UNIVERSIDADE FEDERAL DO PARANÁ

Leonardo Trigueiro dos Santos

Abordagem da máquina de vetor suporte otimizada por evolução diferencial aplicada à previsão de ventos

CURITIBA

2013

Leonardo Trigueiro dos Santos

Abordagem da máquina de vetor suporte otimizada por evolução diferencial aplicada à previsão de ventos

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Engenharia Elétrica, pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Paraná.

Orientador: Leandro dos Santos Coelho

CURITIBA

2013

Santos, Leonardo Trigueiro dos
Abordagem da máquina de vetor suporte otimizada por evolução diferencial aplicada à previsão de ventos / Leonardo Trigueiro dos Santos. – Curitiba, 2013.
77 f. : il.; graf., tab.

Dissertação (mestrado) – Universidade Federal do Paraná, Setor de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica.
Orientador: Leandro dos Santos Coelho

1. Energia eólica. I. Coelho, Leandro dos Santos. II. Título.

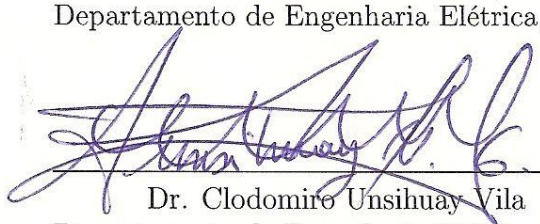
CDD 621.312136

Termo de Aprovação

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre em Engenharia Elétrica, pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Paraná, pela seguinte banca examinadora:

Dr. Leandro dos Santos Coelho
Departamento de Engenharia Elétrica, UFPR

Dr. Alexandre Rasi Aoki
Departamento de Engenharia Elétrica, UFPR



Dr. Clodomiro Unsihuay Vila
Departamento de Engenharia Elétrica, UFPR

Dr. Roberto Zanetti Freire
Programa de Pós Graduação em
Engenharia de Produção e Sistemas, PUCPR

Curitiba, 28 de Fevereiro de 2013.

Dedicatória

Dedico esta dissertação a meus pais,
cujo exemplo de vida e dedicação tem sido um referencial importante;
Ao meu orientador,
por nortear meu crescimento pessoal e profissional.

Agradecimentos

O professor doutor Leandro dos Santos Coelho, pela orientação e incentivo.

O meu tio Otávio Luiz dos Santos, pela consideração, carinho e suporte sempre presentes durante estes anos.

O coordenador do curso de mestrado em Engenharia Elétrica da UFPR, professor doutor Evelio Martín García Fernández, pelo apoio sempre manifestado.

Aos membros da banca pelo tempo e empenho desprendidos afim de avaliar e aprimorar o trabalho.

A Agência Financiadora CAPES, pelo fomento disponibilizado.

Todos os colegas do mestrado em Engenharia Elétrica da UFPR.

Epígrafe

”Nothing is more pratical than a good theory”
Vladimir Vapnik, idealizador da máquina de vetor suporte.

Resumo

As fontes de energia eólica são reconhecidas por não emitir resíduos na atmosfera mas apresentam algumas outras questões ambientais que não podem ser negligenciadas, possuem benefícios sociais e são economicamente competitivas, o que indica um crescimento na aplicação desta tecnologia. A previsão da capacidade energética gerada em parques eólicos, sendo principalmente orientada a viabilidade de instalação de novos parques eólicos, gerenciamento de sistemas e planejamento de manutenções, é de interesse dos operadores do sistema e companhias de energia elétrica. O objetivo geral deste trabalho é levantar o desempenho da máquina de vetor suporte otimizada pela evolução diferencial na identificação das séries de ventos, para avaliar a viabilidade do uso destas técnicas na previsão a curto prazo da geração de energia elétrica proveniente destas fontes eólicas. A técnica a ser aplicada neste trabalho, a máquina de vetor suporte à mínimos quadrados (LS-SVM, do inglês *Least squares support vector machine*). Uma vertente da máquina de vetor suporte original que substitui as inequações da teoria original por equações, torna o método atrativo computacionalmente.Buscando um refinamento do processo de aprendizado, a evolução diferencial é utilizado em conjunto ao de regressão. Sabendo que em sua essência, otimizar é maximizar uma propriedade desejada do sistema enquanto simultaneamente minimiza uma característica indesejável. O que são estas propriedades e quão efetivamente podem ser melhoradas depende do problema em questão. Dentre diversas opções existentes, dadas as características da otimização a ser realizada, optou-se por uma vertente da evolução diferencial (ED) idealizada por Price e Storn, pesquisadores da Universidade de Berkeley, que a desenvolveram para ser um otimizador versátil, confiável e eficiente. Esta vertente é dita alto adaptativa por otimiza seus próprios parâmetros durante as iterações da ED e é conhecida SADE, do inglês *Self-adaptive Differential Evolution*. Para a realização dos testes com os algoritmos foram utilizadas Séries temporais de ventos reais, medidas pelo (*Research Laboratory of Renewable Energy* ,RERL) em três localidades distintas nos Estados Unidos da America. Neste contexto, as séries adotadas foram as seguintes: Barnstable, Orleans e Paxton. Os testes apontam bons resultados para uma previsão um passo a frente mas não obteve um bom resultado para uma previsão N passos a frente mas é mostrado que com alguns ajustes das entradas e uma melhor análise da correlação do erro esse algoritmo tem potencial para vir a ser aplicado na identificação e previsão de séries de ventos. Alguns fatores devem ser observados quanto a identificação de sistemas em geral, como o fato do modelo matemático ser sempre uma representação aproximada. Portanto não existe um modelo único e ideal para um sistema e sim famílias de modelos com características e desempenhos variados.

Palavras-chave: Séries de ventos, LS-SVM, Metaheurísticas, Evolução diferencial, Identificação, Otimização, Máquina de Vetor Suporte.

Abstract

Electricity generation from wind sources has been recognized as environmentally friendly, socially beneficial and economically competitive, what increases the application of this technology. The predicted energy capacity generated in wind farms is primarily oriented according to the feasibility of the installation of new wind farms, systems management and maintenance planning, and also according to the interest of the system operators and power companies. The analysis and time series forecasting aim to understand, define and exploit the statistical dependence on sequentially sampled data. A goal is to be able to obtain some information about the time series, in order to perform the prediction of future values. A strand from the concepts of machine learning and statistical learning is the support vector machines and their variations, among them, the technique to be applied in this work: Least squares support vector machine (LS-SVM). One variant of support vector machine that replaces the original inequalities from the original theory with equalities, making the method computationally attractive. Besides looking for a refinement of the learning process, an optimization algorithm is used in addition to the regression. Knowing that in essence, to optimize is to maximize a desired property of the system, while simultaneously minimizing an undesirable characteristic. What these properties are and how effectively they can be improved depends on the problem at hand. Among a wide range of well known optimization algorithms, given the characteristics of the optimization to be performed, we opted for the algorithm developed by Price and Storn, called differential evolution. This optimizer was developed to be a versatile, reliable and also efficient one. Considering the unpredictable nature of the wind series, the study of heuristics and metaheuristics focused on these series prediction, becomes vital for the development and implementation of new technologies. For the tests with the algorithms we used time series of real winds, measured by the Research Laboratory of Renewable Energy (RERL), in three different locations in the United States: Barnstable, Orleans and Paxton. Some factors should be observed for the identification of systems in general, for example, the fact that the mathematical model is always an approximate representation. Therefore there is no ideal model for a system, but families of models varying in features and performances.

Key-words: Wind series, LS-SVM, Metaheuristics, Differential Evolution, Identification, Optimization, Support Vector Machine.

Lista de Figuras

Lista de Tabelas

Lista de Siglas

$\varepsilon - SV$	*Support vector regression based on* ε*-insensitive*
ED	Evolução diferencial
ICEO	*International Contest on Evolutionary Optimization*
IEEE	*Institute of Electrical and Electronics Engineers*
KKT	*Karush-Kuhn-Tucker*
LS-SVM	*Least squares support vector machine*
ME	Erro médio
MEA	Erro médio absoluto
MLP	*Multilayer perceptron*
MSE	Erro médio quadrático
RBF	*Radial basis function*
RERL	***Research laboratory of renewable energy*** do centro de energia eólica da universidade de Massachusetts
RMSE	Raiz Erro médio quadrático
SADE	*Self-adaptive diferencial evolution*
SVM	*Support vector machine*

Lista de Símbolos

x	Instância
y	Etiqueta $\in [-1,1]$
X	Espaço de entrada
Y	Espaço de saída $\in [-1,1]$
P	Distribuição desconhecida
m	Número de pares (X_i, Y_i)
g	Função que prevê Y a partir de X
$R(g)$	Risco de g
t	função alvo
$R_n(g)$	Risco empírico
g_n	Função retornada pelo algoritmo de aprendizagem
G	Geração da ED
λ	Parâmetro regularizador
D	dados de treinamento
n	Dimensões de x
$F(\cdot)$	Função de classificação
w	Vetor de pesos
b	bias ou polarização
$Q(\cdot)$	Função a ser minimizada na etapa de treinamento da SVM
ε_i	Variáveis de folga
C	Parâmetro que determina o compromisso entre a largura da margem e o número de erros de medição
Φ	Função de mapeamento não linear do espaço original para um espaço característico de alta dimensionalidade
α	Multiplicadores de Lagrange
β	Multiplicadores de Lagrange

ξ	Variáveis de folga
$W(\cdot)$	Problema dual
$K(\cdot)$	Função *Kernel*
γ	Parâmetro que determina a relação entre o ajuste da função aos dados de entrada e a capacidade de generalização do modelo
η	Multiplicadores de Lagrange
NP	Tamanho da população da ED
x_best, G	Melhor vetor de parâmetros da geração G
v	Vetor mutante
x_r	Vetores escolhidos aleatoriamente a partir do intervalo $[0, NP-1]$
F	Fator que controla a amplificação da variação diferencial
u	Vetor experimental
CR	Constante de *crossover* $\in [0,1]$
p_t	valor observado
p	valor previsto

Sumário

1 Introdução

A questão energética é um tópico de estudo muito importante na atualidade pois a qualidade de vida da sociedade esta intimamente relacionada ao consumo de energia elétrica. E o crescimento da demanda energética em países em desenvolvimento, como o Brasil, em razão da melhora na qualidade de vida fazem com que a segurança no suprimento de energia e os custos ambientais para atender esse crescimento sejam alguns dos aspectos que vem ganhando notoriedade na política de planejamento energético de todas as economias emergentes. A inserção de fontes renováveis de energia, como a eólica, aparece com o intuito de minimizar esta questão e vem crescendo em aplicação e pesquisa nos últimos anos [1].

A geração de eletricidade proveniente de fontes eólicas embora reconhecida como amigáveis ao meio ambiente do ponto de vista da emissão de substâncias nocivas à atmosfera apresenta alguns aspectos ambientais que não podem ser negligenciados como, por exemplo, a influência do ruído produzido pelos geradores em animais como os pássaros [2]. Essa tecnologia possui ainda benefícios sociais e é economicamente competitiva. [2]. Contudo, um crescimento na aplicação desta tecnologia é evidente e seu potencial energético é grande. A natureza imprevisível das séries de ventos torna complexa a tarefa de realizar o estudo da viabilidade econômica para a implementação de novos parques eólicos, o que torna a previsão de ventos vital para o desenvolvimento e inclusão desta tecnologia [3].

A previsão das séries temporais de ventos tem, geralmente, como propósito a previsão da capacidade energética gerada em parques eólicos sendo principalmente orientada à viabilidade de instalação de novos parques eólicos, gerenciamento de sistemas e planejamento de manutenções, sendo de interesse dos operadores do sistema e companhias de energia elétrica. Nos últimos anos surgem as primeiras integrações efetivas entre duas linhas de pesquisas bem complementares, a matemática e os fenômenos físicos envolvendo os ventos [4] [3].

Para solucionar o problema da previsão de séries temporais muitos métodos foram desenvolvidos, como aqueles baseados em modelos estatísticos, no aprendizado estatístico ou em sistemas *fuzzy* [5], [6], [7]. Seguindo a vertente conhecida como Aprendizado de máquina [8], um campo da Inteligência Artificial que se ocupa do desenvolvimento de técnicas e métodos capazes de dotar um computador da habilidade de aprender, surgem

as soluções provenientes da teoria de aprendizado estatístico conhecidas como máquinas de vetor suporte divulgadas pela primeira vez em 1992, introduzidas por Boser, Guyon e Vapnik [9] que compreendem um conjunto de métodos de aprendizado supervisionados usados para classificação e regressão [8].

A máquina de vetor suporte foi originalmente desenvolvida para classificação [10] e logo foi estendida para aplicações de regressão [11]. A forma inicial da SVM é um classificador binário, separa os dados em duas categorias, onde a saída da função alvo resulta em um número positivo ou negativo. Cada dado é representado por um vetor de n dimensões, e cada um destes dados pertence a apenas uma das duas classes.

A fim de aprimorar os resultados obtidos com o método aplicado a regressão foi proposto o uso conjunto de um algoritmo de otimização, características do espaço de busca como a de poder ser diferenciável tornaram possível e confiável a aplicação da evolução diferencial (ED) como algoritmo otimizador. Price e Storn desenvolveram a ED para ser um otimizador versátil e confiável e eficiente para otimização contínua. A primeira publicação da ED surgiu como um relatório técnico em 1995 [12] desde então a ED tem se destacada em eventos como na competição internacional de algoritmos evolutivos da IEEE (ICEO, *International Contest on Evolutionary Optimisation*) nos anos de 1996 e 1997 além de uma vasta aplicação em problemas reais [13].

1.1 Objetivos

Os objetivos desta pesquisa foram divididos em geral e específicos, os quais são sumariados nas subseções a seguir.

1.1.1 Objetivo geral

Avaliar o desempenho da máquina de vetor suporte otimizada pela evolução diferencial na identificação de séries de ventos, para previsão de geração de energia eólica em um horizonte de previsão de curto prazo. Validando a aplicação do métodos em conjuntos de dados eólicos reais.

1.1.2 Objetivos específicos

Visando alcançar o objetivo geral, os seguintes objetivos específicos foram traçados:

- Implementar a máquina de vetor suporte para identificar as séries de ventos e possibilitar a previsão da energia gerada por esta série a curto prazo;
- Implementar a evolução diferencial para aprimorar os parâmetros da máquina de vetor suporte buscando uma identificação mais confiável;
- Aplicar as metaheurísticas citadas nos tópicos anteriores aos conjuntos de dados reais de ventos obtidos junto ao RERL;
- Analisar os resultados obtidos dos experimentos.

1.2 Estrutura da dissertação

O restante desta dissertação está organizada da seguinte maneira.

O capítulo 2 denominado *Aprendizado de máquina* detalha a concepção e os fundamentos da máquina de vetor suporte (SVM) que compreende um conjunto de métodos de aprendizado supervisionados usados para classificação e regressão.

O capítulo 3 denominado *Evolução Diferencial (ED)* apresenta a metaheurística de otimização proposta para trabalhar com os problemas que envolvem otimização global sobre espaços contínuos e que dizem respeito, em geral, a tarefa de otimizar certos parâmetros de um sistema da maneira pertinente.

O capítulo 4 denominado *Materiais e métodos* detalha os componentes utilizados, a metodologia aplicada. Além disso, também discorre brevemente sobre as séries de ventos e a iteração entre os algoritmos propostos.

O capítulo 5 denominado *Características gerais e discussões* apresenta os resultados obtidos e discorre sobre as aplicações do método proposto, o que compreende a previsão de ventos principalmente orientada à viabilidade de instalação de novos parques eólicos, gerenciamento de sistemas e planejamento de manutenções.

Finalmente, o capítulo 6 denominado *Conclusão*, encerra o documento desta dissertação, retomando e resumindo os objetivos do projeto e os resultados obtidos com uma breve análise sobre a aplicação das metaheurísticas ao problema de previsão.

2 Aprendizado de máquina

Aprendizado de máquina é considerado uma área da Inteligência Artificial que se ocupa do desenvolvimento de técnicas e métodos capazes de dotar um computador da habilidade de aprender [8]. O processo de aprendizado e inferência indutiva podem ser sintetizados pelo seguinte procedimento:

1. Observar o fenômeno;
2. Construir um modelo deste fenômeno;
3. Fazer predições utilizando o modelo.

Sendo esta definição geral, ela pode ser adotada como o objetivo da ciência natural. O objetivo da máquina de aprendizado é, mais especificamente, automatizar este processo e o objetivo da teoria de aprendizado é formalizar estes conceitos.

2.1 Teoria de aprendizado estatístico

O objetivo da teoria de aprendizado estatístico é prover uma estrutura para o estudo do problema de inferência, ganhar conhecimento, fazer predições, tomar decisões ou construir modelos a partir de conjuntos de dados, ou seja, são considerações de natureza estatística sobre fenômenos latentes.

A teoria de inferência está apta a dar uma definição formal de palavras, tais como "aprendizado", "generalização", "*overfitting*", e tudo o que pode caracterizar o desempenho dos algoritmos de aprendizado, por este motivo tem sido utilizada para ajudar no desenvolvimento destes algoritmos [14].

2.1.1 Aprendizado e inferência

Para fins introdutórios, será considerado o caso particular em que a estrutura de aprendizado supervisionado é utilizada em reconhecimento de padrões. Nesta estrutura, os dados são constituídos de pares, instância x e etiqueta y, onde a etiqueta y corresponde

a um dos valores [-1,1]. O algoritmo de aprendizagem constrói uma função de mapeamento das instâncias x em etiquetas y, válida também para conjuntos de dados desconhecidos.

Possuindo alguns dados de treinamento, sempre é possível construir uma função que descreve exatamente o conjunto de dados. Porém, na presença de ruído, esta pode não ser a melhor opção, podendo acarretar na diminuição da desempenho da função para dados desconhecidos, usualmente referido como *overfitting* da função aos dados de entrada. A idéia geral do modelo dos algoritmos de aprendizado é seguir o conceito de "regularidade" na observação do fenômeno. Podendo então ser generalizado de um passado observado para um futuro [14], [15].

Quando analisada a coleção de possíveis modelos que descrevem o fenômeno, tipicamente busca-se o modelo que descreve bem os dados com a menor complexidade possível. O que incita questionamentos de como medir e quantificar a simplicidade de um modelo. Muitos métodos foram desenvolvidos para este fim, mas nenhum tido como ideal. Freqüentemente pode-se utilizar como indicador de complexidade o tamanho do descritivo do modelo em linguagem de codificação. Por exemplo em Física, existe uma tendência a selecionar modelos com um número reduzido de constantes o que corresponde a um modelo matemático mais simples.

Em Estatística clássica, o número de parâmetros livres de um modelo é medida de simplicidade e a escolha de uma medida específica depende do problema a ser tratado, ou seja, a seleção do modelo esta atrelado ao conhecimento *a priori* sobre o fenômeno em estudo.

Esta carência pela opção ideal pode ser formalizada no que conhece-se como "*No free lunch theorem*", que em sua essência descreve o fato de que sem o conhecimento de como o passado, isto é dados de treinamento, é relacionado ao futuro, isto é, dados desconhecidos, a previsão torna-se inviável. E mais, se não existirem restrições *a priori* no possível fenômeno é esperado que seja impossível fazer a generalização e encontrar o melhor algoritmo.

Dai a necessidade de fazer considerações, como o fato do fenômeno observado poder ser descrito por um modelo simples. Contudo, a simplicidade não é uma noção absoluta por tanto dados não substituem conhecimento, ou em termos pseudo-matemáticos [14], [16]:

$$\text{GENERALIZAÇÃO} = \text{DADOS} + \text{CONHECIMENTO } \textit{A PRIORI}$$

2.1.2 Considerações sobre o modelo

Para fazer considerações precisas sobre o modelo, inicialmente deve-se assumir que o futuro observado (dados desconhecidos) esta relacionado ao passado (dados de teste). Com isso o fenômeno é dito suficientemente estacionário.

O núcleo dessa teoria é a premissa da existência de um modelo probabilístico do fenômeno, com esse modelo probabilístico a relação entre passado e futuro observado é de que são ambos amostras independentes da mesma distribuição.

A consequência imediata ao assumir tais premissas, de maneira genérica, é que o algoritmo será mais consistente, ou seja, quanto maior for o número de dados obtidos, a previsão do algoritmo ficará mais próxima do ótimo. Infelizmente, todos os algoritmos ditos consistentes podem ter um comportamento arbitrário quando somente um conjunto finito de dados de treinamento encontra-se disponível.

O que indica que a generalização apenas pode vir com uma adição de conhecimento específico sobre o conjunto de dados. Cada algoritmo de aprendizado incorpora um conhecimento, ou consideração, específico sobre as características do classificador ótimo. E funciona melhor quando esta consideração é realizada com base em características do fenômeno em questão [14, 16].

2.1.3 Formalização

Considerando um espaço de entrada X e um espaço de saída Y, restringindo-os a uma classificação binária, e escolhendo Y = [-1,1]. E sendo uma distribuição desconhecida P [15]. Observando-se a sequência de m pares (X_i,Y_i) amostrados de acordo com P e com o objetivo de construir uma função g:$X \rightarrow Y$ que faça a previsão de Y a partir de X. Necessita-se de um critério para a escolha da função g. Este critério é dado pela menor probabilidade de erro $P(g(X) \neq Y)$. Define-se assim o "Risco" de g, tal que

$$R(g) = P(g(X) \neq Y) \tag{2.1}$$

O objetivo é identificar a função alvo, t, ou classificador de Bayer. Função esta que minimiza o risco. Mas sendo P desconhecido, fica inviável medir o risco diretamente e, conseqüentemente, determinar t precisamente para o conjunto de dados. Apenas é possível medir a aceitação de uma função candidata para o conjunto de dados e esta incerteza é denominada "Risco Empírico" [14, 15]:

$$Rn(g) = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}_{g(X_i)\neq Y_i}. \tag{2.2}$$

É usual a escolha deste parâmetro como critério para seleção da estimativa para a função desejada [17]. Com isso pode-se definir as estratégias mais comummente adotadas para definir a função alvo, de maneira aproximada, dentre as geradas pelos algoritmos desenvolvidos. Assim, define-se g_n como a função retornada pelo algoritmo de aprendizado.

Por não ser possível computar $R(g)$, mas apenas aproximá-la por $R_n(g)$, torna-se impossível tratar da minimização de $R_n(g)$, pois em um espaço de dimensão infinita sempre será possível construir uma função g_n. Esta função prevê perfeitamente os rótulos Y_i para o conjunto de treinamento, mas que para outros pontos terá um comportamento oposto ao da função Alvo. Desta forma torna-se necessário prever esta situação de $overfitting$.

2.1.4 Algoritmos

Os algoritmos são responsáveis por gerar diferentes funções g para serem testadas a fim de determinar a função que melhor descreve os dados de entrada, minimizando os riscos. Por não ser possível computar a aceitação de g para todos os resultados possíveis do fenômeno em estudo mas apenas para uma região amostral, os dados de entrada, analisar a influência do Risco Empírico pode gerar um sobre ajuste da função, ou seja, a função g que melhor descreve os dados de entrada, região amostral, possui uma baixa capacidade de generalização com um comportamento insatisfatório aos dados desconhecidos.

Para contornar o problema do Risco Empírico há essencialmente duas maneiras para proceder: a primeira é restringir as classes de funções em que a minimização será realizada e a segunda é modificar o critério a ser minimizado, sob a pena de tornar a função mais complexa. A seguir são apresentadas as características dos algoritmos baseados de acordo com o foco do critério a ser otimizado.

Minimização do Risco Empírico

Os algoritmos baseados na minimização do Risco Empírico são os mais diretos e geralmente eficiente. Tem como base a premissa de escolher dentre as possíveis funções g a que minimiza o Risco Empírico do modelo [15]. Contudo são mais suscetíveis ao sobre ajuste da função, tal que

$$g_n = \underset{g \in G}{argmin}\ \mathrm{R}_n(g). \tag{2.3}$$

Este método funciona melhor se a Função alvo pertencer a G. Contudo é raro ter esta certeza, então há uma tendência de aumentar o modelo ao máximo possível para evitar o $overfitting$ da função [14, 16].

Minimização do Erro Estrutural

Consiste em idealizar a escolha de uma sequência infinita das possíveis funções $gi : i$=1,2,3,... que aumentam gradualmente a complexidade da função, minimizando o Erro Empírico e adicionam uma penalidade pelo tamanho do modelo. Dando preferência assim aos modelos com um erro de estimação menor, ou seja, com melhor capacidade de generalização da função aos dados desconhecidos e tentando minimizar o tamanho do modelo [14, 15]

$$g_n = \underset{g \in G_d, d \in N}{argmin}\ \mathrm{R}_n(g) + pen(d,m). \tag{2.4}$$

Estes algoritmos são menos diretos e tendem a dar preferência a modelos onde a estimação do erro seja pequeno e as medidas de tamanho, ou capacidade, do modelo sejam menores, graças a adição da penalidade $pen(d,m)$ [14, 16].

Regularização

De fácil implementação, esta aproximação consiste em escolher o maior e mais complexo modelo g procurando definir este modelo de maneira a regularizar a função, tipicamente utilizando a norma $\|g\|$. Então minimizando o Risco Empírico regularizado, tal que

$$g_n = \underset{g \in G_d, d \in N}{argmin}\ \mathrm{R}_n(g) + \lambda ||g||^2. \tag{2.5}$$

Quando comparado a minimização do Risco Empírico nota-se o surgimento de um parâmetro livre λ, denominado parâmetro regularizador, que permite a correta escolha entre adequação e complexidade do modelo. Sintonizar λ é um problema difícil e que freqüentemente exige dados de validação extras para o modelo de maneira a executar esta tarefa [14].

2.2 Máquina de vetor suporte - SVM

A máquina de vetor suporte (SVM) foi proposta em 1992, introduzida por Boser, Guyon e Vapnik [9] e compreende um conjunto de métodos de aprendizado supervisionados usados para classificação e regressão [8]. Pertencente a uma família de classificadores lineares generalizados. Em outras palavras a máquina de vetor suporte (SVM) é uma ferramenta de classificação, regressão e previsão que utiliza a teoria do aprendizado de máquina para maximizar a precisão enquanto automaticamente evita o *overfitting* da função aos dados de entrada. Pode também ser definida como um sistema que utiliza a hipótese de uma função ser linearmente separável em um espaço de alta dimensionalidade, dotado da capacidade de mapear os dados para este espaço característico e treinado a partir de algoritmos de aprendizagem derivados da teoria de aprendizado estatístico.

A abordagem da SVM utiliza o princípio da minimização do risco estrutural, o que tem se mostrado superior ao tradicional princípio da minimização do risco empírico [10], dotando a SVM de uma promissora capacidade de generalização, habilidade esta que é o principal objetivo do aprendizado estatístico [8].

2.2.1 SVM para classificação de dados

A máquina de vetor suporte foi originalmente desenvolvida para classificação [10] e logo foi estendida para aplicações de regressão [11]. A forma inicial da SVM é de um classificador binário, separa os dados em duas categorias onde a saída da função alvo resulta em um número positivo ou negativo. Cada dado é representado por um vetor de n dimensões e cada um destes dados pertence a apenas uma das duas classes. Um classificador linear faz a separação através de um hiperplano, por exemplo, a Figura 2.1 mostra dois grupos de dados distintos, representados por quadrados e círculos e alguns dos possíveis hiperplanos separadores representados pelas retas que separam os conjuntos[18].

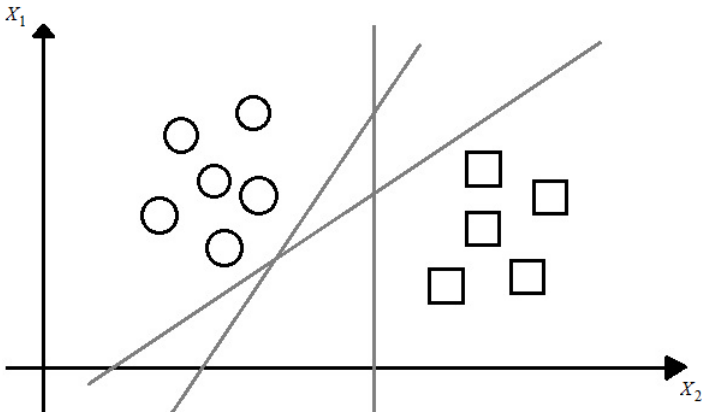


Figura 2.1: Possíveis separadores lineares

Fonte: *Adaptado de* [18]

Existem muitos separadores lineares que classificam os dados. E para selecionar um destes hiperplanos como ideal a SVM utiliza um conceito denominado máxima margem. Optando pelo separador linear que mais se distancia dos dois grupos existentes, esta margem pode ser descrita como a distância entre o hiperplano separador e os pontos mais próximos a ele de cada categoria. Tal hiperplano, ou máxima margem, tem melhor capacidade de generalização de dados desconhecidos. Para lidar com um conjunto de dados que não seja linearmente separável a SVM faz o mapeamento de um espaço de entrada para um espaço característico de alta dimensionalidade onde, neste espaço, os dados tornam-se linearmente separáveis. Aplicando a transformação inversa na equação do hiperplano encontrado no espaço característico, tem-se uma função não linear que separa os dados no espaço de entrada. Contudo, ao realizar este mapeamento para um espaço característico de alta dimensionalidade, operações básicas e recorrentes na SVM, como o produto interno, tornam-se complexas e custosas computacionalmente. Para contornar este problema a máquina de vetor suporte lança mão de um artifício matemático que utiliza funções conhecidas, e com um menor custo computacional, para inferir o cálculo do produto interno no espaço característico. Estas funções são ditas ***Kernels***.

Classificador de margem rígida

Para melhor entender como a SVM calcula o hiperplano de máxima margem e suporta classificadores não lineares, é importante ter a noção do funcionamento do algoritmo de margem rígida onde o conjunto de entrada é livre de ruído e pode ser corretamente classificado por uma função linear. O conjunto de dados $\boldsymbol{D}$, ou dados de treinamento, podem ser descrito matematicamente como sendo

$$D = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\} \tag{2.6}$$

onde x_i é um vetor real de n dimensões, y_i pode ser 1 ou -1 denotando a classe a qual x_i pertence, e sendo m o tamanho do conjunto de dados. A função de classificação $F(x)$ tem a forma

$$\begin{aligned} & F(x) = w \cdot x - b \\ & \text{Sujeito a} \qquad \begin{aligned} & w \cdot x_i - b < 0 \ se \ y_i = -1 \\ & w \cdot x_i - b > 0 \ se \ y_i = +1 \end{aligned} \end{aligned} \tag{2.7}$$

onde w é um vetor de pesos e b é a polarização, o que é computada pelo SVM durante o processo de treinamento. Estas condições podem ser revistas tal que:

$$y_i(w \cdot x_i - b) > 0, \forall (x_i, y_i) \in D \tag{2.8}$$

Se existir uma função linear F que classifique corretamente cada ponto em D, então D é dito linearmente separável. Tendo n hiperplanos que dividam os conjuntos satisfatoriamente, considera-se então a restrição da máxima margem e, para atingir esta restrição, a equação pode ser revista como:

$$y_i(w \cdot x_i - b) \geq 1, \forall (x_i, y_i) \in D \tag{2.9}$$

A distância de um hiperplano a um vetor x_i é formulado como $\frac{|F(x_i)|}{\|w\|}$. Logo, a margem vem a ser

$$margem = \frac{1}{\|w\|} \tag{2.10}$$

Para facilitar o entendimento os componentes e medidas relevantes a delimitação do hiperplano separador podem ser vistas na Figura 2.2.

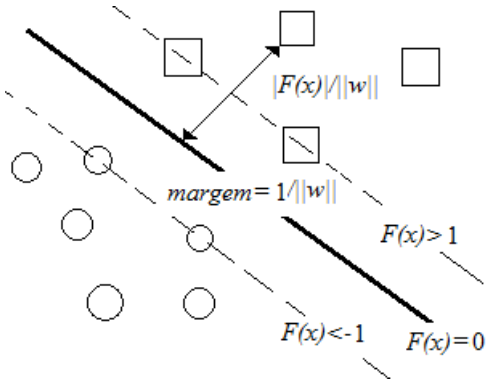


Figura 2.2: Função de classificação da SVM: o hiperplano maximizando a margem no espaço bidimensional

Fonte: *Adaptado de* [18]

Os valores de x_i mais próximos a função $F(x)$ terão como resultado 1 de acordo com a eq. 2.9. Estes vetores que satisfazem a equação 2.9 são ditos vetores de suporte. Maximizar a margem do hiperplano separador pode então ser descrito como uma minimização de $||w||$. Portanto, o problema a ser trabalhado na etapa de treinamento da SVM é

$$\begin{aligned} &\text{minimizar} && Q(w) = \tfrac{1}{2}||w||^2 \\ &\text{sujeito a} && y_i(w \cdot x_i - b) \geq 1, \forall (x_i, y_i) \in D \end{aligned} \tag{2.11}$$

Neste contexto o fator $\frac{1}{2}$ é utilizado por uma conveniência matemática.

Classificador de margem relaxada

O foco do problema em questão continua por ser os casos linearmente separáveis. Contudo, o problema de otimização descrito pela eq. 2.11 não terá solução se D não for linearmente separável. Para lidar com os casos onde isto não acontece, onde aparecem erros de medição e ruído no conjunto de dados de entrada, o classificador de margem relaxada consegue maximizar a margem. Para tal, o método introduz a noção de variáveis de folga, ε_i que mede o grau dos erros de classificação no conjunto de dados. Apresenta-se assim o problema de otimização para classificadores de margem relaxada, onde

$$\begin{aligned} &\text{minimizar} && Q(w, b, \varepsilon_i) = \tfrac{1}{2}||w||^2 + C\textstyle\sum_i \varepsilon_i \\ &\text{sujeito a} && y_i(w \cdot x_i - b) \geq 1 - \varepsilon_i, \forall (x_i, y_i) \in D \\ & && \varepsilon_i \geq 0 \end{aligned} \tag{2.12}$$

Devido a adição das variáveis de folga ε_i na eq. 2.12, dados com erros de medição e ruído podem ser trabalhados e minimizados enquanto acontece a maximização da margem, considerando C um parâmetro que determina o compromisso entre a largura da margem e o número de erros de medição admitidos [11].

A solução do problema de otimização da equação 2.12 é dada pelo ponto de sela do Lagrangiano

$$\Phi(w,b,\alpha,\xi,\beta) = \frac{1}{2}\|w\|^2 + C\sum_i \xi_i - \sum_{i=1}^{l} \alpha_i(y^i[w^T x^i + b] - 1 + \xi_i) - \sum_{j=1}^{l} \beta_j \xi_j \tag{2.13}$$

onde α e β são os multiplicadores de Lagrange. O Lagrangiano deve ser minimizado com relação w, b e x e maximizado em relação a α e β. A dualidade do Lagrangiano clássico permite a transformação do problema primal dado na eq. 2.13 no problema dual dado por

$$\max_{\alpha} W(\alpha,\beta) = \max_{\alpha,\beta} (\min_{w,b,\xi} \Phi(w,b,\alpha,\beta,\xi)) \tag{2.14}$$

Tal que o mínimo em relação a w,b e ξ do lagrangiano, Φ, é dado por

$$\begin{aligned} &\frac{\partial \Phi}{\partial b} = 0 \Rightarrow \textstyle\sum_{i=1}^{l} \alpha_i y_i = 0 \\ &\frac{\partial \Phi}{\partial w} = 0 \Rightarrow w = \textstyle\sum_{i=1}^{l} \alpha_i y_i x_i \\ &\frac{\partial \Phi}{\partial \xi} = 0 \Rightarrow \alpha_i + \beta_i = C. \end{aligned} \tag{2.15}$$

Então das equações 2.13, 2.14 e 2.15 o problema dual passa a ser,

$$\max_{\alpha} W(\alpha) = \max_{\alpha} -\frac{1}{2}\sum_i^l \sum_j^l \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle + \sum_{k=1}^{l} \alpha_k \tag{2.16}$$

e a solução do problema é dada por,

$$\alpha^* = \operatorname*{argmin}_{\alpha} \frac{1}{2}\sum_i^l \sum_j^l \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle - \sum_{k=1}^{l} \alpha_k \tag{2.17}$$

com as seguintes restrições,

$$\begin{aligned} &0 \leq \alpha_i \leq C \quad i = 1,...,l \\ &\textstyle\sum_{j=1}^{l} \alpha_j y_j = 0 \end{aligned} \tag{2.18}$$

2.2.2 *Kernels*

Quando o conjunto de dados não é linearmente separável, nos casos em que não existe um hiperplano reto que realize a separação das classes, a SVM linear deve ser estendida a sua forma não linear para assim poder realizar o aprendizado destas funções, permitindo a classificação de dados linearmente não separáveis. O processo de encontrar uma função de classificação usando uma SVM não linear consiste em dois passos: primeiro os vetores de entrada são transformados em vetores característicos de alta dimensionalidade onde os dados de treinamento podem então ser linearmente separáveis. E um segundo passo onde a SVM busca o hiperplano de máxima margem neste novo espaço característico. O hiperplano separador torna-se uma função linear no espaço característico transformado mas uma função não linear no espaço de entrada original. Seja x um vetor de n dimensões no espaço de entrada e $\Phi(\cdot)$ uma função de mapeamento não linear do espaço original para um espaço característico de alta dimensionalidade, o hiperplano representando o limiar de decisão no espaço característico é definido como

$$w \cdot \Phi(x) - b = 0 \tag{2.19}$$

onde w denota o vetor de pesos que permite mapear os dados de treinamento no espaço característico de alta dimensionalidade, e b é o bias da função. Nota-se que na função de mapeamento no problema de otimização e também na função de classificação sempre aparece o produto interno entre pares de vetores no espaço característico transformado. Calcular o produto interno neste espaço característico pode vir a ser um problema complexo e dispendioso computacionalmente dada a dimensão do problema. Para evitar este problema, as funções ***Kernels*** são utilizadas. A função ***Kernel***(K), no espaço original, substitui o cálculo do produto interno no espaço característico

$$K(u,v) = \Phi(u) \cdot \Phi(v) \tag{2.20}$$

sendo u e v vetores no espaço original e $\Phi(\cdot)$ a função de transformação não linear para o espaço característico de alta dimensionalidade. Para que determinada função seja uma função ***Kernel*** válida ela deve atender aos princípios básicos do teorema de Mercer [19]. Este teorema garante que uma determinada função válida pode ser representada como produto interno de pares de vetores em algum espaço de alta dimensionalidade, portanto o produto interno pode ser calculado utilizando a função ***Kernel*** apenas com os vetores de entrada no espaço original, sem a necessidade de aplicar qualquer transformação nestes

vetores a fim de aumentar sua dimensionalidade. A seguir apresenta-se uma lista das funções *Kernels* utilizadas:

- *Leave-one-out*;
- *Generalized cross-validate*;
- *V-fold cross-validation*.

Nota-se então que a função *Kernel* é um tipo de função de similaridade entre os dois vetores onde a função de saída é maximizada quando os dois vetores tornam-se equivalentes. Por isso, a SVM pode aprender uma função a partir de dados de qualquer formato além de vetores, desde que seja possível estabelecer a existência de uma função de similaridade entre qualquer par dos objetos de dados.

2.3 Máquina de vetor suporte à mínimos quadrados - LS-SVM

Uma variante da máquina de vetor suporte clássica (SVM) foi proposta por Suykens e Vandewalle (1999). Ainda que mantendo as mesmas características básicas e a mesma qualidade na solução encontrada quando comparada com sua predecessora, a máquina de vetor suporte à mínimos quadrados, LS-SVM (*Least squares support vector machine*) [20] considera restrições de igualdade no lugar das restrições de desigualdades na abordagem da máquina de vetor suporte clássica. Como resultado foi obtido um algoritmo que, independe da dimensão do conjunto de treinamento reduzem-se os problemas em aplicar o método para grandes conjuntos de dados. Além disso, diminuindo-se o custo computacional no treinamento e utilização da máquina de vetor suporte à mínimos quadrados [21]. Sendo assim, a solução provém da resolução de um conjunto de equações lineares ao invés da programação quadrática. Enquanto na SVM clássica muitos dos valores de suporte são iguais a zero (valores diferentes de zero indicam os vetores de suporte), na LS-SVM os valores de suporte são proporcionais aos erros [20].

2.3.1 Formulação primal-dual da LS-SVM

A estrutura padrão para a estimação da máquina de vetor suporte à mínimos quadrados é baseada em uma abordagem primal-dual. Dado o conjunto de dados $\{x_i, y_i\}_{i=1}^{N}$ a meta é estimar o modelo na forma,

$$y = w^T \Phi(x_i) + b \tag{2.21}$$

onde $x \in \mathbb{R}^n$, $y \in \mathbb{R}$ e $\Phi(.)\mathbb{R}^n \to \mathbb{R}^{n_h}$ é o mapeamento para um espaço característico de alta dimensionalidade. O seguinte problema de otimização é então formulado,

$$\begin{array}{ll} \underset{w,b,e}{min} & \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{i=1}^{n} e_i^2 \\ s.a. & y_i = w^T \Phi(x_i) + b + e_i,\ i = 1,...,N \end{array} \tag{2.22}$$

com a utilização das funções definidas *Kernels* não é necessário calcular explicitamente o mapeamento não linear $\Phi(.)$ Portanto do Lagrangiano temos,

$$L(w,b,e;\alpha) = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{i=1}^{N} e_i^2 - \sum_{i=1}^{N} \alpha_i (w^T \Phi(x_i) + b + e_i - y_i) \tag{2.23}$$

onde $\alpha_i \in \mathbb{R}$ são os multiplicadores de Lagrange e as condições de otimalidade são dadas por,

$$\begin{array}{l} \frac{\partial L}{w} = 0 \to w = \sum_{i=1}^{N} \alpha_i \Phi(x_i) \\ \frac{\partial L}{b} = 0 \to \sum_{i=1}^{N} \alpha_i = 0 \\ \frac{\partial L}{e_i} = 0 \to \alpha_i = \gamma e_i,\ i = 1,...,N \\ \frac{\partial L}{\alpha_i} = 0 \to y_i = w^T \Phi(x_i) + b + e_i \end{array} \tag{2.24}$$

O modelo da LS-SVM resultante no espaço dual torna-se então

$$y(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i) + b. \tag{2.25}$$

Normalmente o treinamento do modelo da máquina de vetor suporte à mínimos quadrados involve uma seleção ótima dos parâmetros de ajuste α, parâmetro da função *Kernel*. E γ, parâmetro que provém uma relação entre adequação aos dados de entrada e capacidade de generalização do modelo [22].

2.4 Máquina de vetor suporte aplicada à regressão

Supondo um conjunto de dados de treinamento $\{(x_1,y_1),...,(x_l,y_l)\} \subset X \cdot \mathbb{R}$, onde X denota o espaço dos padrões de entrada ($ex. X = \mathbb{R}^d$). Isto significa, por exemplo, as taxas de câmbio de alguma moeda medidas em dias consecutivos juntamente com seu

respectivo indicadores econômico[23]. Na ε-SV regression [16], a meta do algoritmo é determinar uma função $f(x)$ que tenha no máximo um desvio ε de todos os reais valores de y_i para todo o conjunto de treinamento, ao mesmo tempo sendo esta função o mais plana e achatada possível. Em outras palavras os erros não são relevantes desde que sejam menores que ε, mas não serão aceitos desvios maiores que este. Isto pode vir a ser importante, por exemplo, se for pretendido ter a certeza de não perder mais de ε quando se aplica em taxas de câmbio da moeda [11].

Para fins de melhor compreensão será feito o descritivo do caso com uma função linear f, onde esta assume a forma

$$f(x) = \langle w, x \rangle + b \tag{2.26}$$

onde $w \in X$, $b \in \mathbb{R}$, e $\langle ., . \rangle$ indica o produto interno em X. O achatamento da função descrita em 2.26 é feito através de uma busca pelo menor w. Uma das maneiras de se fazer isso é através da minimização da norma de w, isto é $||w||^2 = \langle w, w \rangle$. Pode-se escrever este problema como um problema de otimização convexa dado por

$$\begin{aligned} &\text{Minimizar} && \tfrac{1}{2}||w||^2 \\ &\text{Sujeito a} && y_i - \langle w, x_i \rangle - b \leq \varepsilon \\ & && \langle w, x_i \rangle + b - y_i \leq \varepsilon \end{aligned} \tag{2.27}$$

É assumido na eq.2.27 que existe uma função f que aproxima todos os pares (x_i, y_i) com uma precisão ε, ou em outras palavras, a otimização convexa é possível. Algumas vezes, contudo, isto pode não ser o caso ou pode-se também permitir alguns erros. Análogo a SVM de margem relaxada, pode-se adicionar as variáveis de folga ξ_i, ξ_i^* para lidar com o caso da otimização convexa da função, eq.2.27, não ser possível. Formula-se então o problema como sendo

$$\begin{aligned} &\text{Minimizar} && \tfrac{1}{2}||w||^2 + C\textstyle\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\ &\text{Sujeito a} && y_i - \langle w, x_i \rangle - b \leq \varepsilon + \xi_i \\ & && \langle w, x_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \\ & && \xi_i, \xi_i^* \geq 0. \end{aligned} \tag{2.28}$$

A constante $C > 0$ determina o compromisso entre o achatamento de f e quanto

maior pode ser o desvio tolerado acima de ε. O que corresponde a lidar com a função custo $|\xi|_\varepsilon$ denominada ε-*insensitive* que pode ser descrita como

$$\begin{aligned} |\xi|_\varepsilon := \quad & 0, \quad se \quad |\xi| \leq \varepsilon \\ & |\xi| - \varepsilon, \quad caso \text{ contrário} \end{aligned} \tag{2.29}$$

Será então apresentada a abordagem dual do problema de otimização, dado que na maioria dos casos a otimização do problema proposto na eq.2.28 poder ser resolvido mais facilmente em uma abordagem dual. Esta abordagem permite a extensão da SVM para funções não lineares.

2.4.1 Problema dual

A idéia principal é construir uma função de Lagrange a partir da função objetivo original, denominada *primal*, e suas respectivas restrições através da introdução de um segundo conjunto de variáveis, *dual*. Pode ser demonstrado que esta função tem um ponto de sela referente as variáveis *primal* e *dual* da solução. Para detalhes consultar Mangasarian (1969). Neste caso,

$$\begin{aligned} & L := \tfrac{1}{2}||w||^2 + C\textstyle\sum_{i=1}^{l}(\xi_i + \xi_i^*) - \sum_{i=1}^{l}(\eta_i\xi_i + \eta_i^*\xi_i^*) \\ & -\textstyle\sum_{i=1}^{l}\alpha_i(\varepsilon + \xi_i - y_i + \langle w, x_i\rangle + b) \\ & -\textstyle\sum_{i=1}^{l}\alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle w, x_i\rangle - b) \end{aligned} \tag{2.30}$$

onde L é o Lagrangiano e $\eta_i, \eta_i^*, \alpha_i, \alpha_i^*$ são os multiplicadores de Lagrange. Por tanto as variáveis *dual* na eq.2.30 deve satisfazer as restrições de positividade, isto é

$$\alpha^{(*)_i}, \eta_i^{(*)} \geq 0 \tag{2.31}$$

note que por $\alpha_i^{(*)}$ esta sendo feita uma referência a α_i e α_i^*.

Isso vem da condição do ponto de sela, onde as derivadas parciais de L em relação as variáveis primais (w, b, ξ_i, ξ_i^*) devem ser iguais a zero para garantir a otimalidade

$$\partial_b L = \sum_{i=1}^{l}(\alpha_i^* - \alpha_i) = 0 \tag{2.32}$$

$$\partial_w L = w - \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) x_i = 0 \tag{2.33}$$

$$\partial_{\xi^{(*)}} L = C - \alpha_i^{(*)} - \eta_i^{(*)} = 0 \tag{2.34}$$

Substituindo as equações 2.32,2.33 e 2.34 na eq.2.30 temos o problema de otimização dual

$$\begin{aligned} \text{Maximizar} \quad & -\tfrac{1}{2}\textstyle\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\left\langle x_i, x_j \right\rangle \\ & -\varepsilon \textstyle\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*) \\ \text{Sujeito a} \quad & \textstyle\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0 \quad e \quad \alpha_i, \alpha_i^* \in [0, C] \end{aligned} \tag{2.35}$$

Derivando 2.35 elimina-se as variáveis *dual* η_i, η_i^* através da condição 2.34 pode-se reformular $\eta_i^{(*)} = C - \alpha_i^{(*)}$. A eq.2.33 pode ser então reescrita como,

$$w = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) x_i, \quad assim \quad f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) \langle x_i, x \rangle + b \tag{2.36}$$

sendo denominado vetor suporte expandido *Support Vector expansion*, isto é w pode ser completamente descrita como uma combinação linear dos conjuntos de treinamento x_i. Neste sentido, a complexidade da função de representação pelos vetores de suporte é independente da dimensionalidade do espaço de entrada X e dependente apenas do número de vetores de suporte. Além disso, o algoritmo completo pode ser descrito em função do produto interno dos dados. Mesmo avaliando a $f(x)$, torna-se necessário o cálculo explícito de w. Esta observação é tratada durante a abordagem da aproximação não linear do algoritmo.

Cálculo de b

O cálculo de b pode ser realizado baseado na condição de Karush-Kuhn-Tucker(KKT) [24] que define o produto interno entre as variáveis *dual* e as restrições como zero no ponto da solução

$$\begin{aligned} \alpha_i(\varepsilon + \xi_i - y_i + \langle w, x_i \rangle + b) &= 0 \\ \alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle w, x_i \rangle - b) &= 0 \end{aligned} \tag{2.37}$$

e

$$\begin{aligned}(C-\alpha_i)\xi_i &= 0\\ (C-\alpha_i^*)\xi_i^* &= 0\end{aligned} \tag{2.38}$$

Isso permite que várias conclusões sejam tiradas. Primeiramente apenas as amostras (x_i, y_i) com um $\alpha^{(*)} = C$ extrapolam o limite $\varepsilon - insensitive$. Segundo $\alpha_i \alpha_i^* = 0$, isto é, não pode existir um conjunto de variáveis *dual* onde α_i *e* α_i^* sejam simultaneamente diferentes de zero, o que permite concluir também que,

$$\varepsilon - y_i + \langle w, x_i \rangle + b \geq 0 \quad e \quad \xi_i = 0 \quad se \quad \alpha_i < C \tag{2.39}$$

$$\varepsilon - y_i + \langle w, x_i \rangle + b \leq 0 \quad se \quad \alpha_i > 0 \tag{2.40}$$

em conjunto com a análise similar sobre α_i^* tem-se

$$\begin{aligned}&max\{-\varepsilon + y_i - \langle w, x_i \rangle \,|\alpha_i < C \;\; ou \;\; \alpha_i^* > 0\} \leq b \leq\\ &min\{-\varepsilon + y_i - \langle w, x_i \rangle \,|\alpha_i > 0 \;\; ou \;\; \alpha_i^* < C\}\end{aligned} \tag{2.41}$$

se algum $\alpha^{(*)} \in (0, C)$ as inequações tornam-se equações.

2.4.2 Formulação não linear

Em uma próxima etapa configura-se o algoritmo de vetores de suporte em uma extensão não linear. O que pode ser feito, por exemplo, processando os conjuntos de treinamento x_i usando um mapeamento $\Phi : X \to F$ para algum espaço característico F. Como constatado previamente, o algoritmo de vetores de suporte apenas depende do produto interno entre os conjuntos x_i. Por isso é suficiente saber que $k(x, x') := \langle \Phi(x), \Phi(x') \rangle$ em vez de realizar o cálculo explícito de Φ. Assim, a abordagem do problema de otimização passa a ser,

$$\begin{aligned}\text{Maximizar} \quad & -\tfrac{1}{2}\textstyle\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)k(x_i, x_j)\\ & -\varepsilon\textstyle\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*)\\ \text{Sujeito a} \quad & \textstyle\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0 \quad e \quad \alpha_i, \alpha_i^* \in [0, C]\end{aligned} \tag{2.42}$$

também a expansão de eq.2.36 pode ser formulada como

$$\begin{aligned} w &= \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\Phi(x_i) \quad e \\ f(x) &= \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)k(x_i, x) + b \end{aligned} \tag{2.43}$$

A diferença para o caso linear é que w agora não é mais dado de forma explícita. Nota-se também que na abordagem não linear o problema de otimização passa a ser a determinação da função mais achatada no espaço característico e não mais no espaço de entrada [11]. Esta última abordagem, máquina de vetor suporte para regressão com uma formulação não linear, foi a escolhida para ser aplicada neste trabalho, dada a natureza não linear e a rápida dinâmica das séries de ventos.

3 Evolução Diferencial

Em sua essência, otimizar é maximizar uma propriedade desejada do sistema enquanto simultaneamente minimiza uma característica indesejável. O que são estas propriedades e quão efetivamente podem ser melhoradas depende do problema em questão. Sintonizar um rádio em uma estação por exemplo, é uma tentativa de minimizar a distorção no sinal de rádio. A função matemática que representa esta otimização é dita função objetivo pois determinar seus valores mais extremos é a meta da otimização. Quando o foco da otimização é a minimização da função objetivo pode-se denominar a função como sendo a função custo. E para o caso especial onde o ponto ótimo da otimização é situado em zero tem-se a função erro. Para o caso em que a otimização foca em uma função com propriedades a serem maximizadas é comum referir-se a função objetivo como sendo função de adaptação ou função de *fitness*. Contudo, com uma mudança no sinal da função objetivo pode-se converter uma função de adaptação em uma função custo, ou seja o problema de maximização torna-se um problema de minimização. Por isso é comum que a otimização apenas seja tratada como um problema de minimização.

3.1 Conceito

Os problemas que envolvem otimização global sobre espaços contínuos dizem respeito, em geral, a tarefa de otimizar certos parâmetros de um sistema da maneira pertinente. Pensando nisso, Price e Storn desenvolveram a ED para ser um otimizador versátil, confiável e também eficiente. A primeira publicação da ED surgiu como um relatório técnico em 1995 [12], desde então a ED tem se destacado para aplicações em problemas reais [13]. A evolução diferencial (ED) é um método de pesquisa direto e paralelo que utiliza NP vetores de dimensão D, tal que

$$x_{i,G}, i = 1, 2, ..., NP. \tag{3.1}$$

Como população para cada geração G. Onde G é a geração, x é um indivíduo da população e NP é o tamanho da população que não muda durante o processo de minimização. O vetor de população inicial é selecionado de maneira aleatória e deve cobrir todo o espaço de busca, ou seja, adota-se um gerador de números aleatórios com

distribuição uniforme. Por via de regra, assume-se uma distribuição de probabilidade uniforme para todas as decisões aleatórias, a menos que seja indicado de outra forma. Sendo assim, essa solução preliminar é disponibilizada. Onde a idéia crucial por trás da evolução diferencial é uma nova abordagem para a geração de vetores de parâmetros experimentais. A ED gera novos parâmetros pela adição das diferenças ponderadas de dois membros da população a um terceiro membro. Se o vetor resultante gera um valor mais baixo quando aplicado na função objetivo que um membro da população pré-determinado, o novo vetor gerado substitui o vetor com o qual foi comparado. A comparação pode, mas não precisa, ser parte do processo de geração mencionado. Em adição o melhor vetor de parâmetros $x_{best,G}$ este é avaliado em toda geração G de modo a acompanhar o progresso feito durante o processo de minimização. A princípio, duas formulações da ED se mostraram promissoras e foram descritas em detalhes em [12].

3.1.1 Primeira abordagem da ED

A primeira variante da evolução diferencial apresentada por Price e Storn em 1995 tem seu funcionamento descrito como se segue.

Mutação

Para cada vetor $x_{i,G}, i = 0,1,2,...,NP-1$ um vetor mutante v é gerado de acordo com,

$$v_{i,G} = x_{r1,G} + F \cdot (x_{r2,G} - x_{r3,G}), \quad i = 0,1,2,...,NP-1 \tag{3.2}$$

sendo $r_1 \neq r_2 \neq r_3 \in [0,NP-1]$ inteiros e mutuamente diferentes e $F > 0$. Os inteiros $r1, r2$ *e* $r3$ são escolhidos de maneira aleatória a partir do intervalo $[0,NP-1]$ e são diferentes do atual índice i. F é um fator, que na formulação clássica é real, constante e ainda controla a amplificação da variação diferencial $(x_{r2,G} - x_{r3,G})$. A Figura 3.1 mostra um exemplo bidimensional que ilustra os vetores que fazem parte desta primeira abordagem.

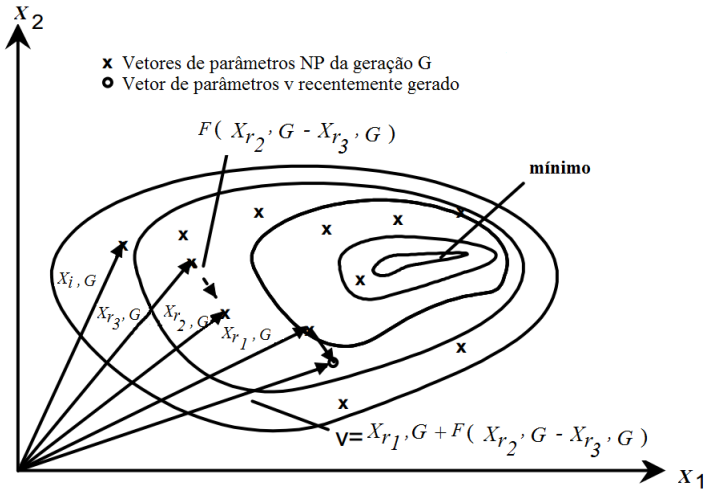


Figura 3.1: Exemplo bidimensional de uma função objetivo com as linhas de contorno e o processo para determinar v na primeira abordagem da ED

Fonte: *Adaptado de* [12]

Crossover

Para aumentar a diversidade dos vetores de parâmetro, o sistema de *crossover* é introduzido para este fim. Assim, o vetor experimental

$$u_{i,G+1} = (u_{1i,G+1}, u_{2i,G+1}, ..., u_{Di,G+1}) \tag{3.3}$$

é formado, onde

$$j = 1, 2, ..., D$$

$$u_{ji,G+1} = \begin{array}{ll} v_{ji,G+1} & para \quad randb(j) \leq CR \quad ou \quad j = rnbr(i) \\ x_{ji,G} & para \quad randb(j) > CR \quad ou \quad j \neq rnbr(i) \end{array} \tag{3.4}$$

e $randb(j)$ é a j-ésima avaliação do gerador de números aleatórios com distribuição uniforme e saída pertencente ao intervalo $[0,1]$. CR é a constante de *crossover* $\in [0,1]$ determinada pelo usuário. $rnbr(i)$ é um índice escolhido de maneira aleatória pertencente ao intervalo $[1,2,...,D]$ o que garante que $u_{ji,G+1}$ retira pelo menos um parâmetro de $v_{i,G+1}$. Este processo pode ser melhor observado na Figura 3.2 com um mecanismo de *crossover* aplicado a um vetor de dimensão $D = 7$.

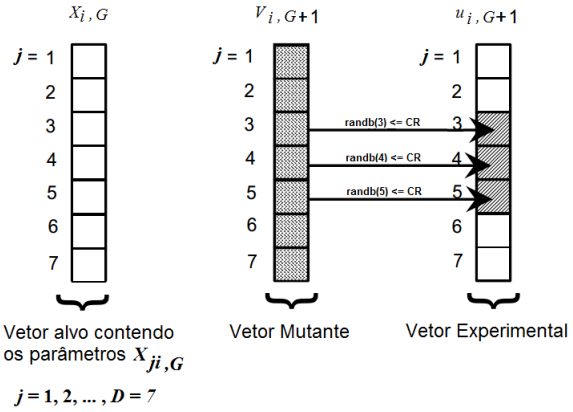


Figura 3.2: Exemplo da operação de *crossover* para $D = 7$

Fonte: *Adaptado de* [12]

Seleção

Para decidir se o novo vetor u deve vir a ser membro da geração $G+1$, ele será comparado ao $x_{i,G}$. Se o vetor u tiver um menor custo na função objetivo então $x_{i,G}$, $x_{i,G+1}$ passa a ser u, caso contrário o antigo valor $x_{i,G}$ é retido.

3.1.2 Segunda abordagem da ED

Basicamente, a segunda abordagem da evolução diferencial funciona da mesma maneira da anterior, porém gera o vetor de mutação v de acordo com a equação,

$$v_{i,G} = x_{i,G} + \lambda \cdot (x_{best,G} - x_{i,G}) + F \cdot (x_{r2,G} - x_{r3,G}), \quad i = 0,1,2,...,NP-1 \tag{3.5}$$

A Eq. 3.5 introduz uma variável de controle adicional denominada λ. A idéia desta adição é prover meios de melhorar os ganhos do sistema pela incorporação do atual melhor resultado obtido, $x_{best,G}$. Essa característica pode ser útil para funções objetivos ditas não críticas. A Figura 3.3 ilustra o processo de geração do vetor v definido na eq.3.5, a construção de u a partir de v e $x_{i,G}$ e o processo de decisão, idêntico ao da primeira abordagem.

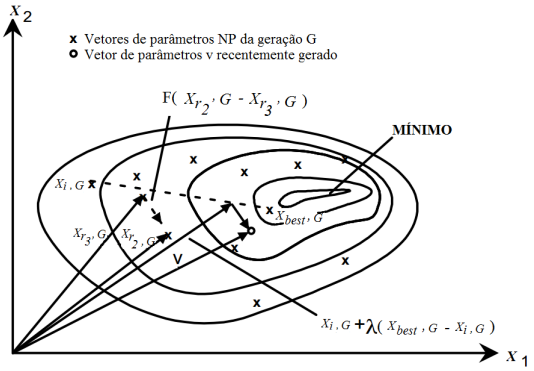


Figura 3.3: Exemplo bidimensional de uma função objetivo com as linhas de contorno e o processo para determinar v na segunda abordagem da ED

Fonte: *Adaptado de* [12]

3.1.3 Outras variantes da ED

Os dois métodos descritos anteriormente não são as únicas variantes da evolução diferencial que provaram ser úteis. E para classificar as diferentes variantes, a notação

$$DE/x/y/z$$

é introduzida, sendo que x especifica o vetor a ser mutado o que mais comumente pode ser *rand*, escolha aleatória do vetor de população, ou *best*, o vetor com menor custo encontrado na população atual. y denota a quantidade de vetores diferença usados e z define o mecanismo de *crossover* utilizado, a variante demonstrada aqui é a binomial, descrita na notação como *bin*.

Usando esta notação, a primeira abordagem apresentada pode ser escrita como

$$DE/rand/1/bin$$

3.1.4 SADE - *Self-adaptive diferencial evolution*

O desempenho do algoritmo de evolução diferencial tradicional é dependente da seleção da estratégia e parametrização para a geração do vetor experimental. Contudo a escolha desta estratégia e a correspondente seleção de parâmetros para um problema específico pode demandar um processamento muito alto, resultando em um aumento do custo computacional [25].

Uma abordagem adaptativa evita o aumento do custo computacional excessivo da abordagem que envolve tentativa e erro na procura pelo vetor experimental mais apropriado e também pelos valores dos parâmetros associados a este vetor [26]. Por estes motivos decidiu-se pela abordagem adaptativa da ED neste trabalho. O cerne da abordagem adaptativa pode ser descrito como se segue.

Estratégia de adaptação

Ao invés de implementar a busca por tentativa e erro que é computacionalmente custosa, nesta abordagem um grupo de estratégias candidatas é mantido incluindo várias estratégias efetivas de geração do vetor experimental, contemplando diversas características distintas. Durante a evolução, Para cada vetor alvo na população atual, uma estratégia será escolhida dentre as pertencentes ao grupo, de acordo com uma probabilidade obtida de experiências anteriores que geraram soluções promissoras. Esta estratégia é então aplicada para realizar a operação de mutação. Quanto mais bem sucedida uma estratégia for nas gerações anteriores maior a probabilidade de ser escolhida na geração atual para gerar novas soluções. Algumas das estratégias comummente utilizadas no grupo de possíveis soluções são [25]:

- DE/rand-to-best/1/bin;
- DE/best/1/bin;
- DE/best/2/bin;
- DE/rand/1/bin.

Adaptação de parâmetros

Na abordagem convencional da ED a escolha dos valores numéricos para os parâmetros F, CR e NP são altamente dependentes do problema em questão. Na abordagem adaptativa apenas NP é deixado para que o usuário escolha, porque este parâmetro esta fortemente ligado a complexidade do problema em questão e, de fato, o tamanho da população NP não necessita ser ajustado com precisão. Assim, alguns poucos valores tipicamente usados podem ser escolhidos sem interferir no desenvolvimento do algoritmo [26].

4 Materiais e métodos

4.1 Materiais

Os experimentos simulados e os algoritmos empregados foram todos criados e executados no ambiente MATLAB. As séries apresentadas como dados reais foram obtidas de terceiros não existindo então uma etapa de aquisição de dados durante o projeto por parte do autor e todos os dados obtidos foram devidamente tratados e filtrados pelo laboratório que disponibilizou os dados (*Research Laboratory of Renewable Energy* ,RERL).

4.1.1 Séries temporais de ventos

A questão energética é um tópico de estudo muito importante na atualidade pois a qualidade de vida da sociedade esta intimamente relacionada ao consumo de energia elétrica. E o crescimento da demanda energética em países em desenvolvimento, como o Brasil, em razão da melhora na qualidade de vida fazem com que a segurança no suprimento de energia e os custos ambientais para atender esse crescimento sejam alguns dos aspectos que vem ganhando notoriedade na política de planejamento energético de todas as economias emergentes. A inserção de fontes renováveis de energia, como a eólica, aparece com o intuito de minimizar esta questão e vem crescendo em aplicação e pesquisa nos últimos anos. A geração de eletricidade proveniente de fontes eólicas embora reconhecida como amigáveis ao meio ambiente do ponto de vista da emissão de substâncias nocivas à atmosfera apresenta alguns aspectos ambientais que não podem ser negligenciados como, por exemplo, a influência do ruído produzido pelos geradores em animais como os pássaros [2].

A previsão das séries de ventos tem como propósito a previsão da capacidade energética gerada em parques eólicos, de maneira direta ou indireta, primeiramente através da estimação da série de ventos e da previsão da capacidade energética associada a esta série. A previsão de ventos é principalmente orientada à viabilidade de instalação de novos parques eólicos, gerenciamento de sistemas e planejamento de manutenções, sendo de interesse dos operadores do sistema e companhias de energia [4] [3].

Desde os primórdios a previsão de ventos despertou expectativas no setor energé-

tico, onde um grupo de discussão do *Pacific Ocean Laboratory* esclareceu as vantagens e importâncias da previsão de séries de ventos para a área energética em meados dos anos 70. E sua evolução vem sendo mantida pela competição de interesses comerciais. Já nos anos 80, aplicações como a utilização conjunta de um preditor para séries de ventos e um controlador on/off atrelado a um gerador a diesel em um sistema autônomo para economia de insumos, denominado de sistema Vento/Diesel foi proposto [27]. Durante os anos 90, o aumento da capacidade energética proveniente de fontes eólicas instaladas por todo o mundo, principalmente na Europa e nos Estados Unidos da América, chamou mais uma vez a atenção das companhias energéticas e dos pesquisadores sobre o problema da previsão da série de ventos. Motivados pela necessidade de integração desta energia incerta e oscilante proveniente de fontes eólicas à rede elétrica, inicia-se também uma tentativa de refinamento dos modelos utilizados para previsão com a adição de fenômenos físicos referentes ao ventos.

Nos últimos anos, uma atenção especial foi dada ao desenvolvimento de ferramentas para operações online e tratamento de incertezas existentes na previsão. Também surgem as primeiras integrações efetivas entre duas linhas de pesquisas complementares, a matemática e os fenômenos físicos envolvendo os ventos [4].

4.1.2 Recursos computacionais

Descrição do computador e software onde as simulações foram executadas:

- Processador: 3^a Geração do Processador Intel Core i7-3632QM 2.2GHz, 8 *Threads*, 6Mb Cache);
- Sistema operacional: Windows 8 Single Language (Português);
- Memória RAM1: 8 GB de SDRAM DDR3 a 1600 MHz;
- Armazenamento: Disco Rígido 1TB SATA (5400 RPM) com 32GB mSATA SSD (para Intel Smart Response);
- Placa de vídeo: Placa de Vídeo AMD Radeon HD 7730M, 128-bit, 2GB;
- *Software*: MATLAB R2012b;
- LS-SVM *software*: Lssvmlab.

4.2 Métodos

O uso da LS-SVM foi proposto pois o custo computacional desta vertente é inferior em comparação com a SVM original. A idéia do uso em conjunto com a evolução diferencial veio da necessidade de escolher parâmetros para a máquina de vetor suporte e na tentativa de minimizar a influência da escolha destes parâmetros na qualidade da identificação, e por consequência da previsão, um otimizador foi colocado. A vertente conhecida como SADE foi então a escolhida por também ter a capacidade de otimizar os parâmetros da ED eliminando a necessidade de realizar qualquer seleção de parâmetro durante o processo de identificação e previsão.

4.2.1 Integração dos algoritmos propostos

A integração entre os dois algoritmos é apresentada no fluxograma mostrado na Figura 4.1, que detalha o comportamento da SADE. A integração ocorre de maneira em que a SADE envia os parâmetros otimizados para a máquina de vetor suporte que por sua vez executa a identificação da série de ventos e retorna para o valor do erro médio quadrático obtido no conjunto de teste da série de vento, esse valor é então utilizado pelo otimizador para avaliar a estratégia empregada e os parâmetros selecionados. O processo se repete até a evolução diferencial atingir o número máximo de gerações selecionado na inicialização do algoritmo.

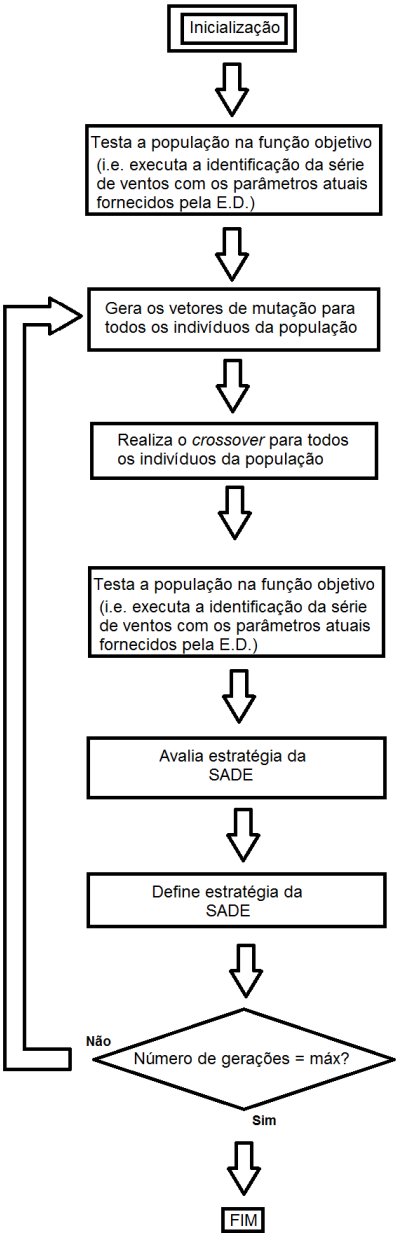


Figura 4.1: Fluxograma de integração dos algoritmos

Fonte: *O autor*(2012)

5 Características gerais e discussões

A geração de eletricidade proveniente de fontes eólicas, ainda que apresente algumas questões ambientais importantes a serem consideradas, vem sendo reconhecida como amigáveis ao meio ambiente, com benefícios sociais e economicamente competitiva, o que indica um crescimento na aplicação desta tecnologia. Contudo a natureza imprevisível das séries de ventos torna difícil a tarefa de realizar o estudo da viabilidade econômica para a implementação de novos parques eólicos, o que torna a previsão de ventos vital para o desenvolvimento e inclusão desta tecnologia na previsão de produção de energia a curto prazo. [3].

Para a realização da identificação e previsão da série de ventos utilizando a máquina de vetor suporte à mínimos quadrados. A divisão do conjunto de dados, série temporal amostrada, é realizada em três partes distintas: Primeiro, Os dados de treinamento usados para a construção do modelo matemático. Segundo, Dados de validação utilizados para a seleção de parâmetros do sistema e avaliação da aderência do modelo aos dados de entrada bem como sua capacidade de generalização. Por fim, em terceiro, os dados de teste dão a noção exata da capacidade do modelo em trabalhar com dados desconhecidos, os dados de teste não fazem parte de nenhuma etapa da implementação do algoritmo responsável pela geração do modelo matemático [28].

Estes dados de teste são os meios pelos quais pode-se avaliar a qualidade da solução obtida. Pois são o meio para a comparação entre modelos e ferramentas da identificação de séries de ventos para a medida de desempenho, neste trabalho uma comparação entre o algoritmo proposto e a rede neural perceptron multi camadas foi realizada para comparação e avaliação do desempenho do algoritmo na previsão de séries de ventos. As principais medidas de avaliação da função gerada são, o erro médio (ME), o erro médio absoluto (MAE, do inglês *Mean Absolute Error*), o erro médio quadrático (MSE, do inglês *Mean Squared Error*) e a raiz do erro médio quadrático (RMSE, do inglês *Root Mean Squared Error*) [4]. A medida de desempenho aqui adotada foi o erro médio quadrático,

$$MSE = \frac{1}{n}\sum_{i=1}^{m}(p_i - p_{ti})^2 \tag{5.1}$$

onde p_t é o valor observado e p o valor previsto e m o número total de pontos avaliados. A seleção das funções *Kernel* a ser aplicada em cada um destes conjuntos de testes foram feitas através do algoritmo da SADE.

Visando detalhar a análise de validação do modelo obtido, testes de correlação orientados a sistemas não lineares foram aplicados permitindo uma análise gráfica da capacidade do preditor em captar a dinâmica do sistema. Uma elucidação mais detalhada a cerca dos testes de correlação em sistemas não lineares pode ser obtida em [29] e [30]. A Tabela 5.1 é utilizada para realizar a análise simplificada dos gráficos de correlação obtidos, esta análise conjunta das correlações obtidas se faz necessária quando o sistema identificado é não linear. Na análise da Tabela 5.1, u representa os dados de entrada e e representa os dados de resíduo. Esta análise é dita simplificada por não abordar a análise do termo exato negligenciado pelo preditor ao descrever a dinâmica do processo. A tabela completa e a análise detalhada é descrita em [29].

Tabela 5.1: Análise simplificada de correlação

u^2,e^2	u^2,e	u,e	Característica
$=0$	$=0$	$=0$	Processo modelado sem tendências
$\neq 0$	$=0$	$=0$	Dinâmica não mapeada ou ruído interno
$\neq 0$	$\neq 0$	$\neq 0$	Processo não pode ser modelado pelo algoritmo
$\neq 0$	$\neq 0$	$=0$	Dinâmica não mapeada ou ruído interno
$\neq 0$	$=0$	$\neq 0$	Dinâmica não mapeada ou ruído interno

Fonte: *Adaptado de* [29]

As séries temporais testadas foram disponibilizadas pelo (*Research Laboratory of Renewable Energy*, RERL), no próprio site do laboratório [31], os conjuntos de dados já foram previamente filtrados por um software de controle de qualidade do próprio laboratório, esta filtragem é também responsável pelo tratamento de lacunas e *outliers* presentes nos conjuntos de dados obtidos das medições iniciais. Contudo os dados antes da filtragem também são disponibilizados para quem deseja aplicar seu próprio método de filtragem.

Foram selecionados três séries de ventos ao acaso dentre todas as disponibilizadas, cada uma referente a uma região dos Estados Unidos da América. Sendo elas, *Paxton*, *Orleans* e *Barnstable*. A Tabela 5.2 fornece uma descrição global da localidade onde os dados foram obtidos e as condições de amostragem utilizadas.

Tabela 5.2: Descrição dos conjuntos de dados testados

Parâmetros	Paxton Muni, MA	Orleans, MA	Barnstable, MA
Latitude [N]:	42,30324	41,76028	41,66483
Longitude [E]:	71,89727	69,9927	70,30457
Fuso horário [H]:	-5	-5	-5
Altura [m]:	317	15	21
Intervalo entre amostragens [s]:	600	600	600
Tempo de amostragem [s]:	2	2	2

Fonte: *Adaptado de* [31]

A Tabela 5.3 apresenta o período de amostragem em cada uma das três localidades, além das datas de geração dos relatórios finais após a obtenção e tratamento dos dados.

Tabela 5.3: Características de amostragem dos dados

Localidade	Período	Data relatório
Paxton Muni, MA	2003-06-24 a 2007-01-08 08:00:00	28/02/2007 12:28
Orleans, MA	2003-10-27 a 2003-12-31 23:50:00	19/06/2007 15:53
Barnstable, MA	2005-04-01 a 2005-12-31 23:50:00	01/11/2006 23:16

Fonte: *Adaptado de* [31]

A seguir são apresentados os gráficos de cada uma das séries descritas anteriormente já devidamente filtradas pelo laboratório que as disponibilizou. Os dados apresentados nestes gráficos ainda serão divididos em dois conjuntos, um conjunto de treinamento e outro de teste. Ficando este segundo conjunto desconhecido para o algoritmo e atuará como um conjunto de dados novos para o sistema.

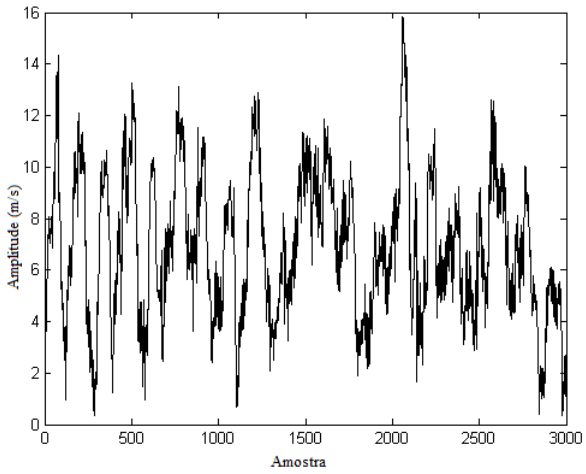


Figura 5.1: Série original de Paxton.

Fonte: *O autor*(2012)

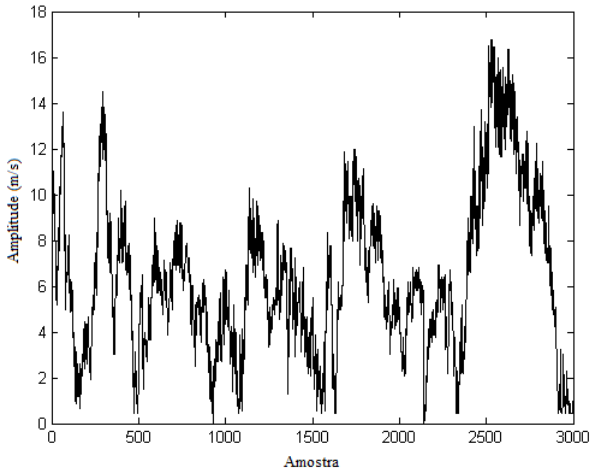


Figura 5.2: Série original de Orleans.

Fonte: *O autor*(2012)

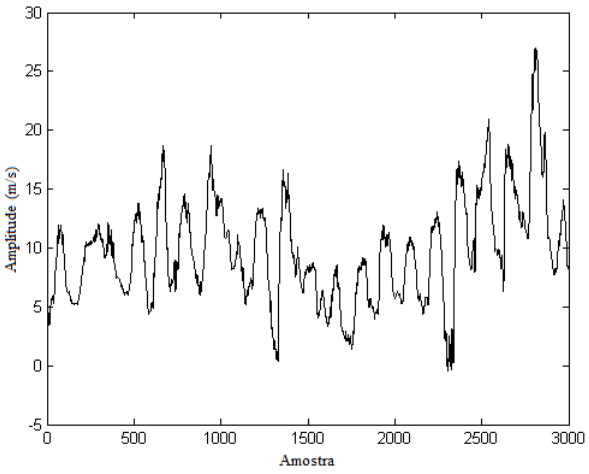


Figura 5.3: Série original de Barnstable.

Fonte: *O autor*(2012)

Nesta etapa os gráficos comparativos dos conjuntos de teste entre a saída real e a estimada pelo uso conjunto da SADE com a LS-SVM serão analisados. A fim de complementar os resultados apontados, outros gráficos como o de aderência da função aos dados de treinamento, e uma análise da capacidade do algoritmo em descrever a dinâmica do processo serão apresentados e discutidos. Dentre os fatores que influenciam a resposta obtida do estimador, considera-se o horizonte de previsão como uma das influências mais significativas. Assim, uma análise separada será realizada de acordo com o número de passos à frente considerados neste horizonte. Para tal, dois horizontes de previsão foram escolhidos para teste, sendo com 1 e 20 passos à frente. A discussão dos resultados serão divididos de acordo com a quantidade de passos à frente.

5.1 Previsão um passo à frente

A Figura 5.4 representa a comparação entre a saída real e a prevista pelo algoritmo proposto para a localidade de Paxton considerando um horizonte de previsão de 1 passo à frente. Uma primeira análise, a partir dos instantes iniciais até pouco mais da metade dos dados, indica uma boa predição da saída quando a função alvo se mostra mais comportada, ou seja, quando não existe variações bruscas na dinâmica do sistema real. A partir deste ponto, entre os instantes 60 e 70, o preditor consegue acompanhar a dinâmica do processo mas logo o erro tende a aumentar e a previsão se torna inadequada.

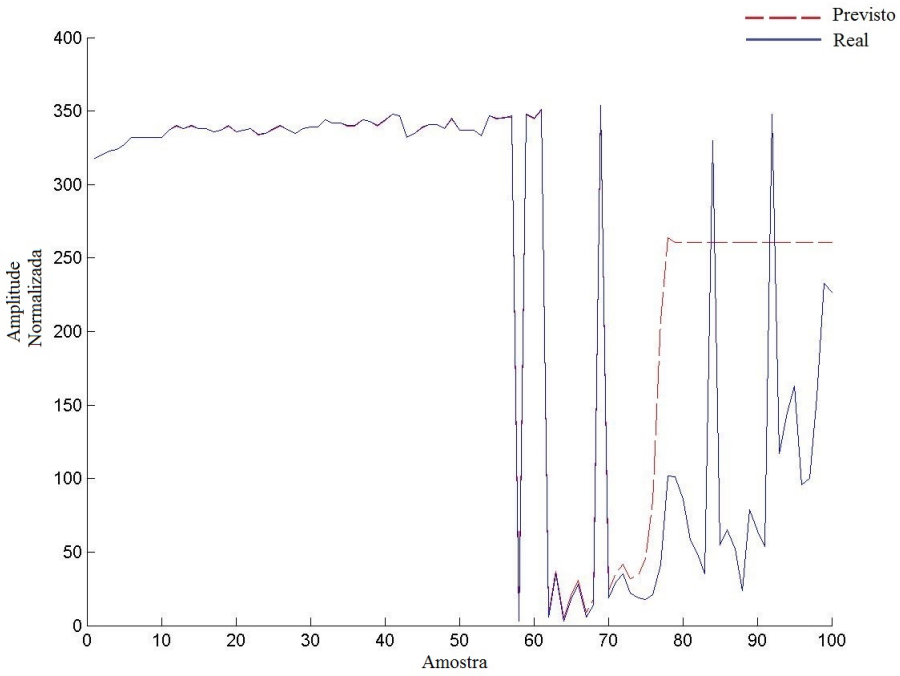


Figura 5.4: Comparativo real versus estimado para a localidade de Paxton

Fonte: *O autor*(2012)

Um dos motivos pelo qual o erro (Figura 5.5) tende a aumentar pode ser visto na Figura 5.6 que representa a aderência da função aos dados de entrada. Percebe-se um sobre ajuste da função aos dados de treinamento e por consequência um erro maior na extrapolação realizada pela função obtida aos dados desconhecidos.

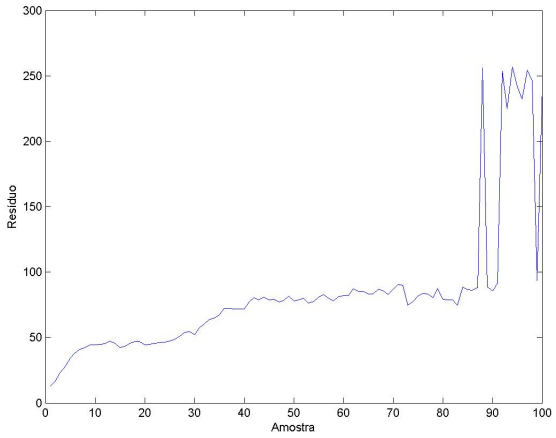


Figura 5.5: Erro absoluto Para a localidade de Paxton

Fonte: *O autor*(2012)

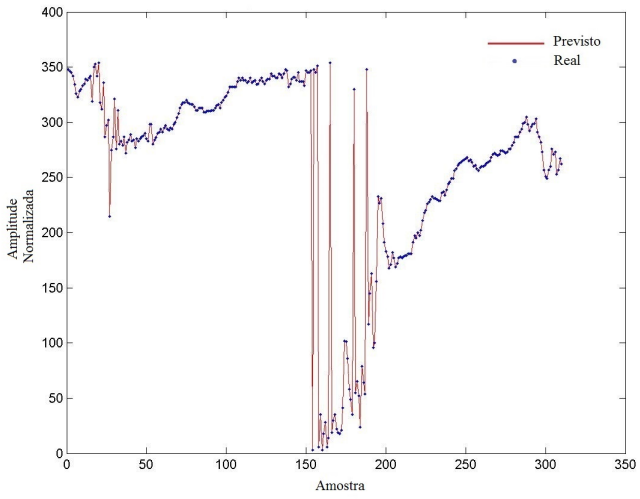


Figura 5.6: Aderência da função aos dados de treinamento Para a localidade de Paxton

Fonte: *O autor*(2012)

Outro motivo é a incapacidade do algoritmo de mapear satisfatoriamente a dinâmica do sistema, como visto na Figura 5.7. Uma análise detalhada dos gráficos de correlação, utilizando a Tabela 5.1, deixa claro que na faixa que compreende os atrasos, *lags*, no intervalo $[-15\ 15]$ todos os gráficos testados apresentaram valores fora da faixa dos **5%** de erros admitidos.

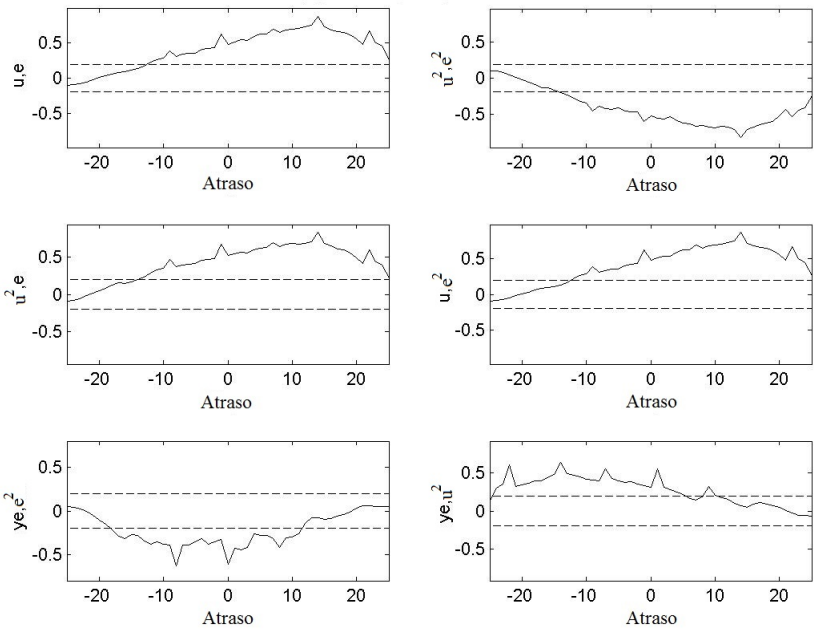


Figura 5.7: Correlação para a localidade de Paxton

Fonte: *O autor*(2012)

Contudo, com um aumento do número de iterações da SADE como algoritmo de otimização, o que permitiu um melhor aproveitamento das características adaptativas do algoritmo, resultasse em uma melhor parametrização do preditor. Com isso outros resultados mais promissores foram obtidos. Na Figura 5.8 percebe-se que a saída prevista coincide com boa aproximação da saída real.

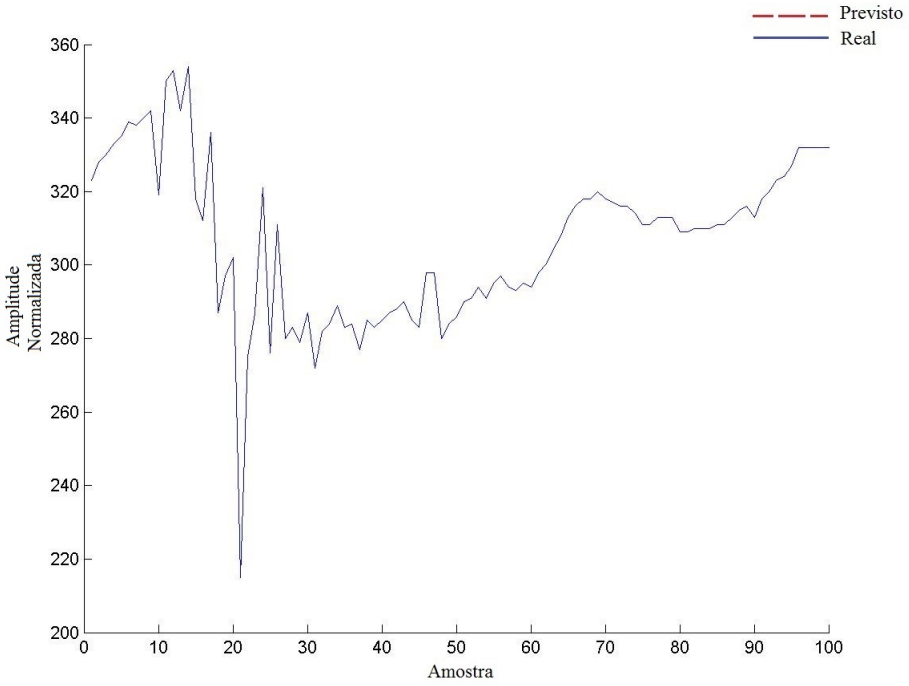


Figura 5.8: Segundo comparativo real versus estimado para a localidade de Paxton

Fonte: *O autor*(2012)

Para uma análise comparativa dos resultados, a Figura 5.9 é o resultado da simulação do problema para a localidade de Paxton utilizando a rede neural perceptron multi camadas. Comparativamente podemos ver que com o aumento das iterações da SADE o otimizador foi o fator responsável por um importante papel na parametrização da máquina de vetor suporte. Com isso o conjunto proposto atingiu uma qualidade melhor na identificação. Na primeira tentativa, com parâmetros aleatórios a máquina de vetor suporte não teve um bom resultado na identificação, como visto na Figura 5.4. Já com o uma maior influência da SADE, como visto na Figura 5.8, o resultado final obtido foi melhor que o apresentado pela rede neural na Figura 5.9

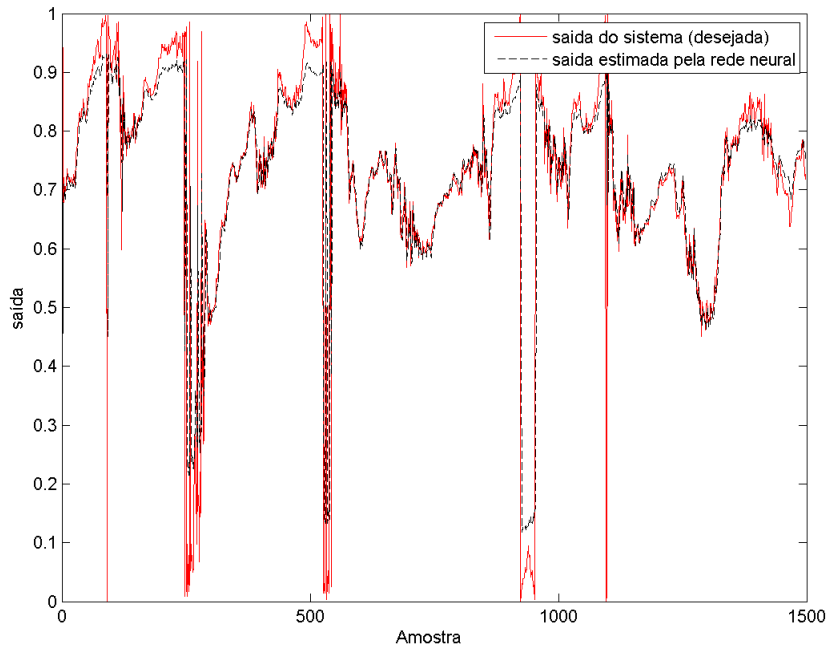


Figura 5.9: Comparativo real versus estimado, pela rede neural, para a localidade de Paxton

Fonte: *O autor*(2012)

Uma análise da correlação (Figura 5.10) corrobora o resultado apresentado. Mostrando que toda a dinâmica do processo foi agora capturada pelo algoritmo.

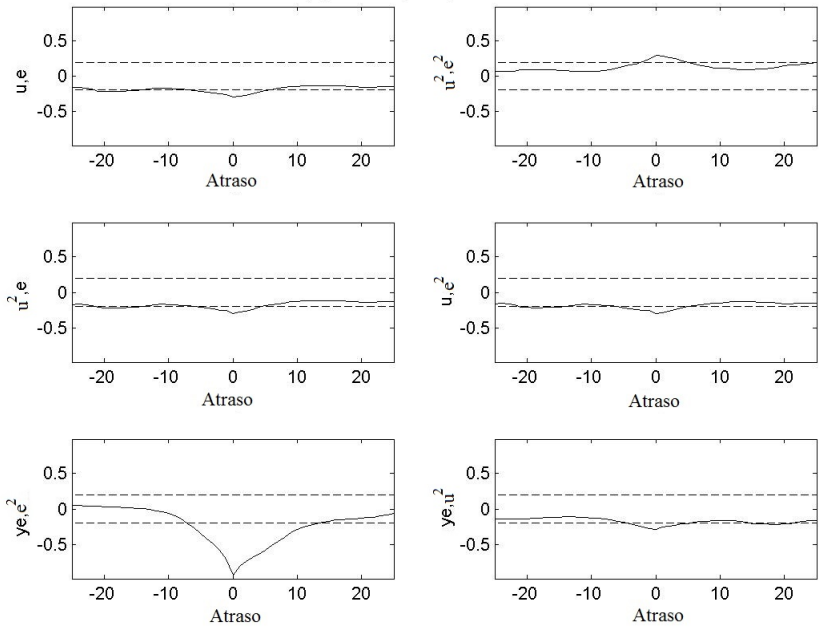


Figura 5.10: Segunda análise de correlação para a localidade de Paxton

Fonte: *O autor*(2012)

Para a localidade de Barnstable a comparação entre a saída real e estimada, Figura 5.11, apresenta a capacidade do algoritmo em acompanhar a tendência apresentada mas não atingiu uma boa aproximação do sistema real em sua amplitude, o que também pode ser visto no gráfico de erro apresentado na Figura 5.12. Sugerindo a necessidade de acrescentar mais dados de entrada ao problema de identificação.

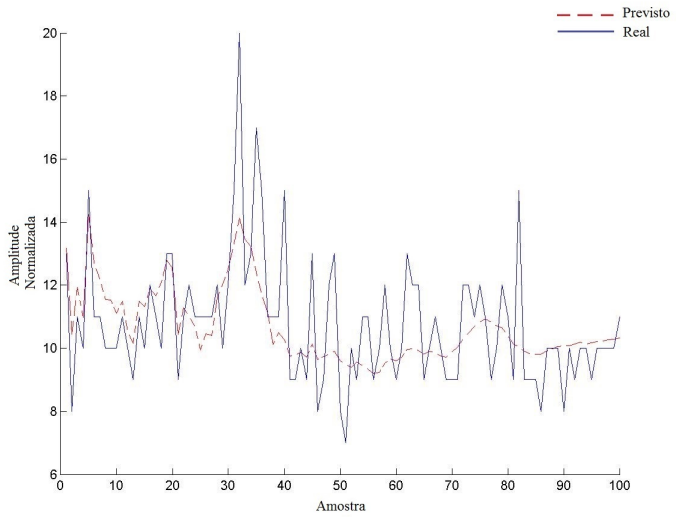


Figura 5.11: Comparativo real versus estimado para a localidade de Barnstable

Fonte: *O autor*(2012)

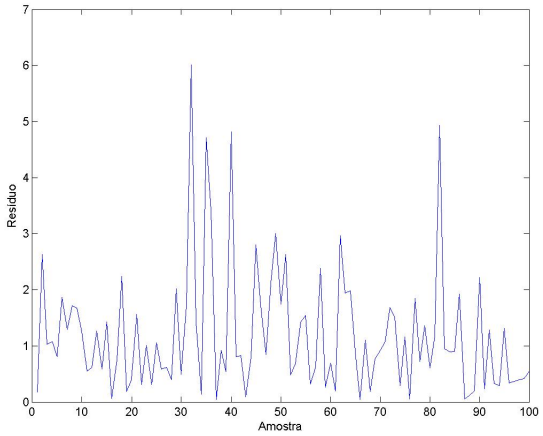


Figura 5.12: Erro absoluto Para a localidade de Barnstable

Fonte: *O autor*(2012)

Ainda que a necessidade de mais dados de entrada ao problema seja clara, a comparação com a solução obtida da simulação utilizando a rede neural perceptron multi camadas (Figura 5.13) mostra que ambos os algoritmos, o proposto e a rede neural, não foram capazes de realizar uma identificação satisfatória do conjunto de dados apresentados.

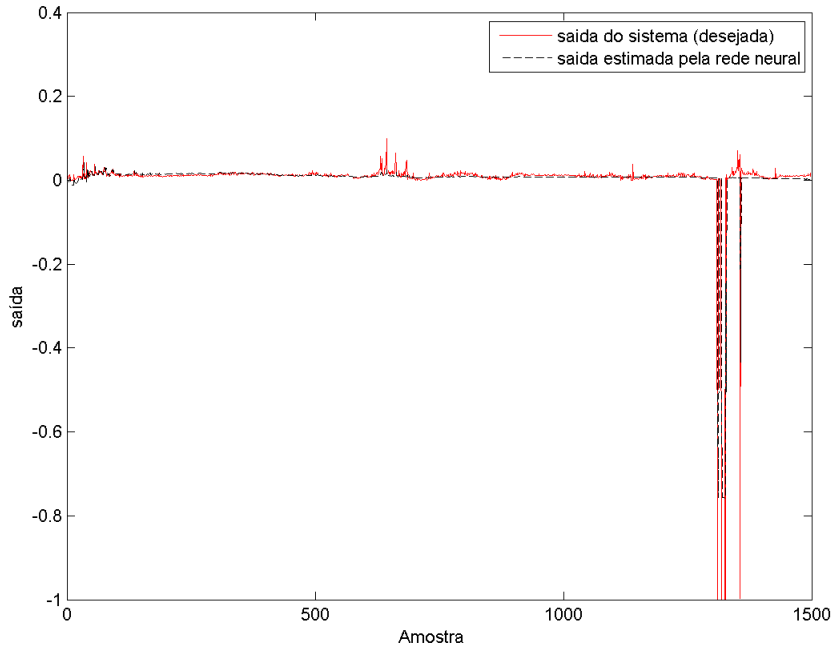


Figura 5.13: Comparativo real versus estimado, pela rede neural, para a localidade de Barnstable

Fonte: *O autor*(2012)

No gráfico de aderência da função aos dados de entrada, Figura 5.14, percebe-se a identificação das tendências e constata-se que o sobre ajuste foi evitado, o que indica que o algoritmo selecionou uma função alvo mais simples e que bem representa os dados de treinamento.

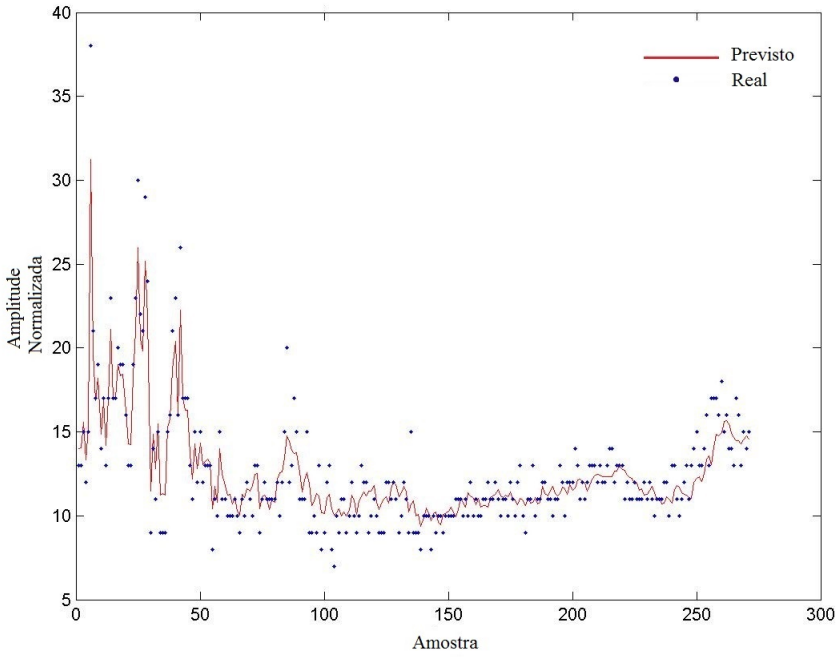


Figura 5.14: Aderência da função aos dados de treinamento Para a localidade de Barnstable

Fonte: *O autor*(2012)

A divergência entre o gráfico real e estimado constatada na Figura 5.11 é explicada pela análise de correlação feita na Figura 5.15, onde alguns pontos foram salientados, indicando que parte da dinâmica do sistema não pode ser mapeada completamente.

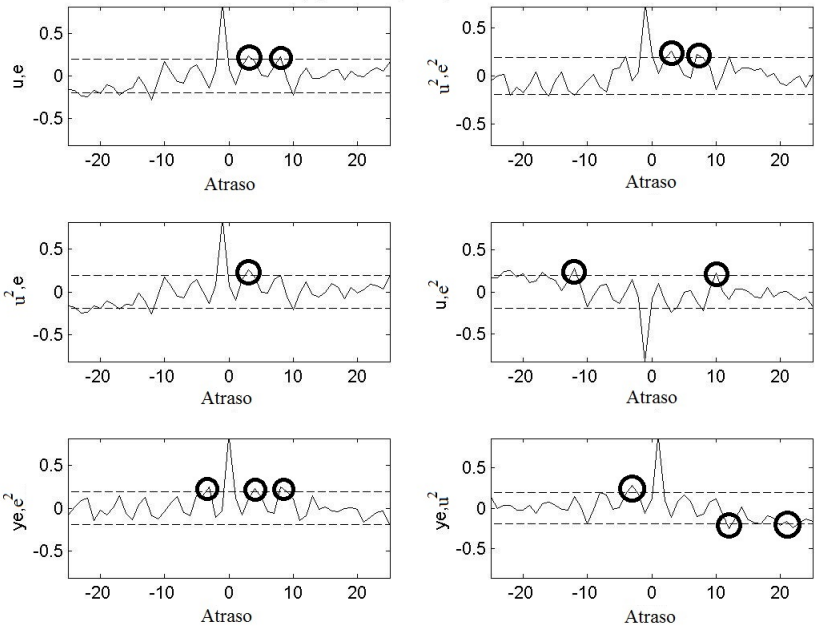


Figura 5.15: Correlação para a localidade de Barnstable

Fonte: *O autor*(2012)

Por fim a análise da saída real e estimada da localidade de Orleans, apresentada na Figura 5.16, apresentou os melhores resultados dentre as localidades testadas. Além de captar a tendência, o sinal estimado convergiu para os valores reais em toda a região abrangida pelos dados de teste, o que pode ser confirmado no gráfico de erro apresentado na Figura 5.17. A comparação com a rede neural perceptron multi camadas (Figura 5.18)mais um vez indica que algoritmo proposto apresenta resultados promissores na identificação e previsão de energia a curto prazo quando comparado ao resultado obtido pela aplicação da rede neural.

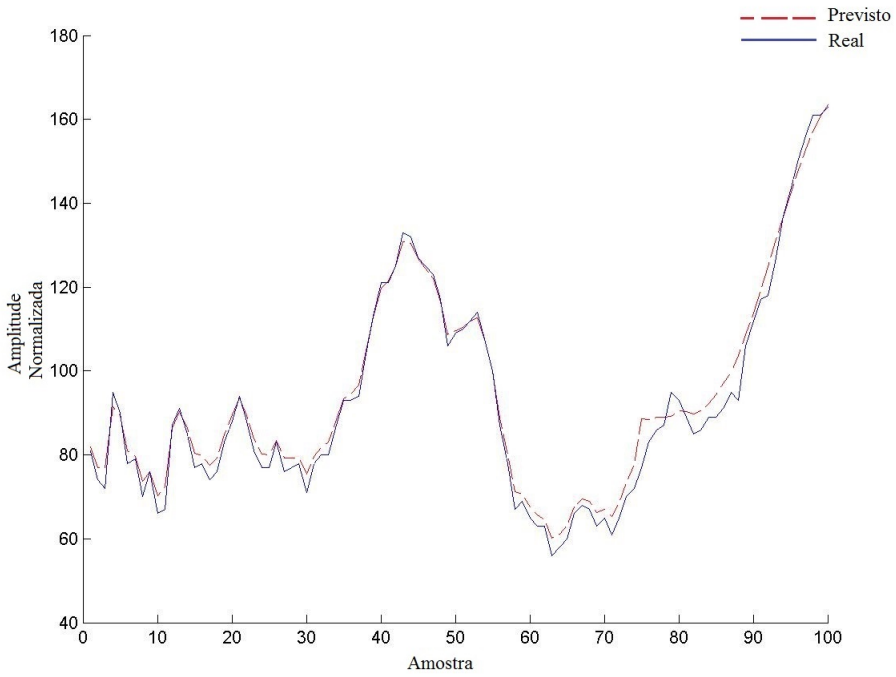


Figura 5.16: Comparativo real versus estimado para a localidade de Orleans

Fonte: *O autor*(2012)

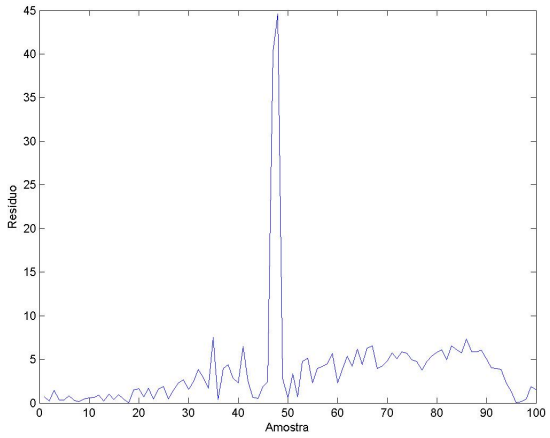


Figura 5.17: Erro absoluto Para a localidade de Orleans

Fonte: *O autor*(2012)

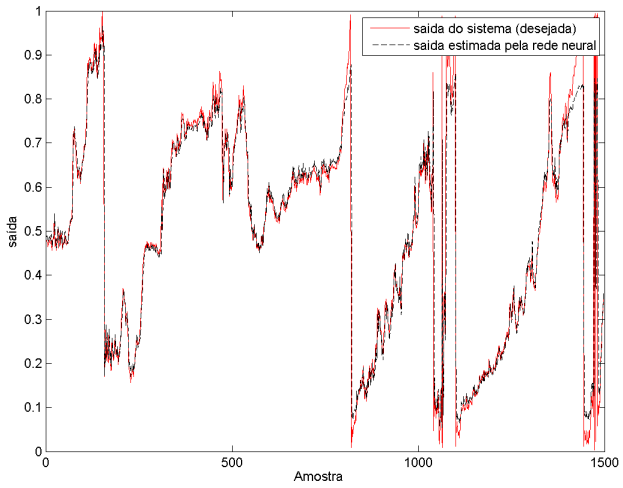


Figura 5.18: Comparativo real versus estimado, pela rede neural, para a localidade de Orleans

Fonte: *O autor*(2012)

Esse bom resultado se deve a boa aderência da função aos dados de treinamento sem que houvesse um sobre ajuste da função aos dados de entrada e sem existir pontos, do conjunto de treinamento, distantes da função estimada para representar esses dados, como mostrado na Figura 5.19.

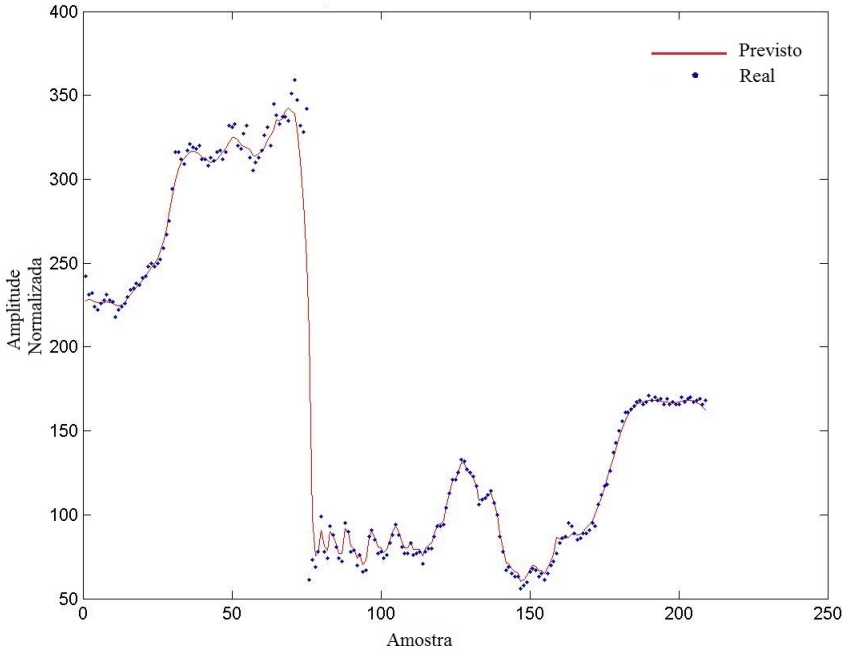


Figura 5.19: Aderência da função aos dados de treinamento Para a localidade de Orleans

Fonte: *O autor*(2012)

Porém a não total coincidência entre a função alvo e a estimada se deve aos pontos salientados na Figura 5.20, o teste de correlação mostrou alguns poucos pontos onde não foi possível identificar a dinâmica do sistema.

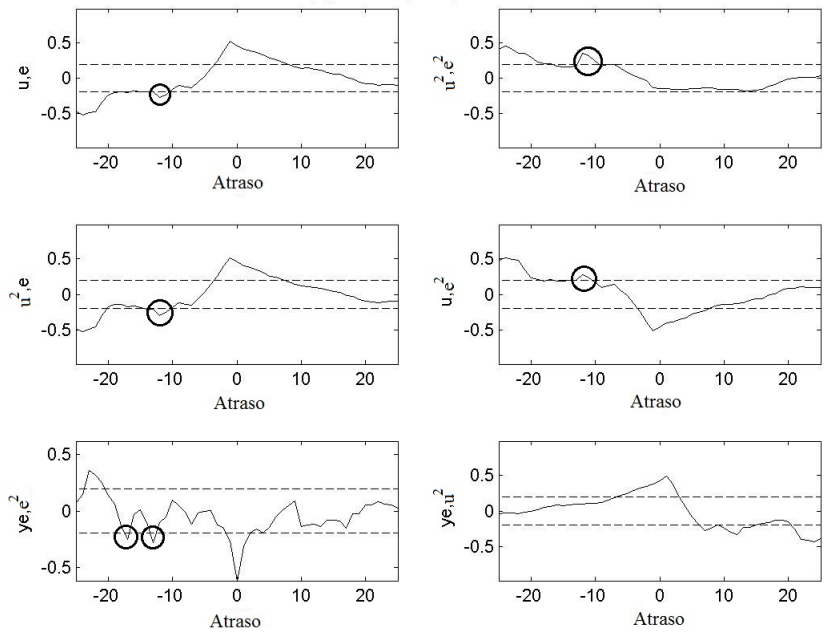


Figura 5.20: Correlação para a localidade de Orleans

Fonte: *O autor*(2012)

5.2 Previsão N passos à frente

A estimação com um horizonte de previsão maior implica em um erro acumulado elevado, dificultando o mapeamento da dinâmica do sistema e aumentando a dificuldade de se obter um modelo confiável do sistema real.

Para a localidade de Paxton algumas características que já ficaram evidentes durante a análise com um horizonte de previsão de 1 passo à frente, como o sobre ajuste da função aos dados de entrada, apareceram novamente porém com o agravante do acúmulo do erro (Figura 5.21) ficou inviável realizar a previsão com o horizonte de previsão aumentado, como mostrado na Figura 5.22

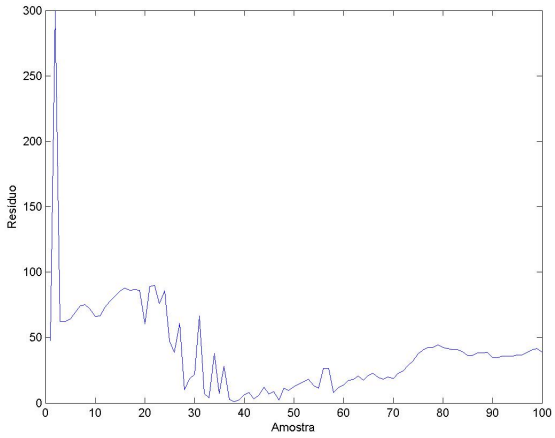


Figura 5.21: Erro absoluto para a localidade de Paxton com um horizonte de previsão de 20 passos à frente

Fonte: *O autor*(2012)

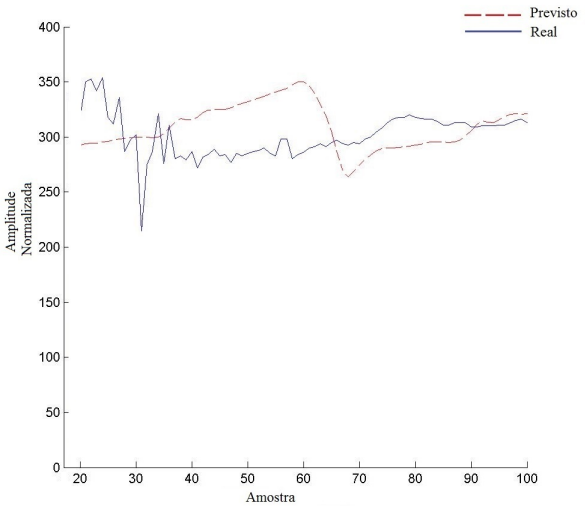


Figura 5.22: Comparativo real versus estimado para a localidade de Paxton com um horizonte de previsão de 20 passos à frente

Fonte: *O autor*(2012)

No conjunto de dados referentes a Barnstable a dificuldade de realizar a identificação considerando o aumento do horizonte de previsão pode ser visto na menor capacidade do preditor em mapear as tendências do sistema, como constatado pelo apresentado na Figura 5.23 e pelo erro apresentado na Figura 5.24.

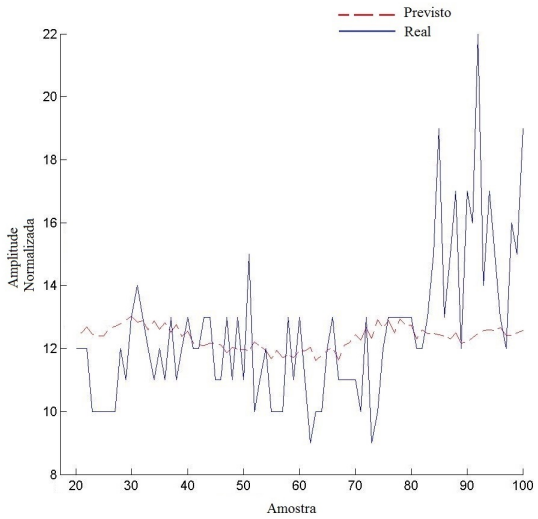


Figura 5.23: Comparativo real versus estimado para a localidade de Barnstable com um horizonte de previsão de 20 passos à frente

Fonte: *O autor*(2012)

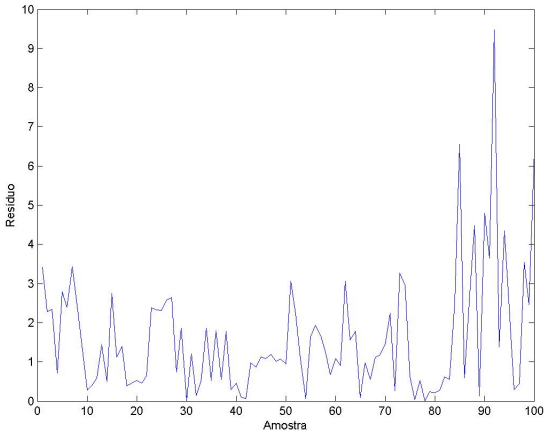


Figura 5.24: Erro absoluto para a localidade de Barnstable com um horizonte de previsão de 20 passos à frente

Fonte: *O autor*(2012)

Essa dificuldade com relação a tendência também pode ser vista na tentativa de mapear a dinâmica completa do sistema, na análise feita a partir da Figura 5.25. Neste caso, os gráficos de correlação estão sensivelmente inferiores aos obtidos com um horizonte de previsão de um passo à frente.

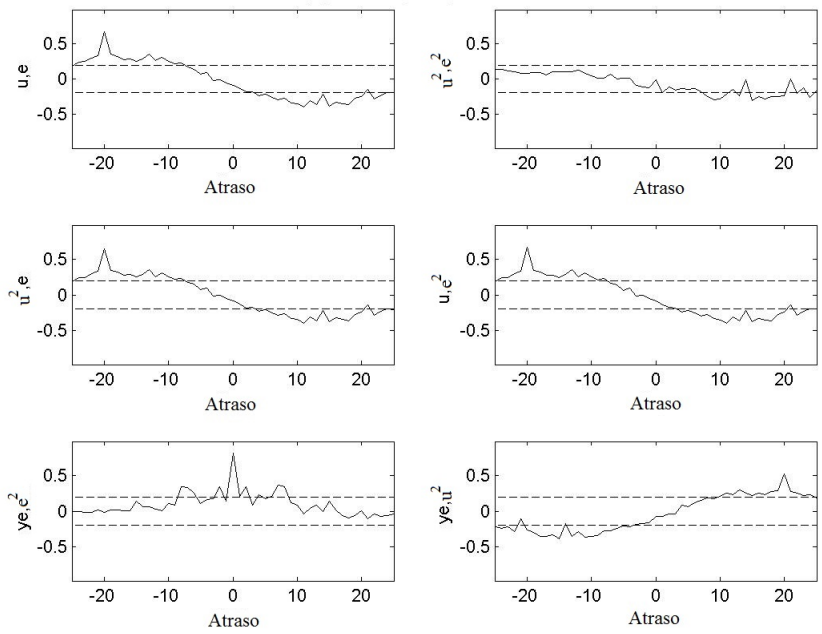


Figura 5.25: Correlação para a localidade de Barnstable com um horizonte de previsão de 20 passos à frente

Fonte: *O autor*(2012)

Já na localidade de Orleans, assim como no horizonte de previsão de um passo à frente, os resultados foram os mais promissores. Ainda com o acumulo do erro (Figura 5.26) a identificação se mostrou eficaz como visto na Figura 5.27.

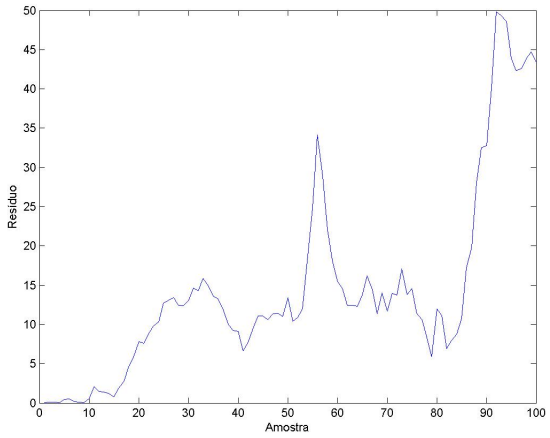


Figura 5.26: Erro absoluto para a localidade de Orleans com um horizonte de previsão de 20 passos à frente

Fonte: *O autor*(2012)

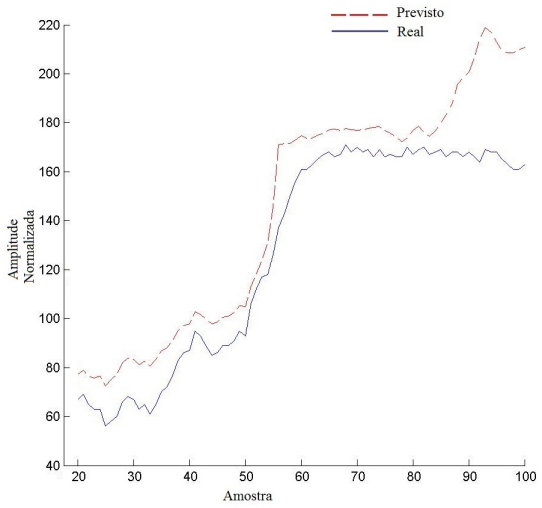


Figura 5.27: Comparativo real versus estimado para a localidade de Orleans com um horizonte de previsão de 20 passos à frente

Fonte: *O autor*(2012)

Contudo, ainda que demonstrando uma previsão aceitável para o conjunto de dados desconhecido, a capacidade de mapear a dinâmica do sistema foi reduzida com o aumento do horizonte de previsão, como observado na Figura 5.28 que contém os gráficos de correlação. A Figura 5.28 indica uma correlação periódica da entrada com o resíduo, confirmando a correlação mas indicando a necessidade de tratamento do resíduo na entrada.

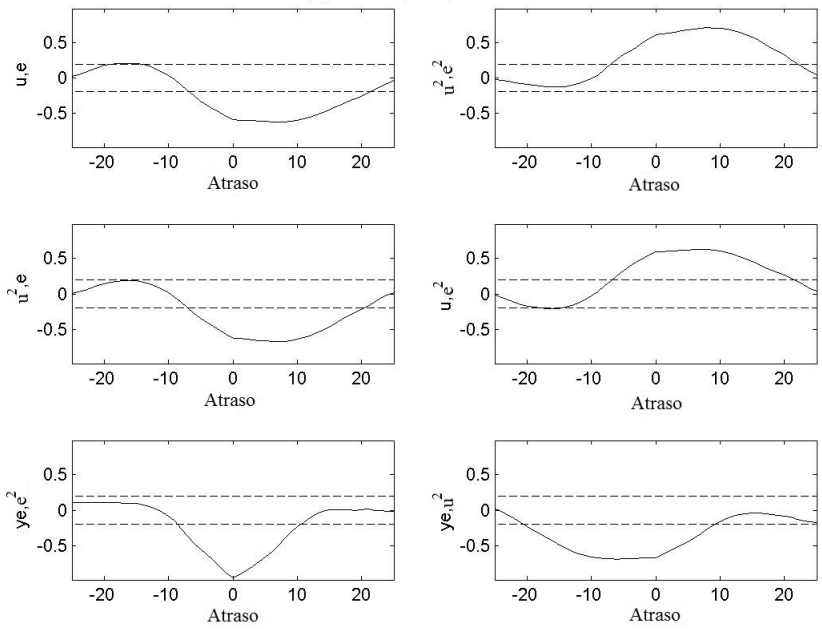


Figura 5.28: Correlação para a localidade de Orleans com um horizonte de previsão de 20 passos à frente

Fonte: *O autor*(2012)

6 Conclusão e trabalhos futuros

Cada vez mais a questão energética tem se mostrado um ponto importante de estudo e planejamento nos países emergentes. Isso acontece porque a qualidade de vida da sociedade esta intimamente ligada ao consumo de energia elétrica e os países em desenvolvimento, como o Brasil, em que o padrão de qualidade de vida vem crescendo em um ritmo acelerado questões como o suprimento de energia e os custos ambientais da geração de energia são cada vez mais relevantes. A inserção de fontes renováveis de energia, como a eólica, aparece com o intuito de minimizar esta questão e vem crescendo em aplicação e pesquisa nos últimos anos. A geração de eletricidade proveniente de fontes eólicas embora reconhecida como amigáveis ao meio ambiente do ponto de vista da emissão de substâncias nocivas à atmosfera apresenta alguns aspectos ambientais que não podem ser negligenciados. Contudo ainda há um crescimento na aplicação desta tecnologia. Um dos maiores problemas referentes à energia eólica é a incerteza do vento. Esta questão é atualmente um tema importante, chamando a atenção de toda a indústria de energia eólica e de serviços públicos. As previsões de vento de curto prazo podem ajudar nesta questão e irão se tornar vitais à medida que mais fontes de energia renováveis são adicionadas à rede elétrica.

O algoritmo proposto com a finalidade de ser o preditor das séries de vento foi a máquina de vetor suporte a mínimos quadrados, LS-SVM. As características deste algoritmo levam em consideração a capacidade de generalização do modelo e a adaptação do modelo obtido ao conjunto de dados dito de treinamento permitiem evitar ou ao menos minimizar o problema do sobre ajuste da função aos dados de entrada. O baixo custo computacional exigido pela LS-SVM é uma outra característica desejável quando é pensado o uso conjunto de duas metaheurísticas. Além disso, o fato de que a máquina de vetor suporte possuir características lineares de variação dos parâmetros ainda que aplicada a identificação de funções não lineares permitiu que fosse aplicada uma otimização idealizada para espaços contínuos.

Quanto a aplicação da SADE como otimizador, o uso de uma abordagem adaptativa garante um melhor desempenho uma vez que a parametrização é uma das etapas de maior influência na obtenção de bons resultados. Este otimizador, ao longo das iterações, atualiza e converge os parâmetros para os valores ideais e com isso resulta em uma melhor

aproximação do preditor, melhorando o desempenho do conjunto.

Com a análise dos dados realizadas no capítulo 5 . Pode-se dizer que o método adotado atingiu as expectativas e o modelo obtido é uma representação razoável dos sistemas em questão considerando um horizonte de previsão médio e curto. Contudo uma comparação direta entre os modelos encontrados pela heurística adotada e por outras abordagens existentes, maneira ideal para comprovar a eficácia do método proposto, é inviável uma vez que não existe um padrão para comparação definido dado a complexidade desta análise e a dificuldade do intercâmbio destes bancos de dados das séries temporais de ventos. Mas o que pode-se constatar, quando analisamos as Figuras 5.10, 5.20 e 5.15 onde observa-se que quanto menos pontos não mapeados na dinâmica do sistema, mais precisa é a representação do sistema real. Percebe-se, sob esta ótica, que o conjunto proposto para realizar a identificação obteve resultados promissores.

Uma análise criteriosa dos gráficos de correlação obtidos indica que o algoritmo ainda pode ser aprimorado com a adição, por exemplo, das séries de erro envolvidas na predição o que reduziria os pontos não mapeados da dinâmica do sistema e viabilizaria o uso de horizontes maiores de previsão e uma precisão maior nas predições feitas para os horizontes atuais. Com estas possibilidades, visando o melhoramento do algoritmo proposto, esta e algumas outras idéias estão retratadas nos ítens sugeridos na seção trabalhos futuros, a seguir.

Como sugestão para possíveis trabalhos futuros, tem-se:

- Analisar a definição de uma família de modelos e métodos para identificação que melhor se adequariam as características de cada sistema;
- Analisar a correlação a fim de estabelecer os conjuntos de dados de entrada omitidos a fim de aprimorar os resultados do preditor;
- Estudar a manipulação da série temporal a fim de reduzir ou melhorar a qualidade das considerações realizadas para a definição do modelo;
- Estudar a implementação de uma ferramenta online de previsão para o problema das séries temporais de ventos e geração de energia a curto prazo.

Referências

1 MARTINS, F.; GUARNIERI, R.; PEREIRA, E. O aproveitamento da energia eólica. *Revista Brasileira de Ensino de Física*, v. 30, p. 1 – 13, 2008.

2 WAYE, K.; OHRSTROM, E. Psycho-acoustic characters of relevance for annoyance of wind turbine noise. *Journal of Sound and Vibration*, v. 250, n. 1, p. 65 – 73, 2002. ISSN 0022-460X. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0022460X01939057>.

3 MONFARED, M.; RASTEGAR, H.; KOJABADI, H. M. A new strategy for wind speed forecasting using artificial intelligent methods. *Renewable Energy*, v. 34, p. 845–848, 2009.

4 COSTA, A. et al. A review on the young history of the wind power short-term prediction. *Renewable and Sustainable Energy Reviews*, v. 12, p. 1725–1744, 2008.

5 RUBIO, G. et al. A heuristic method for parameter selection in ls-svm aplication to time series prediction. *International Journal of Forecasting*, v. 27, p. 725–739, 2010.

6 PINSON, P.; KARINIOTAKIS, G. Wind power forecasting using fuzzy neural networks enhanced with on-line prediction risk assessment. In: *Power Tech Conference Proceedings, 2003 IEEE Bologna*. Bologna: [s.n.], 2003. v. 2, p. 8.

7 HERRERA, L. et al. Recursive prediction for long term time series forecasting using advanced models. *Neurocomputing*, v. 70, n. 16-18, p. 2870 – 2880, 2007. ISSN 0925-2312. <ce:title>Neural Network Applications in Electrical Engineering</ce:title> <ce:title>Selected papers from the 3rd International Work-Conference on Artificial Neural Networks (IWANN 2005)</ce:title> <xocs:full-name>3rd International Work-Conference on Artificial Neural Networks (IWANN 2005)</xocs:full-name>. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0925231207001622>.

8 JAKKULA, V. *Tutorial on Support Vector Machine (SVM)*. Washington, 2006.

9 BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: *Proceedings of the 5th Annual workshop on Computational learning theory*. New York, NY, USA: ACM, 1992. (COLT '92), p. 144–152. ISBN 0-89791-497-X. Disponível em: <http://doi.acm.org/10.1145/130385.130401>.

10 BURGES, C. J. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, Springer Netherlands, v. 2, p. 121–167, 1998. ISSN 1384-5810. 10.1023/A:1009715923555. Disponível em: <http://dx.doi.org/10.1023/A:1009715923555>.

11 SMOLA, A. J.; SCHöLKOPF, B. A tutorial on support vector regression. *Statistics and Computing*, Springer Netherlands, v. 14, p. 199–222, 2004. ISSN 0960-3174. 10.1023/B:STCO.0000035301.49549.88. Disponível em: <http://dx.doi.org/10.1023/B:STCO.0000035301.49549.88>.

12 STORN, R.; PRICE, K. *Differential Evolution A simple and efficient adaptive scheme for global optimization over continuous spaces*. Netherland, 1995.

13 PRICE, K.; STORN, R. *Differential Evolution A Practical Approach to Global Optimization*. Berlin: Springer, 2005.

14 BOUSQUET, O. et al. *Introduction to Statistical Learning Theory*. Berlin: Springer Berlin Heidelberg, 2004.

15 PEDNAULT, E. P. D. *Statistical Learning Theory*. USA, 1997.

16 VAPNIK, V. N. *The Nature of Statistical Learning*. New York: Springer-Verlag New York, Inc., 1995.

17 VAPNIK, V. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, v. 10, n. 5, p. 988 –999, set. 1999. ISSN 1045-9227.

18 YU, H.; KIM, S. Svm tutorial classification, regression and ranking. In: ROZENBERG, G.; THOMAS; KOK, J. (Ed.). *Handbook of Natural Computing*. Berlin: Springer Berlin Heidelberg, 2012. p. 479–506. ISBN 978-3-540-92909-3.

19 MINH, H. Q.; NIYOGI, P.; YAO, Y. Mercer theorem, feature maps, and smoothing. In: LUGOSI, G.; SIMON, H. U. (Ed.). *Learning Theory*. Berlin: Springer Berlin Heidelberg, 2006. (Lecture Notes in Computer Science, v. 4005), p. 154–168.

20 SUYKENS, J.; VANDEWALLE, J. Least squares support vector machine classifiers. *Neural Processing Letters*, Springer Netherlands, v. 9, p. 293–300, 1999. ISSN 1370-4621. 10.1023/A:1018628609742. Disponível em: <http://dx.doi.org/10.1023/A:1018628609742>.

21 ZHANG, M.; FU, L. Unbiased least squares support vector machine with polynomial kernel. In: *Signal Processing, 2006 8th International Conference on*. Guilin: [s.n.], 2006. v. 3.

22 ESPINOZA, M.; SUYKENS, J. Least squares support vector machines and primal space estimation. *Decision and Control, 2003. Proceedings. 42nd IEEE Conference*, v. 4, p. 3451–3456, 2003.

23 HUANG, W.; NAKAMORI, Y.; WANG, S.-Y. Forecasting stock market movement direction with support vector machine. *Computers & Operations Research*, v. 32, n. 10, p. 2513 – 2522, 2005. ISSN 0305-0548. <ce:title>Applications of Neural Networks</ce:title>. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0305054804000681>.

24 KUHN, H.; TUCKER, A. Nonlinear programming. In: STATISTICAL LABORATORY OF THE UNIVERSITY OF CALIFORNIA, BERKELEY. *2nd Berkeley Symposium on Mathematical Statistics and Probabilistics*. California, 1951. p. 481–492.

25 HUANG, A. K. Q. V. L.; SUGANTHAN, P. N. Self-adaptive differential evolution algorithm for constrained real-parameter optimization. In: *IEEE Congress on Evolutionary Computation*. Washington: [s.n.], 2006.

26 QIN, V. L. H. A. K.; SUGANTHAN, P. N. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Transactions on Evolutionary Computation*, v. 13, p. 398–417, 2009.

27 WANG, L.; SINGH, C.; KUSIAK, A. (Ed.). *Wind Power Systems: Applications of Computational Intelligence*, (Green Energy and Technology). Berlin: Springer Berlin Heidelberg, 2010.

28 MOHANDES, M. et al. Support vector machines for wind speed prediction. *Renewable Energy*, v. 29, p. 939–947, 2004.

29 BILLINGS, S. A.; VOON, W. S. F. Correlation based model validity tests for non-linear models. *International Journal of Control*, v. 44, n. 1, p. 235–244, 1986. Disponível em: <http://www.tandfonline.com/doi/abs/10.1080/00207178608933593>.

30 BILLINGS, S. A.; VOON, W. S. F. A prediction-error and stepwise-regression estimation algorithm for non-linear systems. *INT. J. CONTROL*, v. 44, p. 803–822, 1986.

31 RERL. *Fonte de dados para teste.* Site. Disponível em: *<http://www.ceere.org/rerl/publications/resource/data/Orleans/>*.