Self-Generating Prototypes using a Adaptative Distance Approach

Cristiano de S. Pereira1, Dayvid Victor R. de Oliveira1 , Guilherme R. Magalhães1, George D. C. Cavalcanti1

1Centro de Informática – Universidade Federal de Pernambuco (UFPE)  
Cidade Universitária– 50.740-560– Recife – PE – Brazil

{cdsp,dvro,grm, gdcc}@cin.ufpe.br

**Abstract.** This article presents an analysis of the use of prototype-based classifiers. Different techniques have been proposed using this approach. We devote a section to the analysis of the prototype-selection technique of the "creative" class known as Self-Generating Prototypes (SGP), where we discuss the advantages and disadvantages of this method. A change in the SGP algorithm is proposed to use adaptive distances in the implementation and classification steps of the test dataset as a way to improve the hit rate and reduce the disadvantages. To compare the results we use the classical algorithm K-NN.

**Resumo.** Este artigo apresenta uma análise da utilização de classificadores baseados em protótipos. Diferentes técnicas foram propostas utilizando essa abordagem. Dedicamos uma seção à análise da técnica de seleção do tipo “creative” conhecida como Self-Generating Prototypes (SGP), onde discutimos as vantagens e desvantagens deste método. Uma modificação no algoritmo do SGP é proposta para utilizar distâncias adaptativas nas fases de execução e classificação do conjunto de testes como forma de melhorar a taxa de acerto e atenuar as desvantagens. Para a comparação de resultados utilizamos o algoritmo clássico K-NN.

# 1. Introdução

# Seleção de Protótipos é uma técnica de aprendizagem de máquina, cujo propósito principal é escolher vetores de padrões (protótipos) a partir do conjunto de dados de treinamento para alcançar uma melhor disposição dos dados e separação das classes. O objetivo deste tipo de técnica é encontrar o menor conjunto de protótipos que minimiza o erro de classificação. A classificação pode ser obtida pela estratégia do “protótipo mais próximo”. Uma das principais vantagens de classificadores baseados em protótipos é a baixa demanda por espaço de armazenamento e recursos computacionais.

# A literatura separa as estratégias para seleção de protótipos em dois tipos: “Seletivas” e “Criativas”. O primeiro esquema é chamado puramente seletivo, pois o conjunto resultante da seleção é formado por dados presentes na base de treinamento original. Em esquemas “criativos”, novos protótipos são gerados durante o processo de redução, combinando dados e fazendo ajustes durante o treinamento supervisionado. Técnicas “criativas” podem ser usadas para gerar novos dados e adicioná-los à base atenuando problemas de desbalanceamento entre classes (grande diferença na quantidade de instâncias de cada classe) e de poucas instâncias na base.

# A abordagem para seleção de protótipos proposta por Fayed baseada em geração automática, conhecida como Self–Generating Prototypes (SGP) [Fayed et al. 2007]. A principal vantagem dessa técnica é a abstenção de parâmetros não requerendo um posicionamento inicial. Os protótipos são gerados utilizando um conjunto de regras simples e de fácil entendimento que dividem, combinam e removem grupos de dados, agindo assim como um esquema de redução. Outra propriedade conhecida do SGP é o baixo tempo gasto no processo de treinamento. O trabalho apresentado neste artigo busca combinar as vantagens de técnicas de seleção de protótipos a fim de avaliar o resultado de tais técnicas.

# A próxima seção é dedicada ao SGP apresentando informações detalhadas do algoritmo e de sua variação o SGP2. A Seção 3 descreve a medida de distância adaptativa . Uma justificativa da tentativa de junção de técnicas é apresentana na seção 4. A Seção 5 descreve os experimentos, dados utilizados, suposições realizadas e a análise dos resultados obtidos. Finalmente a Seção 6 sumariza as considerações finais e sugestão de trabalhos futuros.

# 2. Self-Generating Prototypes (SGP)

Alguns dos métodos de aprendizagem de máquina dependem da escolha correta do número de protótipos por classe e sua localização. Muitas vezes torna-se difícil encontrar valores ótimos para os parâmetros e o algoritmo deve rodar um grande número de vezes usando diferentes conjuntos de parâmetros para tanto. Isto leva um longo tempo e aumenta o *overhead* computacional da aplicação. A abordagem do SGP (Self-Generationg Prototypes) [Fayed et al. 2007] tenta superar esses problemas sendo sua principal vantagem que o número de protótipos e suas localizações são obtidas durante a a fase de treinamento sem a intervenção humana e é indepente da ordem dos dados.

A idéia principal do método é formar um certo número de grupos, inicialmente formados pelos padrões do conjunto de dados de treinamento de mesma classe, e fazer do centróide de cada grupo o protótipo representante. Posteriromente utilizando sucessivas separações, deslocamento de padrões entre os grupos e operações de mesclagem (merge) como etapa de poda (prunning). Todas estas operações são bastantes simples e podem ser classificadas de acordo com as quatro situações abaixo:

* Se para todos os padrões de um grupo o protótipo mais próximo é o centróide do grupo, então nenhuma operação é realizada.
* Se para todos os padrões de um grupo o protótipo mais próximo é de uma classe diferente da do grupo, ele é dividido em dois subgrupos. Essa divisão é feita separando os padrões pelo hiperplano que passa pelo centroide do grupo, e cujo vetor normal é a primeira componente principal gerada pelos padrões do grupo.

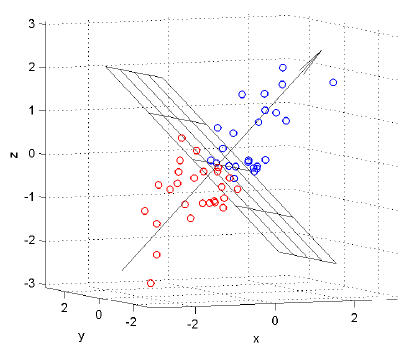


Figura 1 - Etapa da divisão de grupos. A seta representa a primeira componente principal.

* Se para alguns padrões de um grupo o protótipo mais próximo é diferente do centróide, mas da mesma classe, esses padrões são deslocados do grupo original para o grupo do protótipo mais próximo
* Se para alguns padrões de um grupo o protótipo mais próximo não é o centróide e é de uma classe diferente, estes padrões são removidos do grupo original e formam um novo grupo, sendo o centróide computado como um novo protótipo.

Em cada um dos casos aprensentados, os centróides de cada grupo é recomputado ao final de uma iteração para atualizar os protótipos de saída. O processo inteiro é repetido até que nenhuma mudança ocorra com os grupos.

## 2.1. O Algoritmo SGP1

1. Carregue sua Base de treinamento e divida-a em grupos onde e número total de conjuntos
2. Compute os protótipos iniciais, onde
3. Faça (Número de classes)
4. Compute as distâncias de cada elemento do grupo para os centróides representantes.
5. Determine e guarde o índice do protótipo mais próximo de cada instância daquele grupo.
6. Se para toda instância do grupo o padrão mais próximo é o centróide do grupo ao qual ela pertence, vá para o passo 10.
7. Se para todas as instâncias de um grupo a classe do protótipo mais próximo for diferente da do protótipo centróide, faça , crie um novo grupo separe o grupo por um hiperplano que passa pelo centróide e cujo vetor normal é a primeira componente principal, formando dois novos subgrupos e . Após essa etapa atualize os centróides dos novos supgrupos. As classes dos protótipos do novos subgrupos é igual a classe do protótipo do grupo antes da divisão. Vá para o passo 4.
8. Se para algumas instâncias do grupo, o protótipo mais próximo é o representante de outro grupo mas possui a mesma classes, remova esses padrões de e inclua-os no grupo do protótipo mais próximo. Atualize os protótipos de cada grupo.
9. Se para algumas instâncias do grupo, o protótipo mais próximo é o representante de outro grupo e ele é de uma classe diferente, faça , remova esses padrões de e crie um novo grupo contendo esses padrões. Atualize os protótipos de cada grupo, , .
10. Se (chegou no último grupo) e nenhuma mudança foi realizada nos grupos ou protótipos, então PARE.
11. Se não for o último grupo, então incremente k e vá para o passo 4.
12. Se k = M, então faça k = 1 e vá para o passo 4.

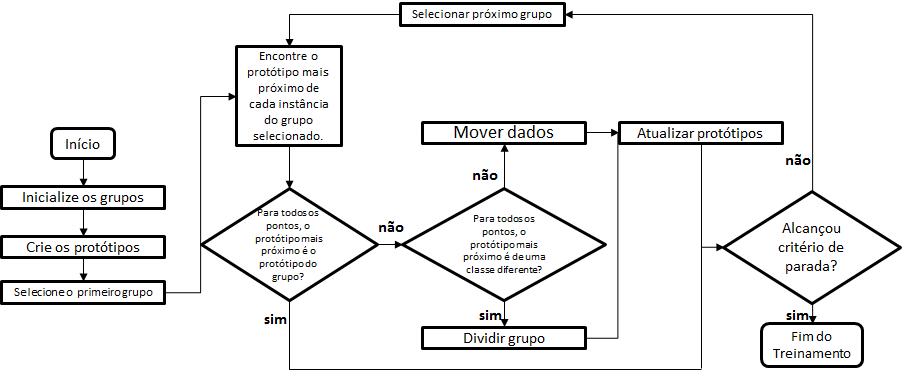


Figura 2 - Fluxo de Passos do SGP

## 2.2. O SGP2

Após a execução do algoritmo acima, um passo de merging pode ser aplicado para reduzir o número de protótipos. Os Grupos A e B podem ser fundidos se ambos tiverem a mesma classe e se o segundo protótipo mais próximo dos padrões do grupo A for o protótipo (centróide) do grupo B (Pb), e o segundo protótipo mais próximo dos padrões do grupo B for o centróide do grupo A (Pa).

Um passo de *prunnig* (poda) também pode ser utilizado, se o segundo protótipo mais próximo de todos os padrões de um certo grupo possuem a mesma classe, o grupo e seu protótipo são removidos. O algoritmo SGP2 é o algoritmo SGP com as etapas de merging e prunnig.

Para evitar o sobre-ajuste e perda de generalização introduziu-se [Fayed et al. 2007]dois parâmetros ao SGP chamados de e . O representa um limite para o tamanho relativo de um grupo em relação ao maior grupo. Por exemplo, se é 0.1 e o tamanho do maior grupo é 200, todos os grupos com tamanho menor que são descartados. O segundo parâmetro é um limiar para a taxa de classificações incorretas de um grupo. Se o número de padrões classificados erroneamentes em um grupo dividido pelo tamanho do grupo é menor que , então o grupo permanece inalterado. Este parâmetros reduzem o número de protótipos e aumentam a capacidade de generalização. Como recomendado em [Pereira e Cavalcanti 2008] em nossos experimentos, os valores do parâmetros foram variados no intervalo [0.01, 0.2].

# 3. Regra adaptativa do “vizinho” mais próximo

Seguindo a estratégia Adaptative Nearest Neighbor [Wang et al. 2007], vamos assumir que os padrões a serem classificados são representados como vetores em um espaço euclidiano d-dimensional . Dado um conjunto de instâncias de treinamento e o padrão de consulta , a regra do K-NN primeiramente encontra os vizinhos mais próximos de , denotados por ,...,, e associa a classe majoritária entre ,...,, onde é a classe correspondente de . Sem conhecimento a priori a distância euclidiana

(1)

Para definir a distância adapatativa local entre um padrão de consulta e um exemplo de treinamento , nós primeiramente construímos a maior esfera centrada em que exclui todos os padões de treinameto de outras classes. Isto pode ser facilmente conseguido fazendo o raio da esferea , onde é um número pequeno arbitrário. Note que dependendo da métrica que é utilizado, as regiões definidas por pontos com distância a menor que pode não ser uma esfera. No entanto, por simplicidade, nos referimos covenientemente a tais regiões como esferas quando isso não causa confusão. A distância adaptativa local entre e o exemplo de treinamento é definida como

(2)

Notamos que, de acordo com a medida local da distância adaptativa, cada padrão de treinamento é associado como um padrão . A maior esfera associada com cada padrão de treinamento define a maior região esférica dentro da qual sua classe pode ser genralizada para outros padões de treinamento de forma confiável, i.e., sem cometer um erro. É fácil perceber que as esferas associadas com padrões de treinamento mais na parte interior das classes terão relativamente um raio maior que aquelas próximas as regiões de fronteira. Como resultado do fator de escalamento , padrões de treinamento que estão muito distantes de um padrão de consulta usando a distância euclidiana podem, tornar-se próximos do padão de consulta utilizando a distância adaptativa se a esfera associada a elas for grande o suficiente. Próximo a região de fronteira onde diferentes classes podem se sobrepor ou o nível de ruído é alto esta característica é benéfica porque tende a identificar exemplos de treinamento com esferas relativamente grandes como vizinhos mais próximos. Já para um padrão de consulta localizado em uma posição distante das outras classes, a medida da distância adptativa raramente muda o resultado da classificação porque todos os vizinhos próximos são da mesma classe.

# 4. Motivos para utilizar o SGP com distâncias adptativas

Apesar das vantagens apresentadas pelo SGP, ele possui alguns aspectos a serem melhorados. O algoritmos SGP pode apresentar problemas quando aplicado a domínios desbalanceados. As vezes clusters inteiros podem desaparecer como na Figura abaixo:

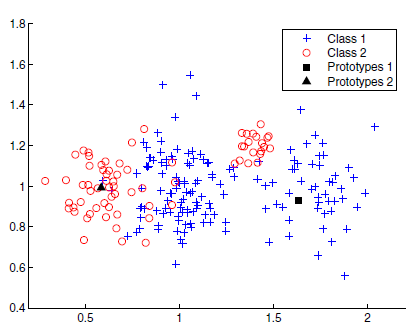


Figura 3 - Simulação do problema do SGP em um conjunto de dados 2-D. Note que o pequeno cluster da classe círculo não é representados por nenhum protótipo e consequentemente todos os padrões do cluster são classificados erroneamente.

Por isso é interessante adaptar o SGP para utilizar alguma técnicas de ajuste-fino. Nossa abordagem é de utilizar distâncias adaptativas para verificar mudanças nas classificações dos protótipos.

# 5. Experimentos e Resultados

## 5.1. Bases de dados

Utilizamos como base de dados para validar o funcionamento da nossa implementação do SGP as seguintes bases do UCI Machine Learning Repositório:

Cancer

Diagnóstico de câncer de mama. Formando por descrição de células obtidas por exame microscópico. Composto por 699 instâncias, possui nove características e duas classes (Tumor maligno ou benigno).

Diabetes

Predizer se um indivíduo indío da raça da tribo Pima possui ou não diabetes baseado em características como idade, pressão sanguínea, índice de massa corpórea, etc. Composto por 768 instâncias, possui oito características e duas classes (positivo ou negativo).

Para a divisão das bases em conjuntos de treinamento e teste, utilizamos uma estratégia de validação cruzada separando a base em 10 folds (mantendo a proporção entre as classes da base original) e utilizando sempre um para teste e os nove restantes para treinamento. Para as taxas de acerto tivemos que obter a média e desvio padrão para a classificação de cada fold.

## 5.2. Experimentos

Experimentamos para o conjunto de dados acima descrito vária versões do SGP: O SGP1 que é versão mais simples sem etapa de prunning e merge, o SGPM que inclui a etapa de merge; SGP2 com etapa de merge e prunning e finalmente o ASGP que baseia-se no SGP1 e utiliza distância adptativa para avaliar os protótipos mais próximos e para a classificação pela regra adptativa do vizinho mais próximo.

Para execução do SGP2, utilizamos os parâmetros Rmin = 0.2 e Rmis = 0.2, para obtermos os casos extremos de podagem.

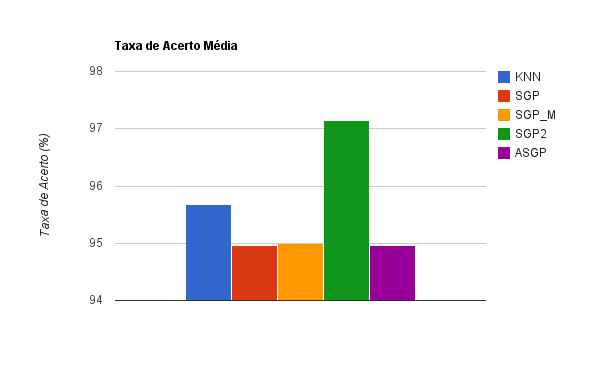
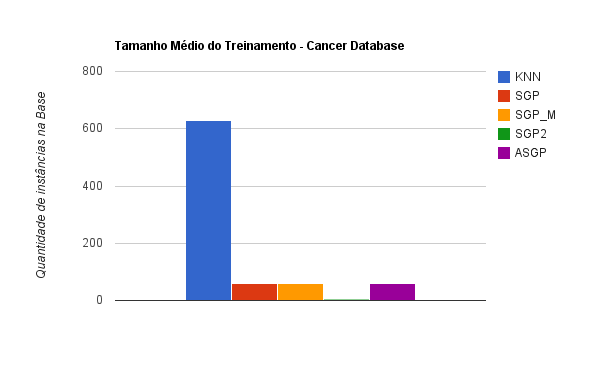
Para cada um dos algoritmos avaliamos a taxa de acerto e o número de protótipos gerados no conjunto final.

## 5.3. Resultados

A seguir apresentamos os resultados obtidos em cada base.

Tabela 1. Cancer Dataset

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Algoritmo** | KNN | SGP1 | SGP1M | SGP2 | ASGP |
| **Taxa de Acerto** | 95,68 ± 2,13 | 94,96 ± 1,60 | 95,00 ± 1,50 | **97,14 ± 1,16** | 94,96 ± 1,60 |
| **Tamanho da Base** | 629 ± 0,57 | 60,5 ± 5,25 | 58 ± 5,30 | **2 ± 0** | 60,5 ± 5,25 |



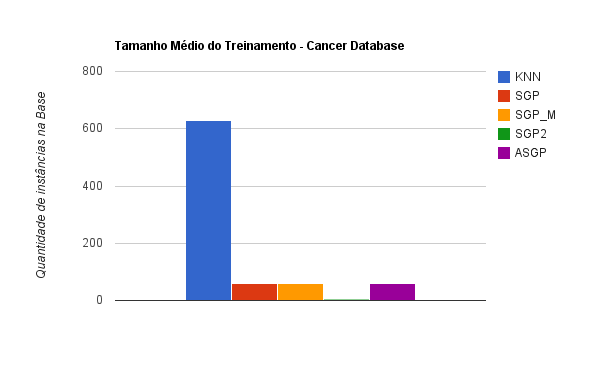
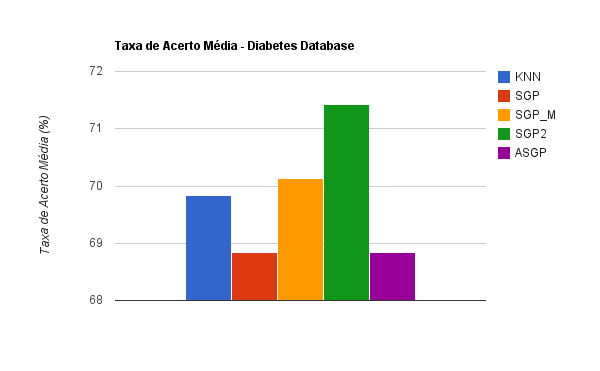


Figura 4. Gráficos Cancer Dataset

Tabela 2. Diabetes Dataset

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Algoritmo** | KNN | SGP1 | SGP1M | SGP2 | ASGP |
| **Taxa de Acerto** | 69,83 ± 3,24 | 68,83 ± 5,16 | 70,13 ± 24,72 | **71,43 ± 4,49** | 68,83 ± 5,16 |
| **Tamanho da Base** | 691 | 294 ± 8,03 | 270 ± 8,86 | **96 ± 28,60** | 294 ± 8,02 |



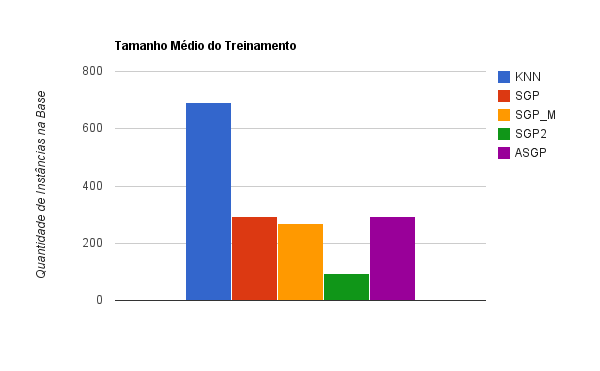


Figura . Gráficos Diabetes Dataset

Dos resultados podemos concluir que o SGP provê uma boa redução no conjunto de treinamento e mantém uma boa taxa de acerto. Como foi utilizada validação cruzada, os dados obtidos possuem uma boa representatividade do desempenho do algoritmos.

Para o Cancer dataset, observa-se que o SGP1 diminuiu em 10 vezes a base de dados, e a taxa de acerto foi muito semelhante. O SGP1 com merging aumentou minimamente a taxa de acerto, e diminuiu a base de dados em 2 instâncias, em média. E o melhor desempenho obtido foi o SGP2 que com apenas 2 protótipos, ou seja, aproximadamente 0.31% da base original, e ainda assim obteve a maior taxa de acerto média de 97.17%, chegando a picos de até 98.57%.

O Adaptative SGP, proposto neste artigo obteve exatamente o mesmo desempenho do SGP1. Essa observação é justificada pelo fato de classificadores baseados em protótipos utilizarem apenas o primeiro protótipo mais próximo para classificação e o fato da redução do número de protótipos diminuir a eficácia do fator de escalonamento uma vez que temos pouquíssimos pontos no interior das classes.

Para o Diabetes Dataset, observa-se que o SGP1, em média, obteve uma taxa de acerto menor do que o KNN, mas reduziu a base de dados em 57.45%, tornando a comparação menos custosa. Em alguns momentos o SGP1 obteve taxas melhores que o KNN, com picos de 75.32%. O SGP1 com merge obteve uma taxa de acerto média superior ao KNN e ao SGP1 e diminuiu a base em 60.92%.

Assim como na base do Cancer, o SGP2 obteve o melhor desempenho na base Diabetes. O SGP2, obteve a maior taxa de acerto média, 71.43%, chegando a um pico de 81.82%, uma excelente taxa para o conjunto de dados abordado. Observamos que o prunning teve um papel fundamental nesta base de dados, pois mesmo diminuindo a base para 13.9% da base original, o desempenho não foi comprometido.

Pelo mesmo motivo mencionado na base anterior, observamos que o Adaptative SGP não apresentou mudanças em relação ao SGP original.

# 6. Conclusão

Verificamos que o SGP é uma técnica de seleção de protótipos bastante interessante e que apresenta como principal vantagem a independência de intervenção humana (ajustes de parâmetros), sendo os parâmetros do SGP2 pouco influentes para a taxa de acerto, sendo mais eficazes com relação ao número de protótipos. Outra característica, comprovada empiricamente, é que os protótipos gerados pelo SGP1 por sí sós, representam toda a base de dados original, sendo a taxa de acerto para quando o conjunto de testes está contido no conjunto de treinamento original de 100%.

A idéia báseica do SGP é dividir os padrões em grupos, realocar padrões entre os grupos, dividir e criar grupos e por fim, eliminar grupos desprezíveis. O algorítmo é bastante simples e em geral, obtém performace superior á várias técnicas de protótipos.

# 7. Referências

Wang, J., Neskovic, P. e Cooper. L. (2007) “Improving nearest neighbor rule with a simple adaptive distance measure”, Em: Pattern Recognition Letters, vol. 28, p. 207–213.

Pereira, C. de S. e Cavalcanti, G. D. C.(2008) “Prototype selection: Combining self-generating prototypes and Gaussian mixtures for pattern classification”,Em: IJCNN, p. 3505-3510.

Hatem, S. R. H., Fayed, A. e Atiya, A. F.(2007) “Self-generating prototypes for pattern classification”, Em: Pattern Recognition, vol. 40, no. 5, p. 1498–1509.