEST-BEST	366.2 (0.4)	403.6 (0.5)	317.9 (2.3)	400.3 (1.5)	422.9 (2.2)
Total	5518.6 (195.0)	9079.2 (321.7)	9840.2 (601.7)	8409.9 (490.7)	3799.5 (72.6)

Foi possível verificar que o tempo individual de cada uma das abordagens foi sempre menor do que a a fase de geração, fazendo com que a seleção dinâmica possua uma complexidade computacional aceitável. Na série temporal Henon, por exemplo, a abordagem com maior tempo médio (DSFC-B ALL-BEST) chegou a ter apenas cerca de 4% do tempo de geração. A menor diferença ocorreu na base EEG3, onde a abordagem DSFC-B BEST-BEST levou em média aproximadamente 51% do tempo de geração.

Entre as variações do mesmo algoritmo de seleção dinâmica, as disparidades não foram significativas. Tal cenário ocorreu porque a diferença entre as abordagens com o mesmo algoritmo de seleção dinâmica dizem somente respeito ao número de combinadores utilizados na seleção, o que não é suficiente para impactar a complexidade computacional do modelo. Por outro lado, como previsto, foi constatado que as abordagens do algoritmo DSFC-A são mais eficientes do que o DSFC-B. Afinal, o DSFC-B possui um laço de repetição a mais, responsável por calcular o comportamento da saída dos combinadores

Tabela 21 – Tempo de processamento, parte II

Fase	Quasi-Periodic	Laser	EEG1	EEG2	EEG3
Geração	932.1	889.4	721.8	713.1	688.9
	(33.7)	(32.5)	(35.2)	(34.6)	(26.7)
DSFC-A ALL-ALL	211.9	210.8	206.3	206.1	203.9
	(0.4)	(0.5)	(1.3)	(1.4)	(1.4)
DSFC-A ALL-BEST	211.1	210.1	205.5	205.5	203.1
	(0.6)	(0.6)	(1.5)	(1.3)	(0.3)
DSFC-A BEST-ALL	211.8	210.6	206.1	206.1	203.7
	(1.0)	(0.6)	(0.4)	(1.3)	(0.5)
DSFC-A BEST-BEST	211.6	210.3	205.4	205.8	203.2
	(1.1)	(0.6)	(0.5)	(1.9)	(0.4)
DSFC-B ALL-ALL	336.6	383.5	341.6	398.8	324.4
	(0.7)	(0.8)	(2.2)	(2.7)	(2.5)
DSFC-B ALL-BEST	367.2	401.5	362.6	404.3	354.5
	(0.9)	(0.7)	(2.4)	(2.7)	(1.5)
DSFC-B BEST-ALL	332.1	379.6	338.7	397.3	327.6
	(0.8)	(1.0)	(2.1)	(2.3)	(2.3)
DSFC-B BEST-BEST	364.1	397.3	360.6	402.2	355.7
	(1.0)	(0.9)	(2.5)	(2.6)	(2.3)
Total	3178.6	3293.2	2948.6	3139.2	2865.0
	(35.0)	(34.3)	(36.1)	(40.7)	(28.0)

para cada padrão de teste. Em geral, as abordagens DSFC-B tiveram quase o dobro de tempo das abordagens DSFC-A.

Foi discutido na seção anterior que as abordagens DSFC-A BEST-BEST e DSFC-B BEST-BEST dividiram o melhor desempenho do modelo quando se levou em consideração as menores taxas de erro. O tempo de processamento, então, pode ser utilizado para constatar que a abordagem DSFC-A BEST-BEST é a melhor escolha para seleção dinâmica de combinadores de previsão de curto alcance, de acordo com as bases de dados testadas. Nas dez séries temporais utilizadas para validar o arcabouço proposto, DSFC-A BEST-BEST obteve o melhor desempenho: foi estatisticamente superior aos melhores preditores base e combinadores em relação à taxa de erro na maioria dos cenários testados e foi mais eficiente do que as variações do algoritmo DSFC-B.

É importante notar que o tamanho da base de dados influencia no tempo de processamento. Sendo assim, é natural que a fase de geração leve mais tempo que a fase de seleção dinâmica, por utilizar uma base de dados maior, de treinamento. Entretanto, essa comparação tem o intuito de verificar o quão mais complexo seria o arcabouço com a adição dos algoritmos de seleção dinâmica. Foi observado que esse tempo adicional é suficientemente pequeno para justificar sua utilização.

5.2 **Previsão de Longo Alcance**

Os resultados para a previsão de longo alcance são apresentados a seguir. Toda a configuração das simulações numéricas foi semelhante à previsão de curto alcance, com

exceção do horizonte de previsão, dado por 10.

5.2.1 Preditores Base

A tabela 22 (página 90) mostra a média e o desvio-padrão do MSE dos preditores de aprendizado de máquina na base de testes, após 30 execuções do método. A tabela 23 (página 90) faz o mesmo para os preditores estatísticos.

Tabela 22 – MSE dos preditores de aprendizado de máquina (30 execuções, teste)

Base de Dados	FANN-1	FANN-2	DBN	SDAE	SVR
Mackey-Glass	1.74e-06	1.24e-06	**1.04e-06**	1.37e-06	2.08e-04
	(2.87e-07)	(2.37e-07)	**(2.06e-07)**	(3.57e-07)	(1.03e-05)
Lorenz	**4.85e-06**	3.56e-05	1.18e-05	2.74e-05	9.07e-02
	(4.00e-06)	(1.45e-05)	(6.47e-06)	(2.26e-05)	(7.78e-03)
Henon	**1.04e+00**	1.05e+00	1.05e+00	1.06e+00	1.05e+00
	(3.86e-02)	(4.33e-02)	(3.01e-02)	(4.63e-02)	(2.22e-02)
Rossler	5.27e-04	**7.73e-05**	1.54e-04	2.30e-04	1.13e+00
	(1.03e-03)	**(6.01e-05)**	(1.53e-04)	(5.84e-04)	(4.92e-01)
Periodic	8.82e-05	8.64e-05	8.79e-05	**8.63e-05**	3.99e-04
	(5.84e-06)	(5.28e-06)	(5.92e-06)	**(6.06e-06)**	(1.21e-05)
Quasi-Periodic	**3.69e-02**	4.00e-02	3.88e-02	4.39e-02	7.50e-02
	(3.77e-03)	(6.03e-03)	(7.45e-03)	(1.11e-02)	(2.60e-03)
Laser	4.74e+01	2.78e+01	**2.70e+01**	3.13e+01	2.79e+02
	(2.78e+01)	(2.22e+01)	**(1.01e+01)**	(1.30e+01)	(1.85e+02)
EEG1	2.71e+05	2.76e+05	2.74e+05	2.80e+05	**2.61e+05**
	(9.89e+03)	(8.20e+03)	(1.22e+04)	(1.56e+04)	**(4.93e+03)**
EEG2	3.88e-01	3.89e-01	3.88e-01	**3.85e-01**	4.43e-01
	(2.03e-02)	(2.16e-02)	(1.28e-02)	**(1.63e-02)**	(1.90e-02)
EEG3	6.45e-02	**6.41e-02**	6.60e-02	6.74e-02	8.40e-02
	(3.35e-03)	**(3.89e-03)**	(5.37e-03)	(6.05e-03)	(2.68e-03)

Assim como ocorreu na previsão de curto alcance, os preditores advindos da aprendizado de máquina tiveram um melhor resultado geral. Em apenas uma base de dados, Henon, um preditor estatístico alcançou uma menor taxa de erro média, ainda assim por margens muito pequenas. Interessante notar que na previsão de longo alcance a variabilidade dos melhores preditores base foi maior, apontando ainda mais a necessidade de alguma técnica de combinação e posterior algoritmo de seleção dinâmica.

A previsão de longo alcance é uma tarefa mais difícil do que a previsão de curto alcance. À medida que o horizonte de previsão aumenta, normalmente os modelos têm mais dificuldade em captar as correlações entre os pontos da série temporal, sendo mais suscetíveis a maiores taxas de erro. Tal conjuntura ocorreu nas simulações numéricas realizadas. A previsão de longo alcance alcançou um menor desempenho em todas as bases de dados testadas, em maior ou menor grau. A maior disparidade ocorreu na base de dados Henon. O melhor preditor na previsão de curto alcance obteve um MSE médio na ordem de e-11, enquanto que no longo alcance esse erro aumentou para e+00. Observando a

Tabela 23 – MSE dos preditores estatísticos (30 execuções, teste)

Base de Dados	AR	MA	ARMA	ARIMA	GARCH
Mackey-Glass	8.18e-02	**4.87e-02**	4.82e-01	1.23e-01	1.92e-01
	(0.00e+00)	**(2.12e-17)**	(1.69e-16)	(5.65e-17)	(5.65e-17)
Lorenz	1.24e-01	3.60e+01	1.16e-01	**1.12e-01**	4.06e+01
	(4.23e-17)	(1.45e-14)	(2.82e-17)	**(4.23e-17)**	(1.45e-14)
Henon	1.23e+00	**1.03e+00**	2.33e+00	1.18e+00	1.34e+00
	(9.03e-16)	**(0.00e+00)**	(1.81e-15)	(6.78e-16)	(2.26e-16)
Rossler	3.58e+00	6.83e+01	**3.92e-02**	9.59e+00	6.36e+01
	(1.81e-15)	(2.89e-14)	**(7.06e-18)**	(7.23e-15)	(2.89e-14)
Periodic	4.75e-03	4.18e-02	**8.90e-03**	9.45e-03	3.26e-02
	(0.00e+00)	(2.12e-17)	**(7.06e-18)**	(0.00e+00)	(2.12e-17)
Quasi-Periodic	1.33e-01	**9.14e-02**	1.38e-01	1.43e-01	2.82e-01
	(5.65e-17)	**(1.41e-17)**	(1.13e-16)	(2.82e-17)	(5.65e-17)
Laser	4.20e+03	**2.57e+03**	5.63e+03	3.89e+03	5.25e+03
	(2.78e-12)	**(4.63e-13)**	(9.25e-13)	(9.25e-13)	(0.00e+00)
EEG1	6.32e+05	**2.95e+05**	6.38e+05	6.16e+05	6.76e+05
	(1.18e-10)	**(5.92e-11)**	(1.18e-10)	(2.37e-10)	(3.55e-10)
EEG2	8.13e-01	**4.41e-01**	3.81e+00	2.33e+00	8.14e-01
	(4.52e-16)	**(1.69e-16)**	(4.52e-16)	(1.36e-15)	(2.26e-16)
EEG3	1.74e-01	**9.71e-02**	1.92e-01	1.80e+00	2.14e-01
	(0.00e+00)	**(2.82e-17)**	(8.47e-17)	(4.52e-16)	(8.47e-17)

figura 13, é possível supor que a causa para a queda de desempenho tenha sido a natureza da série, com muito mais ciclos de sazonalidade.

As figuras 41, 42, 43, 44, 45, 46, 47, 48, 49 e 50 mostram a curva de aptidão em uma rodada do PSO para cada um dos modelos de aprendizado de máquina. Assim como ocorreu na previsão de curto alcance, a busca pelos melhores parâmetros do modelo seguiu o comportamento esperado da otimização por enxame de partículas, com as fases de exploração e explotação.

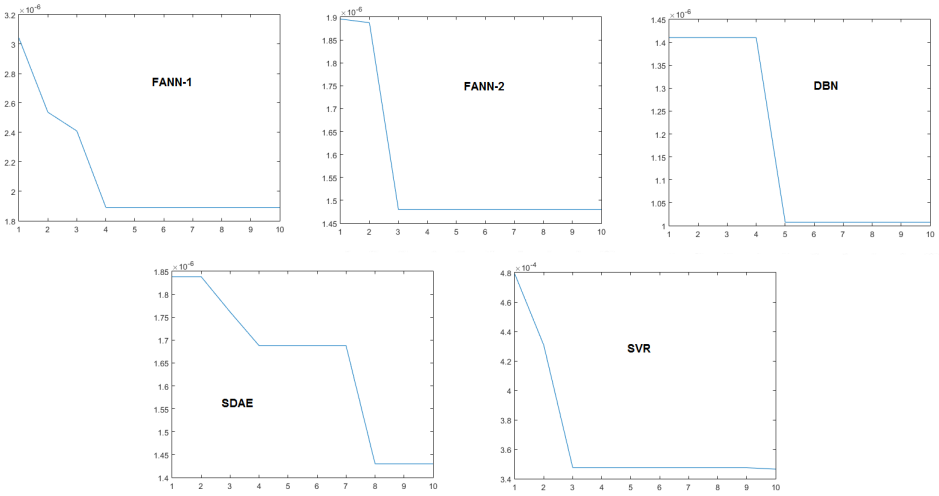


Figura 41 – Mackey-Glass - Curva de aptidão do PSO

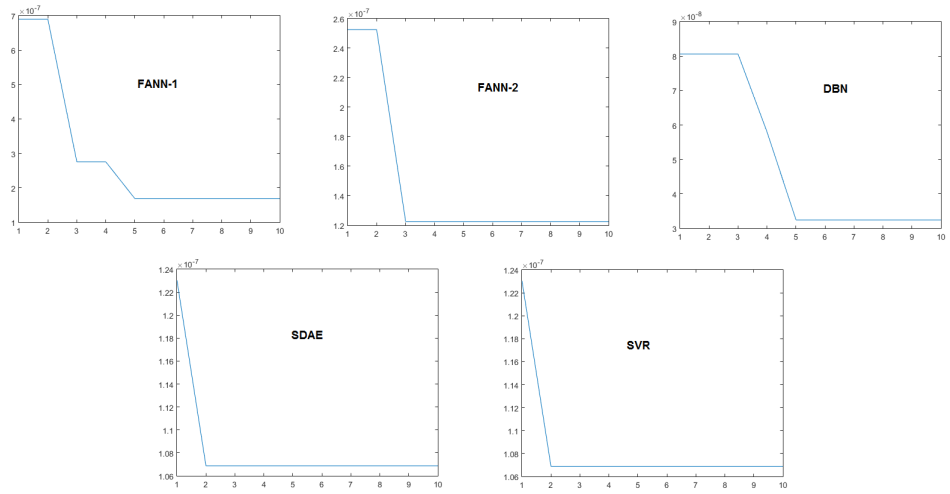


Figura 42 – Lorenz - Curva de aptidão do PSO

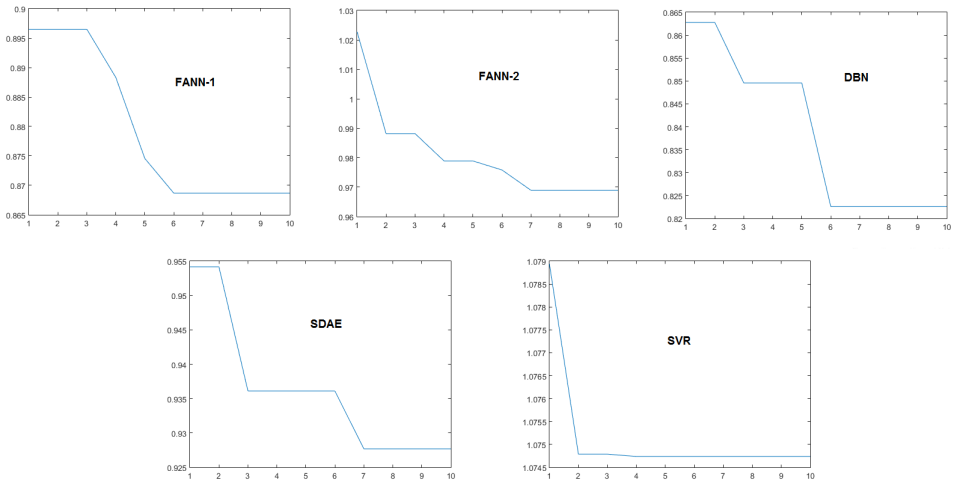


Figura 43 – Henon - Curva de aptidão do PSO

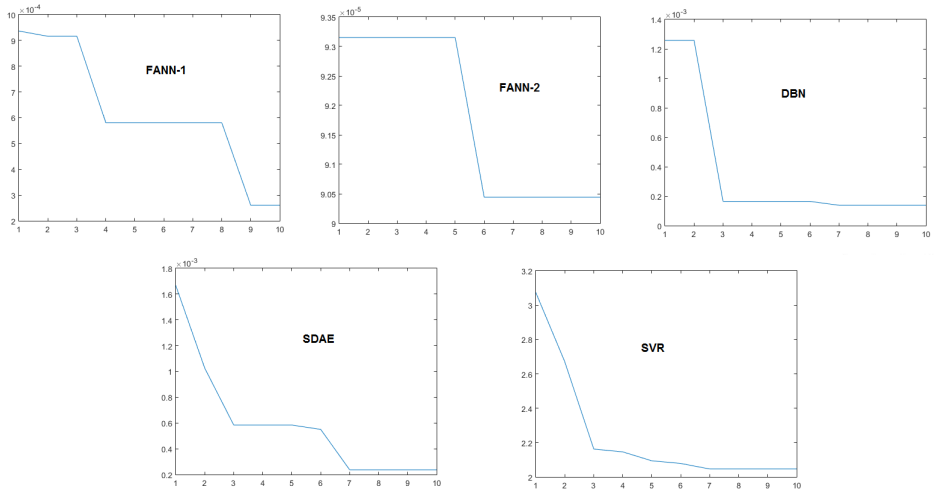


Figura 44 – Rossler - Curva de aptidão do PSO

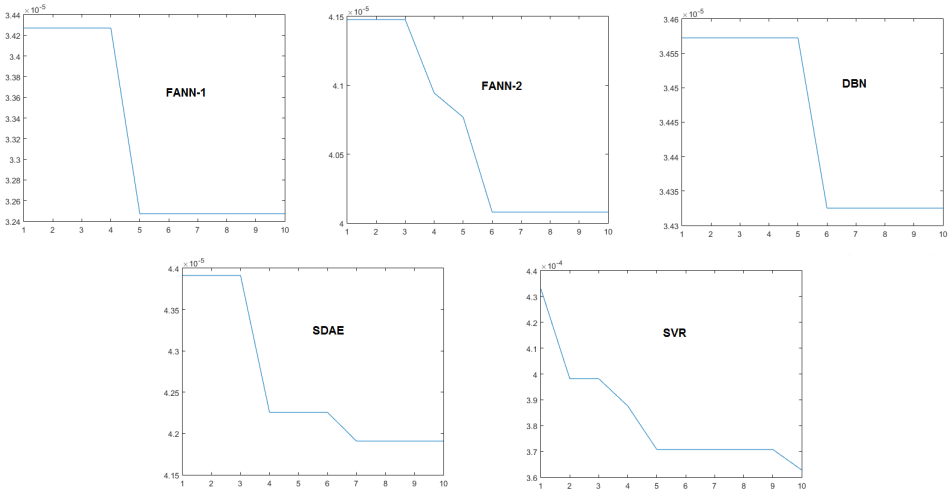


Figura 45 – Periodic - Curva de aptidão do PSO

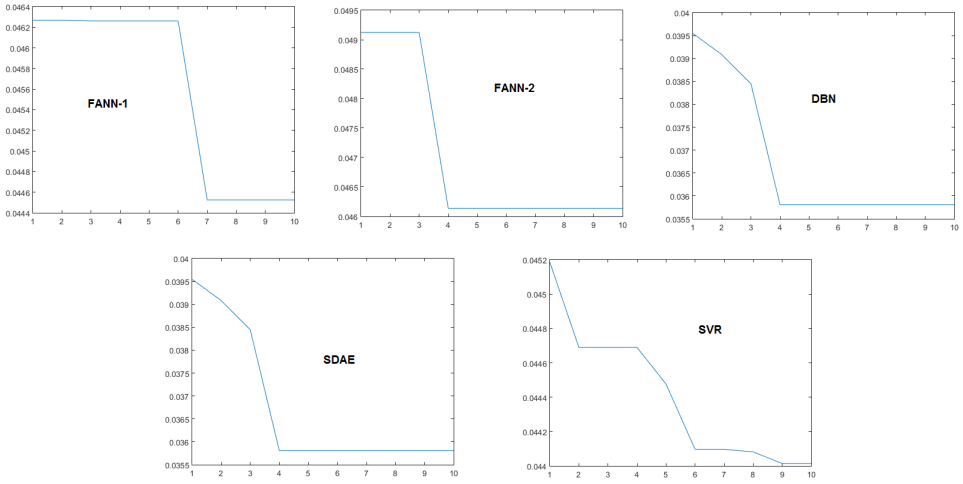


Figura 46 – Quasi-Periodic - Curva de aptidão do PSO

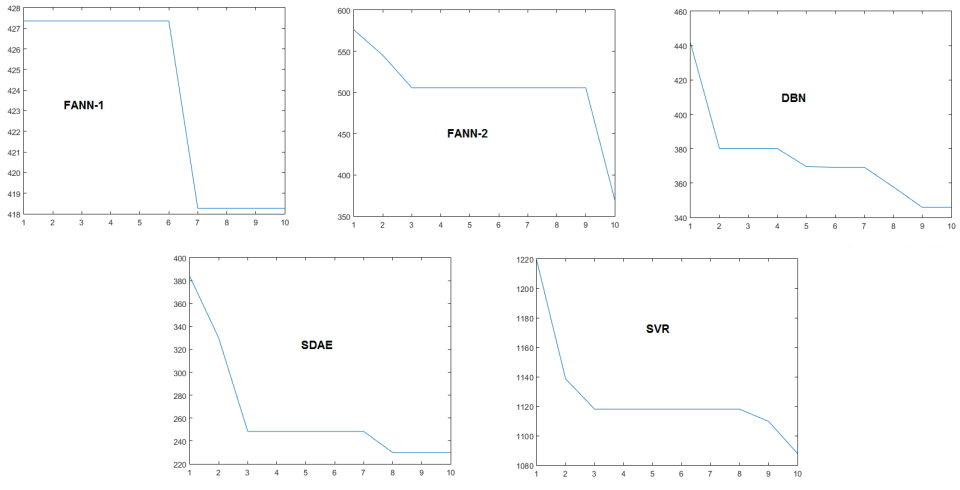


Figura 47 – Laser - Curva de aptidão do PSO

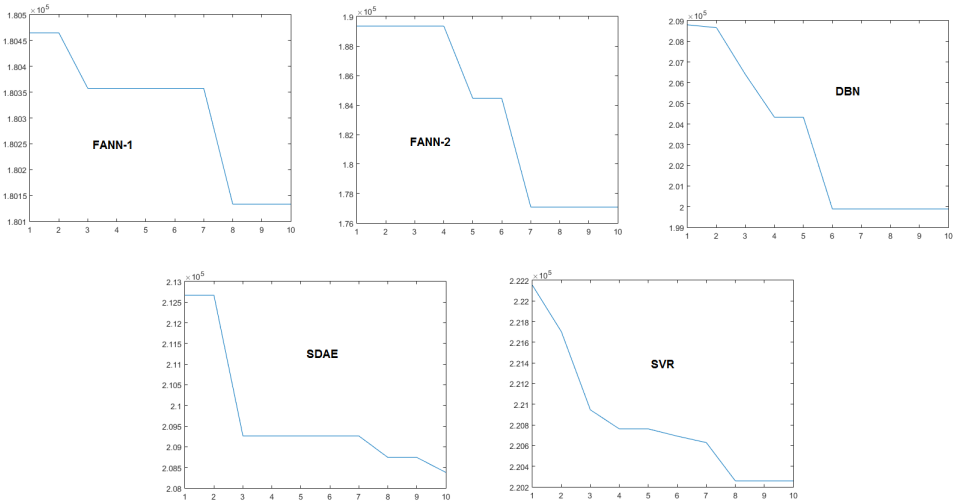


Figura 48 – EEG1 - Curva de aptidão do PSO

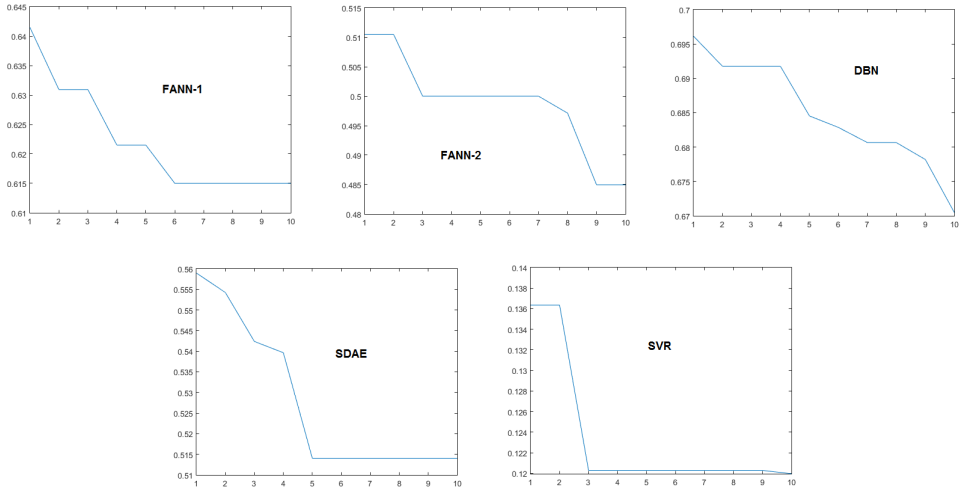


Figura 49 – EEG2 - Curva de aptidão do PSO

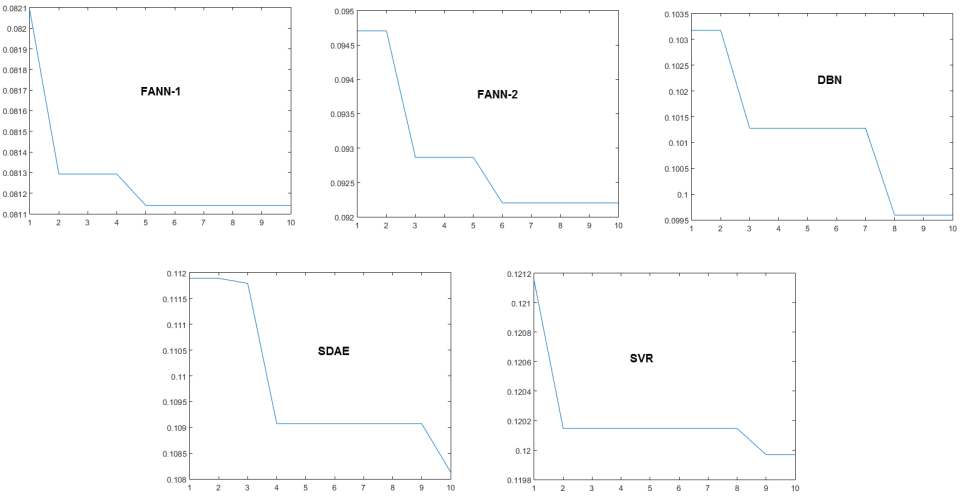


Figura 50 – EEG3 - Curva de aptidão do PSO

5.2.2 Combinadores

A tabela 24 (página 95) mostra a média e o devio-padrão dos combinadores que utilizaram todos os dez preditores do modelo, após 30 execuções. A tabela 25 (página 95) faz o mesmo para os combinadores que utilizaram como entrada os cinco melhores preditores do modelo, definidos através de seu desempenho em uma base de dados de validação.

Apesar de não ter havido correspondência estrita entre quais combinadores foram melhores em cada série temporal, certos padrões puderam ser identificados quando da comparação da previsão de longo alcance com a previsão de curto alcance. Em geral, os combinadores que tiveram como entrada os melhores preditores base alcançaram menores taxas de erro. Essa característica fica mais evidente com os combinadores estatísticos. Na maioria dos cenários, o desempenho das médias e da mediana aumenta substancialmente quando o número de entradas da combinação é limitado. Nesse sentido, é possível mais uma vez constatar que o softmax com os melhores preditores acaba sendo a melhor opção para combinação. Em contraponto aos outros combinadores testados, a versão do softmax com todos os preditores alcançou bons resultados. Como discutido na previsão de curto alcance, esse combinador é capaz de penalizar os preditores com maiores taxas de erro em sua combinação linear.

Como esperado, os melhores combinadores em cada série temporal obtiveram em geral melhor desempenho do que os melhores preditores base. Entretanto, os combinadores não foram capazes de recuperar o desempenho alcançado na previsão de curto alcance, pela dificuldade do problema.

5.2.3 Seleção Dinâmica

A tabela 26 (página 96) mostra a média e o desvio-padrão do MSE das quatro abordagens testadas do DSFC-A para a previsão de longo alcance (dez passos), após 30 execuções. A tabela 27 (página 96) faz o mesmo para o DSFC-B. Mais uma vez, o primeiro termo ALL ou BEST implica no uso de todos ou dos melhores preditores do modelo como entrada para os combinadores, respectivamente. O segundo termo indica o uso de todos ou dos melhores combinadores na seleção dinâmica. Os melhores preditores e combinadores foram determinados pelo melhor desempenho em uma base de dados de validação.

As tabelas 28, 29, 30, 31, 32, 33, 34, 35, 36 e 37 mostram a comparação do MSE do método proposto com o melhor preditor individual e com o melhor combinador para cada base de dados testada. As tabelas expressar se o método foi estatisticamente superior ($>$), equivalente ($=$) ou inferior ($<$), além de indicar o *p-value* a 5% de nível de confiança. A tabela 38 mostra o resumo das diferenças de desempenho entre as abordagens testadas e os melhores preditores e combinadores em cada uma das séries temporais.

Tabela 24 – MSE dos combinadores para todos os preditores (30 execuções, teste)

Base de Dados	Média	Média Aparada	Média Winsorizada	Mediana	RBLC	Softmax
Mackey-Glass	2.08e-02 (1.49e-05)	1.41e-02 (1.37e-05)	1.53e-02 (1.18e-05)	3.39e-03 (2.82e-05)	5.52e-03 (1.11e-05)	**1.01e-06 (1.99e-07)**
Lorenz	5.23e-01 (1.82e-03)	1.85e-02 (3.26e-04)	1.93e-02 (3.13e-04)	3.97e-03 (3.09e-04)	3.93e-01 (3.85e-02)	**8.62e-06 (8.38e-06)**
Henon	1.09e+00 (7.70e-03)	1.06e+00 (7.66e-03)	1.06e+00 (7.94e-03)	**1.03e+00 (1.04e-02)**	1.07e+00 (1.21e-02)	1.08e+00 (7.96e-03)
Rossler	2.52e+00 (3.20e-02)	9.31e-01 (4.15e-02)	9.30e-01 (3.96e-02)	1.42e-01 (2.78e-02)	7.74e+00 (1.79e-01)	**7.72e-05 (7.25e-05)**
Periodic	1.83e-03 (6.56e-06)	1.04e-03 (5.81e-06)	1.13e-03 (6.00e-06)	2.56e-04 (6.33e-06)	2.07e-03 (8.39e-06)	**8.44e-05 (3.54e-06)**
Quasi-Periodic	6.82e-02 (2.70e-03)	6.42e-02 (2.81e-03)	6.41e-02 (2.86e-03)	6.65e-02 (2.92e-03)	5.90e-02 (2.82e-03)	**4.77e-02 (3.52e-03)**
Laser	6.97e+02 (2.92e+01)	5.57e+02 (3.02e+01)	5.94e+02 (2.66e+01)	2.74e+02 (7.41e+01)	2.59e+02 (2.34e+01)	**1.71e+01 (8.09e+00)**
EEG1	3.23e+05 (6.64e+03)	3.22e+05 (6.70e+03)	3.25e+05 (6.67e+03)	**2.78e+05 (7.31e+03)**	3.49e+05 (9.64e+03)	3.00e+05 (6.93e+03)
EEG2	5.47e-01 (6.61e-03)	4.63e-01 (5.67e-03)	4.57e-01 (5.70e-03)	**3.94e-01 (4.53e-03)**	4.58e-01 (1.68e-02)	4.37e-01 (6.66e-03)
EEG3	1.10e-01 (1.61e-03)	8.43e-02 (1.38e-03)	8.50e-02 (1.34e-03)	7.74e-02 (1.88e-03)	**7.51e-02 (1.93e-03)**	8.23e-02 (1.81e-03)

Tabela 25 – MSE dos combinadores para os melhores preditores (30 execuções, teste)

Base de Dados	Média	Média Aparada	Média Winsorizada	Mediana	RBLC	Softmax
Mackey-Glass	9.36e-07 (1.00e-07)	9.36e-07 (1.00e-07)	9.16e-07 (9.78e-08)	**8.90e-07 (9.81e-08)**	5.22e-03 (1.64e-05)	1.01e-06 (1.99e-07)
Lorenz	1.06e-05 (4.09e-06)	1.06e-05 (4.09e-06)	1.02e-05 (4.16e-06)	9.89e-06 (4.56e-06)	2.49e+00 (2.10e-03)	**8.62e-06 (8.38e-06)**
Henon	1.04e+00 (1.60e-02)	1.04e+00 (1.60e-02)	**1.04e+00 (1.59e-02)**	1.04e+00 (1.77e-02)	1.04e+00 (1.80e-02)	1.04e+00 (1.59e-02)
Rossler	7.61e-05 (9.82e-05)	7.61e-05 (9.82e-05)	6.35e-05 (6.89e-05)	**5.20e-05 (4.05e-05)**	4.27e+00 (2.10e-02)	7.72e-05 (7.25e-05)
Periodic	8.47e-05 (3.01e-06)	8.47e-05 (3.01e-06)	8.46e-05 (2.97e-06)	8.45e-05 (2.96e-06)	1.05e-03 (1.86e-05)	**8.44e-05 (3.54e-06)**
Quasi-Periodic	3.50e-02 (3.57e-03)	3.50e-02 (3.57e-03)	3.49e-02 (3.49e-03)	3.48e-02 (3.41e-03)	3.60e-02 (4.13e-03)	**3.48e-02 (3.43e-03)**
Laser	1.37e+01 (1.26e+01)	1.37e+01 (1.26e+01)	1.24e+01 (8.53e+00)	**1.14e+01 (3.82e+00)**	5.40e+01 (2.57e+01)	1.68e+01 (8.21e+00)
EEG1	2.66e+05 (4.50e+03)	2.66e+05 (4.50e+03)	**2.65e+05 (4.66e+03)**	2.66e+05 (5.29e+03)	2.74e+05 (7.62e+03)	2.66e+05 (4.48e+03)
EEG2	**3.83e-01 (1.02e-02)**	**3.83e-01 (1.02e-02)**	3.84e-01 (1.10e-02)	3.86e-01 (1.41e-02)	3.89e-01 (9.96e-03)	3.84e-01 (1.02e-02)
EEG3	6.18e-02 (2.45e-03)	6.18e-02 (2.45e-03)	6.18e-02 (2.33e-03)	6.18e-02 (2.14e-03)	**6.09e-02 (2.10e-03)**	6.17e-02 (2.40e-03)

Tabela 26 – MSE dos DSFC-A (30 execuções, teste)

Base de Dados	ALL-ALL	ALL-BEST	BEST-ALL	BEST-BEST
Mackey-Glass	1.01e-06	1.01e-06	8.06e-07	**8.05e-07**
	(1.98e-07)	(1.98e-07)	(1.15e-07)	**(1.15e-07)**
Lorenz	**8.62e-06**	**8.62e-06**	8.98e-06	9.02e-06
	(8.38e-06)	**(8.38e-06)**	(3.69e-06)	(3.73e-06)
Henon	1.04e+00	1.04e+00	1.04e+00	**1.04e+00**
	(8.66e-03)	(1.05e-02)	(2.03e-02)	**(1.69e-02)**
Rossler	7.70e-05	7.70e-05	**5.48e-05**	**5.48e-05**
	(7.24e-05)	(7.24e-05)	**(5.04e-05)**	**(5.04e-05)**
Periodic	**8.35e-05**	**8.35e-05**	8.40e-05	8.40e-05
	(3.41e-06)	**(3.41e-06)**	(2.68e-06)	(2.66e-06)
Quasi-Periodic	4.37e-02	4.77e-02	3.92e-02	**3.47e-02**
	(3.07e-03)	(3.52e-03)	(4.38e-03)	**(3.36e-03)**
Laser	1.61e+01	1.67e+01	**1.11e+01**	1.11e+01
	(5.52e+00)	(7.66e+00)	**(3.28e+00)**	(3.41e+00)
EEG1	3.01e+05	2.80e+05	2.77e+05	**2.66e+05**
	(5.47e+03)	(6.90e+03)	(1.16e+04)	**(4.78e+03)**
EEG2	3.85e-01	3.90e-01	3.86e-01	**3.81e-01**
	(7.03e-03)	(4.07e-03)	(8.18e-03)	**(9.75e-03)**
EEG3	7.27e-02	7.61e-02	6.22e-02	**6.06e-02**
	(1.71e-03)	(1.71e-03)	(3.01e-03)	**(2.23e-03)**

Tabela 27 – MSE dos DSFC-B (30 execuções, teste)

Base de Dados	ALL-ALL	ALL-BEST	BEST-ALL	BEST-BEST
Mackey-Glass	5.52e-03	1.01e-06	**7.95e-07**	8.03e-07
	(1.11e-05)	(1.99e-07)	**(1.22e-07)**	(1.17e-07)
Lorenz	3.93e-01	**7.43e-06**	8.53e-06	8.55e-06
	(3.85e-02)	**(3.81e-06)**	(3.68e-06)	(3.66e-06)
Henon	1.07e+00	1.04e+00	1.04e+00	**1.04e+00**
	(1.21e-02)	(1.02e-02)	(1.96e-02)	**(1.67e-02)**
Rossler	7.74e+00	7.63e-05	**5.13e-05**	5.24e-05
	(1.79e-01)	(7.17e-05)	**(4.62e-05)**	(4.68e-05)
Periodic	2.07e-03	**8.35e-05**	8.40e-05	8.40e-05
	(8.39e-06)	**(3.43e-06)**	(2.66e-06)	(2.66e-06)
Quasi-Periodic	5.90e-02	4.77e-02	3.93e-02	**3.47e-02**
	(2.82e-03)	(3.52e-03)	(4.41e-03)	**(3.35e-03)**
Laser	2.59e+02	1.64e+01	1.09e+01	**1.05e+01**
	(2.34e+01)	(7.56e+00)	(3.53e+00)	**(3.59e+00)**
EEG1	3.49e+05	2.80e+05	2.77e+05	**2.66e+05**
	(9.64e+03)	(6.90e+03)	(1.06e+04)	**(4.80e+03)**
EEG2	4.58e-01	3.90e-01	3.86e-01	**3.81e-01**
	(1.68e-02)	(4.07e-03)	(8.21e-03)	**(9.75e-03)**
EEG3	7.51e-02	7.62e-02	6.21e-02	**6.06e-02**
	(1.93e-03)	(1.72e-03)	(2.82e-03)	**(2.22e-03)**

Tabela 28 – Mackey-Glass - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	= (5.78e-02)	< (9.63e-04)
DSFC-A ALL-BEST	= (5.78e-02)	< (9.63e-04)
DSFC-A BEST-ALL	> (1.73e-06)	> (1.97e-05)
DSFC-A BEST-BEST	> (1.73e-06)	> (1.97e-05)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (5.56e-02)	< (9.63e-04)
DSFC-B BEST-ALL	> (1.73e-06)	> (1.24e-05)
DSFC-B BEST-BEST	> (1.73e-06)	> (1.97e-05)

Tabela 29 – Lorenz - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-1	Melhor Combinador - Softmax BEST
DSFC-A ALL-ALL	> (3.61e-03)	= (2.85e-01)
DSFC-A ALL-BEST	> (3.61e-03)	= (2.85e-01)
DSFC-A BEST-ALL	> (1.11e-03)	= (7.66e-01)
DSFC-A BEST-BEST	> (9.63e-04)	= (7.34e-01)
DSFC-B ALL-ALL	> (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	> (1.83e-03)	= (5.24e-01)
DSFC-B BEST-ALL	> (1.59e-03)	= (9.92e-01)
DSFC-B BEST-BEST	> (2.11e-03)	= (9.59e-01)

Tabela 30 – Henon - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-1	Melhor Combinador - Mediana ALL
DSFC-A ALL-ALL	= (9.43e-01)	< (5.75e-06)
DSFC-A ALL-BEST	= (6.73e-01)	< (3.88e-06)
DSFC-A BEST-ALL	= (9.43e-01)	< (1.96e-03)
DSFC-A BEST-BEST	= (7.66e-01)	< (9.63e-01)
DSFC-B ALL-ALL	< (1.25e-04)	< (1.73e-06)
DSFC-B ALL-BEST	= (6.29e-01)	< (3.88e-06)
DSFC-B BEST-ALL	= (9.43e-01)	< (3.16e-03)
DSFC-B BEST-BEST	= (7.04e-01)	< (1.11e-01)

Tabela 31 – Rossler - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-2	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	= (9.70e-01)	= (5.45e-02)
DSFC-A ALL-BEST	= (9.70e-01)	= (5.45e-02)
DSFC-A BEST-ALL	= (1.06e-01)	= (3.39e-01)
DSFC-A BEST-BEST	= (8.35e-02)	= (3.60e-01)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (9.86e-01)	= (5.71e-02)
DSFC-B BEST-ALL	> (2.43e-02)	= (3.82e-01)
DSFC-B BEST-BEST	> (1.85e-02)	= (3.82e-01)

Tabela 32 – Periodic - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - SDAE	Melhor Combinador - Média BEST
DSFC-A ALL-ALL	= (9.59e-01)	= (2.80e-01)
DSFC-A ALL-BEST	= (5.71e-02)	< (5.71e-04)
DSFC-A BEST-ALL	= (6.58e-01)	< (2.43e-02)
DSFC-A BEST-BEST	= (5.44e-01)	= (5.72e-01)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	= (5.71e-02)	< (5.71e-04)
DSFC-B BEST-ALL	= (6.58e-01)	< (2.43e-02)
DSFC-B BEST-BEST	= (5.44e-01)	= (5.72e-01)

Tabela 33 – Quasi-Periodic - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-1	Melhor Combinador - Softmax BEST
DSFC-A ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-A ALL-BEST	< (1.73e-06)	< (1.73e-06)
DSFC-A BEST-ALL	< (1.11e-03)	< (3.18e-06)
DSFC-A BEST-BEST	> (1.04e-03)	= (6.56e-02)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	< (1.73e-06)	< (1.73e-06)
DSFC-B BEST-ALL	< (1.20e-03)	< (2.88e-06)
DSFC-B BEST-BEST	> (7.71e-04)	> (3.00e-02)

De uma maneiral geral, o desempenho dos algoritmos de seleção dinâmica diminuiu na previsão de longo alcance. Mais uma vez o pior desempenho foi alcançado pela abordagem DSFC-B ALL-ALL. Também similar à previsão de curto alcance, é possível constatar que as melhores abordagens de acordo com o MSE foram o DSFC-A BEST-BEST e o DSFC-B BEST-BEST. Ambas as variações atingiram os mesmos resultados, tendo

Tabela 34 – Laser - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	> (1.02e-05)	< (1.36e-05)
DSFC-A ALL-BEST	> (3.72e-05)	< (1.64e-05)
DSFC-A BEST-ALL	> (1.73e-06)	= (2.80e-01)
DSFC-A BEST-BEST	> (1.73e-06)	= (2.71e-01)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	> (3.41e-05)	< (1.36e-05)
DSFC-B BEST-ALL	> (1.73e-06)	= (3.18e-01)
DSFC-B BEST-BEST	> (1.73e-06)	> (1.85e-02)

Tabela 35 – EEG1 - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-A ALL-BEST	< (1.73e-06)	< (1.92e-06)
DSFC-A BEST-ALL	< (2.60e-06)	< (3.52e-06)
DSFC-A BEST-BEST	< (1.60e-04)	< (8.73e-03)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	< (1.73e-06)	< (1.92e-06)
DSFC-B BEST-ALL	< (2.88e-06)	< (2.60e-06)
DSFC-B BEST-BEST	< (1.60e-04)	< (3.61e-03)

Tabela 36 – EEG2 - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - DBN	Melhor Combinador - Mediana BEST
DSFC-A ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-A ALL-BEST	< (1.73e-06)	< (1.92e-06)
DSFC-A BEST-ALL	< (2.60e-06)	< (3.52e-06)
DSFC-A BEST-BEST	> (1.60e-04)	> (8.73e-03)
DSFC-B ALL-ALL	< (1.73e-06)	< (1.73e-06)
DSFC-B ALL-BEST	< (1.73e-06)	< (1.92e-06)
DSFC-B BEST-ALL	< (2.88e-06)	< (2.60e-06)
DSFC-B BEST-BEST	> (1.60e-04)	> (3.61e-03)

Tabela 37 – EEG3 - Teste de Wilcoxon em relação ao MSE

Modelo	Melhor Preditor - FANN-2	Melhor Combinador - RBLC BEST
DSFC-A ALL-ALL	< (2.35e-06)	< (1.73e-06)
DSFC-A ALL-BEST	< (1.92e-06)	< (1.73e-06)
DSFC-A BEST-ALL	= (6.44e-01)	< (9.32e-06)
DSFC-A BEST-BEST	> (1.29e-03)	> (3.32e-04)
DSFC-B ALL-ALL	< (1.92e-06)	< (1.73e-06)
DSFC-B ALL-BEST	< (1.92e-06)	< (1.73e-06)
DSFC-B BEST-ALL	= (3.49e-01)	< (1.36e-05)
DSFC-B BEST-BEST	> (1.29e-03)	> (4.90e-04)

sido estatisticamente superiores ao melhor preditor em sete base de dados e ao melhor combinador em cinco séries temporais. Entretanto, diferentemente do que ocorreu na previsão de curto alcance, as melhores abordagens da seleção dinâmica não conseguiram alcançar o desempenho do melhor combinador em uma base de dados, EEG1. Há de se salientar, entretanto, que a previsão de longo alcance em EEG1 tratou-se de um cenário único nas simulações numéricas realizadas: nenhuma abordagem foi sequer melhor que o principal preditor individual. Ainda assim, pode-se observar que DSFC-A BEST-BEST

Tabela 38 – Resultados no teste de Wilcoxon

Modelo	Preditores			Combinadores		
	>	=	<	>	=	<
DSFC-A ALL-ALL	2	4	4	0	3	7
DSFC-A ALL-BEST	2	4	4	0	3	7
DSFC-A BEST-ALL	3	4	3	1	3	6
DSFC-A BEST-BEST	7	2	1	5	4	1
DSFC-B ALL-ALL	1	0	9	0	0	10
DSFC-B ALL-BEST	2	4	4	0	2	8
DSFC-B BEST-ALL	4	3	3	1	3	6
DSFC-B BEST-BEST	7	2	1	5	4	1

e o DSFC-B BEST-BEST garantiram um desempenho mínimo em nove das dez séries temporais, fazendo-se necessário quando não se tem conhecimento de qual combinador atingirá as menores taxas de erro.

As figuras 51, 52, 53, 54, 55, 56, 57, 58, 59 e 60 (páginas 100 a 103) mostram as previsões de uma rodada de execução da melhor abordagem dos dois algoritmos propostos de seleção dinâmica, para previsão de longo alcance de cada série temporal. Na previsão, a linha azul com pontos é a saída desejada e a linha vermelha a saída produzida. Em cada figura, o eixo vertical do primeiro gráfico é dado pelos valores desejados e predições. O eixo horizontal é dado pelos pontos do conjunto de teste. Assim como ocorreu na previsão de curto alcance, as melhores abordagens dos métodos DSFC-A e DSFC-B obtiveram previsões bastante semelhantes.

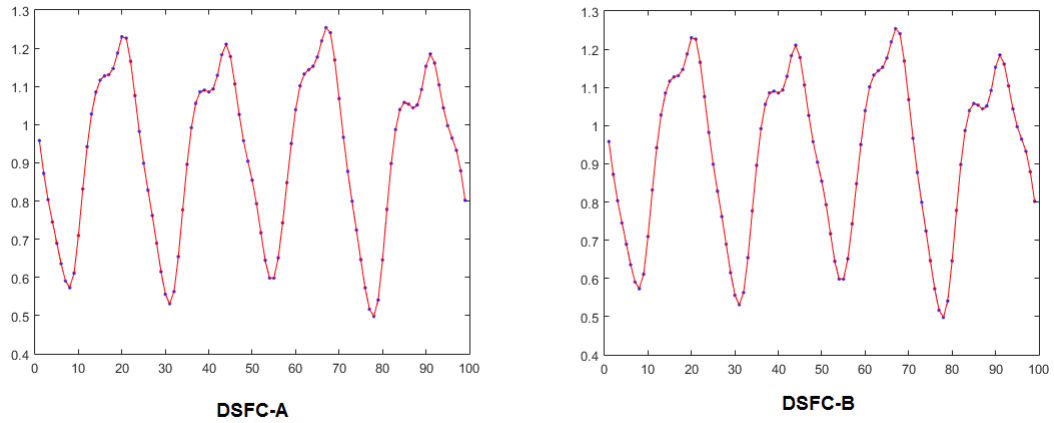


Figura 51 – Mackey-Glass - Previsão das melhores abordagens para DSFC-A e DSFC-B

Apesar da seleção dinâmica ter sido estatisticamente superior aos melhores preditores e combinadores nas bases EEG2 e EEG3, a previsão de longo alcance não conseguiu corresponder visualmente aos valores esperados. O mesmo ocorreu na série temporal EEG1. Entretanto, é possível argumentar que essas bases de dados são séries temporais reais, difíceis de serem previstas mesmo em situaçãoes com horizontes de previsão menores. Dentre as séries artificiais, a base de dados Henon foi a mais difícil de ser prevista. Isso

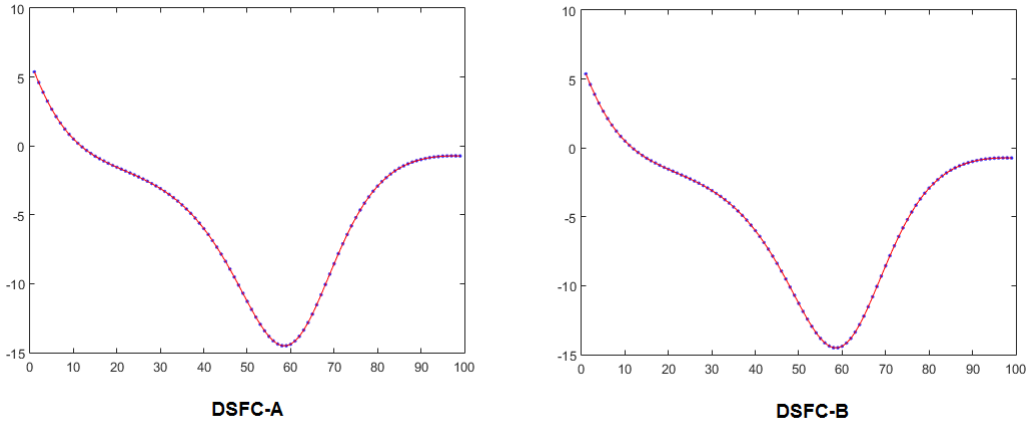


Figura 52 – Lorenz - Previsão das melhores abordagens para DSFC-A e DSFC-B

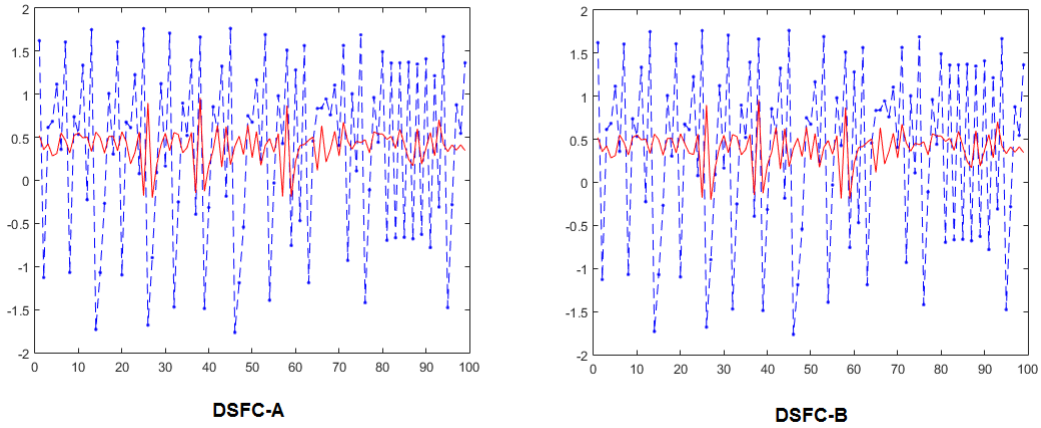


Figura 53 – Henon - Previsão das melhores abordagens para DSFC-A e DSFC-B

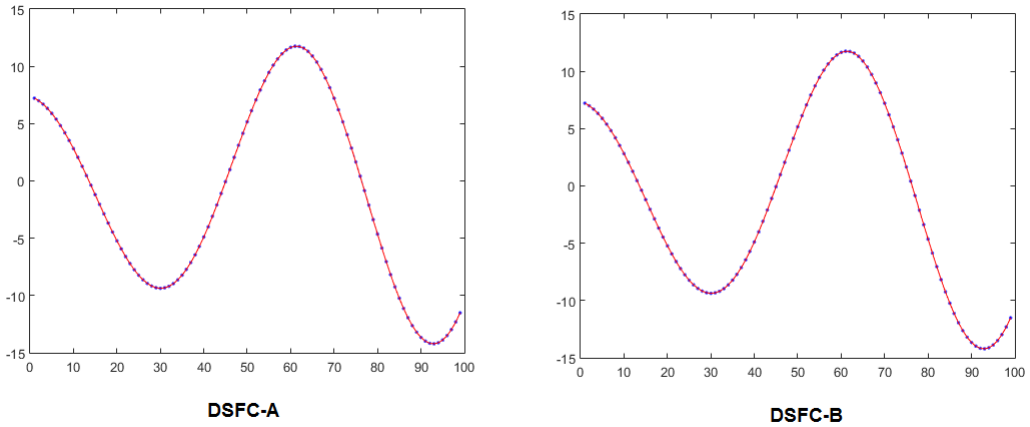


Figura 54 – Rossler - Previsão das melhores abordagens para DSFC-A e DSFC-B

pode ser resultado da sua natureza excessivamente sazonal, como pode ser visto na figura 53.

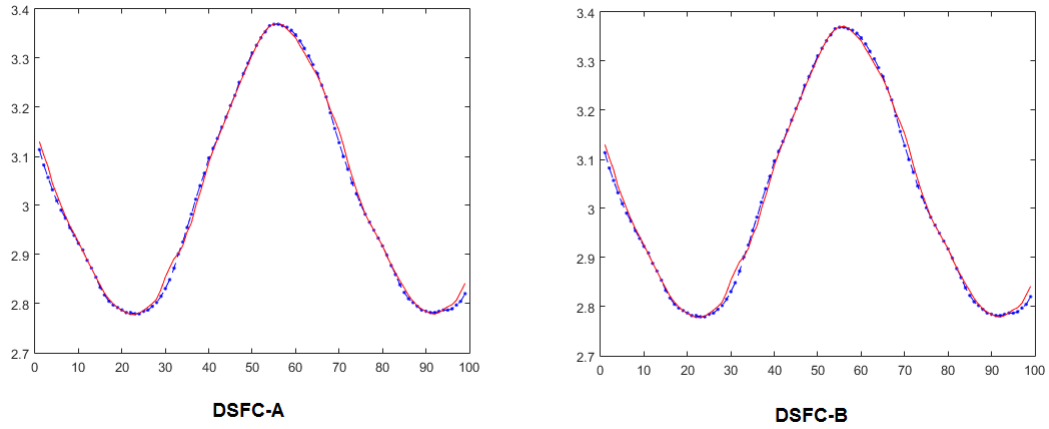


Figura 55 – Periodic - Previsão das melhores abordagens para DSFC-A e DSFC-B

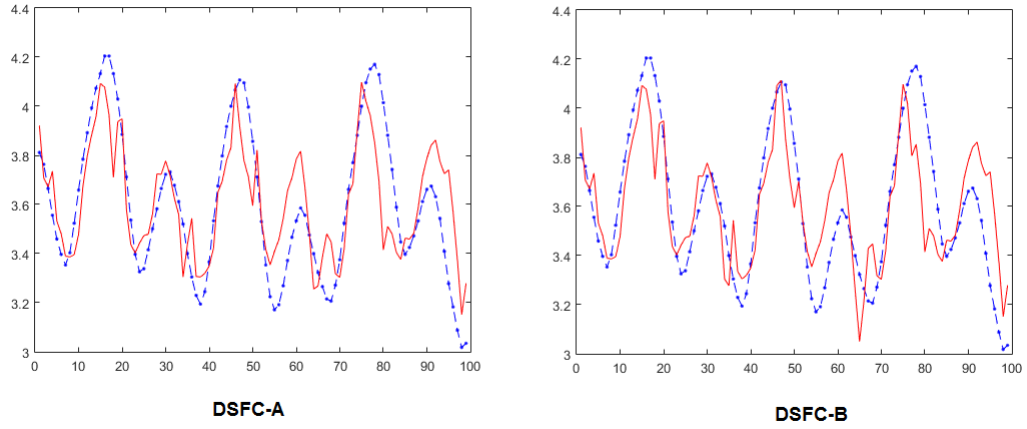


Figura 56 – Quasi-Periodic - Previsão das melhores abordagens para DSFC-A e DSFC-B

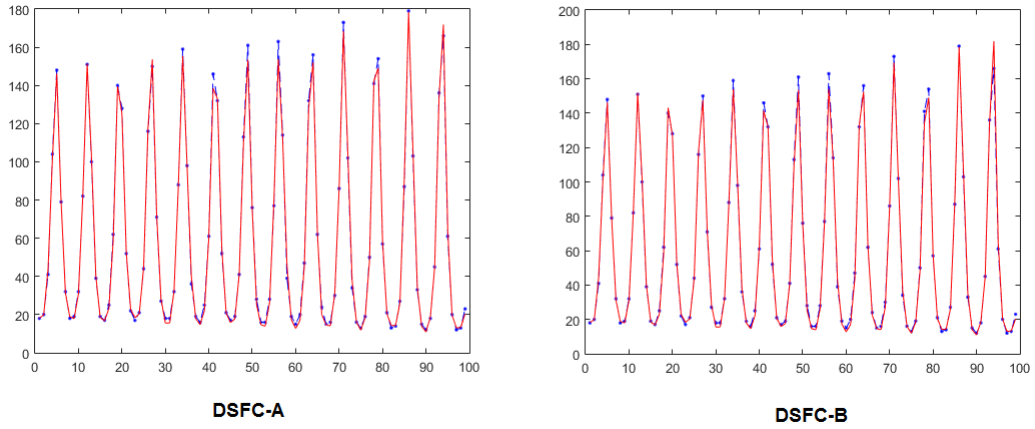


Figura 57 – Laser - Previsão das melhores abordagens para DSFC-A e DSFC-B

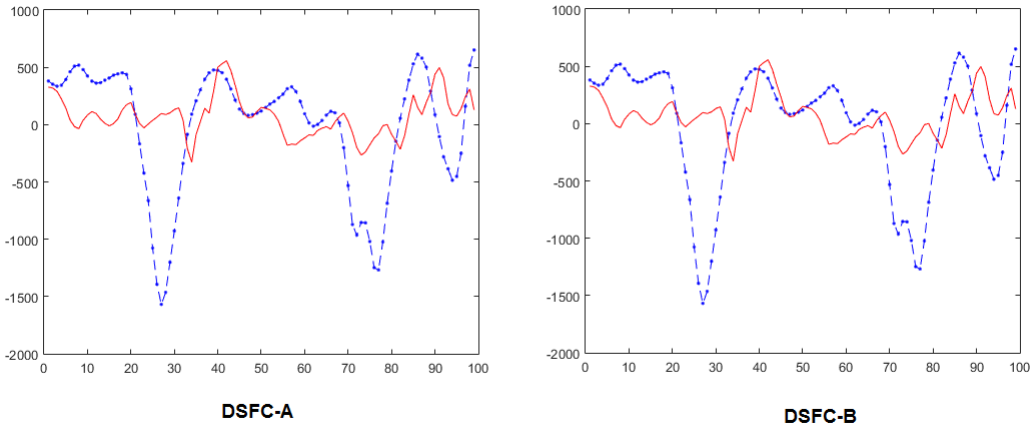


Figura 58 – EEG1 - Previsão das melhores abordagens para DSFC-A e DSFC-B

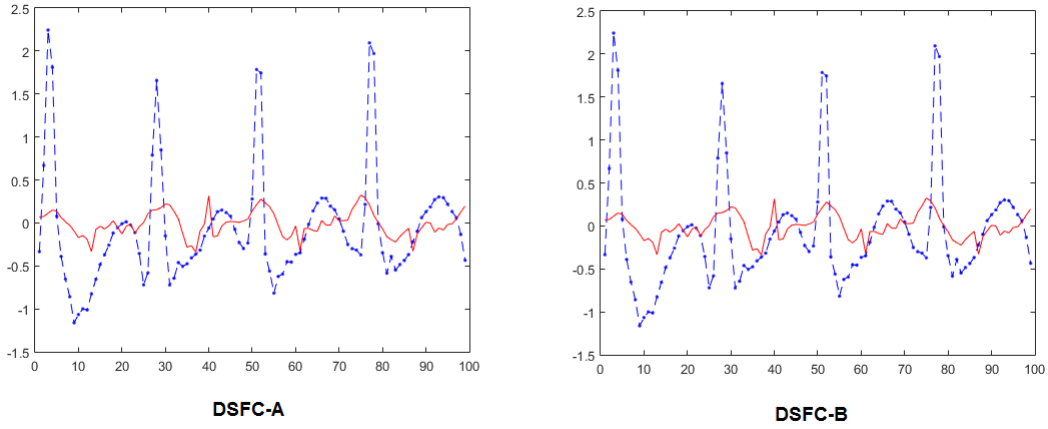


Figura 59 – EEG2 - Previsão das melhores abordagens para DSFC-A e DSFC-B

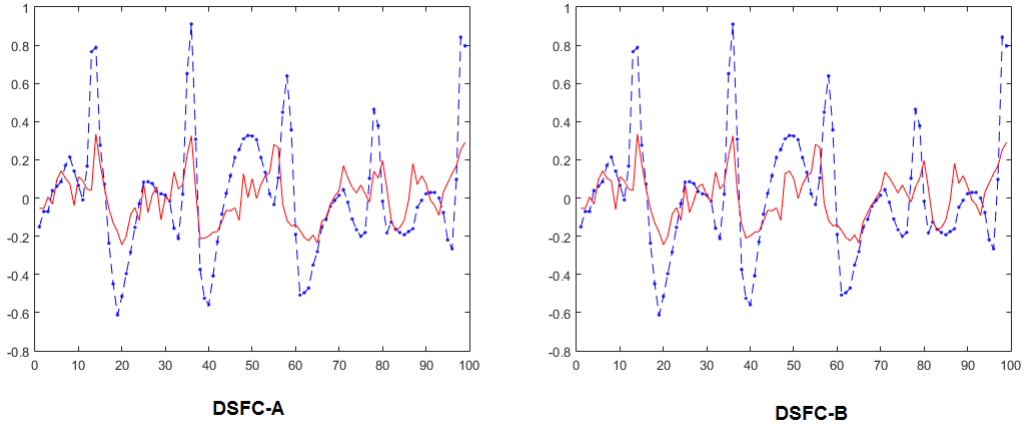


Figura 60 – EEG3 - Previsão das melhores abordagens para DSFC-A e DSFC-B

5.2.4 Tempo de Processamento

As tabelas 39 e 40 mostram a média e o desvio-padrão do tempo de processamento em segundos para a fase de geração e para cada uma das abordagens de seleção dinâmica, após 30 execuções. Com um horizonte de previsão mais alto, mais uma vez todas as abordagens de seleção dinâmica atingiram tempo de processamento menores que o tempo de geração. Nesse cenário, inclusive, o tempo de geração em algumas bases de dados chegou a aumentar, enquanto que em geral o tempo da seleção dinâmica manteve-se estável quando comparado à previsão de curto alcance. Na série Lorenz, por exemplo, o DSFC-A BEST BEST foi executado em aproximadamente 1% do tempo de geração.

Tabela 39 – Tempo de processamento, parte I

Fase	Mackey-Glass	Lorenz	Henon	Rossler	Periodic
Geração	3800.9 (237.2)	20429.9 (1676.0)	609.5 (31.9)	3067.0 (228.4)	1534.9 (92.8)
DSFC-A ALL-ALL	212.0 (1.2)	205.8 (0.5)	207.0 (1.4)	205.9 (0.5)	206.1 (0.5)
DSFC-A ALL-BEST	211.5 (1.3)	205.5 (0.5)	206.3 (1.3)	205.1 (0.6)	205.5 (0.5)
DSFC-A BEST-ALL	211.9 (0.8)	205.9 (0.4)	207.0 (1.4)	205.8 (0.5)	206.1 (0.4)
DSFC-A BEST-BEST	211.4 (0.6)	205.4 (0.5)	206.4 (1.2)	205.4 (0.6)	205.6 (0.5)
DSFC-B ALL-ALL	332.3 (0.7)	366.0 (2.5)	410.5 (1.7)	266.4 (0.9)	389.9 (0.8)
DSFC-B ALL-BEST	360.6 (2.2)	382.4 (2.9)	411.4 (1.6)	300.6 (1.3)	409.2 (1.0)
DSFC-B BEST-ALL	356.1 (2.1)	361.1 (0.6)	405.1 (11.8)	262.0 (0.8)	406.4 (1.1)
DSFC-B BEST-BEST	391.3 (2.1)	377.0 (0.6)	409.7 (2.6)	291.2 (0.7)	409.7 (0.8)
Total	6088.1 (235.7)	22739.0 (1674.9)	3072.9 (33.9)	5009.4 (226.7)	3973.4 (92.8)

Assim como ocorreu na previsão de curto alcance, em relação ao erro de previsão, o melhor desempenho foi alcançado por DSFC-A BEST-BEST e DSFC-B BEST-BEST. O tempo de processamento pode, então, mais uma vez ser utilizado como métrica para determinar qual abordagem atingiu os melhores resultados. Nesse sentido, o DSFC-A BEST-BEST pode ser considerado o melhor algoritmo para previsão de longo alcance, de acordo com essas métricas e as bases de dados testadas. Devido à ausência de um laço de repetição extra, o tempo de processamento das variações do DSFC-A chega a ter em média metade do tempo de execução das variações do DSFC-B.

5.3 **Comparação com a Literatura**

Os resultados alcançados pelo método proposto foram comparados com alguns trabalhos da literatura que utilizaram algumas das bases de dados testadas. Os trabalhos

Tabela 40 – Tempo de processamento, parte II

Fase	Quasi-Periodic	Laser	EEG1	EEG2	EEG3
Geração	776.9	876.2	646.7	679.2	647.7
	(46.1)	(43.0)	(27.4)	(29.8)	(38.6)
DSFC-A ALL-ALL	204.2	206.1	204.3	204.3	206.6
	(0.4)	(0.4)	(1.3)	(0.5)	(0.9)
DSFC-A ALL-BEST	203.5	205.7	203.4	203.9	206.1
	(0.7)	(1.5)	(1.5)	(1.0)	(0.6)
DSFC-A BEST-ALL	204.3	206.0	204.2	204.5	206.8
	(0.6)	(0.6)	(1.1)	(0.6)	(0.9)
DSFC-A BEST-BEST	203.7	205.5	204.2	203.9	206.2
	(0.5)	(0.5)	(1.9)	(0.5)	(0.7)
DSFC-B ALL-ALL	408.5	374.8	408.4	408.4	397.3
	(2.4)	(2.0)	(3.6)	(1.6)	(2.1)
DSFC-B ALL-BEST	406.9	386.7	407.4	407.0	404.5
	(1.5)	(2.4)	(3.9)	(1.0)	(1.1)
DSFC-B BEST-ALL	399.8	351.9	398.9	408.4	394.5
	(1.9)	(2.6)	(4.5)	(0.7)	(4.3)
DSFC-B BEST-BEST	404.9	370.9	403.9	406.9	406.3
	(0.6)	(2.3)	(3.6)	(0.8)	(2.3)
Total	3212.6	3183.8	3081.5	3126.6	3076.0
	(48.0)	(43.8)	(31.9)	(29.4)	(39.3)

relacionados nessa análise são de naturezas distintas, prevendo séries caóticas com técnicas desde inteligência de enxames (LIN; CHEN; LIN, 2009) a lógica difusa (MIRANIAN; ABDOLLAHZADE, 2013), passando por modelos híbridos (BODYANSKIY; VYNOKUROVA, 2013). A tabela 41 (página 106) mostra a comparação da média de três medidas de erro de previsão alcançadas pelo DSFC-A BEST-BEST e DSFC-B BEST-BEST (NMSE, NRMSE e RMSE) com os resultados da literatura, na previsão de curto alcance. É importante frisar que essa comparação deve ser feita com ressalvas, uma vez que os experimentos alheios não foram reproduzidos, e a maioria deles são resultados alcançados após uma única rodada de execução.

Em medidas absolutas, a seleção dinâmica alcançou melhor desempenho do que todos os trabalhos mostrados em quatro das séries temporais testadas (Mackey-Glass, Lorenz, Rossler e Henon). Nas séries Rossler e Henon, inclusive, o desempenho do método proposto em relação ao NMSE teve ganho bastante significativo. Na série Laser, o método proposto teve desempenho inferior em relação a duas técnicas: uma rede neural recursiva bayesiana e um modelo SVR com lógica difusa. Apesar disso, os resultados alcançados nesse trabalho mostram-se competitivos. Ademais, há de se salientar que o método proposto não se resume a utilização dos preditores individuais e das combinações que foram testadas. A seleção dinâmica apresentada nesse trabalho atua mais como um arcabouço, sendo possível modificar tanto os preditores quanto os combinadores. Sendo assim, a seleção dinâmica pode ser feita com uma combinação de diversas técnicas que já foram testadas ou que venham a ser utilizadas no futuro, tendendo a apresentar melhores resultados do que modelos individuais, como discutido nessa análise.

Tabela 41 – Comparação com a literatura: previsão de curto alcance

Base de Dados	Método	NMSE	NRMSE	RMSE
Mackey-Glass	DSFC-A BEST-BEST	1.48e-05	2.50e-03	8.27e-04
	DSFC-B BEST-BEST	1.45e-05	2.50e-03	8.20e-04
	Yang et al. (YANG; YU; PEDRYCZ, 2017)			3.80e-03
	Ardalani-Farsa et al. (ARDALANI-FARSA; ZOLFAGHARI, 2011)			1.30e-03
	Chandra et al. (CHANDRA; ZHANG, 2012)	2.79e-04		6.33e-03
	Li et al. (LI; HAN; WANG, 2012)		1.94e-01	
	Miranian et al. (MIRANIAN; ABDOLLAHZADE, 2013)			7.90e-04
	Yilmaz et al. (YILMAZ; OYSAL, 2010)			1.09e-03
Lorenz	DSFC-A BEST-BEST	2.57e-11	2.72e-06	2.27e-05
	DSFC-B BEST-BEST	2.89e-11	2.77e-06	2.37e-05
	Ardalani-Farsa et al. (ARDALANI-FARSA; ZOLFAGHARI, 2011)			2.96e-02
	Chandra et al. (CHANDRA; ZHANG, 2012)			6.36e-03
	Li et al. (LI; HAN; WANG, 2012)			2.23e-01
	Miranian et al. (MIRANIAN; ABDOLLAHZADE, 2013)	6.40e-05		
	Bodyanskiy et al. (BODYANSKIY; VYNOKUROVA, 2013)			1.89e-01
Henon	DSFC-A BEST-BEST	3.31e-11	4.32e-06	5.85e-06
	DSFC-B BEST-BEST	3.37e-11	4.34e-06	5.90e-06
	Mirikitani et al. 1 (MIRIKITANI; NIKOLAEV, 2010)	7.20e-04		
	Mirikitani et al. 2 (MIRIKITANI; NIKOLAEV, 2010)	6.80e-04		
	Miranian et al. (MIRANIAN; ABDOLLAHZADE, 2013)	4.40e-04		
Rossler	DSFC-A BEST-BEST	9.13e-11	5.67e-06	7.31e-05
	DSFC-B BEST-BEST	9.15e-11	5.68e-06	7.35e-05
	Mirikitani et al. 1 (MIRIKITANI; NIKOLAEV, 2010)	1.01e-03		
	Mirikitani et al. 2 (MIRIKITANI; NIKOLAEV, 2010)	8.10e-04		
	Miranian et al. (MIRANIAN; ABDOLLAHZADE, 2013)	1.50e-05		
Laser	DSFC-A BEST-BEST	1.40e-03	2.51e-02	1.86
	DSFC-B BEST-BEST	1.30e-03	2.49e-02	1.84
	Mirikitani et al. 1 (MIRIKITANI; NIKOLAEV, 2010)	4.36e-03		
	Mirikitani et al. 2 (MIRIKITANI; NIKOLAEV, 2010)	6.00e-04		
	Miranian et al. (MIRANIAN; ABDOLLAHZADE, 2013)	5.30e-04		

5.4 Discussão

A introdução desse trabalho levantou questionamentos acerca da seleção dinâmica para combinadores de previsão de séries temporais. Após a proposição de um arcabouço para resolução do problema, realização de simulações numéricas e apresentação dos resultados, esta sessão tem como objetivo sugerir considerações a respeito dessas questões.

- Quais e quantos preditores base devem ser utilizados na combinação?

  Em se tratando de comitês, o número de especialistas base tem um significativo impacto na acurácia do modelo. Com o crescente aumento de poder computacional, os comitês podem utilizar um grande número de técnicas para combinação dos resultados. Para problemas de classificação, um arcabouço teórico recentemente sugeriu que há um número ideal de classificadores base, sendo que uma quantidade maior ou menor pode deteriorar o desempenho do modelo de combinação. De acordo com a chamada "lei dos retornos decrescentes na construção do comitê" os autores mostraram que um número igual de classificadores e rótulos da base de dados implica em maior desempenho (BONAB; CAN, 2016). Para a combinação de preditores de séries temporais, não foram encontrados trabalhos que concentraram-se em investigar a relação entre o número de especialistas individuais e a acurácia da combinação das previsões. Pode ser constatado, entretanto, que normalmente os comitês construídos para classificação e reconhecimento de padrões tendem a possuir mais especialistas base do que em problemas de previsão de séries temporais. Por exemplo, Kuncheva et al. utilizam dez variações de árvores de decisão para a construção do comitê (KUNCHEVA; RODRIGUEZ et al., 2007), enquanto que Adhikari et al. utilizaram apenas quatro preditores em sua combinação proposta (ADHIKARI, 2015);

  Os resultados mostraram que, de maneira geral e de acordo com as séries temporais testadas, as menores taxas de erro foram obtidas pelos combinadores que utilizaram como entrada um número reduzido de preditores base. Nesses casos, dentre os dez preditores do modelo, foram selecionados os cinco melhores de acordo com seu desempenho em uma base de validação independente. É possível argumentar que, já que a combinação de preditores geralmente trata-se de uma média ponderada, um número grande de modelos individuais pode forçar a influência negativa de um preditor com alta taxa de erro. Esse problema pode ser limitado pelo uso de combinadores que utilizem uma base de dados independente para mensurar o desempenho dos preditores.

  É possível advogar que os resultados corroboraram a quantidade mais comum de modelos individuais em trabalhos correlatos, em torno de quatro ou cinco preditores. Importante salientar que é viável utilizar um número maior de modelos, conquanto que a combinação seja limitada por aqueles com um melhor desempenho em uma

base de dados independente. Inclusive, um número maior de modelos pode ampliar a diversidade do comitê, além de propiciar a possibilidade de diferentes preditores alcançarem bons resultados em séries temporais distintas. Nesse sentido, as simulações numéricas desse trabalho utilizam modelos heterogêneos, tanto estatísticos quanto de aprendizado de máquina. Nas séries temporais testadas, os modelos de redes neurais *deep learning* tiveram destaque. Porém, dado o teorema de que não há almoço grátis, o ideal é compor o comitê com técnicas de diferentes origens.

- Quais e quantos combinadores devem ser utilizados?

  O arcabouço proposto nesta tese tem o objetivo de realizar uma seleção dinâmica a partir das saídas de combinadores de previsão. Nesse sentido, mais uma vez é preciso definir quais e quantos combinadores devem ser utilizados. Assim como na definição dos preditores base, os resultados mostraram que o ideal é usar combinadores que divirjam entre si. Uma forma de implementar esse cenário é a utilização de combinadores heterogêneos. As simulações numéricas aqui apresentadas utilizaram combinadores com ou sem a necessidade de uma base de dados de validação para definição dos pesos da média ponderada. Como o processo de seleção dinâmica utiliza os melhores combinadores a partir de seu desempenho em uma base de dados independente, é prudente afirmar que podem ser utilizados no arcabouço tantos combinadores quanto for razoável a implementação dos modelos em termos de recursos computacionais.

- Na seleção dinâmica, devem ser levados em consideração somente os especialistas com melhor desempenho de validação ou todos os modelos gerados? Devem ser usados todos os combinadores do modelo ou apenas aqueles com melhor desempenho de validação?

  Essas questões tentaram ser respondidas pela variação dos métodos que foram executadas nas simulações numéricas. Referindo-se aos termos ALL e BEST, os dois algoritmos propostos de seleção dinâmica foram testados no contexto de utilizar todos ou os melhores preditores e combinadores.

  Em geral, a seleção dinâmica obteve resultados satisfatórios. Os métodos propostos selecionam o combinador com melhor desempenho em um subconjunto da base de treinamento semelhante a determinado padrão de teste. A similaridade, nos algoritmos implementados, é dada pela distância euclideana entre as previsões dos combinadores para o padrão de teste e para a região de competência selecionada. A seleção da região de competência é determinada pelos k vizinhos mais próximos do padrão de teste. Os métodos tendem a desbancar o desempenho dos combinadores porque é improvável que algum deles seja superior aos demais em todas as regiões de competência. Ao estender o raciocínio descrito pelas equações 3.1, 3.2 e 3.3 no problema de classificação para o problema de previsão de séries temporais, a seleção

dinâmica inclina-se a obter no mínimo o desempenho do melhor modelo do comitê. Para cada padrão de teste, os algoritmos buscam o melhor combinador pela região de competência mais adequada, definida pela experiência de desempenho em uma base de dados conhecida. Essa situação é justamente a que ocorreu nos experimentos realizados, onde os método propostos alcançaram pelo menos o desempenho do melhor combinador na maioria dos cenários testados.

Os resultados mostraram que a melhor abordagem de ambos os algoritmos propostos foi o uso dos melhores preditores e combinadores: DSFC-A BEST-BEST e DSFC-B BEST-BEST. A utilização dos melhores preditores é consequência do melhor desempenho dos combinadores quando da utilização dos preditores base com menores taxas de erro em uma base de dados independente. O melhor desempenho com a utilização dos melhores combinadores pode ser entendido como sendo nesse mesmo sentido. Os combinadores com maiores taxas de erro são descartados e as chances de se produzir uma saída mais distante do valor desejado diminuem. Adicionalmente, o melhor desempenho com um número reduzido de combinadores pode estar relacionado com a baixa dimensionalidade dos dados, implicando em regiões de competência mais limitadas. Como o DSFC-A BEST-BEST e o DSFC-B BEST-BEST alcançaram resultados semelhantes quanto às taxas de erro, os recursos computacionais consumidos podem ser utilizados como um critério adicional para escolha da melhor abordagem. Por esse ponto de vista, nas bases de dados testadas, o DSFC-A BEST-BEST foi o melhor método, pois além de ser competitivo em relação as saídas desejadas também é menos computacionalmente custoso quando comparado ao DSFC-B.

É possível ponderar que, em se tratando da seleção dinâmica de combinadores de previsão, o comitê pode ser formado por tantos modelos (preditores e combinadores) quanto for aceitável computacionalmente em relação aos recursos disponíveis, desde que se utilizem as melhores técnicas de acordo com o desempenho em uma base de dados independente. Esta tese advoga que o número de preditores não deve ultrapassar a quantidade sugerida e mais comum na literatura: cinco modelos base. Quanto aos combinadores, os resultados indicaram que um número reduzido é suficiente para que a seleção dinâmica funcione e produza na maioria dos cenários pelo menos um desempenho estatisticamente semelhante aos melhores preditores e combinadores. Nas simulações numéricas, o DSFC-A BEST-BEST e o DSFC-B BEST-BEST foi melhor do que o melhor preditor e o melhor combinador na maioria dos cenários.

- Qual o comportamento da seleção dinâmica quando se varia o alcance das previsões?

  Na previsão de curto alcance, as melhores variações (DSFC-A BEST-BEST e DSFC-B BEST-BEST) foram estatisticamente superiores ao melhor combinador em sete base de dados, sendo que nas outras três não foram piores. Já na previsão de

longo alcance, essas abordagens foram superadas em uma base de dados. Porém, foram estatisticamente superiores ao melhor combinador em cinco séries temporais e não foram superadas em outras quatro. O desempenho das melhores abordagens diminuiu moderadamente quando se aumentou o horizonte de previsão. Tal conjuntura parece ter sido parcialmente causada pelo próprio tipo de problema, já que na previsão de longo alcance há maiores incertezas acerca da correlação entre os dados e maior acúmulo de erros. Por outro lado, na previsão de longo alcance houve mais variabilidade em relação aos melhores preditores e combinadores, gerando oscilação na seleção dinâmica. Ainda assim, pelo custo computacional e pelas taxas de erro de previsão, as melhores abordagens dos algoritmos propostos de seleção dinâmica foram satisfatórias.

6 CONCLUSÃO E TRABALHOS FUTUROS

Este capítulo compreende as considerações finais e as propostas de trabalhos futuros.

6.1 Considerações Finais

Baseado em problemas de classificação e reconhecimento de padrões, este trabalho propôs um arcabouço de seleção dinâmica de combinadores de previsão de séries temporais. Inicialmente, preditores base com um bom grau de diversidade produzem suas respectivas saídas. A diversidade é alcançada através da utilização de modelos heterogêneos e validação cruzada. Segundo, são realizadas combinações das predições individuais. Um método de seleção dinâmica é então utilizado para selecionar a combinação mais promissora para cada padrão de teste. Foram desenvolvidos dois algoritmos de seleção dinâmica, o DSFC-A (*Dynamic Selection of Forecast Combiners - Acuracy*) e o DSFC-B (*Dynamic Selection of Forecast Combiners - Behavior*). Ambos utilizam a regra de vizinhos mais próximos para determinar um subconjunto de exemplos de validação que sejam mais semelhantes ao padrão de teste. O DSFC-A utiliza como norma a acurácia dos combinadores e o DSFC-B o comportamento de seu padrão de saída. Nesses algoritmos, é preciso definir quais preditores serão utilizados na combinação e quais combinadores serão utilizados na seleção dinâmica. Para apontar qual seria a melhor abordagem quanto a esse aspecto, foram testadas as variações possíveis do DSFC-A e do DSFC-B no que diz respeito ao uso de todos ou melhores preditores e todos ou melhores combinadores. A definição dos melhores preditores e combinadores para determinada série temporal é dada pelo desempenho dos modelos em uma base de dados de validação, independente. Sendo ALL e BEST as alcunhas para a utilização de todos ou melhores preditores e combinadores, foram testadas as seguintes variações: DSFC-A ALL-ALL, DSFC-A ALL-BEST, DSFC-A BEST-ALL, DSFC-A BEST-BEST, DSFC-B ALL-ALL, DSFC-B ALL-BEST, DSFC-B BEST-ALL e DSFC-B BEST-BEST. Em cada variação, o primeiro termo indica se foram utilizados todos ou os melhores preditores e o segundo termo indica se foram utilizados todos ou os melhores combinadores.

O arcabouço proposto independe dos preditores e combinadores. Para testar o método, foram utilizados preditores base advindos da aprendizado de máquina e da estatística. São eles: uma rede neural *feedforward* com uma camada escondida, uma rede neural *feedforward* com duas camadas escondidas, uma rede neural *deep learning* DBN, uma rede neural *deep learning* SDAE, um modelo SVR e modelos estatísticos lineares e não-lineares (AR, MA, ARMA, ARIMA e GARCH). Buscou-se com esses modelos atingir o maior espectro possível de preditores presentes na literatura. Nas simulações numéricas, os parâmetros dos modelos de aprendizado de máquina foram definidos pelo algoritmo de

otimização por enxame de partículas (PSO). Os seguintes combinadores foram utilizados: média, média aparada, média winsorizada, mediana, RBLC e softmax. Esses dois últimos combinadores necessitam de uma base de dados extra para cálculo dos pesos da combinação linear. Dos dez preditores base, um subconjunto de cinco modelos foram utilizados nas abordagens BEST. Em relação aos combinadores, o subconjunto dos melhores modelos foi composto por três dos seis combinadores implementados.

Para testar o método proposto, séries temporais caóticas foram utilizadas: Mackey-Glass, Lorenz, Rossler, Henon, Periodic, Quasi-Periodic, Laser e três séries produzidas a partir de exames de eletroencefalograma. A previsão de séries caóticas tem importância para várias áreas de atuação humana como astronomia e processamento de sinais, sendo que algumas das que foram testadas também funcionam como *benchmark* em diversas pesquisas. O desempenho do modelo foi observado em previsão de curto alcance e de longo alcance (horizontes de previsão um e dez, respectivamente).

De acordo com os resultados, foi observado que houve grande variabilidade quanto aos melhores preditores e combinadores para cada base de dados. Esse cenário evidenciou a necessidade de um algoritmo automático de seleção dinâmica. Foi possível constatar, entretanto, que os modelos *deep learning* se destacaram como preditores base. Quanto aos combinadores, o softmax em geral obteve as menores taxas de erro. Os resultados mostraram que, de acordo com os erros de previsão, as melhores abordagens do método de seleção dinâmica foram o DSFC-A BEST-BEST e o DSFC-B BEST-BEST. Isso significa que as menores taxas de erro foram obtidas quando da utilização do subconjunto dos cinco melhores preditores e dos três melhores combinadores, para cada base de dados. Esse comportamento é explicado pelo fato de que os modelos com altas taxas de erro não são incluídos no arcabouço, aumentando as chances de sucesso da seleção dinâmica. O número de preditores, inclusive, fica de acordo com as recomendações encontradas na literatura. Com o suporte de testes estatísticos, ficou constatado que na maioria dos cenários testados (incluindo previsão de curto e longo alcance), o DSFC-A BEST-BEST e o DSFC-B BEST-BEST atingiram menores erros de previsão do que o melhor preditor base o melhor combinador para cada série temporal. É importante salientar que o arcabouço pode ser utilizado com qualquer preditor base e combinador presentes na literatura ou que venham a ser desenvolvidos no futuro.

Já que o DSFC-A BEST-BEST e o DSFC-B BEST-BEST obtiveram o melhor desempenho no que diz respeito aos erros de previsão, o custo computacional de cada modelo foi também analisado. Sob essa ótica, foi possível constatar que o DSFC-A BEST-BEST acaba sendo a melhor abordagem de seleção dinâmica por consumir menos recursos computacionais. Recursos esses que foram sempre menores do que na criação do comitê de preditores e combinadores.

6.2 Trabalhos Futuros

Abaixo, alguns trabalhos futuros que podem ser derivados desta tese:

- Propor novos algoritmos de seleção dinâmica, baseados nas técnicas para problemas de classificação apresentadas no capítulo 3. Por exemplo, seria possível introduzir medidas baseadas em probabilidade ou oráculo. Se tais medidas forem concorrentes, é factível a aplicação de algum método de otimização multi-objetivo.
- Implementar mais preditores e combinadores no sentido de tentar aumentar o desempenho dos algoritmos de seleção dinâmica. Com mais modelos, aumenta a probabilidade do comitê conter as melhores técnicas para determinada base de dados.
- Investigar a adição dos melhores preditores na seleção dinâmica. É factível haver séries temporais em que a melhor saída seja dada por um preditor base e não por um combinador. Nesse sentido, os algoritmos de seleção dinâmica poderiam fazer uso dessas saídas para ter seu desempenho aumentado.
- Investigar se é possível de antemão determinar qual a quantidade mais indicada de preditores e combinadores para o arcabouço. Esta tese usou de empirismo e recomendações da literatura para chegar a esse valor.
- Testar o método proposto em mais séries temporais de uso prático.

REFERÊNCIAS

ADHIKARI, R. A neural network based linear ensemble framework for time series forecasting. *Neurocomputing*, Elsevier, v. 157, p. 231–242, 2015.

ADHIKARI, R.; AGRAWAL, R. A novel weighted ensemble technique for time series forecasting. In: *Advances in Knowledge Discovery and Data Mining*. [S.l.]: Springer, 2012. p. 38–49.

ALMEIDA, L. M.; GALVÃO, P. S. Ensembles with clustering-and-selection model using evolutionary algorithms. In: IEEE. *Intelligent Systems (BRACIS), 2016 5th Brazilian Conference on*. [S.l.], 2016. p. 444–449.

ANDRAWIS, R. R.; ATIYA, A. F.; EL-SHISHINY, H. Forecast combinations of computational intelligence and linear models for the nn5 time series forecasting competition. *International Journal of Forecasting*, Elsevier, v. 27, n. 3, p. 672–688, 2011.

ANDRIENKO, N.; ANDRIENKO, G. *Exploratory analysis of spatial and temporal data: a systematic approach*. [S.l.]: Springer Science & Business Media, 2006.

ANDRZEJAK, R. G. et al. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Physical Review E*, APS, v. 64, n. 6, p. 061907, 2001.

ARAÚJO, R. d. A.; OLIVEIRA, A. L.; MEIRA, S. A prediction model for high-frequency financial time series. In: IEEE. *Neural Networks (IJCNN), 2015 International Joint Conference on*. [S.l.], 2015. p. 1–8.

ARDALANI-FARSA, M.; ZOLFAGHARI, S. Residual analysis and combination of embedding theorem and artificial intelligence in chaotic time series forecasting. *Applied Artificial Intelligence*, Taylor & Francis, v. 25, n. 1, p. 45–73, 2011.

ARMSTRONG, J. S. Combining forecasts. In: *Principles of forecasting*. [S.l.]: Springer, 2001. p. 417–439.

ARMSTRONG, J. S. *Principles of forecasting: a handbook for researchers and practitioners*. [S.l.]: Springer Science & Business Media, 2001. v. 30.

ATYABI, A.; SHIC, F.; NAPLES, A. Mixture of autoregressive modeling orders and its implication on single trial eeg classification. *Expert Systems with Applications*, Elsevier, v. 65, p. 164–180, 2016.

AXELROD, R. M. *The evolution of cooperation*. [S.l.]: Basic books, 2006.

BAO, Y.; XIONG, T.; HU, Z. Multi-step-ahead time series prediction using multiple-output support vector regression. *Neurocomputing*, Elsevier, v. 129, p. 482–493, 2014.

BATES, J. M.; GRANGER, C. W. The combination of forecasts. *Or*, JSTOR, p. 451–468, 1969.

BENGIO, Y. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, Now Publishers Inc., v. 2, n. 1, p. 1–127, 2009.

BENGIO, Y.; COURVILLE, A.; VINCENT, P. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 35, n. 8, p. 1798–1828, 2013.

BENGIO, Y. et al. Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, MIT; 1998, v. 19, p. 153, 2007.

BERGH, F. V. D. *An analysis of particle swarm optimizers*. Tese (Doutorado) — University of Pretoria, 2006.

BODYANSKIY, Y.; VYNOKUROVA, O. Hybrid adaptive wavelet-neuro-fuzzy system for chaotic time series identification. *Information Sciences*, Elsevier, v. 220, p. 170–179, 2013.

BONAB, H. R.; CAN, F. A theoretical framework on the ideal number of classifiers for online ensembles in data streams. In: ACM. *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. [S.l.], 2016. p. 2053–2056.

BONTEMPI, G.; TAIEB, S. B.; BORGNE, Y.-A. L. Machine learning strategies for time series forecasting. In: *Business Intelligence*. [S.l.]: Springer, 2013. p. 62–77.

BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: ACM. *Proceedings of the fifth annual workshop on Computational learning theory*. [S.l.], 1992. p. 144–152.

BOURLARD, H.; KAMP, Y. Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, Springer, v. 59, n. 4-5, p. 291–294, 1988.

BOX, G. E. et al. *Time series analysis: forecasting and control*. [S.l.]: John Wiley & Sons, 2015.

BRADLEY, B. A. et al. A curve fitting procedure to derive inter-annual phenologies from time series of noisy satellite ndvi data. *Remote Sensing of Environment*, Elsevier, v. 106, n. 2, p. 137–145, 2007.

BRITTO, A. S.; SABOURIN, R.; OLIVEIRA, L. E. Dynamic selection of classifiers—a comprehensive review. *Pattern Recognition*, Elsevier, v. 47, n. 11, p. 3665–3680, 2014.

BUNN, D. W. A bayesian approach to the linear combination of forecasts. *Operational Research Quarterly*, JSTOR, p. 325–329, 1975.

CANUTO, A. d. P.; FAIRHURST, M.; PINTRO, F. Ensemble systems and cancellable transformations for multibiometric-based identification. *IET biometrics*, IET, v. 3, n. 1, p. 29–40, 2014.

CAVALCANTE, R. C.; MINKU, L. L.; OLIVEIRA, A. L. Fedd: Feature extraction for explicit concept drift detection in time series. In: IEEE. *Neural Networks (IJCNN), 2016 International Joint Conference on*. [S.l.], 2016. p. 740–747.

CAVALIN, P. R.; SABOURIN, R.; SUEN, C. Y. Dynamic selection approaches for multiple classifier systems. *Neural Computing and Applications*, Springer, v. 22, n. 3-4, p. 673–688, 2013.

CHANDRA, R.; ZHANG, M. Cooperative coevolution of elman recurrent neural networks for chaotic time series prediction. *Neurocomputing*, Elsevier, v. 86, p. 116–123, 2012.

CHAO, J.; SHEN, F.; ZHAO, J. Forecasting exchange rate with deep belief networks. In: IEEE. *Neural Networks (IJCNN), The 2011 International Joint Conference on*. [S.l.], 2011. p. 1259–1266.

CHAOTIC. *Chaotic time series database*. 2016. Disponível em: <http://www.physics.emory.edu/faculty/weeks//research/tseries1.html>.

CHEN, J.; JIN, Q.; CHAO, J. Design of deep belief networks for short-term prediction of drought index using data in the huaihe river basin. *Mathematical Problems in Engineering*, Hindawi Publishing Corporation, v. 2012, 2012.

CLERC, M.; KENNEDY, J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE transactions on Evolutionary Computation*, IEEE, v. 6, n. 1, p. 58–73, 2002.

COWPERTWAIT, P. S.; METCALFE, A. V. *Introductory time series with R*. [S.l.]: Springer Science & Business Media, 2009.

DAYAN, P. et al. The helmholtz machine. *Neural computation*, MIT Press, v. 7, n. 5, p. 889–904, 1995.

DEIHIMI, A.; ORANG, O.; SHOWKATI, H. Short-term electric load and temperature forecasting using wavelet echo state networks with neural reconstruction. *Energy*, Elsevier, v. 57, p. 382–401, 2013.

DEMŠAR, J. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, JMLR. org, v. 7, p. 1–30, 2006.

DIETTERICH, T. G. Ensemble learning. *The handbook of brain theory and neural networks*, MIT Press: Cambridge, MA, v. 2, p. 110–125, 2002.

DONATE, J. P. et al. Time series forecasting by evolving artificial neural networks with genetic algorithms, differential evolution and estimation of distribution algorithm. *Neural Computing and Applications*, Springer, v. 22, n. 1, p. 11–20, 2013.

EEG. *EEG time series database*. 2016. Disponível em: <https://vis.caltech.edu/~rodri/data.htm>.

EGRIOGLU, E.; ALADAG, C. H.; YOLCU, U. Fuzzy time series forecasting with a novel hybrid approach combining fuzzy c-means and neural networks. *Expert Systems with Applications*, Elsevier, v. 40, n. 3, p. 854–857, 2013.

ELLIOTT, G.; TIMMERMANN, A. *Handbook of Economic Forecasting SET 2A-2B*. [S.l.]: Elsevier, 2013.

ELMAN, J. L. Finding structure in time. *Cognitive science*, Elsevier, v. 14, n. 2, p. 179–211, 1990.

ENGLE, R. F. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal of the Econometric Society*, JSTOR, p. 987–1007, 1982.

FAN, G.-F. et al. Electric load forecasting by the svr model with differential empirical mode decomposition and auto regression. *Neurocomputing*, Elsevier, v. 173, p. 958–970, 2016.

FERREIRA, A. A.; LUDERMIR, T. B.; AQUINO, R. R. D. An approach to reservoir computing design and training. *Expert systems with applications*, Elsevier, v. 40, n. 10, p. 4172–4182, 2013.

FIRMINO, P. R. A.; NETO, P. S. de M.; FERREIRA, T. A. Error modeling approach to improve time series forecasters. *Neurocomputing*, Elsevier, v. 153, p. 242–254, 2015.

FLAKE, G. W. *The computational beauty of nature: Computer explorations of fractals, chaos, complex systems, and adaptation*. [S.l.]: MIT press, 1998.

FREITAS, P. S.; RODRIGUES, A. J. Model combination in neural-based forecasting. *European Journal of Operational Research*, Elsevier, v. 173, n. 3, p. 801–814, 2006.

GALAR, M. et al. Dynamic classifier selection for one-vs-one strategy: Avoiding non-competent classifiers. *Pattern Recognition*, Elsevier, v. 46, n. 12, p. 3412–3424, 2013.

GHEYAS, I. A.; SMITH, L. S. A novel neural network ensemble architecture for time series forecasting. *Neurocomputing*, Elsevier, v. 74, n. 18, p. 3855–3864, 2011.

GIACINTO, G.; ROLI, F. Methods for dynamic classifier selection. In: IEEE. *International Conference on Image Analysis and Processing, 1999*. [S.l.], 1999. p. 659–664.

GIACINTO, G.; ROLI, F. Dynamic classifier selection based on multiple classifier behaviour. *Pattern Recognition*, Pergamon, v. 34, n. 9, p. 1879–1881, 2001.

HASHEM, S.; SCHMEISER, B. Improving model accuracy using optimal linear combinations of trained neural networks. *IEEE Transactions on Neural Networks*, IEEE, v. 6, n. 3, p. 792–794, 1995.

HASSAN, R. et al. A comparison of particle swarm optimization and the genetic algorithm. In: *Proceedings of the 1st AIAA multidisciplinary design optimization specialist conference*. [S.l.: s.n.], 2005. p. 1–13.

HAYKIN, S. A comprehensive foundation. *Neural Networks*, v. 2, n. 2004, 2004.

HENTSCHEL, L. All in the family nesting symmetric and asymmetric garch models. *Journal of Financial Economics*, Elsevier, v. 39, n. 1, p. 71–104, 1995.

HINTON, G. E. Training products of experts by minimizing contrastive divergence. *Neural computation*, MIT Press, v. 14, n. 8, p. 1771–1800, 2002.

HINTON, G. E.; OSINDERO, S.; TEH, Y.-W. A fast learning algorithm for deep belief nets. *Neural computation*, MIT Press, v. 18, n. 7, p. 1527–1554, 2006.

HINTON, G. E.; SALAKHUTDINOV, R. R. Reducing the dimensionality of data with neural networks. *Science*, American Association for the Advancement of Science, v. 313, n. 5786, p. 504–507, 2006.

HO, T. K.; HULL, J. J.; SRIHARI, S. N. Decision combination in multiple classifier systems. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 16, n. 1, p. 66–75, 1994.

HOETING, J. A. et al. Bayesian model averaging: a tutorial. *Statistical science*, JSTOR, p. 382–401, 1999.

HURVICH, C. M.; TSAI, C.-L. Regression and time series model selection in small samples. *Biometrika*, Biometrika Trust, v. 76, n. 2, p. 297–307, 1989.

JAEGER, H. The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, v. 148, p. 34, 2001.

JOSE, V. R. R.; WINKLER, R. L. Simple robust averages of forecasts: Some empirical results. *International Journal of Forecasting*, Elsevier, v. 24, n. 1, p. 163–169, 2008.

KAPP, M. N.; SABOURIN, R.; MAUPIN, P. A dynamic model selection strategy for support vector machine classifiers. *Applied Soft Computing*, Elsevier, v. 12, n. 8, p. 2550–2565, 2012.

KELLERT, S. H. *In the wake of chaos: Unpredictable order in dynamical systems*. [S.l.]: University of Chicago press, 1994.

KENNEDY, J.; EBERHART, R. Particle swarm intelligence. In: IEEE. *Neural Networks, IEEE International Conference on*. [S.l.], 1995. p. 1942–1948.

KENNEL, M. B.; ISABELLE, S. Method to distinguish possible chaos from colored noise and to determine embedding parameters. *Physical Review A*, APS, v. 46, n. 6, p. 3111, 1992.

KEOGH, E. et al. Segmenting time series: A survey and novel approach. *Data mining in time series databases*, World Scientific Publishing, v. 57, p. 1–22, 2004.

KITTLER, J. et al. On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 20, n. 3, p. 226–239, 1998.

KRISTJANPOLLER, W.; MINUTOLO, M. C. Forecasting volatility of oil price using an artificial neural network-garch model. *Expert Systems with Applications*, Elsevier, v. 65, p. 233–241, 2016.

KUNCHEVA, L.; RODRIGUEZ, J. J. et al. Classifier ensembles with a random linear oracle. *Knowledge and Data Engineering, IEEE Transactions on*, IEEE, v. 19, n. 4, p. 500–508, 2007.

KUNCHEVA, L. I. Clustering-and-selection model for classifier combination. In: IEEE. *Knowledge-Based Intelligent Engineering Systems and Allied Technologies, 2000. Proceedings. Fourth International Conference on*. [S.l.], 2000. v. 1, p. 185–188.

KUREMOTO, T. et al. Time series forecasting using a deep belief network with restricted boltzmann machines. *Neurocomputing*, Elsevier, v. 137, p. 47–56, 2014.

LANDASSURI-MORENO, V. M.; BULLINARIA, J. A. Neural network ensembles for time series forecasting. In: ACM. *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*. [S.l.], 2009. p. 1235–1242.

LAROCHELLE, H. et al. Exploring strategies for training deep neural networks. *The Journal of Machine Learning Research*, JMLR. org, v. 10, p. 1–40, 2009.

LASER. *Laser generated time series*. 2016. Disponível em: <http://www-psych.stanford.edu/~andreas/Time-Series/SantaFe.html>.

LEMKE, C.; GABRYS, B. Meta-learning for time series forecasting and forecast combination. *Neurocomputing*, Elsevier, v. 73, n. 10, p. 2006–2016, 2010.

LI, D.; HAN, M.; WANG, J. Chaotic time series prediction based on a novel robust echo state network. *Neural Networks and Learning Systems, IEEE Transactions on*, IEEE, v. 23, n. 5, p. 787–799, 2012.

LIAN, C. et al. Ensemble of extreme learning machine for landslide displacement prediction based on time series analysis. *Neural Computing and Applications*, Springer, v. 24, n. 1, p. 99–107, 2014.

LIMA, T. P. F. de; SERGIO, A. T.; LUDERMIR, T. B. Improving classifiers and regions of competence in dynamic ensemble selection. In: IEEE. *Intelligent Systems (BRACIS), 2014 Brazilian Conference on*. [S.l.], 2014. p. 13–18.

LIN, C. et al. Libd3c: ensemble classifiers with a clustering and dynamic selection strategy. *Neurocomputing*, Elsevier, v. 123, p. 424–435, 2014.

LIN, C.-J.; CHEN, C.-H.; LIN, C.-T. A hybrid of cooperative particle swarm optimization and cultural algorithm for neural fuzzy networks and its prediction applications. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, IEEE, v. 39, n. 1, p. 55–68, 2009.

LIU, H. et al. Forecasting models for wind speed using wavelet, wavelet packet, time series and artificial neural networks. *Applied Energy*, Elsevier, v. 107, p. 191–208, 2013.

LIU, Y.; WANG, R. Study on network traffic forecast model of svr optimized by gafsa. *Chaos, Solitons & Fractals*, Elsevier, v. 89, p. 153–159, 2016.

LORENZ, E. N. Deterministic nonperiodic flow. *Journal of the atmospheric sciences*, v. 20, n. 2, p. 130–141, 1963.

MAASS, W.; NATSCHLÄGER, T.; MARKRAM, H. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural computation*, MIT Press, v. 14, n. 11, p. 2531–2560, 2002.

MAKRIDAKIS, S.; WINKLER, R. L. Averages of forecasts: Some empirical results. *Management Science*, INFORMS, v. 29, n. 9, p. 987–996, 1983.

MATLAB. *Statistics and Machine Learning Toolbox*. 2016. Disponível em: <https://www.mathworks.com/products/statistics/>.

MELIN, P. et al. A new approach for time series prediction using ensembles of anfis models. *Expert Systems with Applications*, Elsevier, v. 39, n. 3, p. 3494–3506, 2012.

MICHALSKI, R. S.; CARBONELL, J. G.; MITCHELL, T. M. *Machine learning: An artificial intelligence approach*. [S.l.]: Springer Science & Business Media, 2013.

MIHANDOOST, S.; AMIRANI, M. C. Cyclic spectral analysis of electrocardiogram signals based on garch model. *Biomedical Signal Processing and Control*, Elsevier, v. 31, p. 79–88, 2017.

MIRANIAN, A.; ABDOLLAHZADE, M. Developing a local least-squares support vector machines-based neuro-fuzzy model for nonlinear and chaotic time series prediction. *Neural Networks and Learning Systems, IEEE Transactions on*, IEEE, v. 24, n. 2, p. 207–218, 2013.

MIRIKITANI, D. T.; NIKOLAEV, N. Recursive bayesian recurrent neural networks for time-series modeling. *Neural Networks, IEEE Transactions on*, IEEE, v. 21, n. 2, p. 262–274, 2010.

MITCHELL, T. M. Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, v. 45, p. 37, 1997.

MORETTI, F. et al. Urban traffic flow forecasting through statistical and neural network bagging ensemble hybrid modeling. *Neurocomputing*, Elsevier, v. 167, p. 3–7, 2015.

NABIHA, A.; NADIR, F. New dynamic ensemble of classifiers selection approach based on confusion matrix for arabic handwritten recognition. In: IEEE. *Multimedia Computing and Systems (ICMCS), 2012 International Conference on*. [S.l.], 2012. p. 308–313.

NN3. *NN3 Forecasting Competition*. 2016. Disponível em: <http://www.neural-forecasting-competition.com/NN3/>.

NN5. *NN5 Forecasting Competition*. 2016. Disponível em: <http://www.neural-forecasting-competition.com/NN5/>.

NÓBREGA, J. P.; OLIVEIRA, A. L. A combination forecasting model using machine learning and kalman filter for statistical arbitrage. In: IEEE. *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*. [S.l.], 2014. p. 1294–1299.

OLIVEIRA, M.; TORGO, L. Ensembles for time series forecasting. In: *Proceedings of the Sixth Asian Conference on Machine Learning*. [S.l.: s.n.], 2014. p. 360–370.

POULTNEY, C. et al. Efficient learning of sparse representations with an energy-based model. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2006. p. 1137–1144.

QI, M.; ZHANG, G. P. An investigation of model selection criteria for neural network time series forecasting. *European Journal of Operational Research*, Elsevier, v. 132, n. 3, p. 666–680, 2001.

REID, D. J. *A comparative study of time series prediction techniques on economic data*. [S.l.]: University of Nottingham, Library Photographic Unit, 1969.

ROJAS, I. et al. Soft-computing techniques and arma model for time series prediction. *Neurocomputing*, Elsevier, v. 71, n. 4, p. 519–537, 2008.

ROMEU, P. et al. Time-series forecasting of indoor temperature using pre-trained deep neural networks. In: SPRINGER. *International Conference on Artificial Neural Networks*. [S.l.], 2013. p. 451–458.

RÖSSLER, O. E. An equation for continuous chaos. *Physics Letters A*, Elsevier, v. 57, n. 5, p. 397–398, 1976.

SABOURIN, M. et al. Classifier combination for hand-printed digit recognition. In: IEEE. *Document Analysis and Recognition, 1993., Proceedings of the Second International Conference on*. [S.l.], 1993. p. 163–166.

SAMANTA, B. Prediction of chaotic time series using computational intelligence. *Expert Systems with Applications*, Elsevier, v. 38, n. 9, p. 11406–11411, 2011.

SANTANA, A. et al. A dynamic classifier selection method to build ensembles using accuracy and diversity. In: IEEE. *Neural Networks, 2006. SBRN'06. Ninth Brazilian Symposium on*. [S.l.], 2006. p. 36–41.

SANTOS, E. M. D.; SABOURIN, R.; MAUPIN, P. Ambiguity-guided dynamic selection of ensemble of classifiers. In: IEEE. *Information Fusion, 2007 10th International Conference on*. [S.l.], 2007. p. 1–8.

SANTOS, E. M. D.; SABOURIN, R.; MAUPIN, P. A dynamic overproduce-and-choose strategy for the selection of classifier ensembles. *Pattern Recognition*, Elsevier, v. 41, n. 10, p. 2993–3009, 2008.

SERGIO, A.; LUDERMIR, T. B. Deep learning for wind speed forecasting in northeastern region of brazil. In: IEEE. *2015 Brazilian Conference on Intelligent Systems (BRACIS)*. [S.l.], 2015. p. 322–327.

SERGIO, A. T.; LIMA, T. P. de; LUDERMIR, T. B. Dynamic selection of forecast combiners. *Neurocomputing*, Elsevier, v. 218, p. 37–50, 2016.

SERGIO, A. T.; LUDERMIR, T. B. Reservoir computing optimization with a hybrid method. In: IEEE. *2014 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2014. p. 2653–2660.

SERRE, T. et al. A quantitative theory of immediate visual recognition. *Progress in brain research*, Elsevier, v. 165, p. 33–56, 2007.

SHAMAN, J.; YANG, W.; KANDULA, S. Inference and forecast of the current west african ebola outbreak in guinea, sierra leone and liberia. *PLoS currents*, Public Library of Science, v. 6, 2014.

SHI, H. et al. Trend prediction of fdi based on the intervention model and arima-garch-m model. *AASRI Procedia*, Elsevier, v. 3, p. 387–393, 2012.

SHIN, H.; SOHN, S. Y. Combining both ensemble and dynamic classifier selection schemes for prediction of mobile internet subscribers. *Expert Systems with Applications*, Elsevier, v. 25, n. 1, p. 63–68, 2003.

SMOLA, A.; VAPNIK, V. Support vector regression machines. *Advances in neural information processing systems*, v. 9, p. 155–161, 1997.

SORJAMAA, A. et al. Methodology for long-term prediction of time series. *Neurocomputing*, Elsevier, v. 70, n. 16, p. 2861–2869, 2007.

SORJAMAA, A.; LENDASSE, A. Time series prediction using dirrec strategy. In: *ESANN*. [S.l.: s.n.], 2006. v. 6, p. 143–148.

STOSIC, D. et al. Foreign exchange rate entropy evolution during financial crises. *Physica A: Statistical Mechanics and its Applications*, Elsevier, v. 449, p. 233–239, 2016.

SVM, L. *Lib SVM Toolbox*. 2016. Disponível em: <https://www.csie.ntu.edu.tw/~cjlin/libsvm/>.

TANEJA, K. et al. Time series analysis of aerosol optical depth over new delhi using box–jenkins arima modeling approach. *Atmospheric Pollution Research*, Elsevier, 2016.

TIAMPO, K. F.; SHCHERBAKOV, R. Seismicity-based earthquake forecasting techniques: Ten years of progress. *Tectonophysics*, Elsevier, v. 522, p. 89–121, 2012.

TIMMERMANN, A. Forecast combinations. *Handbook of economic forecasting*, Elsevier, v. 1, p. 135–196, 2006.

TRELEA, I. C. The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information processing letters*, Elsevier, v. 85, n. 6, p. 317–325, 2003.

UTGOFF, P. E.; STRACUZZI, D. J. Many-layered learning. *Neural Computation*, MIT Press, v. 14, n. 10, p. 2497–2529, 2002.

VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer Science & Business Media, 2013.

VINCENT, P. et al. Extracting and composing robust features with denoising autoencoders. In: ACM. *Proceedings of the 25th international conference on Machine learning*. [S.l.], 2008. p. 1096–1103.

VOYANT, C. et al. Meteorological time series forecasting based on mlp modelling using heterogeneous transfer functions. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2015. v. 574, n. 1, p. 012064.

WANG, X. et al. Research on the relation of eeg signal chaos characteristics with high-level intelligence activity of human brain. *Nonlinear biomedical physics*, BioMed Central Ltd, v. 4, n. 1, p. 2, 2010.

WOLPERT, D. H. The lack of a priori distinctions between learning algorithms. *Neural computation*, MIT Press, v. 8, n. 7, p. 1341–1390, 1996.

WOODS, K.; BOWYER, K.; JR, W. P. K. Combination of multiple classifiers using local accuracy estimates. In: IEEE. *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*. [S.l.], 1996. p. 391–396.

XI, X. et al. Fast time series classification using numerosity reduction. In: ACM. *Proceedings of thhe 23rd international conference on Machine learning*. [S.l.], 2006. p. 1033–1040.

YAN, W. Toward automatic time-series forecasting using neural networks. *Neural Networks and Learning Systems, IEEE Transactions on*, IEEE, v. 23, n. 7, p. 1028–1039, 2012.

YANG, X.; YU, F.; PEDRYCZ, W. Long-term forecasting of time series based on linear fuzzy information granules and fuzzy inference system. *International Journal of Approximate Reasoning*, Elsevier, v. 81, p. 1–27, 2017.

YANG, X.-H. et al. Chaotic bayesian optimal prediction method and its application in hydrological time series. *Computers & Mathematics with Applications*, Elsevier, v. 61, n. 8, p. 1975–1978, 2011.

YAO, X.; LIU, Y. Making use of population information in evolutionary artificial neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 28, n. 3, p. 417–425, 1998.

YILMAZ, S.; OYSAL, Y. Fuzzy wavelet neural network models for prediction and identification of dynamical systems. *Neural Networks, IEEE Transactions on*, IEEE, v. 21, n. 10, p. 1599–1609, 2010.

YUAN, C.; LIU, S.; FANG, Z. Comparison of china's primary energy consumption forecasting by using arima (the autoregressive integrated moving average) model and gm (1, 1) model. *Energy*, Elsevier, v. 100, p. 384–390, 2016.

ZHAI, Y. et al. Water demand forecasting of beijing using the time series forecasting method. *Journal of Geographical Sciences*, Springer, v. 22, n. 5, p. 919–932, 2012.

ZHANG, G.; PATUWO, B. E.; HU, M. Y. Forecasting with artificial neural networks:: The state of the art. *International journal of forecasting*, Elsevier, v. 14, n. 1, p. 35–62, 1998.

ZHANG, G. P. A neural network ensemble method with jittered training data for time series forecasting. *Information Sciences*, Elsevier, v. 177, n. 23, p. 5329–5346, 2007.

ZHANG, L. et al. Iterated time series prediction with multiple support vector regression models. *Neurocomputing*, Elsevier, v. 99, p. 411–422, 2013.

ZHIQIANG, G.; HUAIQING, W.; QUAN, L. Financial time series forecasting using lpp and svm optimized by pso. *Soft Computing*, Springer, v. 17, n. 5, p. 805–818, 2013.

APÊNDICE A – OTIMIZAÇÃO POR ENXAME DE PARTÍCULAS

A afirmação "pensar é social" pode ser utilizada como motivador da utilização de um algoritmo como a Otimização por Enxame de Partículas (PSO, do inglês *Particle Swarm Optimization*. Ela vem da ideia de que uma iteração grupal pode melhorar a capacidade cognitiva de um conjunto de indivíduos, que por sua vez aprendem e compartilham conhecimento com seus vizinhos. O assim denominado Modelo Cultural Adaptativo (AXELROD, 2006) é baseado em três processos básicos: avaliar, comparar e imitar. Um sistema classifica estímulos em positivos ou negativos, define um referencial e aprende de acordo com esse mesmo referencial.

É nesse sentido que o PSO apresenta-se como uma técnica de otimização global baseada em uma população de soluções. De forma geral, o algoritmo é baseado no comportamento social de revoadas dos pássaros, onde um indivíduo imita as ações do melhor do grupo (ou mais indicado). O processo tem início com a definição da população de soluções. Cada indivíduo, por vezes chamado de "partícula", é uma solução possível. Cada partícula possui uma posição e uma velocidade, e o processo de atualização é baseado na melhor experiência pessoal e na melhor experiência do grupo. O PSO foi criado por Kennedy e Eberhart em 1995 (KENNEDY; EBERHART, 1995).

Seja s o tamanho do enxame, n a dimensão do problema e t o instante de tempo atual. Cada partícula $1 \leq i \leq s$ tem uma posição $x_i(t) \in R^n$ no espaço da solução e uma velocidade $v_i(t) \in R^n$, que por sua vez controla a magnitude, o sentido e a direção do movimento. Cada partícula mantém a melhor posição individual $y_i(t) \in R^n$ visitada até o instante de tempo t. Por outro lado, o enxame como um todo mantém em memória a melhor posição $\hat{y}(t) \in R^n$ visitada até agora por cada uma das partículas.

Ao longo do algoritmo, a velocidade de cada partícula é guiada por duas variáveis, ou pontos de busca: a melhor posição individual visitada até agora (termo cognitivo da otimização, $y_i(t)$) e a melhor posição global visitada até agora (termo social da otimização, $\hat{y}(t)$). Matematicamente, a nova velocidade de cada partícula é dada pela equação A.1, enquanto que a equação A.2 determina sua nova posição.

$$v_{ij}(t+1) = w * v_{ij}(t) + c_1 * r_1 * (y_{ij}(t) - x_{ij}(t)) + c_2 * r_2 * (\hat{y}_j(t) - x_{ij}(t)) \tag{A.1}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{A.2}$$

onde o termo j indica a dimensão da partícula, $1 \leq i \leq s$ e $1 \leq j \leq n$.

O chamado termo de momentum (ou inércia), w, provoca uma busca mais exploratória nas primeiras iterações e com nível maior de explotação nas últimas iterações. Essa variável é uma escalar que geralmente decresce linearmente de 0,9 a 0,4. Os termos r_1 e r_2 são variáveis aleatórias uniformemente variando de 0 a 1. Essas variáveis estão relacionadas com os dois termos da equação (cognitivo e social). c_1 e c_2 são os coeficientes de aceleração individual e local, respectivamente. Os coeficientes têm valores fixos e iguais, sendo responsáveis pelo controle do movimento da partícula em cada iteração.

A equação A.3 mostra como a melhor posição individual é atualizada, enquanto que a melhor posição global é atualizada de acordo com a equação A.4. É importante frisar que a velocidade é determinada por um intervalo com os limites mínimo e máximo, evitando assim a "explosão" do enxame. O algoritmo 12 apresenta o algoritmo do PSO padrão. ff é a função de aptidão do algoritmo (ou *fitness*). A função de aptidão define se determinada posição de uma partícula tem desempenho melhor ou pior do que outra, determinando o curso de atualização das posições e das velocidades.

$$y_i(t+1) = \begin{cases} y_i(t), & \text{se } f(x_i(t+1)) > f(y_i(t)) \\ x_i(t+1), & \text{se } f(x_i(t+1)) < f(y_i(t)) \end{cases} \tag{A.3}$$

$$\hat{y}(t+1) = argmin f(y_i(t+1)) \tag{A.4}$$

Pseudocodigo 12 PSO

```
1: Inicie aleatoriamente a população de partículas
2: while critério de parada não for alcançado do
3:     for cada partícula i da população do
4:         if f(x_i(t)) < f(y_i(t)) then
5:             y_i(t) = x_i(t)
6:         end if
7:         if f(y_i(t)) < f(ŷ(t)) then
8:             ŷ(t) = y_i(t)
9:         end if
10:    end for
11:    Atualize velocidade e posição de cada partícula de acordo com equações A.1 e A.2
12: end while
```

Trabalhos como o de Van den Bergh (BERGH, 2006), Clerc et al. (CLERC; KENNEDY, 2002) e Trelea (TRELEA, 2003) analisam matematicamente a convergência do PSO. Tais trabalhos tiveram como resultado orientações sobre quais parâmetros afetam a convergência, divergência ou oscilação do algoritmo, e esses estudos deram origem a diversas variações do PSO original. Entretanto, Pedersen [PC10] mostra que essas análises são simplistas, no sentido em que assumem que o enxame possui somente uma partícula, que não utilizam variáveis estocásticas e que os pontos de atração permanecem constantes

durante o processo de otimização. Adicionalmente, certas análises permitem um número infinito de iterações, impossível na realidade. Sendo assim, determinar as capacidades de convergência do PSO ainda depende de resultados empíricos.