

ALGORITMO SUBIDA DA ENCOSTA PARA OTIMIZAÇÃO DE PARÂMETROS DE MÁQUINAS DE VETORES SUPORTES

Francisco Carlos Monteiro Souza, Ricardo B. C. Prudêncio

Centro de Informática – Universidade Federal de Pernambuco (CIN – UFPE)
CEP 50732-970 – Recife – PE – Brasil
{fcms, rbcp}@cin.ufpe.br

Abstract – Máquinas de Vetores Suporte (SVM) surgiu recentemente como uma técnica poderosa para resolução de problemas classificação de padrão e regressão, mas seu desempenho depende principalmente da seleção de parâmetros do mesmo. Seleção de parâmetros para SVM é de uma natureza muito complexa e muito difícil de resolver por meio de técnicas de otimização convencional, ocasionando a restrição de sua aplicação em certo grau. Este artigo apresenta um sistema híbrido para otimização da seleção do parâmetro de regularização C do SVM e o parâmetro γ (gamma) do Kernel RBF utilizando o algoritmo de busca meta-heurística Subida da Encosta e uma versão otimizada do mesmo, onde o processo de busca foi aplicado em uma grade de busca composta por 38 problemas de benchmark, contendo o valor de desempenho da combinação de 399 parâmetros distintos executados no SVM.

Keywords – Máquinas de Vetor Suporte, Otimização de parâmetros, Algoritmo Subida da Encosta.

1 Introdução

Máquinas de Vetores Suporte é técnica de aprendizagem baseada na teoria de aprendizagem estatística. Ultimamente tem sido aplicado em diversos tipos de problemas como reconhecimento de escrita, classificação de imagens, análises de seqüências biológicas, problemas de regressão. Contudo, é observado que o desempenho das SVMs depende fortemente da seleção adequada de seus parâmetros internos, que incluem, por exemplo, a escolha da função kernel, dos parâmetros associados à função, do parâmetro de regularização, entre outros [1].

Nesse contexto, SVM possui um pequeno número de parâmetros que proporciona um efeito de regularização sobre o método, como por exemplo, o parâmetro C com função de regularização do classificador e os parâmetros de largura do kernel RBF γ (gamma) ou a dimensão de um kernel polinomial, dentre outros parâmetros existentes. Uma vez que, os valores desses parâmetros não foram escolhidos adequadamente, pode ocasionar a redução no desempenho geral do sistema. Existe um número muito significativo de configurações de parâmetros que podem ser usados no SVMs, de forma que, tenta encontrar a melhor configuração através de um processo exaustivo é extremamente difícil.

Neste artigo é apresentado um sistema híbrido para a otimização dos parâmetros do SVM, utilizando a técnica de busca meta-heurística Subida da Encosta e uma versão otimizada, a fim de encontrar a melhor configuração de parâmetros. Este artigo está dividido em 6 seções. Na seção 2 foi abordada a teoria de máquinas de vetores. Na seção 3, foi abordado o algoritmo de busca meta-heurística Subida da Encosta. Na seção 4, foi descrito o Sistema para Otimização Parâmetros de SVM. Na seção 5, são mostrados os resultados obtidos. Por último, na seção 6, é apresentada as considerações finais.

2 Máquinas de Vetores Suporte

Máquinas de Vetores Suporte (SVMs) é uma técnica baseada na teoria de aprendizagem estatística e foi proposta por Vapnik [2] e [3] para classificação e reconhecimento de padrões, tais como, reconhecimento facial, reconhecimento de caracteres manuscritos, reconhecimento de texturas e objetos, bioinformática etc. e ainda estendida para suportar regressão, problemas de classificação de múltiplas classes e outras tarefas.

A idéia básica do SVM é separar as classes com superfícies que maximizem a margem entre as mesmas, ou seja, encontrar um hiperplano de separação ótima. Em caso de problemas não linearmente separáveis, os padrões de entrada são convertidos para um vetor de características de alta dimensionalidade com o auxílio de funções de kernel, cujo objetivo é tornar possível a separação dos padrões linearmente no espaço. Uma vez que, o espaço adequado de características é definido, o SVM seleciona o hiperplano especial, chamado de Hiperplano de Margem Máxima, do inglês *Maximum Margin Hyperplane* (MMH) ou simplesmente Hiperplano Ótimo, o qual corresponde à maior distância de seus padrões no conjunto de treinamento. Estes padrões são chamados de Vetores de Suporte, do inglês *Support Vector* (SV) [4].

Segundo [5], máquinas de vetores suporte é uma implementação do método de minimização do risco estrutural equivalente a minimização do erro de generalização que consiste a imposição de um limite para minimizar os erros de classificação na etapa de teste. Este princípio indutivo é baseado no fato de que a taxa de erro de uma máquina de

aprendizagem sobre dados de teste (taxa de erro de generalização) é limitada pela soma da taxa de erro de treinamento e por um termo de dependência da Dimensão VC. Essa dimensão mede a capacidade de um espaço de hipóteses. A capacidade é uma medida da complexidade e mede a força expressiva, a riqueza ou a flexibilidade de um conjunto de funções [6].

3 Busca Meta-Heurística

Uma meta-heurística é um método ou algoritmo que combina funções objetivos com heurísticas. De acordo com [7], essa combinação é muitas vezes realizada utilizando estatísticas obtidas de amostras em um espaço de busca ou baseiam-se em algum fenômeno natural ou processo físico. Como por exemplo, a analogia de algoritmos genéticos com o comportamento natural e evolução das espécies da teoria de Charles Darwin ou a metáfora de um processo térmico utilizado em metalurgias para *Simulated Annealing*.

Nesse contexto técnicas meta-heurísticas são algoritmos que visam à resolução problemas de otimização de modo a alcançar soluções ótimas ou aproximadas ao ótimo, normalmente são utilizadas em problemas que não podem ser resolvidos com algoritmos convencionais ou que possuem grande complexidade. Muitos desses se enquadram em problemas de otimização e busca onde não se sabe se existe uma solução ótima que seja em um tempo hábil e um custo computacional aceitável.

3.1 Subida da Encosta

O Algoritmo Subida da Encosta do inglês *Hill Climbing* (HC) é baseado na idéia de subida do Monte Everest em meio de um nevoeiro denso durante uma crise de amnésia, onde o objetivo é chegar no pico mais alto apenas examinando os locais mais próximos (vizinhos) esquecendo os caminhos anteriores e termina quando alcança um pico em que nenhum vizinho tenha o valor mais alto.

Segundo [8], o algoritmo de busca da encosta é simplesmente um laço repetitivo que se movimenta de forma contínua no sentido do valor crescente, isto é, encosta acima e termina quando encontra o pico mais alto. Subida da Encosta é uma técnica de busca heurística que combina um método de busca geral *generate-and-teste* (gere-e-teste) com o uso de funções heurísticas para a avaliação dos estados gerados pelo método gere-e-teste, a fim de encontrar o melhor caminho a seguir na busca e retornar um resultado satisfatório. Portanto, o algoritmo consiste em selecionar um estado inicial (nó corrente), avaliá-lo e em seguida compará-lo com todos os estados que podem ser gerados a partir do nó corrente. O melhor estado dentre todos é escolhido para ser o novo nó corrente. Esse processo é repetido até encontrar o estado-objetivo ou chegar à condição de parada.

A subida da encosta pode ser comparada com uma busca gulosa, isso porque captura o próximo estado vizinho sem decidir com antecedência o caminho que seguirá em seguida. Este processo causa alguns problemas no algoritmo como, por exemplo, cair em máximos locais, picos e platôs.

O maior problema do algoritmo está relacionado à convergência prematura, principalmente quando a busca encontra um máximo local, pois se trata de um pico mais alto que cada um de seus estados vizinhos, embora seja mais baixo que o máximo global o algoritmo fica paralisado quando alcança esse estado. Os picos resultam em uma seqüência de máximos locais, o que ocasiona uma difícil movimentação para algoritmos gulosos e por fim um platô trata-se de uma região onde a função de avaliação é plana.

Existem alguns métodos para resolver quando o algoritmo alcança um ponto em que não há progressos, como os reinícios aleatórios ou movimentos laterais se considerado que a natureza do problema seja uma planície.

4 Sistema para Otimização Parâmetros de SVM utilizando Subida da Encosta

A seleção de parâmetros é um problema bem comum em técnicas de aprendizagem de máquina, pois, influencia diretamente na qualidade dos modelos. Apesar do SVM possuir uma performance eficaz para maioria dos problemas de classificação e regressão, este é sensível a seleção adequada dos parâmetros. Muitas estratégias para seleção e otimização do processo vêm sendo aplicada para esse tipo de problema, focando principalmente em como definir os valores dos parâmetros ideais, visando alcançar um desempenho satisfatório para determinados conjuntos de dados, sendo que algumas dessas estratégias possuem resultados interessantes outras nem tanto, visto que, é uma tarefa onde requer tratamento de vários fatores, como robustez, tempo, erros, custo computacional.

Este artigo apresenta um estudo sobre a técnica de busca meta-heurística Subida da Encosta e uma versão otimizada, evidenciando seu desempenho, aplicabilidade, bem como esta contribuição para a solução do problema de otimização de parâmetros para SVM, a fim de identificar a melhor configuração de parâmetros em um espaço de busca. Após a

implementação desse algoritmo foi realizado experimentos e análise dos resultados, verificando a aplicabilidade, desempenho e a veracidade do modelo. Na Figura 1 é exibida a arquitetura do sistema híbrido com SVM e o algoritmo de busca e otimização, bem como as etapas que constituíram o ciclo dessa pesquisa.

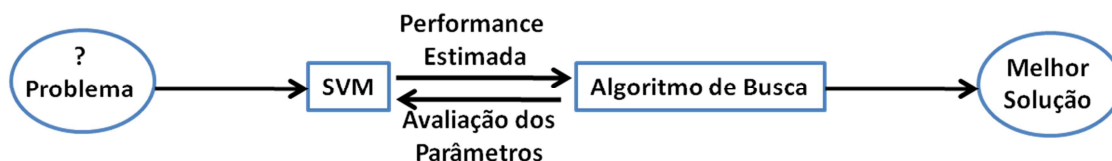


Figura 1 – Arquitetura do Sistema Híbrido SVM e Algoritmos de Busca Meta-Heurística.

Para a solução do problema, é realizada a combinação do SVM com a técnica de busca (Figura 1), iniciando sua execução a partir de pontos aleatórios do espaço de busca contendo os parâmetros C e γ (parâmetro gamma do kernel RBF) do SVM. O espaço de busca foi estabelecido com 21 valores para C e 19 valores para γ totalizando 399 combinações distintas como é sugerido na literatura (Quadro 1).

Quadro 3.1 – Número de Parâmetros do Espaço de Busca

Parâmetros	C	γ
Valores	$2^{-5}, \dots, 2^{15}$	$2^{-15}, \dots, 2^3$
Total	21	19

O SVM funciona como uma função *fitness* para o algoritmo de busca, assim, avaliando a configuração dos parâmetros iniciais sugeridos pelo algoritmo Subida da Encosta através do cálculo de desempenho do SVM. O resultado retornará para o algoritmo a fim de verificar se essa é a melhor configuração, caso não seja, então, o mesmo sugere uma nova combinação de acordo com a estratégia de busca. Este processo é repetido até que o algoritmo de busca encontre a melhor configuração dos parâmetros para o SVM ou atinja um determinado critério de parada.

A base de dados utilizada é composta por 38 (trinta e oito) problemas de regressão coletadas do site WEKA. Cada base de Benchmarks é de até 2178 instâncias sendo que dados faltantes foram removidos e a ordem dos exemplos foi alterada para evitar qualquer tipo de tendência da coleta de dados do conjunto original.

Para realização dos experimentos foi implementada a técnica de busca Subida da Encosta foi desenvolvida como especificada em [8] devido sua simplicidade e em diversos tipos de problemas garantir bons resultados, onde em cada passo a solução atual é substituída pelo vizinho com melhor valor (Figura .2).

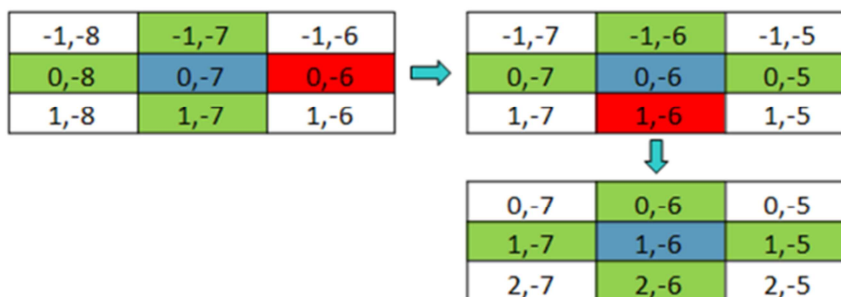


Figura 2 – Processo do Algoritmo Subida da Encosta

O primeiro nó (nó corrente) com a combinação de parâmetros C e γ é selecionado aleatoriamente, a partir dele são selecionados quatro nodos vizinhos e o que apresentar melhor valor de fitness será o novo nó corrente, onde o valor de cada quadrado representa uma combinação de parâmetros C e γ . O nó de cor azul é o inicial e o vermelho o que tem melhor fitness, na próxima iteração ele será o novo nó corrente e serão expandidos seus vizinhos e assim sucessivamente até não encontrar um nó vizinho melhor que o nó corrente.

A condição de parada para o algoritmo Subida da Encosta é encontrada quando forem avaliadas trinta soluções diferentes. Desde modo evita-se um tempo de execução muito alto e potencializa o valor do algoritmo, no sentido que de ele se torna eficiente para o problema com um número de interações relativamente baixo.

A execução do sistema foi realizada de modo off-line, ou seja, o valor de *fitness* para cada combinação de parâmetros foi encontrada anteriormente a partir da execução do SVM aplicando a validação cruzada com 10-folds utilizando a implementação do SVM do software “R statistical” (especificamente o pacote e1071 interface para o LIBSVM, biblioteca para

máquinas de vetores suporte para problemas de regressão). Cada configuração de parâmetro tem como *fitness* o NMSE que se trata do erro médio quadrático normalizado (equação 1) obtido no experimento de validação cruzada. Sendo assim, foi criada uma grade de busca contendo os valores de desempenho das 399 combinações de C e γ e aplicado o algoritmo de busca na mesma.

$$NMSE = \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{(y_i - \bar{y})^2} \quad (1)$$

Onde, y_i é o erro real, \hat{y}_i é o erro esperado e \bar{y} a média dos erros.

5 Resultados

Para análise e obtenção dos resultados foram realizados experimentos com o algoritmo Subida da Encosta para a seleção de parâmetros do SVM, bem como um estudo comparativo com o algoritmo busca aleatória.

O algoritmo Subida da encosta começa sua execução com a combinação de parâmetros C e γ selecionada aleatoriamente em seguida realiza o processo baseado na idéia do melhor vizinho. Resumindo, o processo inicia selecionando uma solução aleatória após isso é gerado quatro soluções da vizinhança baseado na solução corrente, duas da vertical e duas da horizontal, essas soluções são avaliadas e a que tiver o melhor valor será a nova solução corrente. Caso não exista nenhuma solução melhor que a solução corrente o algoritmo termina sua execução com condição de ter completado a avaliação de trinta soluções distintas, se não tiver terminado, um novo nó corrente (C, γ) é gerado aleatoriamente e segue o mesmo processo.

Para a avaliação do método de otimização de parâmetros proposto foi utilizado uma base de dados contendo 38 problemas de *benchmarks* com o espaço de busca com a combinação de 399 parâmetros de C e γ , onde para cada problema foi efetuado 100 execuções e cada iteração até que visite 30 soluções diferentes. Sendo assim, em cada uma das execuções o sistema retorna uma lista contendo o resultado de desempenho das 30 combinações de C e γ para o SVM, ou seja, 3.000 valores de NMSE das combinações de parâmetros selecionada na busca. Após executado as 100 vezes para os 38 problemas, foi calculada a média aritmética para cada um e a média das médias.

No entanto, cada vizinho é incluído na lista das soluções diferentes, pois, cada solução vizinha visitada é uma solução que será executada no SVM, portanto é incluída na lista, mesmo sabendo que das quatro soluções vizinhas, duas não são interessantes, levando em consideração que uma solução será o melhor caminho e a outra já foi adicionada na lista na iteração anterior.

O Gráfico 1 foi construído a partir da média das médias dos 38 problemas e exhibe o desempenho do algoritmo com 100 execuções em relação ao método convencional de busca aleatória. Os números na vertical são intervalos dos valores de desempenho para as configurações de parâmetros C e γ executada, enquanto que os valores na horizontal representam o número de soluções distintas visitadas.

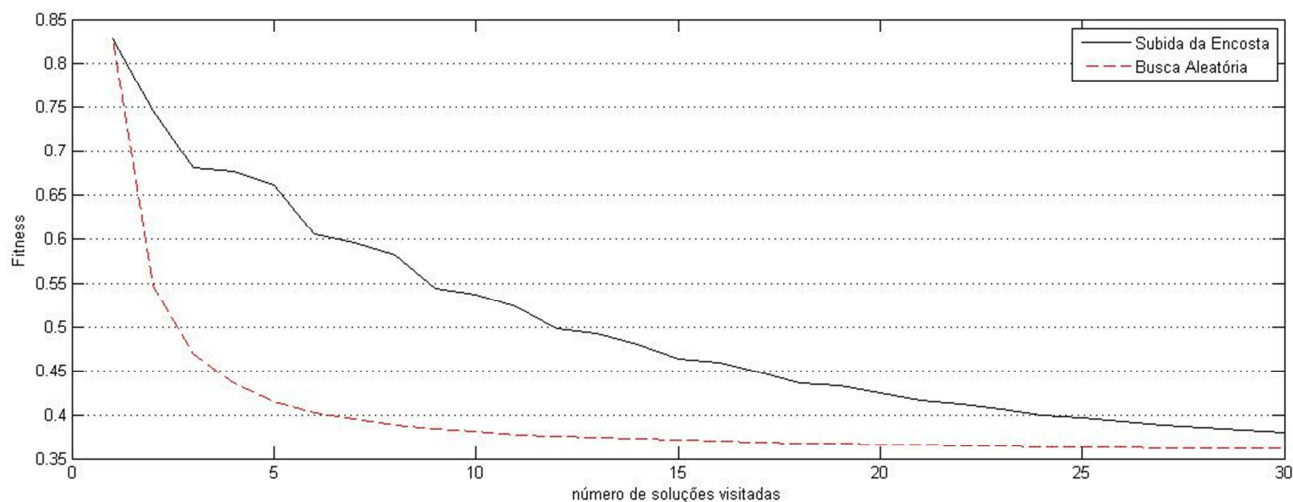


Gráfico 1 – Subida da Encosta x Busca Aleatória

Nota-se que a busca aleatória obteve um desempenho melhor que o algoritmo Subida da Encosta devido à natureza de seu processo ser por busca local, ou seja, para cada solução é definida a vizinhança a ser executada no SVM e acrescentado ao número de soluções visitadas definido para os experimentos, o que ocasiona uma evolução mais lenta.

Para tentar resolver esse problema também foi implementado uma versão do algoritmo Subida da Encosta otimizada, ou seja, a solução corrente inicial deixará de ser totalmente aleatória e passará ser direcionada. A solução corrente inicial é selecionada a partir de um grupo de 38 combinações de parâmetros C e γ considerados boas soluções para os problemas, para encontrar essas combinações os problemas foram executados a priori e armazenado as melhores soluções, com isso, a escolha do nó inicial é realizada aleatoriamente entre as melhores eliminando somente a combinação de parâmetro do problema em questão. Após a escolha do nó corrente é definida a vizinhança dando um pulo sobre os vizinhos que estão mais próximos, para tornar o processo de busca mais rápido. Os nodos gerados a partir do corrente serão testados e o que tiver melhor valor de *fitness* será a novo nó corrente.

O Gráfico 2 abaixo representa o desempenho do algoritmo Subida da Encosta otimizada com passos largos, sem início direcionado e Busca Aleatória, calculado a partir da média das médias da execução dos 38 problemas

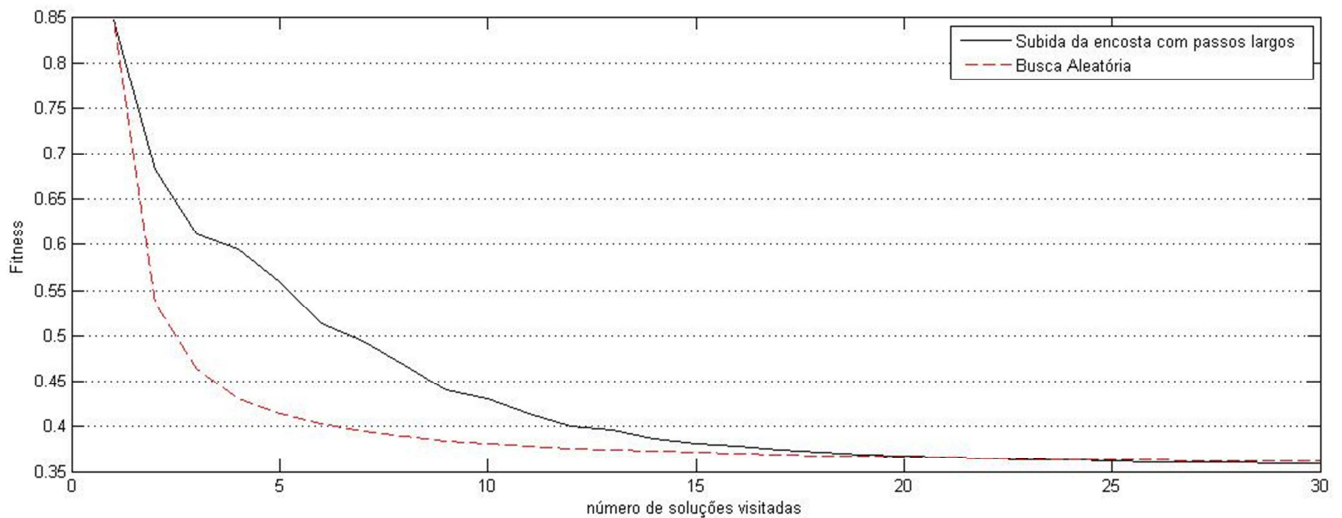


Gráfico 2 – Subida da Encosta com passos largos x Busca Aleatória

O algoritmo Subida da Encosta otimizado obteve uma melhora no seu desempenho, sendo perceptível no Gráfico 2 a regressão da curva com maior velocidade em relação ao Gráfico 1, assim, agilizando o processo da busca. No entanto, nota-se que a Busca Aleatória permanece melhor na maioria das execuções devido à observação sobre o embasamento da Subida da Encosta ser de busca local, uma busca mais refinada.

O Gráfico 3 apresenta o desempenho entre Subida da Encosta otimizada com início direcionado e passos largos e Busca Aleatória, calculado através da média das médias dos 38 problemas com 100 execuções.

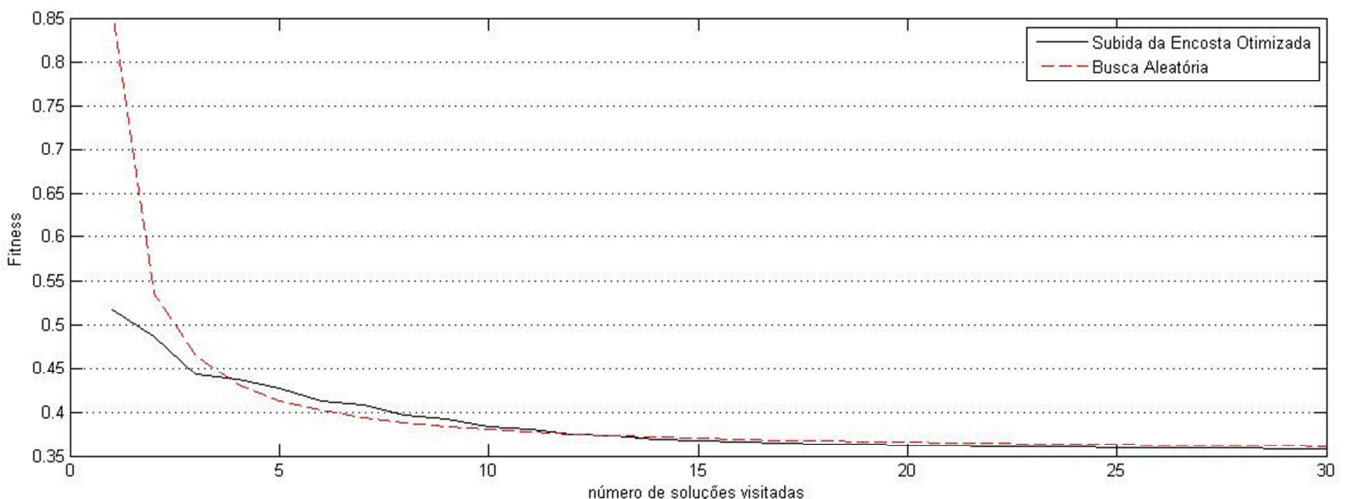


Gráfico 3 – Desempenho Subida da Encosta com início direcionado e passos largos x Busca Aleatória

Como nas outras versões do algoritmo, neste caso a curva não se modifica consideravelmente, no entanto permanece melhor na maior parte do experimento. Para melhor análise foi criado um histograma contendo o número de vitórias entre os algoritmos, em cada iteração de uma execução o algoritmo que obter o melhor valor de desempenho é o vitorioso, deste modo em 30 iterações para 100 execuções o número máximo de vitórias que um algoritmo pode atingir é de 3 mil.

No Gráfico 4 referente ao histograma da quantidade de vitórias entre o algoritmo Subida da Encosta Otimizada e Busca Aleatória.

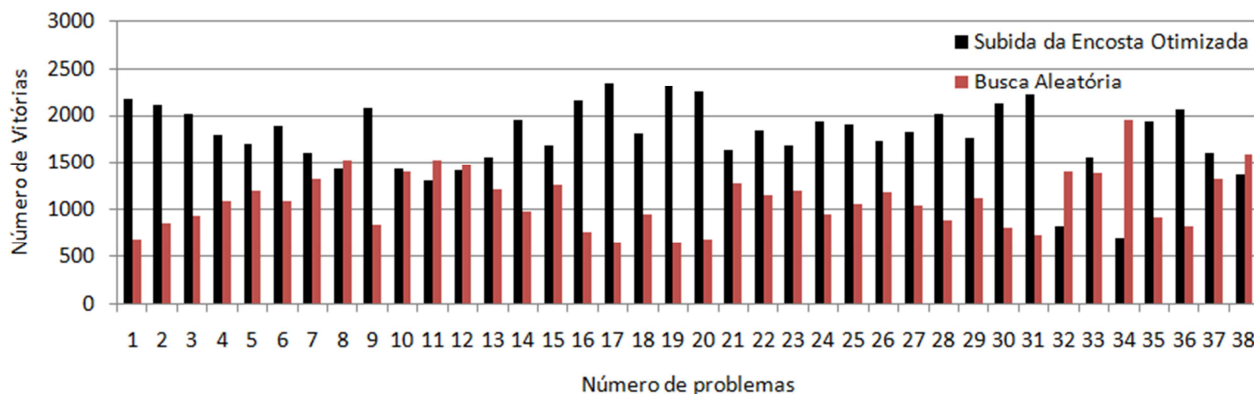


Gráfico 4 – Nº de vitoriosos entre Subida da Encosta Otimizada x Busca Aleatória

Nesta comparação é perceptível que a maioria dos problemas o algoritmo Subida da Encosta otimizado apresenta maior número de vitórias com exceção dos problemas 32 e 40 que são *outliers* da base de dados e apresentaram comportamento diferenciado em relação aos demais problemas.

O Gráfico 5 é apresenta a performance do algoritmo Subida da Encosta tradicional e o otimizado, sendo notável a diferença da evolução da curva e a melhora do algoritmo para a resolução do problema de otimização de parâmetros de SVM.

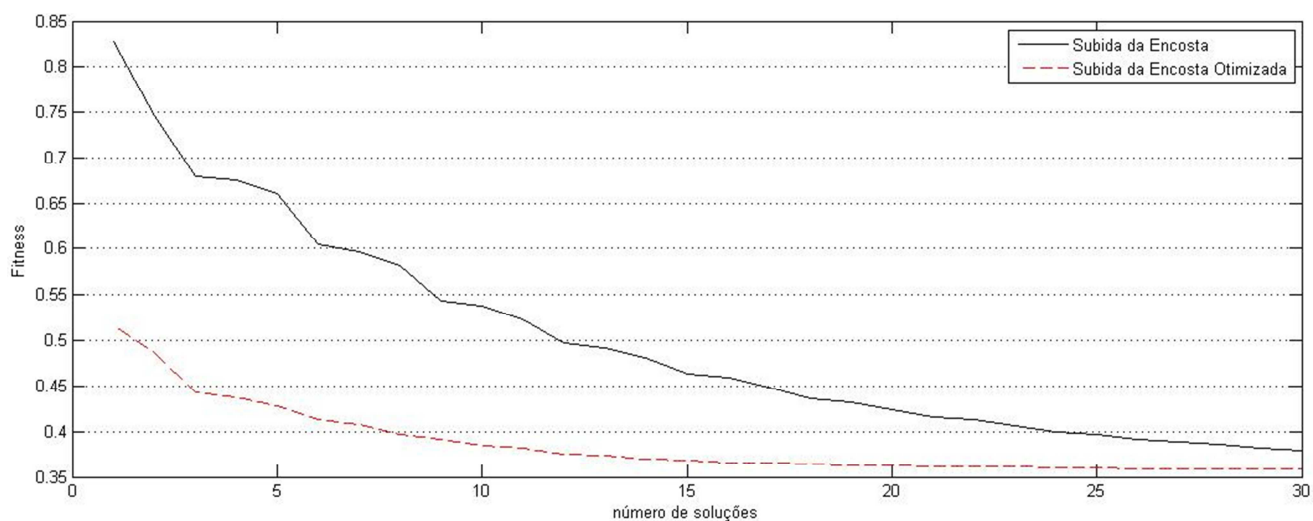


Gráfico 5 – Desempenho Subida da Encosta x Subida da Encosta com inicio direcionado e passos largos

6 Conclusão

Neste artigo foi apresentado um sistema híbrido para seleção de parâmetros de Máquinas de Vetores Suporte utilizando a técnica de busca meta-heurística Subida da Encosta. No início da investigação foram realizados experimentos com o algoritmo para testar a viabilidade do mesmo para o problema. No entanto, foram obtidos resultados pouco satisfatórios, então foi desenvolvido uma versão otimizada do algoritmo para melhorar o processo de busca.

Na maioria dos trabalhos encontrados na literatura os algoritmos para seleção de parâmetros seguem sua busca até alcançar um determinado valor como critério de parada, que normalmente são bem extenso como 100 gerações, 1.000, 3.000 gerações, etc. ou até mesmo até encontrar a melhor solução. Para um sistema híbrido com SVM esse método torna-se muito custoso e demorado principalmente se a base de dados utilizada for muito grande.

Nesse contexto, foi estipulado neste trabalho que a execução do algoritmo termina quando forem testadas 30 combinações de parâmetros C e γ diferentes, sendo este um número considerado muito abaixo do que normalmente é utilizado.

Com os experimentos realizados, foi possível perceber que o algoritmo Subida da Encosta é inviável para a seleção de parâmetros com um critério de parada com poucas iterações. No entanto, a versão otimizada do algoritmo Subida da Encosta de acordo com os experimentos realizados é viável para o problema, principalmente pelos resultados satisfatórios obtidos e a simplicidade do seu processo.

7 Referências

- [1] CRISTIANINI, N.; SHAWE-TAYLOR, J. An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods: Cambridge University Press, 2000.
- [2] BOSER, B.; GUYON, I; Vapnik, V. A training algorithm for optimal margin classifiers. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pages 144–152. ACM Press, 1992.
- [3] Vapnik, V. The nature of statistical learning theory. Springer Verlag, 1995.
- [4] SANTOS, C. R. Análise Automática de Assinaturas Manuscritas Baseada nos Princípios da Grafoscopia. Dissertação de Mestrado, Pontifícia Universidade Católica do Paraná, Brasil, 2004.
- [5] HAYKIN, S., Neural Networks - A Comprehensive Foundation. Prentice-Hall, New Jersey, 2nd edition, 1999.
- [6] VAPNIK, Vladimir N. An Overview of Statistical Learning Theory. IEEE Transactions on Neural Networks, vol. 10, n°. 5, 1999.
- [7] WEISE, T. Global Optimization Algorithms: Theory and Application, 2nd Edition, 2009. p. 22 – 28.
- [8] RUSSEL S.; NORVIG P. Artificial Intelligence A modern Approach, segunda edição, cap 4, 1995.