O Papel da Inferência Seletiva em Aprendizado de Máquina

Ígor Assis Braga, Maria Carolina Monard

Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo – São Carlos, SP – Brasil

{igorab,mcmonard}@icmc.usp.br

Abstract. *One of the challenges in machine learning is the automatization of empirical inference processes. In this paper we deal with selective inference, which is an inference type that remains unexplored in machine learning. Despite the possibility of using more general inference algorithms for solving selective inference, we argue that an algorithm for solving it directly can achieve success even when more general inference algorithms fail. We report on initial experiments conducted on text data sets that support this idea.*

Resumo. *Um dos desafios em aprendizado de máquina é a automatização de processos de inferência empírica. Neste artigo é apresentado um tipo de inferência ainda não explorado em aprendizado de máquina, denominado inferência seletiva. Apesar de algoritmos de inferência mais gerais poderem ser empregados na realização de inferência seletiva, é defendida a ideia de que um algoritmo para realizá-la diretamente pode ter sucesso mesmo quando algoritmos de inferência mais gerais falham. Para corroborar essa ideia, são relatados experimentos iniciais conduzidos em bases de dados textuais.*

1. Introdução

Um dos principais desafios em aprendizado de máquina é o desenvolvimento de algoritmos que realizam inferências a partir de dados empíricos. Desde o final da década de 50, quando o algoritmo Perceptron foi proposto [Rosenblatt 1957], diversos algoritmos de aprendizado foram desenvolvidos para a realização de inferência indutiva. Em contraste, somente no final dos anos 90 foi que surgiram os primeiros algoritmos para a realização de outros tipos de inferência, como a inferência transdutiva.

A maior concentração de pesquisas em inferência indutiva é compreensível, pois, por exemplo, realizar bem inferência indutiva implica em realizar bem inferência transdutiva. Sendo assim, por que se pesquisam algoritmos de inferência transdutiva? A resposta para essa pergunta está na constatação de que realizar bem inferência transdutiva não depende de se realizar bem inferência indutiva. Desse modo, em situações nas quais algoritmos indutivos falham, ainda é possível que um algoritmo transdutivo obtenha sucesso.

Existem outros tipos de inferência empírica sugeridos na literatura que ainda não foram explorados em aprendizado de máquina. Esse é o caso da *inferência seletiva*, tema deste artigo. O motivo pelo qual esse tipo de inferência não recebeu atenção é o mesmo que causou a demora no desenvolvimento de algoritmos transdutivos: realizar bem inferência indutiva ou inferência transdutiva implica em realizar bem inferência seletiva. De maneira análoga, como será argumentado ao longo deste artigo, realizar bem inferência seletiva não depende de se realizar bem os outros dois tipos de inferência.

Com o intuito de contribuir para a exploração de inferência seletiva em aprendizado de máquina, neste artigo são identificados domínios de aplicação de amplo interesse que podem ser beneficiados pelo desenvolvimento de algoritmos seletivos. Além disso, são relatados experimentos em bases de texto que demonstram a possibilidade de se realizar bem inferência seletiva mesmo quando as inferências indutiva e transdutiva falham.

O restante deste artigo está estruturado como se segue. Na Seção 2 é destacada a importância de se desenvolver algoritmos que realizem diretamente um determinado tipo de inferência, usando como exemplo o caso da inferência transdutiva. Na Seção 3 é apresentada a definição de inferência seletiva, além de importantes domínios de aplicação e de uma formulação teórica para um caso particular. Na Seção 4 é proposto um algoritmo inspirado em inferência seletiva a fim de compará-lo experimentalmente com um algoritmo indutivo e com um algoritmo transdutivo empregados na realização de inferência seletiva. Por último, na Seção 5, são apresentadas as considerações finais.

2. Fundamentação

Nesta seção é destacada a filosofia de se realizar um determinado tipo de inferência diretamente, sem depender da realização de tipos de inferência mais gerais. Para isso, as inferências indutiva e transdutiva são definidas, e, em seguida, é mostrado como essa filosofia contribuiu no desenvolvimento de algoritmos mais apropriados para a realização de inferência transdutiva.

Em todos os tipos de inferência considerados ao longo deste artigo, assume-se a existência de uma função $f : X \mapsto Y$. A função f é desconhecida, mas é possível ter acesso a um conjunto $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ tal que $y_i = f(\mathbf{x}_i)$. Esse conjunto, denominado conjunto de treinamento, contém elementos que indicam o valor da função f em n pontos do domínio X. O objetivo geral em inferência empírica é utilizar a informação sobre f disponível no conjunto de treinamento para obter novas informações sobre f.

Diferentes tipos de inferência podem ser definidos de acordo com a magnitude das informações obtidas ao final do processo de inferência. Em *inferência indutiva*, por exemplo, o objetivo é, dado um conjunto de treinamento, encontrar uma função (ou hipótese) $h : X \mapsto Y$ que seja uma "boa aproximação" da função desconhecida f. Idealmente, o que se obtém nesse tipo de inferência é uma estimativa dos valores de f em *todos* os pontos do domínio X.

Agora considere que é necessário algo mais específico: inferir o valor da função desconhecida f somente em *alguns* pontos pré-determinados do domínio X. Em outras palavras, dados um conjunto de treinamento e um conjunto $\{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}\}$ contendo m pontos de interesse, é necessário estimar os valores $y_{n+1}, \ldots, y_{n+m}$ que a função desconhecida f assume nos m pontos de interesse. Como a magnitude das informações obtidas nesse caso é bem menor que na inferência indutiva, Vapnik [1995, p. 293] propôs diferenciar esse tipo de inferência, denominando-o *inferência transdutiva*.

A fim de se realizar inferência transdutiva, poderia-se utilizar um algoritmo indutivo[1] para obter uma função aproximadora h, que em seguida seria utilizada para estimar

[1]Entre os algoritmos de aprendizado que realizam inferência indutiva encontram-se, por exemplo, os baseados em árvores de decisão, o k-Nearest Neighbors (kNN), Support Vector Machines (SVM) e os baseados em redes neurais preditivas [Alpaydin 2004].

o valor de f nos pontos de interesse — Figura 1(a). No entanto, quando os exemplos disponíveis no conjunto de treinamento não são suficientes para se realizar bem inferência indutiva, ainda é possível que a informação disponível seja suficiente para se realizar inferência transdutiva diretamente — Figura 1(b).

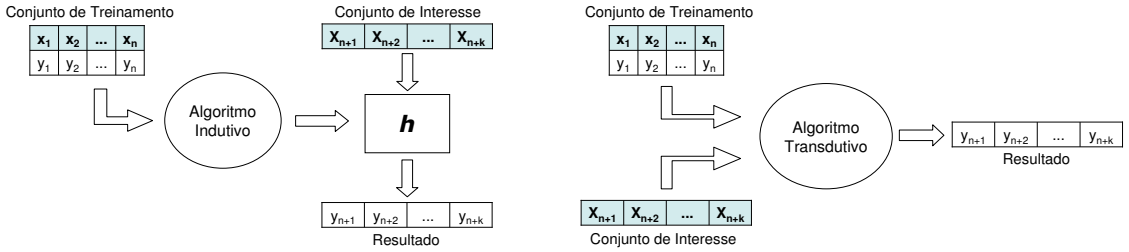

Figura 1. (a) Algoritmo indutivo realizando inferência transdutiva. (b) Algoritmo para realizar inferência transdutiva diretamente.

A busca por algoritmos que realizem inferência transdutiva diretamente é uma consequência da seguinte filosofia de inferência [Vapnik 2006, p. 477]:

> *Ao tentar resolver um problema de interesse, não tente resolvê-lo utilizando a solução de um problema mais geral. Tente obter a resposta desejada diretamente.*

No caso de algoritmos transdutivos já desenvolvidos, como, por exemplo, Transductive Support Vector Machines (TSVM) [Joachims 1999] e Spectral Graph Transducer [Joachims 2003], a vantagem obtida em relação aos algoritmos indutivos é a disponibilidade dos pontos de interesse durante a fase de treinamento. Trabalhos em diferentes áreas de aplicação de aprendizado de máquina mostram que os algoritmos transdutivos são mais eficazes que os algoritmos indutivos empregados na realização de inferência transdutiva. Essa observação é ainda mais forte nos casos em que o conjunto de treinamento contém poucos exemplos descritos em um espaço de alta-dimensionalidade [Joachims 1999, Weston et al. 2003, Audibert 2008, El-Yaniv et al. 2008].

3. Inferência Seletiva

Na seção anterior foram definidos dois tipos distintos de inferência empírica: a inferência indutiva, mais geral, e a inferência transdutiva, mais específica. Também foi destacado, no contexto da inferência transdutiva, a importância de se providenciar algoritmos que realizem um determinado tipo de inferência diretamente.

Nesta seção é apresentado um terceiro tipo de inferência, denominado inferência seletiva, para o qual ainda não se conhecem algoritmos próprios. Além disso, são listadas tarefas de aprendizado nas quais a realização de inferência seletiva deveria ser considerada. Por fim, é apresentada uma formulação teórica dada por Vapnik [2006] para um caso particular de inferência seletiva.

3.1. Definição

Suponha novamente a existência de uma função desconhecida $f : X \mapsto Y$ e de um conjunto de treinamento $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ no qual $y_i = f(\mathbf{x}_i)$. Por vezes, deseja-se selecionar alguns pontos de um conjunto pré-determinado de candidatos $\{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}\}$

de acordo com a função f. Como essa função é desconhecida, a seleção de tais candidatos não pode ser feita dedutivamente. Em *inferência seletiva*, o objetivo é selecionar os candidatos de acordo com a informação sobre f disponível no conjunto de treinamento.

Dependendo da natureza do conjunto Y e das condições que um candidato deve satisfazer para ser selecionado, pode-se obter diferentes definições particulares de inferência seletiva. Os dois casos mencionados a seguir não esgotam todas as possibilidades.

> *Caso I*. Suponha que $Y = \{\oplus, \ominus\}$. Dados um conjunto de treinamento e um conjunto contendo m exemplos candidatos, selecione p exemplos candidatos que são positivos de acordo com a função desconhecida f.
>
> *Caso II*. Suponha que $Y \subseteq \mathbb{R}$. Dados um conjunto de treinamento e um conjunto contendo m pontos candidatos, selecione o ponto candidato $\mathbf{x}$ que leva ao maior valor de $f(\mathbf{x})$ entre os pontos candidatos, isto é, selecione o ponto candidato $\mathbf{x}$ para o qual $f(\mathbf{x}) \geq f(\mathbf{x}_i), i = n+1, \ldots, n+m$.

Contrastando inferência seletiva com as inferências indutiva e transdutiva, definidas na seção anterior, percebe-se que estas são mais gerais que aquela. De fato, um algoritmo indutivo ou transdutivo pode ser utilizado na realização de inferência seletiva, mas a recíproca não é verdadeira. A Figura 2 ilustra esse argumento para o Caso II. Observe que um algoritmo transdutivo, por exemplo, pode estimar os valores da função desconhecida f nos pontos candidatos, e, em seguida, o ponto que leva ao maior valor estimado é selecionado — Figura 2(a). Contudo, o resultado de um possível algoritmo para realizar inferência seletiva diretamente não pode ser utilizado para estimar os valores de f nos pontos candidatos — Figura 2(b).

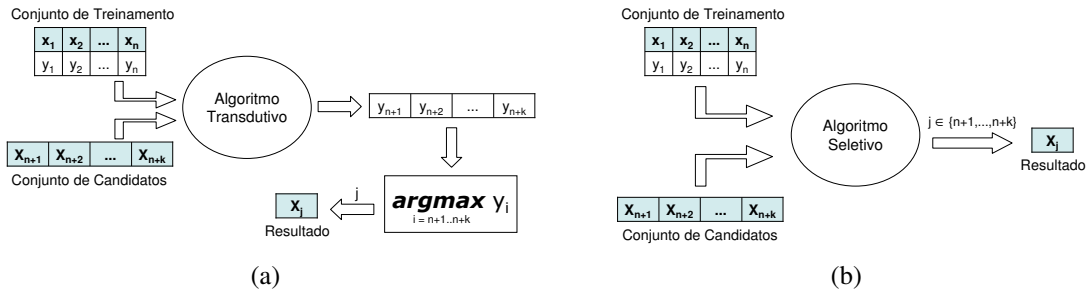

Figura 2. (a) Algoritmo transdutivo realizando inferência seletiva — Caso II. (b) Possível algoritmo para realizar inferência seletiva diretamente.

A definição de inferência seletiva foi sugerida primeiramente por Vapnik [2006, p. 468] pelos mesmos motivos que o levaram a sugerir a inferência transdutiva. Afinal, os exemplos disponíveis no conjunto de treinamento podem não ser suficientes para se realizar bem inferência indutiva ou transdutiva, mas ainda é possível que a informação disponível seja suficiente para se realizar inferência seletiva diretamente.

3.2. Aplicações

A seguir são apresentadas algumas tarefas de aprendizado de máquina nas quais a realização de inferência seletiva deveria ser considerada. Essas tarefas aparecem em domínios de aplicação de grande interesse para pesquisadores na área. As três primeiras tarefas são

exemplos da definição de inferência seletiva para o Caso I, enquanto que a última é um exemplo para o Caso II.

Desenvolvimento de fármacos. Uma etapa importante no desenvolvimento de fármacos é testar compostos químicos quanto a sua capacidade de se ligarem a um determinado receptor no corpo humano. Na tentativa de reduzir os custos envolvidos nessa etapa, os dados resultantes dos testes são analisados para que novos testes sejam feitos em compostos mais promissores. Tal análise pode ser considerada como uma tarefa de aprendizado de máquina quando um conjunto de compostos previamente testados está disponível para treinamento[2]. Dado um conjunto de compostos ainda não testados, é interessante identificar um a um qual se liga e qual não se liga ao receptor de interesse usando, por exemplo, inferência transdutiva. No entanto, a escassa quantidade de compostos de treinamento típica dessa tarefa faz com que a realização de inferência seletiva seja considerada.

Detecção de homologia entre proteínas. A detecção de homologia entre proteínas é considerada uma importante tarefa em bioinformática, pois é possível se fazer considerações sobre a estrutura desconhecida de uma proteína por meio da estrutura conhecida de uma proteína homóloga. Devido à disponibilidade crescente de algoritmos para obter características de proteínas, é possível detectar homologia entre uma proteína com estrutura desconhecida e uma com estrutura conhecida utilizando algoritmos de aprendizado[3]. Apesar de uma proteína poder ser homóloga a várias outras, pode ocorrer de somente uma ou algumas poucas já serem suficientes para a modelagem de sua estrutura. Sendo assim, não é necessário realizar um tipo de inferência mais geral que a inferência seletiva para cada proteína com estrutura desconhecida.

Descoberta de genes relevantes a uma doença. Identificar genes que desempenham papéis importantes em doenças é uma tendência na pesquisa médica. A disponibilidade de algoritmos para gerar características de genes e a tecnologia para obter níveis de expressão gênica tem colaborado para a aplicação de aprendizado de máquina nesta tarefa. O ideal seria poder identificar todos ou grande parte dos genes relevantes a uma doença dentre os genes candidatos, mas é pouca a quantidade de genes de treinamento relevantes tipicamente disponível nesta tarefa. Desse modo, os algoritmos de aprendizado são normalmente utilizados para priorizar genes para estudos futuros [Agarwal e Sengupta 2009], o que pode ser realizado com inferência seletiva.

Busca na *web*. Uma consulta realizada em sistemas de busca na *web* gera uma lista contendo páginas com variados graus de relevância, sendo necessário identificar aquelas com maior relevância para compor os primeiros resultados de busca. Essa identificação pode ser feita usando-se algoritmos de aprendizado que utilizam dados de treinamento obtidos explicitamente por meio de especialistas ou implicitamente por meio dos cliques dos usuários nos resultados de busca [Joachims 2002]. Geralmente, os usuários visualizam no máximo os 10 primeiros resultados de busca [Campos 2007]. Consequentemente, ordenar todas as páginas recuperadas, inclusive as pouco relevantes e as irrelevantes, constitui um trabalho desnecessário. Uma solução mais direta, e que pode requerer menos dados de treinamento, seria investigar as chances de cada página *web* recuperada figurar no topo dos resultados de busca, o que pode ser realizado com inferência seletiva.

[2]Veja, por exemplo, o KDD-Cup 2001: `http://pages.cs.wisc.edu/~dpage/kddcup2001/`.
[3]Veja, por exemplo, o KDD-Cup 2004: `http://kodiak.cs.cornell.edu/kddcup/tasks.html`.

No geral, a realização de inferência seletiva torna-se uma opção a ser considerada em aprendizado de máquina quando o domínio de aplicação impõe um interesse maior sobre determinados candidatos. No Caso I, esse interesse recai sobre exemplos que podem ser identificados como positivos com alta precisão, enquanto que, no Caso II, o interesse é por pontos que assumem o maior valor da função desconhecida f dentre os pontos candidatos. No entanto, se a quantidade de informação disponível no conjunto de treinamento for pouca, a realização de inferência seletiva pode se tornar a única opção, pois realizar tipos de inferência mais gerais nessa condição é mais arriscado.

3.3. Formulação Teórica

Assumindo que a realização de inferência seletiva pode ter um papel importante em aprendizado de máquina, a primeira questão que se levanta é como realizá-la sem depender de tipos de inferência mais gerais. A seguir, é apresentada uma formulação teórica para esse problema dada por Vapnik [2006, p. 341] para o Caso I de inferência seletiva, isto é, assume-se aqui que $Y = \{\oplus, \ominus\}$. Uma formulação análoga para o Caso II também foi proposta, e pode ser encontrada em [Vapnik 2006, p. 344].

Antes de prosseguir, é necessário definir o conceito de classes de equivalência de funções. Considere a existência de um espaço de funções $F \subseteq X \times Y$. Os exemplos do conjunto de treinamento e do conjunto de candidatos particionam F em l classes de equivalência $E_1, E_2, \ldots, E_l$, tal que as funções pertencentes a uma classe de equivalência E_i rotulam os exemplos de treinamento e os exemplos candidatos da mesma maneira. Formalmente, para $\forall f', f'' \in E_i$, tem-se que

$$\begin{gathered} f'(\mathbf{x}_1) = f''(\mathbf{x}_1) = y_{1,i} \\ \vdots \\ f'(\mathbf{x}_n) = f''(\mathbf{x}_n) = y_{n,i} \\ f'(\mathbf{x}_{n+1}) = f''(\mathbf{x}_{n+1}) = y_{n+1,i} \\ \vdots \\ f'(\mathbf{x}_{n+m}) = f''(\mathbf{x}_{n+m}) = y_{n+m,i} \end{gathered}$$

Agora suponha que se deseja realizar inferência seletiva para identificar um único exemplo positivo dentre os candidatos. A formulação teórica dada por Vapnik expressa a probabilidade de cada candidato $\mathbf{x}_i$ ser positivo de acordo com f dados o conjunto de treinamento e o conjunto de candidatos. O candidato que obtiver a maior probabilidade é aquele que deve ser selecionado.

No que se segue, denote o conjunto de treinamento $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ por T, o conjunto de candidatos $\{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}\}$ por C, e a sequência de candidatos obtida omitindo-se o candidato $\mathbf{x}_i$ por C^i, isto é, $C^i = \mathbf{x}_{n+1}, \ldots, \hat{\mathbf{x}}_i, \ldots, \mathbf{x}_{n+m}$. Note que a sequência C^i pode ser rotulada em duas classes de 2^{m-1} maneiras. Denote a r-ésima rotulação $y_{n+1,r}, \ldots, \hat{y}_{i,r}, \ldots, y_{n+m,r}$ por L^i_r.

Suponha que foi definida a probabilidade $P(L^i_r)$ de que a rotulação L^i_r coincidirá com a rotulação da sequência C^i dada pela função desconhecida f. Então, pela lei da probabilidade total, tem-se para um determinado candidato $\mathbf{x}_i$ que

$$P(f(\mathbf{x}_i) = \oplus | T, C) = \sum_{r=1}^{2^{m-1}} P(f(\mathbf{x}_i) = \oplus | T, C, L^i_r) P(L^i_r).$$

Para calcular essa expressão, é preciso ter mais informações sobre o problema em mãos. Considere, por exemplo, que a função desconhecida f pertença a um espaço de funções F conhecido. Nesse caso, uma das classes de equivalência $E_1, E_2, \ldots, E_l$ geradas em F rotula corretamente os exemplos de treinamento e os exemplos candidatos. Suponha *a priori* que a rotulação correta desses exemplos pode ser realizada equiprovavelmente por qualquer uma das classes de equivalência[4]. Sendo assim, os termos do lado direito da equação podem ser calculados. A probabilidade de que a rotulação L_r^i coincidirá com a rotulação da sequência C^i dada pela função desconhecida f é

$$P(L_r^i) = \frac{\#(C^i, L_r^i)}{l}$$

na qual $\#(C^i, L_r^i)$ é o número de classes de equivalência que rotulam a sequência C^i de acordo com a rotulação L_r^i. A probabilidade condicional $P(f(\mathbf{x}_i) = \oplus | T, C, L_r^i)$ é calculada como

0, se não existe uma classe de equivalência que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \oplus;$$
$\frac{1}{2}$, se existe uma classe de equivalência que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \oplus$$
e uma que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \ominus;$$
1, se existe uma classe de equivalência que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \oplus$$
e não existe uma que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \ominus.$$

Uma propriedade interessante dessa formulação é que $P(f(\mathbf{x}_i) = \oplus | T, C) = \frac{1}{2}$ para qualquer candidato $\mathbf{x}_i$ quando o espaço de funções pode rotular os conjuntos de treinamento e de candidatos de todas as formas possíveis ($l = 2^{n+m}$). Isso significa que nenhuma inferência pode ser feita nesse caso, assim como já foi demonstrado para as inferências indutiva e transdutiva [Vapnik 2006, p. 152].

4. Experimento Ilustrativo

Na seção anterior foi apresentado o papel que a inferência seletiva pode desempenhar em aprendizado de máquina. Apesar de existirem tarefas de aprendizado nas quais a inferência seletiva deveria ser fortemente considerada, ainda não se conhecem algoritmos propriamente seletivos. No entanto, a formulação teórica apresentada por Vapnik [2006] pode oferecer um ponto de partida para esse desenvolvimento.

Nesta seção, com o intuito de realizar os primeiros experimentos com inferência seletiva, é proposto um algoritmo seletivo preliminar tendo como base a formulação teórica apresentada na seção anterior. Esse algoritmo é comparado experimentalmente com um algoritmo indutivo e um algoritmo transdutivo empregados na realização de inferência seletiva em nove bases de textos. Para simular uma situação de escassez de informação, os experimentos também são conduzidos em conjuntos de treinamento reduzidos artificialmente. Os resultados demonstram que um algoritmo seletivo pode obter sucesso mesmo quando os algoritmos indutivos e transdutivos falham.

[4]Também é possível levar em consideração a distribuição de exemplos positivos e negativos no problema para obter uma distribuição não uniforme [Vapnik 2006, p. 344].

4.1. Algoritmo

A formulação teórica apresentada na seção anterior expressa a probabilidade de cada exemplo candidato ser positivo de acordo com a função desconhecida f, sendo que o candidato que obtém a maior probabilidade é aquele que deve ser selecionado (Caso I com $p = 1$). Contudo, o cálculo dessas probabilidades impõe sérios problemas computacionais, pois há um somatório que envolve um número de parcelas igual a 2^{m-1}, sendo m o número de exemplos candidatos. Portanto, o algoritmo mais simples para calcular essas probabilidades será exponencial no número de exemplos candidatos.

Com o intuito de realizar os primeiros experimentos com inferência seletiva, é proposto um algoritmo seletivo preliminar baseado no caso em que a formulação teórica é aplicada a um único exemplo candidato. Consequentemente, $r = 1$, $P(L_r^i)$ fica indefinido e $P(f(\mathbf{x}_i) = \oplus|T, C) = P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i)$. A probabilidade condicional $P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i)$ é calculada como

- 0, se não existe uma classe de equivalência que permita a rotulação $T \cup \mathbf{x}_i, \oplus$;
- $\frac{1}{2}$, se existe uma classe de equivalência que permita a rotulação $T \cup \mathbf{x}_i, \oplus$ e uma que permita a rotulação $T \cup \mathbf{x}_i, \ominus$;
- 1, se existe uma classe de equivalência que permita a rotulação $T \cup \mathbf{x}_i, \oplus$ e não existe uma que permita a rotulação $T \cup \mathbf{x}_i, \ominus$.

No final, são selecionados todos os candidatos para os quais $P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i) = 1$.

No Algoritmo 1 apresentado a seguir, os exemplos candidatos para os quais $P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i) = 1$ são selecionados. Para isso, é utilizado o procedimento $SVM(T', \alpha, K)$ que executa o algoritmo indutivo Support Vector Machines [Burges 1998] sobre um conjunto de treinamento T', parâmetro de generalização α e *kernel* K. Os parâmetros α e K definem o espaço de funções F para o algoritmo SVM. Caso esse espaço de funções não acomode um determinado exemplo do conjunto de treinamento, a função obtida aplicada a esse exemplo assume um valor maior que -1 para exemplos negativos e menor que 1 para exemplos positivos. Assim, para que um exemplo candidato $\mathbf{x}_i$ obtenha $P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i) = 1$, é necessário que, ao ser inserido como positivo no conjunto de treinamento, a função obtida $f_\oplus$ assuma um valor maior ou igual a 1 em $\mathbf{x}_i$, e, ao ser inserido como negativo, a função obtida $f_\ominus$ assuma um valor maior que -1.

É importante ressaltar que, embora o algoritmo indutivo SVM esteja sendo utili-

Algoritmo 1: Algoritmo seletivo preliminar

Entrada:
- conjunto de treinamento $T = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$
- conjunto de candidatos $C = \{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}\}$
- parâmetro de generalização α
- *kernel* K

$S \leftarrow \emptyset$;
para cada $\boldsymbol{x}_i \in C$ **faça**
  $f_\oplus \leftarrow SVM(T \cup \{\mathbf{x}_i, \oplus\}, \alpha, K)$;
  se $f_\oplus(\boldsymbol{x}_i) \geq 1$ **então**
    $f_\ominus \leftarrow SVM(T \cup \{\mathbf{x}_i, \ominus\}, \alpha, K)$;
    se $f_\ominus(\boldsymbol{x}_i) > -1$ **então**
      $S \leftarrow S \cup \{\mathbf{x}_i\}$;
    fim
  fim
fim
Saída: conjunto de exemplos selecionados S

zado dentro do algoritmo seletivo proposto, os exemplos candidatos são incluídos um a um no conjunto de treinamento, além de ser necessário avaliar duas funções para decidir se um exemplo será selecionado ou não. Portanto, tal utilização de SVM não pode ser considerada como uma maneira de se realizar inferência indutiva.

4.2. Desenho Experimental

Para a realização dos experimentos com inferência seletiva, foram utilizadas coleções textuais provenientes de fontes variadas, como páginas da *web*, mensagens eletrônicas em grupos de discussão, artigos científicos e críticas de filmes. Essas coleções foram estruturadas em nove tabelas atributo-valor, utilizando unigramas para a descrição dos documentos (exemplos)[5]. As informações relativas às bases utilizadas e às respectivas tabelas atributo-valor estão na Tabela 1. Nessa tabela estão descritos, para cada base de dados: o domínio de origem; o número de documentos (#Doc.); o número original de atributos (#Atributos); o número de atributos após remoção de *stopwords*, *stemming* e corte por frequência (#Atrib. pós-corte); a classe positiva ($\oplus$); e a porcentagem de exemplos positivos na base($\oplus\%$).

Tabela 1. Descrição das bases de dados utilizadas

Base	Domínio	#Doc.	#Atributos	#Atrib. pós-corte	$\oplus$	$\oplus\%$
CBR	Artigos científicos	675	21545	5072	CBR	48
CS	Artigos científicos	823	68047	13336	IA	61
EC	Páginas *web*	1199	12621	3597	Computers	50
ET	Páginas *web*	1069	12941	3941	Transport	50
HARDWARE	Mensagens de *newsgroups*	1943	13398	3958	Mac	50
MOVIE	Críticas de filmes	2000	25302	10669	Positive review	50
SCIENCE	Artigos científicos	797	74739	15580	Physics	50
SPORTS	Mensagens de *newsgroups*	1993	14254	5741	Baseball	50
VEHICLES	Mensagens de *newsgroups*	1984	14048	5362	Motorcycles	50

Fixada uma base, o processo de avaliação é realizado com 10-*fold cross-validation* estratificado, gerando-se dez pares (T, C) para a base, sendo que os conjuntos de candidatos gerados são mutualmente disjuntos. Em cada iteração do processo de *cross-validation*, é realizada inferência seletiva com um dos dez pares de conjuntos (T, C), e a precisão da inferência é avaliada. A precisão da inferência na iteração k é definida como

$$pr_k = \frac{\#\ de\ exemplos\ positivos\ selecionados}{\#\ de\ exemplos\ selecionados} \quad \text{ou} \quad pr_k = \tfrac{1}{2}, se\ \#\ de\ ex.\ sel. = 0$$

O resultado final para uma determinada base é a média de pr_k em todas as iterações.

O processo de avaliação descrito é repetido três vezes, cada vez utilizando um algoritmo diferente para realizar inferência seletiva, porém mantendo-se os mesmos dez pares (T, C). Um dos algoritmos é o algoritmo indutivo Support Vector Machines (SVM). Esse algoritmo obtém uma função que é usada para classificar todos os exemplos candidatos. Nesse caso, todos os exemplos candidatos classificados como positivos são selecionados. O segundo algoritmo é o algoritmo transdutivo Transductive Support Vector Machines (TSVM). Nesse segundo caso, todos os exemplos candidatos rotulados como positivos são selecionados. O terceiro algoritmo é o algoritmo seletivo preliminar proposto anteriormente. O software UniverSVM[6] foi utilizado para executar SVM, TSVM e o procedimento que chama SVM no algoritmo proposto.

[5] As bases podem ser encontradas em `http://www.icmc.usp.br/~igorab/msc/datasets/`.

[6] `http://3t.kyb.tuebingen.mpg.de/bs/people/fabee/universvm.html`

Pelo fato dos três algoritmos nessa avaliação terem parâmetros a serem ajustados, é necessário executá-los variando-se esses parâmetros. O único parâmetro fixo em todas as execuções dos três algoritmos é o *kernel*, que nesta avaliação, foi fixada no *kernel* linear, o qual é tradicionalmente utilizado em bases de texto. Para SVM e o algoritmo seletivo proposto, foi realizada uma execução para cada um dos seguintes valores de α: 10^{-4}, 10^{-3}, ..., 10^{0}, ..., 10^{2}, 10^{3}. Para TSVM, que tem mais parâmetros a serem ajustados, foi realizada uma execução para os mesmos valores de α mencionados, enquanto os outros parâmetros foram mantidos em seu valor padrão. O valor médio de pr_k relatado em cada base para um determinado algoritmo é dado pelo maior valor médio de pr_k atingido utilizando os diferentes valores de α. Por exemplo, se o algoritmo SVM ao ser executado na base CBR atinge o maior valor médio de pr_k para $\alpha = 10^{-}4$, então esse é o valor médio de pr_k que será relatado.

Além de avaliar os três algoritmos usando todos os exemplos da base de dados, foi realizada uma simulação de escassez de informação para verificar como se comportam os algoritmos nessa situação. Para isso, todo o processo mencionado anteriormente foi repetido usando-se 10%, 5% e 1% dos exemplos de treinamento, enquanto o número de exemplos em C era mantido constante. Por exemplo, considere que a base tem um total de 1000 exemplos, então cada um dos *folds* possui 100 exemplos. Nesse caso, cada uma das $k = 10$ iterações com 100% dos exemplos de treinamento considera $|T| = 900$ (9 *folds*) e $|C| = 100$ (1 *fold*). Entretanto, nas $k = 10$ iterações com 10% dos exemplos de treinamento, essa porcentagem de exemplos é extraída aleatoriamente dos 9 *folds* anteriores, ou seja, $|T| = 90$ e $|C| = 100$. Analogamente, $|T| = 45$ e $|T| = 9$ quando, respectivamente, 5% e 1% dos exemplos de treinamento são considerados. Em todos os casos, no entanto, é mantido $|C| = 100$. Os exemplos que integram os conjuntos de treinamento reduzidos foram escolhidos previamente à execução dos algoritmos, a fim de que os experimentos fossem pareados.

4.3. Resultados e Análise

Os resultados obtidos são apresentados na Tabela 2. Em cada base de dados, o comportamento dos algoritmos indutivo (SVM), transdutivo (TSVM) e seletivo (preliminar) podem ser observados quando 100%, 10%, 5% e 1% dos exemplos do conjunto de treinamento estão disponíveis. Em cada um desses casos, é mostrado o valor médio de pr_k atingido quando os algoritmos são executados com o melhor valor do parâmetro α.

Como pode ser observado na Tabela 2, o algoritmo seletivo proposto sempre tem desempenho superior ou igual aos algoritmos indutivo e transdutivo nas faixas de 100%, 10% e 5%. Quando 100% do conjunto de treinamento é utilizado, o algoritmo seletivo supera com boa margem os outros dois algoritmos nas bases EC, HARDWARE e MOVIE. Nas faixas de 10% e 5%, o algoritmo seletivo se destaca, superando os outros dois algoritmos com boa margem em quase todas as bases. Nessas faixas é visível, principalmente nas bases CS e HARDWARE, que um algoritmo seletivo pode obter bons resultados mesmo quando os algoritmos indutivo e transdutivo não obtêm bons resultados.

Na faixa de 1%, no entanto, o algoritmo seletivo só aprensenta resultados melhores nas bases SPORTS e VEHICLES. Nas bases EC, HARDWARE e MOVIE, os resultados são semelhantes. Nas bases CBR, CS, ET e SCIENCE, o algoritmo seletivo obtém piores resultados que os outros dois algoritmos. Nessas quatro bases, o algoritmo seletivo

Tabela 2. Comportamento dos algoritmos indutivo (SVM), transdutivo (TSVM) e seletivo (preliminar) usando 1%, 5%, 10% e 100% do conjunto de treinamento de cada base.

(a) CBR

Alg.	1%	5%	10%	100%
Ind.	0,95	**1,00**	**1,00**	**1,00**
Trans.	**0,98**	**1,00**	**1,00**	**1,00**
Sel.	0,84	**1,00**	**1,00**	**1,00**

(b) CS

Alg.	1%	5%	10%	100%
Ind.	**0,73**	0,87	0,91	0,95
Trans.	0,66	0,81	0,84	0,88
Sel.	0,52	**0,95**	**1,00**	**1,00**

(c) EC

Alg.	1%	5%	10%	100%
Ind.	0,58	0,68	0,73	0,91
Trans.	0,54	0,65	0,72	0,91
Sel.	**0,59**	**0,78**	**0,86**	**0,99**

(d) ET

Alg.	1%	5%	10%	100%
Ind.	**0,65**	0,83	0,98	0,98
Trans.	0,61	0,77	0,87	0,98
Sel.	0,61	**0,93**	**0,98**	**1,00**

(e) HARDWARE

Alg.	1%	5%	10%	100%
Ind.	0,80	0,80	0,91	0,93
Trans.	0,70	0,80	0,87	0,93
Sel.	**0,81**	**0,98**	**0,98**	**1,00**

(f) MOVIE

Alg.	1%	5%	10%	100%
Ind.	**0,72**	0,77	0,89	0,84
Trans.	0,58	0,67	0,74	0,83
Sel.	**0,72**	**0,82**	**0,91**	**0,96**

(g) SCIENCE

Alg.	1%	5%	10%	100%
Ind.	0,84	0,96	0,96	**1,00**
Trans.	**0,88**	0,96	0,97	**1,00**
Sel.	0,60	**0,99**	**1,00**	**1,00**

(h) SPORTS

Alg.	1%	5%	10%	100%
Ind.	0,74	0,93	0,97	0,99
Trans.	0,73	0,94	0,98	0,99
Sel.	**0,87**	**0,98**	**1,00**	**1,00**

(i) VEHICLES

Alg.	1%	5%	10%	100%
Ind.	0,81	0,87	0,94	**1,00**
Trans.	0,87	0,94	0,95	0,95
Sel.	**0,90**	**1,00**	**1,00**	**1,00**

se recusou a selecionar exemplos positivos em diversas iterações do processo de *cross-validation*, o que fez a média de pr_k para o algoritmo seletivo ter uma queda brusca em comparação com os resultados usando 5% dos exemplos de treinamento. Uma correlação importante nesse caso é o fato de que as quatro bases são as menores bases entre as nove consideradas — veja a Tabela 1. Dessa forma o número absoluto de exemplos é muito baixo para essas bases na faixa de 1%, indo de 6 a 10 exemplos de treinamento.

O comportamento do algoritmo seletivo na faixa de 1% precisa ser melhor estudado. No entanto, é importante lembrar que esse algoritmo considera cada exemplo candidato isoladamente. Portanto, assim que seja desenvolvido um algoritmo seletivo que consiga trabalhar com todos os exemplos candidatos conjuntamente, haverá uma quantidade maior de exemplos disponíveis na faixa de 1%.

5. Considerações Finais

Neste artigo foi apresentado um tipo de inferência ainda não explorado em aprendizado de máquina, denominado inferência seletiva. O desenvolvimento de algoritmos de inferência seletiva pode beneficiar tarefas de aprendizado em importantes domínios de aplicação, como a bioinformática e a busca na *web*. Com o intuito de realizar os primeiros experimentos com inferência seletiva em aprendizado de máquina, foi proposto um algoritmo seletivo preliminar baseado em uma formulação teórica dada na literatura.

Os resultados de experimentos em nove bases de texto mostram uma boa vantagem do algoritmo seletivo proposto sobre um algoritmo indutivo (SVM) e um algoritmo transdutivo (TSVM) empregados na realização de inferência seletiva. A vantagem se mostrou maior quando o número de exemplos de treinamento foi reduzido a 10% e a 5% do número de exemplos de treinamento obtidos originalmente com 10-*fold cross-validation*. No entanto, quando o número de exemplos de treinamento foi reduzido a 1%, o algoritmo seletivo obteve piores resultados que os outros dois algoritmos nas quatro menores bases.

Como o algoritmo proposto considera cada exemplo candidato isoladamente, os trabalhos futuros focalizarão o desenvolvimento de um algoritmo seletivo que integre de uma só vez todos os exemplos candidatos. Para isso, serão investigadas novas formulações teóricas para inferência seletiva, a fim de se evitar o somatório exponencial no número de candidatos. Além disso, pretende-se continuar investigando novas aplicações para inferência seletiva em aprendizado de máquina.

Agradecimentos À FAPESP, pelo apoio financeiro a este trabalho.

Referências

Agarwal, S. e Sengupta, S. (2009). "Ranking genes by relevance to a disease". Em *CSB '09: Proceedings of the 8th Annual International Conference on Computational Systems Bioinformatics*, páginas 1–10.

Alpaydin, E. (2004). *Introduction to Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press.

Audibert, J. Y. (2008). Transductive learning and computer vision. NIPS 2008 Workshop on Learning with Data-dependent Concept Spaces (abstract and invited talk).

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

Campos, L. F. B. (2007). Increase of precision on the top of the list of retrieved web documents using global and local link analysis. *Webology*, 4(3). `http://www.webology.ir/2007/v4n3/a44.html`.

El-Yaniv, R., Pechyony, D., e Vapnik, V. (2008). Large margin vs. large volume in transductive learning. *Machine Learning*, 72(3):173–188.

Joachims, T. (1999). "Transductive inference for text classification using support vector machines". Em *ICML '99: Proceedings of the 16th International Conference on Machine Learning*, páginas 200–209.

Joachims, T. (2002). "Optimizing search engines using clickthrough data". Em *KDD '02: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, páginas 133–142.

Joachims, T. (2003). "Transductive learning via spectral graph partitioning". Em *ICML '03: Proceedings of the 20th International Conference on Machine Learning*, páginas 290–297.

Rosenblatt, F. (1957). The perceptron: a perceiving and recognizing automaton. Relatório Técnico 85-460-1, Cornell Aeronautical Laboratory.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag.

Vapnik, V. (2006). *Estimation of Dependences Based on Empirical Data*. Information Science and Statistics. Springer-Verlag, 2^a edição.

Weston, J., Perez-Cruz, F., Bousquet, O., Chapelle, O., Elisseeff, A., e Scholkopf, B. (2003). Feature selection and transduction for prediction of molecular bioactivity for drug design. *Bioinformatics*, 19(6):764–771.