

# Reconhecimento de Padrões em Classificadores – Comparação de Técnicas e Aplicações

R. L. Stange, J. J. Neto

**Resumo** — O reconhecimento de padrão utiliza diferentes abordagens em classificadores, a estatística e a sintática são as abordagens de maior interesse neste trabalho. A escolha de um método de classificação depende da natureza do problema. Este trabalho tem o objetivo de comparar técnicas utilizadas para a construção de classificadores e de suas aplicações. No final, é apresentada uma proposta para construção de classificadores, baseada em métodos híbridos incluindo métodos adaptativos.

**Palavras chaves** — Adaptatividade, Classificadores, Métodos de Reconhecimento de Padrão, Técnicas de Aprendizagem.

## I. INTRODUÇÃO

O PROBLEMA de reconhecimento de padrão é um problema de classificação. O objetivo é discriminar amostras de objetos e classificar corretamente amostras futuras. Objetos ou indivíduos são caracterizados por um conjunto de atributos. Cada atributo possui um valor. Os atributos podem ser categóricos, binários, ordinais ou contínuos. Em alguns casos, é necessária a conversão dos valores de atributos (ex: discretização, binarização).

O estudo em reconhecimento de padrão está em constante desenvolvimento e tem alcançado diversas áreas de aplicação, tais como, biologia, medicina, agricultura, entre outras.

Atualmente, diferentes técnicas provenientes de diversas áreas de pesquisa são utilizadas para o desenvolvimento de algoritmos computacionais para reconhecimento de padrões. As principais abordagens segundo Jain et al. [3] são: *estatística*: baseada nos modelos probabilísticos para geração de padrão e na teoria da decisão (ex: aprendizagem bayesiana); *sintática e estrutural*: fundamentada principalmente na teoria das linguagens formais (ex: aprendizagem de árvores de decisão, inferência gramatical); *neural*: o conjunto de pesos é a base para geração de padrão na abordagem neural. Essa abordagem incorpora as propriedades das redes neurais para classificação (ex: redes neurais artificiais). Alguns autores incluem a abordagem baseada na lógica fuzzy para classificação não-determinística, chamada de abordagem *difusa*. Além dessas, a principal abordagem para aprendizagem não-supervisionada é conhecida como *Clustering* [1].

A finalidade deste trabalho é comparar técnicas para reconhecimento de padrão provenientes de diferentes áreas de aplicação.

## II. ABORDAGEM GERAL SOBRE RECONHECIMENTO DE PADRÃO

A definição de aprendizagem de máquina para Mitchell [5] é “um programa de computador é dito aprender a partir de uma experiência  $E$  com respeito a uma classe de tarefas  $T$  e medida de desempenho  $P$ , se seu desempenho nas tarefas em  $T$ , segundo a medida  $P$ , melhora com a experiência  $E$ ”. Para Witten et al. [13] a aprendizagem é a capacidade que os programas de computador têm de aprender coisas que mudam seu comportamento e melhoram seu desempenho futuramente. Na aprendizagem de máquina, de acordo com Russel et al. [9], dispositivos computacionais se adaptam a novas circunstâncias para detectar padrões.

O campo da aprendizagem de máquina incorpora o reconhecimento de padrões. O reconhecimento de padrões considera a capacidade de programas de computador encontrar regularidades através de experiências e melhorar automaticamente [5].

O reconhecimento de padrão de acordo com Theodoridis et al. [10] é a descoberta automática de regularidades em dados através de algoritmos computacionais e uso dessas regularidades para classificar objetos em categorias ou classes. O termo genérico padrão é utilizado para referir-se a esses objetos. A figura 1 apresenta um diagrama geral para sistemas de reconhecimento de padrão [1].

O processo de reconhecimento de padrão é dividido basicamente em duas fases: treinamento (aprendizagem) e reconhecimento (classificação). A fase de reconhecimento ou classificação tem início com a aquisição dos dados e sensoriamento, nesta tarefa ocorrem as medições das variáveis físicas. Em seguida um pré-processamento é realizado para remoção de ruídos nos dados. Além disso, qualquer outro tratamento realizado sobre os dados capturados ocorre nesta fase. Na extração de características é necessário encontrar uma nova representação para o padrão em termos de funcionalidades. No modelo de aprendizagem é necessário mapear as características entre grupos de padrões e categorias. A classificação usa as características e o modelo de aprendizagem para atribuir uma categoria a um padrão. Em seguida é realizada a avaliação de confiança nas decisões.

R. L. Stange é estudante de Mestrado em Engenharia Elétrica da Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Computação e Sistemas Digitais, Av. Prof. Luciano Gualberto, travessa 3, número 158 - Cidade Universitária - São Paulo - SP - CEP: 05508-900 ([rlstange@usp.br](mailto:rlstange@usp.br)).

J. J. Neto atua no Departamento de Engenharia de Computação e Sistemas Digitais (PCS) da Escola Politécnica (POLI) da Universidade de São Paulo (USP), Av. Prof. Luciano Gualberto, Trav. 3, N. 158 - CEP: 05508-900 - São Paulo/SP, Brasil. ([joao.jose@poli.usp.br](mailto:joao.jose@poli.usp.br)).

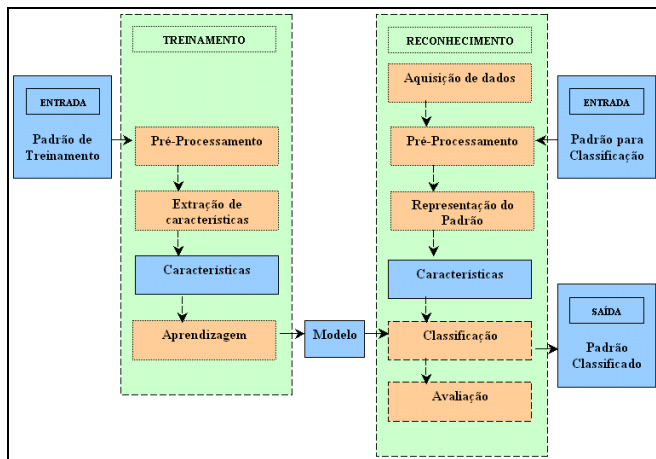


Figura 1 – Sistema de reconhecimento de padrão

Na componente de treinamento a finalidade é inferir, a partir de um conjunto de exemplos de treinamento, um conjunto de regras que define um padrão. O resultado final do treinamento é um modelo de aprendizagem que define o classificador. Na componente de reconhecimento, o classificador é utilizado com a finalidade de classificar cadeias de entrada de acordo com critérios definidos durante sua construção. A cadeia de entrada é uma sequência de estímulos que alimenta o dispositivo. O modelo apresenta duas entradas. Na entrada para treinamento, exemplos de um padrão, chamados exemplos positivos, são inseridos para aprendizagem. Em alguns modelos são inseridos exemplos que não pertencem ao padrão, esses exemplos são conhecidos como exemplos negativos. Na entrada para reconhecimento, um padrão não classificado é inserido para ser classificado de acordo com o modelo de aprendizagem definido durante o treinamento. O critério de classificação é definido a partir das diferentes técnicas de reconhecimento de padrão.

A seguir, são apresentadas técnicas de reconhecimento de padrão utilizando métodos estatísticos e métodos determinísticos.

#### A. Aprendizagem baseada em métodos estatísticos

Os métodos estatísticos tendem a obter resultados rapidamente. Os métodos estatísticos são apropriados quando o foco é evidenciar tendências no espaço amostral e quando as particularidades dos indivíduos são irrelevantes.

Na abordagem estatística, cada padrão é representado em termos de  $d$  características ou medições, onde  $d$  é o número de características. O padrão é visto como um ponto em um espaço  $d$ -dimensional. A finalidade é escolher características que separem vetores de padrão, pertencentes a diferentes categorias, em regiões compactas. A eficácia da representação do espaço (conjunto de características) é determinada por quão bem os padrões de classes distintas podem ser separados. Dado um conjunto de padrões de treinamento, o objetivo é estabelecer fronteiras de decisão no espaço de característica que separe padrões pertencentes a diferentes categorias [1] [3].

Na abordagem baseada em teoria de decisão estatística o Teorema de Bayes é aplicado e as fronteiras de decisão são determinadas pelas distribuições de probabilidade de padrões pertencentes a cada classe [1][5]. Além da teoria da decisão,

uma abordagem baseada na análise discriminante para a classificação pode ser adotada, neste caso podemos utilizar o Discriminante Linear de Fisher [1].

Considere os padrões  $X = (x_1, x_2, \dots, x_d) \in \mathbb{R}^d$  e as classes  $c = \{\omega_1, \omega_2, \dots, \omega_c\}$ , em reconhecimento de padrões. Estamos interessados em associar um padrão a uma classe. A abordagem Bayesiana supõe que as probabilidades de cada classe  $P(\omega_i)$  e as densidades de probabilidade condicionais  $p(x|\omega_i)$  de  $x$  com respeito a cada um das classes  $\omega_i$   $i = 1, 2, \dots, c$ , são conhecidas. Na ausência de qualquer outra informação, poderíamos classificar um padrão  $x$  como sendo da classe  $\omega_i$  de maior probabilidade. Porém, dado que  $x$  foi observado, isto parece uma decisão muito ingênua, pois o classificador acertaria a classificação com probabilidade  $P(\omega_i)$ , porém erraria sempre com probabilidade  $\sum_{i \neq j} P(\omega_j)$ . Como temos as condicionais, podemos utilizar o teorema de Bayes e calcular a probabilidade  $P(\omega_i|x)$ , ou seja [1][10]:

$$P(\omega_i|x) = \frac{P(\omega_i)p(x|\omega_i)}{p(x)}$$

Na qual  $P(\omega_i)$  é a *priori*,  $p(x|\omega_i)$  é a densidade condicional ou verossimilhança,  $p(x) = \sum_{i=1}^c P(\omega_i) p(x|\omega_i)$  é a evidência, e  $P(\omega_i|x)$  é a *posteriori*. Dentre as variações do classificador de Bayes podemos aplicar classificação com mínima taxa de erro e mínimo risco, classificação de Bayes para distribuição gaussiana, classificação Naïve Bayes e outras [1].

#### B. Aprendizagem baseada em métodos determinísticos

Alguns problemas de classificação envolvem atributos nominais ou categóricos, sem qualquer noção natural de semelhança ou mesmo de ordem. Nestes casos, a distribuição de probabilidade não é adequada. Para classificar objetos utilizando atributos nominais, métodos apropriados devem ser adotados. Dentre os principais métodos podemos citar a indução árvores de decisão, o reconhecimento com String, os métodos gramaticais, a inferência gramatical e a aprendizagem baseada em regras.

A seguir é destacado o método de indução de árvores de decisão e na sequência a aprendizagem baseada em regras.

Segundo Duda et al. [1], é natural e intuitivo classificar um padrão através de uma sequência de questões, onde a próxima questão depende da resposta da questão corrente. Uma árvore de decisão é uma sequência de questões. Por convenção, o primeiro nó da árvore fica no topo e é chamado nó *raiz*. Em seguida, ramificações sucessivas são conectadas a outros nós, essas conexões são chamadas *links* ou *ramos*. O número de links em cada nó é chamado *fator* de ramificação. Em árvores binárias o fator de ramificação de cada nó é 2 (dois), em árvores  $n$ -árias o fator de ramificação é  $n$ . Essas conexões ocorrem até alcançar um nó *terminal*, ou *folha*. Um nó folha é atingido quando termina a ramificação. Para classificar um item, dada uma árvore de decisão percorre-se a árvore do nó raiz até um nó folha seguindo o link apropriado para o nó subsequente ou descendente. Cada link deve ser mutuamente distinto e exaustivo, ou seja, um e somente um link deve ser seguido [1]. Cada folha contém uma categoria rotulada, chamada de *classe*. Cada nó intermediário especifica um teste de um atributo ou instancia pertencente à classe. Cada link contém um dos valores possíveis de cada atributo. Dentre os

principais algoritmos para indução de árvores estão: ID3 [8], C4.5 [7], ID5R [14].

No algoritmo ID3, conforme figura 2, *Examples* são os exemplos de treinamento. *Target\_attribute* é o atributo cujo valor será predito pela árvore. *Attributes* é uma lista de atributos que podem ser testados pela árvore de decisão. O algoritmo retorna uma árvore de decisão que classifica corretamente os exemplos dados. Na sequência do algoritmo, o primeiro passo é criar um nó raiz para a árvore. Se os exemplos de treinamento são corretos, retorna uma árvore de nó único (raiz), com rótulo "+". Se os exemplos são incorretos, retorna uma árvore de nó único (raiz), com rótulo "-". Se a lista de atributos testada pela árvore estiver vazia, retorna uma árvore de nó único com o rótulo contendo o valor mais comum do atributo predito pela árvore. Caso contrário, escolhe o atributo da lista de atributos que melhor classifica os exemplos de treinamento, e o nó raiz da sub-árvore recebe o atributo escolhido. Para cada valor possível do atributo é adicionado um novo ramo da árvore abaixo da raiz, correspondendo ao teste "atributo = valor do atributo". Na sequência, são criados subconjuntos de exemplos com o valor do atributo. Se o subconjunto for vazio, adicionar uma folha embaixo deste novo ramo com rótulo = "valor mais comum do atributo, cujo valor foi predito pela árvore nos exemplos". Se o subconjunto for diferente de vazio adicionar a sub-árvore e repetir o algoritmo considerando a lista de atributos, retirando o subconjunto testado. No final, retornar raiz.

```

01. ID3(Examples, Target_attribute, Attributes)
02. Create a Root node for the tree
03. if all Examples are positive, Return the single-node tree Root, with label = +
04. if all Examples are negative, Return the single-node tree Root, with label = -
05. if Attributes is empty, Return the single-node tree Root,
06.   with label = most common value of Target_attribute in Examples
07.
08. Otherwise Begin
09. A ← the attribute from Attributes that best* classifies Examples
10. The decision attribute to Root ← A
11. For each possible value, vi, of A,
12.   Add a new tree branch below Root, corresponding to the test A = vi
13.   Let Examples_vi be the subset of Examples that have value vi to A
14.   if Examples_vi is empty
15.     then below this new branch add a leaf node with label = most common
16.     value of Target attribute in Examples
17.   else below this new branch add the subtree
18.     ID3(Examples_vi, Target_attribute, Attributes - {A})
19.   end
20. return Root
21.
22. "The best attribute is the one with highest information gain"

```

Figura 2 – Algoritmo ID3 (Figura adaptada de Michell, 1997)

Métodos baseados em regras são partes integrantes de sistemas em inteligência artificial, mais sua adoção em reconhecimento de padrões ainda é modesta [1]. Uma visão geral desse método para representação e aprendizagem, concentra-se em classes de regras do tipo "se...então". Mitchell [5] considera o conjunto de regras do tipo se-então uma das formas mais expressiva e legível para representar hipóteses em aprendizagem de máquina. Uma regra pode ser exemplificada em [1]: "se *Swims(x)* e *HasScales(x)* então *Fish(x)*", (lê-se: se um objeto *x* tem a propriedade de que nada e a propriedade de ter escamas então esse objeto é um peixe). Um predicado, como *Swims(x)*, *HasScales(x)* é um teste que retorna um valor lógico, que pode ser verdadeiro ou falso. Os predicados podem ser aplicados a problemas onde os dados são numéricos, não-numérico, sequências linguísticas, entre outros. A tarefa de escolha de predicados apropriados e a avaliação desses predicados dependem fortemente do

problema, em geral, é mais difícil do que aprender a regra. Mitchell [5] considera dois tipos principais de regras se-então, as regras proposicionais e as regras de primeira-ordem. A regra proposicional descreve um caso particular, a desvantagem da lógica proposicional é não fornecer uma maneira para representar as relações gerais entre um grande número de casos. As regras de primeira ordem usam variáveis, o que permite expressar propriedades gerais, pertencentes a todos os indivíduos e propriedades existenciais, pertencentes a alguns indivíduos. Para Russell et al. [9] a lógica de primeira ordem é considerada satisfatória para representar o conhecimento comum. Além disso, de acordo com Mitchell [5], ao contrário da árvore de decisão, que aprende apenas regras proposicionais, as regras de primeira ordem são mais expressivas e permitem representar variáveis e relacionamentos entre elas. Neste caso, árvores de decisão ou técnicas estatísticas, não são capazes de aprender a regra perfeitamente, mesmo tendo em conta um número extremamente grande de exemplos [1].

Na classificação de padrão, dado um conjunto de regras escritas em lógica de primeira ordem, avaliar as proposições e regras para classificar um item desconhecido. Regras proposicionais podem ser aprendidas por algoritmos conhecidos como algoritmos de Cobertura Sequencial, já as regras de primeira ordem utilizam o algoritmo FOIL [5] para aprendizagem.

No algoritmo de cobertura sequencial, a entrada é um conjunto de exemplos de treinamento positivos e negativos e a saída é uma regra que cobre vários dos exemplos positivos e nenhum, ou poucos, negativos. A regra deve ter uma boa taxa de acerto, porém, não necessariamente precisa englobar muitos exemplos para evitar *overfitting*.

A figura 3 mostra o pseudocódigo do algoritmo de cobertura.

Os métodos para aprender um conjunto de regras, [5] dado à função *learn-one-rule* é: construir uma árvore de decisão e fazer a conversão para um conjunto de regras ou utilizar o algoritmo *Sequential Covering*. Neste último: invocar *Learn-one-rule* sobre todos os exemplos; remover os exemplos positivos cobertos pela regra aprendida; repetir o processo até atingir a fração desejada de exemplos positivos cobertos pelas regras.

```

view plain copy to clipboard print ?
01. Sequential Covering (Classe, Atributos, Exemplos, Limiar)
02. regras_aprendidas ← {};
03. regra ← LEARN-ONE-RULE(classe, atributos, exemplos)
04. while PERFORMANCE(regra, exemplos) > limiar do
05.   regras_aprendidas ← regras_aprendidas + regra;
06.   exemplos ← exemplos - {exemplos cobertos pela Regra};
07.   regra ← LEARN-ONE-RULE(classe, atributos, exemplos);
08. regras_aprendidas ← ordene regras_aprendidas por PERFORMANCE;
09. return regras_aprendidas;

```

Figura 3 – Algoritmo *Sequential Covering* para aprendizagem de regras (Figura adaptada de Michell, 1997)

Uma das diferenças entre aprendizagem de árvores e aprendizagem de regras é que o algoritmo *Sequential Covering* aprende as regras uma de cada vez enquanto as árvores de decisão aprendem o conjunto de regras de uma só vez [5].

### C. Aprendizagem baseada em métodos adaptativos

A adaptatividade em sistemas de software permite um comportamento dinâmico através da automodificação da funcionalidade desses sistemas. O estudo da tecnologia adaptativa tem como objetivo propor modelos baseados em dispositivos adaptativos para solução de problemas práticos. A tecnologia adaptativa é o conjunto de aplicações práticas baseadas no conceito de adaptatividade, com a finalidade de solucionar problemas em diversas áreas.

Um dispositivo é adaptativo quando seu comportamento pode mudar dinamicamente em resposta a estímulos de entrada, sem a interferência de agentes externos. Dentre os principais dispositivos adaptativos, consideramos apropriados para solução de problemas de classificação as árvores de decisão adaptativas [6], as tabelas de decisão adaptativas [11] e as gramáticas adaptativas [2].

A indução de árvores de decisão proposta em [6] baseada na teoria dos dispositivos adaptativos pode ser aplicada em problemas de aprendizagem computacional. O resultado é uma árvore de decisão adaptativa com capacidade de automodificação através da inserção e/ou remoção de sub-árvores. As árvores de decisão adaptativas são aplicadas na solução de problemas relacionados a processamento digital de imagens e aprendizagem de máquina [6].

A tabela de decisão adaptativa, cujo formalismo pode ser encontrado em [11], pode apoiar o processo de tomada de decisão e ser aplicada na busca de padrões [12]. Uma tabela de decisão adaptativa é um dispositivo guiado por regras que permite mudanças no conjunto de regras da tabela de forma dinâmica, através de ações adaptativas. As tabelas são formadas por um conjunto de condições, ações, regras e funções adaptativas. Na execução de uma condição em uma tabela de decisão adaptativa, são verificadas as regras não adaptativas e caso uma única delas se aplique, as ações correspondentes são executadas. Caso mais de uma regra não adaptativa satisfaça a condição, as ações correspondentes às condições devem ser aplicadas em paralelo. Porém, se nenhuma regra satisfizer a condição, essa é uma condição excepcional e no caso de uma tabela de decisão convencional não seria possível prosseguir. Porém, na solução adaptativa é verificado se uma regra adaptativa se aplica. Se existir, a ação é executada e o conjunto de regras não adaptativas é alterado. Neste último caso, o comportamento do sistema é modificado e uma vez aplicada à regra, utiliza-se novamente a tabela de decisão adaptativa em sua nova configuração.

As gramáticas adaptativas [2], são capazes de representar linguagens sensíveis ao contexto. O que diferencia estas gramáticas das convencionais é a sua capacidade de automodificação, característica dos dispositivos adaptativos. As modificações acontecem durante a geração da sentença da linguagem, as ações adaptativas são associadas às regras de produção que possibilitam alterar tanto o conjunto de símbolos não-terminais, como as regras de produção da gramática.

### III. PROPOSTA DE UM MÉTODO HÍBRIDO PARA CONSTRUÇÃO DE CLASSIFICADORES

Na literatura a abordagem estatística para reconhecimento de padrão é predominante, porém, cada abordagem tem seu

domínio de aplicação onde uma é mais apropriada que outra. Existem muitas aplicações potenciais para as diferentes abordagens de reconhecimento de padrão [1].

Métodos estatísticos pressupõe uma distribuição de probabilidade associada ao espaço de características. A noção de erro de classificação é formulada em termos estatísticos e probabilísticos. Os métodos estatísticos utilizam variáveis aleatórias sobre um espaço amostral. Quanto menor a variância das variáveis aleatórias, maior a precisão. Assim, se considerarmos problemas de alta de precisão a variância deve ser nula.

Métodos estatísticos são apropriados em casos que o indivíduo é irrelevante, pois esses métodos massificam as amostras e eliminam a importância dos indivíduos. Em problemas em que as características de cada indivíduo são fundamentais, é mais apropriado usar métodos determinísticos auxiliares ou utilizar métodos híbridos.

A segunda opção é promissora e uma proposta para construção de classificadores utilizando métodos híbridos é considerada.

Um método híbrido é apropriado na busca de uma solução considerando um problema de natureza estruturada e dinâmica. Também é adequado onde às relações entre os indivíduos, bem como seu comportamento individual e coletivo, são de grande importância. Suponha uma amostra grande de indivíduos. Os métodos estatísticos têm a função de localizar a região provável da solução procurada, convergindo rapidamente para o espaço de possíveis soluções. Na sequência, métodos determinísticos têm a função de refinar a solução do processo, considerando os aspectos estruturais relacionados aos indivíduos. Assim, métodos estáticos e dinâmicos, dependendo intimamente da natureza do problema, são aplicados para tomada de decisão do processo. Esses métodos incluem métodos não adaptativos (estáticos) e métodos adaptativos (dinâmicos). Para exemplificar, em problemas de processamento da língua natural, um método híbrido pode ser uma alternativa de busca de solução no processo de reconhecimento. Aplicando um método estatístico, a busca por uma possível solução em amostras grandes é reduzida, o seja, o espaço amostral é reduzido, porém a decisão final não é baseada na probabilidade. Em seguida, mecanismos de inferência gramatical e gramática adaptativa são utilizados para tratamento de características estruturais e dinâmicas do problema. A decisão de classificação é baseada nesses métodos. Em geral, métodos híbridos tendem a produzir melhores resultados, quando os indivíduos são relevantes.

É importante destacar que se trata de uma proposta inicial, sem resultados práticos. Porém, a possibilidade de combinar técnicas é muito utilizada e pode agregar as vantagens e capacidades de diferentes métodos tradicionalmente conhecidos.

### IV. CONCLUSÃO

O estudo comparativo entre métodos estatísticos e métodos determinísticos permitiram identificar as características

relevantes em cada uma das abordagens. Métodos estatísticos estão preocupados em buscar regularidades em um espaço amostral sobre uma massa de dados e classificar novos indivíduos com uma taxa de erro aceitável. Esse tipo de abordagem desconsidera a precisão. Métodos determinísticos buscam regularidades em amostras mapeando a estrutura relativa aos indivíduos. Considera cada indivíduo relevante e particular, proporcionando alta precisão na classificação de novos indivíduos. Além disso, métodos estáticos são apropriados quando o fenômeno que descreve o comportamento dos indivíduos é fixo. Em casos onde o fenômeno que descreve o comportamento dos indivíduos é variável, métodos dinâmicos são mais apropriados. Neste trabalho, os métodos dinâmicos correspondem aos métodos adaptativos.

A proposta de uso de adaptatividade em métodos para reconhecimento de padrão considera as seguintes características do problema: problemas com amostras grandes de dados; comportamento coletivo entre os indivíduos, comportamento isolado de cada indivíduo e as relações entre os indivíduos; problemas com comportamento dinâmico.

Em problemas de classificação em que é necessária a obtenção de resultados de alta precisão, devemos considerar: 1) uso de métodos estatísticos para localizar a região provável da solução procurada. 2) uso de métodos determinísticos para refinamento da resolução do processo. 3) uso de métodos estáticos para tratamento de fenômenos de comportamento fixo. 4) uso de métodos dinâmicos (adaptativos) para tratamento de comportamentos variantes no tempo.

Os resultados mostram que a combinação de diferentes métodos e técnicas na construção de classificadores permite: 1) obtenção rápida de resultados utilizando métodos estatísticos com uma estimativa aceitável para a solução, ainda que com baixa porcentagem de acerto. 2) eliminação de imprecisões e flutuações com a introdução de técnicas determinísticas mais onerosas. 3) obtenção eventualmente lenta de refinamentos proporcionados por métodos não estatísticos, resultando em alta porcentagem de acerto.

Em trabalhos futuros, o método híbrido deve ser detalhado, sendo aplicado os conceitos apresentados neste trabalho. Isso inclui delimitar a atuação de cada abordagem e método no processo de reconhecimento.

A combinação de métodos e técnicas em reconhecimento de padrão é comum, porém a capacidade de tratar aspectos dinâmicos de forma expressiva, na busca de uma solução mais precisa, é uma das principais contribuições da aplicação de métodos adaptativos na construção de classificadores.

## VI. REFERÊNCIAS

- [1] DUDA, R. O.; HART, P. E. ; STORK, D. G. *Pattern Classification*. 2ª Edição. Wiley, 2001. 680 p. ISBN: 0471056693.
- [2] IWAI, M. K. Um formalismo gramatical adaptativo para linguagens dependentes de contexto. Tese (Doutorado), Escola Politécnica da USP, São Paulo, 2000.
- [3] JAIN, A.K.; DUIN, R. P.W.; MAO, J.. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 22 (no.1):4–37, 2000.
- [4] NETO, J. J. Adaptive Rule-Driven Devices – General Formulation and Case Study. *Revista de Engenharia de Computação e Sistemas Digitais*, São Paulo, v.1, n.1, p. 45-57, nov. 2003.
- [5] MITCHELL, T. M.. *Machine Learning*. 1ª Edição. McGraw-Hill, 1997. ISBN: 0070428077.
- [6] PISTORI, H. *Tecnologia Adaptativa em Engenharia de Computação: estado da arte e aplicações*. Tese (Doutorado), Escola Politécnica da USP, 2003.
- [7] QUINLAN, J. R. C4.5: Programs for Machine Learning. Morgan-Kaufmann, San Francisco, 1993.
- [8] QUINLAN, J. R. (1986). Induction of decision trees. *Machine Learning*, 1, 81-106.
- [9] RUSSEL, S. J.; NORVIG, P.. *Artificial Intelligence: A Modern Approach*. 2ª Edição. Prentice-Hall, 2002. ISBN - 10: 0137903952
- [10] THEODORIDIS, S.; KOUTROUMBAS, K. *Pattern Recognition*. 3ª Edição. Academic Press, 2006.
- [11] TICHEMRA, A. H. Aplicação da Tecnologia Adaptativa em Sistemas de Tomada de Decisão. I WTA – Workshop sobre Tecnologia Adaptativa, Jan 2007 - [www.pcs.usp.br/~lta](http://www.pcs.usp.br/~lta).
- [12] TICHEMRA, A. H., CAMARGO, R. Descoberta de padrões em bases de dados utilizando Técnicas Adaptativas. III WTA – Workshop sobre Tecnologia Adaptativa, Jan 2009 - [www.pcs.usp.br/~lta](http://www.pcs.usp.br/~lta).
- [13] WINTEN, I. H. e EIBE, F.; *Data Mining: Practical Machine Learning Tools and Techniques*.
- [14] UTGOFF, P. E., Incremental Induction of Decision Trees. *Machine Learning*, Machine Learning, 4, 161-186, 1989.

## Autores

**Renata Luiza Stange** é mestranda em Engenharia Elétrica (Área de concentração: Computação e Sistemas Digitais) pela Escola Politécnica da Universidade de São Paulo. Especialista em Administração de Sistemas de Informação pela Universidade Federal de Lavras e Bacharel em Análise de Sistemas pela Universidade Estadual do Centro-Oeste do Paraná.

**João José Neto** é graduado em Engenharia de Eletricidade (1971), mestre em Engenharia Elétrica (1975), doutor em Engenharia Elétrica (1980) e livre-docente (1993) pela Escola Politécnica da Universidade de São Paulo. Atualmente, é professor associado da Escola Politécnica da Universidade de São Paulo (EPUSP) e coordena o LTA – Laboratório de Linguagens e Tecnologia Adaptativa do Departamento de Engenharia de Computação e Sistemas Digitais da EPUSP. Tem experiência na área de Ciência da Computação, com ênfase nos Fundamentos da Engenharia da Computação, atuando principalmente nos seguintes temas: dispositivos adaptativos, tecnologia adaptativa, autômatos adaptativos, e em suas aplicações à Engenharia de Computação, particularmente em sistemas de tomada de decisão adaptativa, análise e processamento de linguagens naturais, construção de compiladores, robótica, ensino assistido por computador, modelagem de sistemas inteligentes, processos de aprendizagem automática e inferências baseadas em tecnologia adaptativa.