IMPLEMENTAÇÃO DE CLASSIFICAÇÃO BAYESIANA

Gabriela Silva de Oliveira[1], Silvio do Lago Pereira[2]
[1]Aluna do Curso de Análise e Desenvolvimento de Sistemas – DTI/FATEC-SP
[2]Prof. Dr. do Departamento de Tecnologia da Informação – FATEC-SP
gsilvadoliveira@gmail.com, slago@fatecsp.br

Resumo

C*lassificação* é uma técnica de aprendizado de máquina supervisionado que, a partir de exemplos de objetos de um conjunto predefinido de classes, sintetiza modelos preditivos capazes de determinar a classe de um objeto em função dos valores de seus atributos. Esses modelos preditivos têm aplicações práticas em várias áreas como, por exemplo, medicina, economia, negócios e biologia. Há diversas abordagens que podem ser empregadas para predizer a classe de um objeto (e.g., árvores de decisão e redes neurais), mas a abordagem estatística, além de predizer a classe de um objeto, também pode informar a probabilidade de sua predição. Particularmente, classificadores bayesianos são algoritmos estatísticos que classificam objetos analisando a probabilidade de sua pertinência em cada uma das possíveis classes predefinidas e escolhendo aquela mais provável. Evidentemente, todo modelo preditivo está sujeito a erro e modelos com maior acurácia são preferíveis. Nesse contexto, o objetivo deste artigo é descrever a implementação de um classificador bayesiano e avaliar sua acurácia (i.e., taxa de acertos).*

1. Introdução

Aprendizado de Máquina [1] é uma área de pesquisa da Inteligência Artificial que estuda como criar algoritmos capazes de melhorar seu desempenho na realização de uma tarefa, por meio da experiência. Particularmente para a tarefa de *classificação* [2], os algoritmos devem ser capazes de induzir hipóteses, a partir de exemplos de objetos previamente classificados por um especialista, e usá-las para classificar novos objetos posteriormente considerados. As hipóteses induzidas pelos algoritmos de classificação são denominadas *modelos preditivos* e, como elas são induzidas a partir de exemplos já classificados, o aprendizado implementado pelos algoritmos é denominado *aprendizado supervisionado*.

Modelos preditivos têm aplicação prática em várias áreas; por exemplo, em medicina, um modelo preditivo pode ser usado para diagnosticar doenças, a partir de resultados de exames feitos pelos pacientes; em economia e negócios, um modelo preditivo pode ser usado para decidir a aprovação de crédito, a partir de informações pessoais e financeiras dos clientes; em biologia, um modelo preditivo pode ser usado para determinar a espécie de um animal ou planta desconhecido [3].

Há diversas abordagens que podem ser empregadas para a síntese de modelos preditivos, a partir de exemplos previamente classificados por um especialista do domínio de aplicação (e.g., árvores de decisão e redes neurais) [3]. Porém, a abordagem estatística tem a vantagem de possibilitar a síntese de modelos preditivos que, além de predizerem a classe de um objeto, também podem informar a probabilidade de suas predições se confirmarem.

Numa abordagem estatística, a probabilidade de um evento é estimada com base na observação da frequência com que este evento ocorre em um determinado contexto. Por exemplo, no contexto da Figura 1, um conjunto de animais (i.e., *objetos*) previamente classificados por um especialista, como jacarés ou crocodilos, está disponível. Neste contexto, cada animal é descrito por um par de atributos quantitativos contínuos: comprimento da *cauda* e tamanho da *boca*, como na Figura 1-a. Os valores desses atributos podem ser usados para a construção de histogramas. Por exemplo, a Figura 1-b mostra histogramas criados a partir dos dados relativos ao comprimento da cauda, que podem ser sumarizados por distribuições de probabilidades normais, como ilustrado na Figura 1-c.

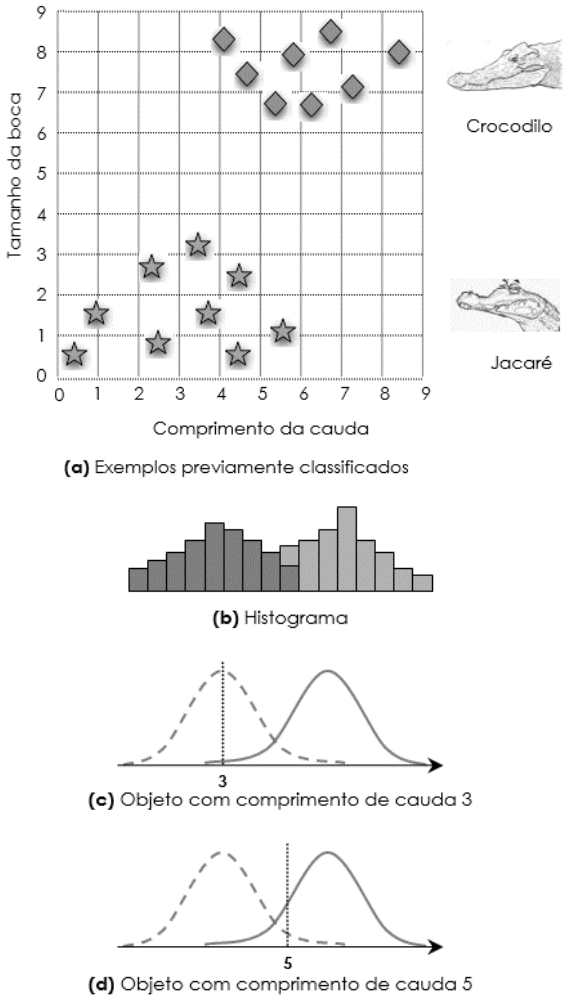

Figura 1 – Classificação de animais, adaptado de [4].

Suponha que um novo animal com comprimento de cauda *k* seja encontrado e deva ser classificado como **jacaré** ou **crocodilo** . Para decidir a classe mais provável desse animal desconhecido, com base nas probabilidades observadas no conjunto de exemplos, podemos verificar

qual classe tem maior probabilidade para a característica observada; ou seja, calcular a probabilidade de ele ser **jacaré**, $P(animal=\textbf{jacaré} \mid cauda=k)$, ou **crocodilo**, $P(animal=\textbf{crocodilo} \mid cauda=k)$, dado que sua cauda tem comprimento k, e escolher a classe com maior probabilidade. Por exemplo, para $k = \mathbf{3}$ (Figura 1-c), a classe mais provável é **jacaré**; por outro lado, para $k = \mathbf{5}$ (Figura 1-d), a classe mais provável é **crocodilo**.

Evidentemente, todo modelo preditivo está sujeito a erro e o objetivo da classificação é obter modelos preditivos que tenham a maior *acurácia* possível (i.e., taxa de acerto da predição). Na prática, é difícil aferir a acurácia real de um modelo preditivo. Assim, a acurácia dos modelos costuma ser avaliada com relação a exemplos também previamente classificados por um especialista, mas que não foram usados para a síntese do modelo.

Neste contexto, o objetivo deste artigo é descrever a implementação de um *classificador bayesiano* e relatar resultados de experimentos que indicam sua acurácia.

O restante deste artigo está organizado do seguinte modo: a Seção 2 introduz a fundamentação teórica do trabalho; a Seção 3 descreve as principais características do classificador bayesiano desenvolvido; a Seção 4 discute resultados de experimentos feitos com o classificador; e a Seção 5 apresenta as conclusões finais do trabalho.

2. Fundamentos Teóricos

Os fundamentos teóricos deste trabalho são o Teorema de Bayes e o algoritmo de classificação bayesiana.

2.1. Teorema de Bayes

O *Teorema de Bayes* [5] estabelece a relação entre uma probabilidade condicional e a sua inversa, isto é, entre a probabilidade de uma hipótese H, dada uma evidência E, e a probabilidade de uma evidência E, dada uma hipótese H. Mais precisamente:

$$P(H \mid E) = \frac{P(E \mid H)P(H)}{P(E)}, \qquad (1)$$

sendo:

- $P(H) \neq 0$ e $P(E) \neq 0$ as probabilidades independentes da hipótese H e da evidência E;
- $P(H \mid E)$ a probabilidade condicional da hipótese H, dada a evidência E;
- $P(E \mid H)$ a probabilidade condicional da evidência E, dada a hipótese H.

Por exemplo, esse teorema pode ser usado para avaliar a confiabilidade do resultado de um exame para diagnosticar uma determinada doença [6]. Considerando que:

- Apenas 1% das pessoas têm a doença, ou seja, a probabilidade de alguém selecionado aleatoriamente ter a doença é $P(H) = 0,01$ e de não ter a doença é $P(\bar{H}) = 0,99$.
- O exame tem resultado positivo para 95% das pessoas que realmente têm a doença, ou seja, $P(E_+ \mid H) = 0,95$. O mesmo exame tem resultado negativo para 95% das pessoas que não têm a doença, ou seja, $P(E_- \mid \bar{H}) = 0,95$.
- Se uma pessoa não tem a doença, a probabilidade de um resultado falso positivo é de 5%, ou seja, $P(E_+ \mid \bar{H}) = 0,05$.

Qual a probabilidade de uma pessoa que teve um exame com resultado positivo realmente ter a doença? Para obter a resposta a esta pergunta, basta calcular $P(H \mid E_+)$.

De acordo com o princípio aditivo da probabilidade:

$$P(E_+) = P(E_+ \mid H)P(H) + P(E_+ \mid \bar{H})P(\bar{H}) \qquad (2)$$

Substituindo o denominador da Equação (1) pela Equação (2), temos:

$$P(H \mid E_+) = \frac{P(E_+ \mid H)P(H)}{P(E_+ \mid H)P(H) + P(E_+ \mid \bar{H})P(\bar{H})} \qquad (3)$$

$$P(H \mid E_+) = \frac{0,95 \times 0,01}{0,95 \times 0,01 + 0,05 \times 0,99} \cong 16,10\% \qquad (4)$$

Portanto, apesar de o exame apresentar acurácia de 95%, apenas $16,10\%$ das pessoas que têm um resultado positivo no exame realmente têm a doença.

2.2. Classificação Bayesiana

O algoritmo de classificação bayesiana considerado neste artigo, *Naïve-Bayes* [4], usa o Teorema de Bayes para prever a *classe* de um objeto, supondo que seus *atributos* são independentes. Apesar de essa suposição nem sempre ser verdadeira, resultados empíricos descritos na literatura mostram que o algoritmo *Naïve-Bayes* tem bons resultados e custo computacional baixo [3,4,5].

O algoritmo pode ser usado para classificar objetos descritos por atributos *qualitativos* (i.e., nominais ou ordinais) e *quantitativos* (i.e., discretos ou contínuos). As probabilidades de ocorrência de valores específicos de atributos qualitativos ou discretos são dadas por frequências relativas (obtidas por contagem). Por outro lado, as probabilidades de ocorrência de valores específicos de atributos contínuos são dadas por distribuições de probabilidades normais (i.e., *gaussianas*), o que pode diminuir a acurácia dos resultados, quando as distribuições reais desses valores não são normais.

Por exemplo, considere o problema de classificar uma pessoa como sendo do *sexo* feminino ou masculino [4], com base no seu *nome* e nas probabilidades que podem ser obtidas a partir dos exemplos previamente classificados apresentados na Tabela I.

Tabela I – Conjunto de exemplos de pessoas.

Nome	Sexo
Reid	Masculino
Reid	Feminino
Morgan	Masculino
Reid	Masculino
Joan	Feminino
Rachel	Feminino
Elizabeth	Feminino

Para decidir o sexo mais provável de uma pessoa chamada *Reid*, considerando os dados na Tabela I, basta observar que:

- A probabilidade de ocorrência do nome *Reid* é $P(nome = \textbf{Reid}) = 3/7$;
- A probabilidade de *Reid* ser do sexo feminino é $P(nome=\textbf{Reid} \mid sexo=\textbf{feminino}) = 1/4$;
- A probabilidade de *Reid* ser do sexo masculino é $P(nome=\textbf{Reid} \mid sexo=\textbf{masculino}) = 2/3$;
- A probabilidade do sexo feminino no conjunto de exemplos é $P(sexo = \textbf{feminino}) = 4/7$.
- A probabilidade do sexo masculino no conjunto de exemplos é $P(sexo = \textbf{masculino}) = 3/7$.

Então, de acordo com o Teorema de Bayes, segue que:

$$P(sexo=\textbf{feminino} \mid nome=\textbf{Reid})$$
$$= \frac{P(nome=\textbf{Reid} \mid sexo=\textbf{feminino})\,P(sexo=\textbf{feminino})}{P(nome=\textbf{Reid})} \quad (5)$$
$$= \frac{1/4 \times 4/7}{3/7} \cong 33{,}3\%$$

$$P(sexo=\textbf{masculino} \mid nome=\textbf{Reid})$$
$$= \frac{P(nome=\textbf{Reid} \mid sexo=\textbf{masculino})\,P(sexo=\textbf{masculino})}{P(nome=\textbf{Reid})} \quad (6)$$
$$= \frac{2/3 \times 3/7}{3/7} \cong 66{,}6\%$$

Portanto, pode-se concluir que, de acordo com os dados na Tabela I, uma pessoa chamada *Reid* é mais provavelmente do sexo *masculino*.

Embora os objetos (i.e., pessoas) nesse exemplo sejam descritos por um único atributo (i.e., *nome*) e a classe (i.e., *sexo*) seja binária, o algoritmo *Naïve-Bayes* também pode ser usado quando os objetos são descritos por vários atributos e há diversas classes. Nesse caso, uma evidência E é dada por um vetor de atributos (cujos elementos $E_1, E_2, ..., E_n$ são considerados independentes) e o Teorema de Bayes pode ser generalizado por:

$$P(H_i \mid \vec{E}) = \frac{P(E_1 \mid H_i)P(E_2 \mid H_i)\cdots P(E_n \mid H_i)P(H_i)}{P(\vec{E})} \quad (7)$$

Nos exemplos considerados na Tabela I, todos os atributos são *qualitativos nominais*; porém, o algoritmo de classificação bayesiana também pode ser usado com atributos *quantitativos contínuos*. Neste caso, em vez de representar as probabilidades pela frequência relativa dos valores dos atributos no conjunto de exemplos, o algoritmo representa as probabilidades usando a seguinte função de distribuição de probabilidades normal:

$$P(E_i = x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad (8)$$

onde μ é a média dos valores dos atributos contínuos e σ é o desvio padrão correspondente, calculados a partir dos dados disponíveis.

3. O Classificador Bayesiano Desenvolvido

O classificador bayesiano, desenvolvido em Python [7], pode ser facilmente usado por quem precisa de um algoritmo de classificação, mas não tem conhecimentos específicos na área de aprendizado de máquina. A janela principal do classificador é apresentada na Figura 2.

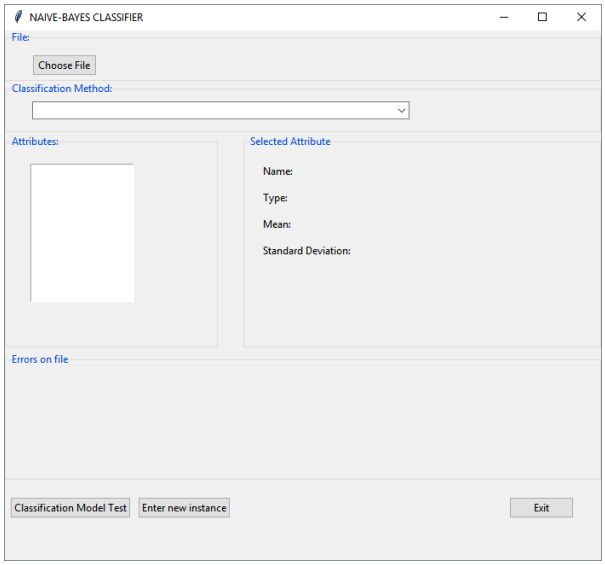

Figura 2 – Janela principal do classificador desenvolvido.

3.1. Interface de Usuário do Classificador

Para usar o classificador, primeiramente o usuário deve selecionar um arquivo do tipo ARFF [8] contendo os exemplos que deverão ser usados para a síntese do modelo preditivo. Para isto, basta que ele clique o botão Choose File, na janela principal do sistema. A partir daí, é aberta uma janela para navegação no sistema de arquivos do computador e escolha do arquivo desejado, como mostra a Figura 3.

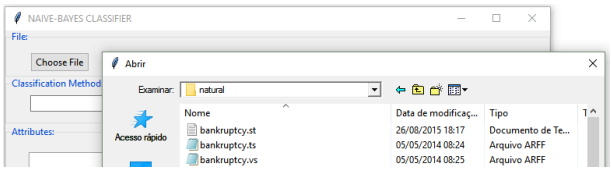

Figura 3 – Janela para navegação no sistema de arquivos.

Assim que um arquivo é selecionado, o sistema lê e compila os dados existentes no arquivo. Caso haja erros de sintaxe na especificação dos exemplos no arquivo, o sistema informa na janela principal os erros encontrados, como mostra a Figura 4 (os exemplos com erros de sintaxe não são usados na síntese do modelo preditivo). Caso contrário, o sistema atualiza sua janela principal com informações sobre o arquivo lido, como mostra a Figura 5, e gera as *tabelas de probabilidades* para os atributos nominais e/ou discretos, bem como as *funções de distribuição normais* para os atributos contínuos.

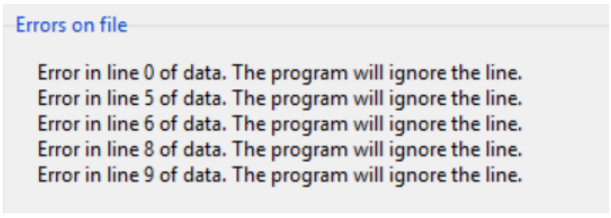

Figura 4 – Janela de erros de sintaxe nos exemplos.

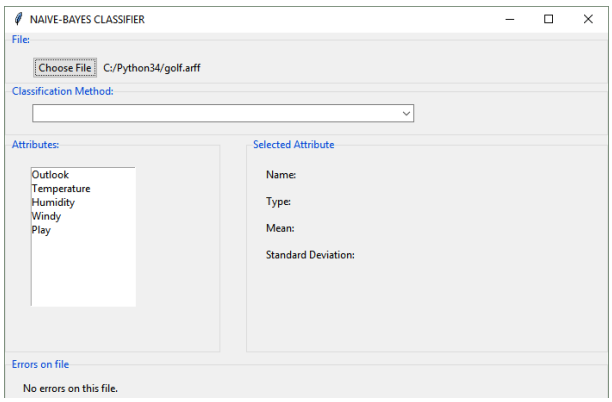

Figura 5 – Janela principal atualizada.

Após a geração das tabelas e funções de probabilidades (i.e., modelo preditivo), o sistema habilita duas outras funcionalidades: a *classificação* de uma nova instância, que deve ser do mesmo tipo dos objetos descritos nos exemplos do arquivo lido, e a *acurácia* do modelo preditivo, que foi sintetizado a partir dos exemplos do arquivo.

Para classificar uma nova instância, o usuário precisa clicar o botão Enter new instance. A janela para entrada da nova instância a ser classificada é exibida na Figura 6.

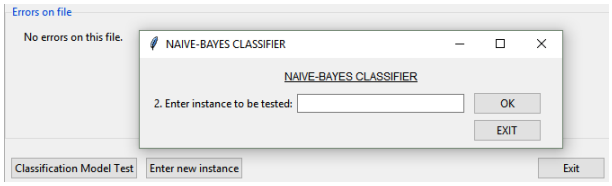

Figura 6 – Janela para entrada da instância a ser classificada.

Depois que o usuário digita os atributos da nova instância a ser classificada e clica o botão OK, o sistema abre uma janela informando a probabilidade de essa instância pertencer a cada uma das possíveis classes consideradas no modelo preditivo, como mostra a Figura 7.

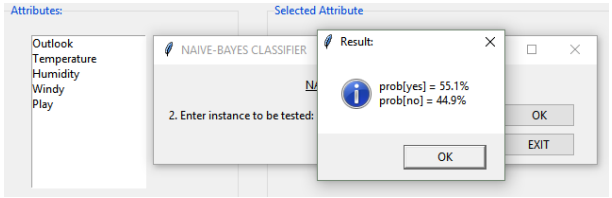

Figura 7 – Resultado da classificação de uma instância.

Para avaliar o modelo preditivo sintetizado, o usuário deve clicar o botão Classification Model Test. A acurácia e *g-measure* [9] são exibidas no IDE do Python (Figura 8).

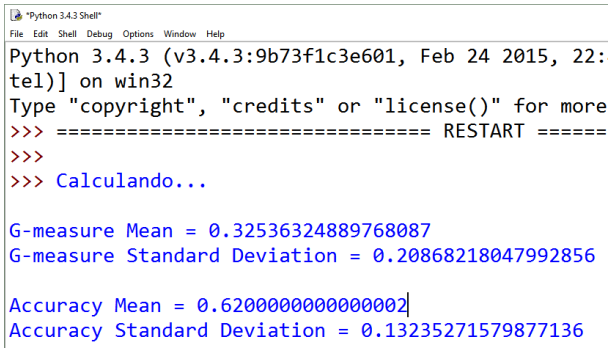

Figura 8 – Resultados do teste do modelo preditivo.

3.2. Formato do Arquivo de Entrada

Para o correto funcionamento do sistema, o arquivo de entrada deve ser formatado de acordo com o padrão ARFF (*Attribute Relation File Format*) [8], amplamente empregado em mineração de dados e, em particular, em sistemas de classificação.

Cada arquivo ARFF representa um conjunto de exemplos descritos pelos mesmos atributos. Dentro do arquivo constam as declarações dos atributos e seus respectivos tipos, seguidos de registros de dados descrevendo os exemplos a serem considerados para a síntese do modelo preditivo. Cada registro de dado é representado por uma sequência de valores correspondentes aos atributos declarados. A ordem dos atributos em cada registro deve ser a mesma ordem na qual os atributos são declarados. O último atributo declarado é, por definição, o atributo de classificação (i.e., que indica a classe de cada exemplo e cujo valor deve ser decidido para uma nova instância). Um exemplo de arquivo ARFF é apresentado na Figura 9.

```
@relation golf
@attribute outlook     {sunny, overcast, rainy}
@attribute temperature numeric
@attribute humidity    numeric
@attribute windy       {true, false}
@attribute play        {yes, no}
@data
sunny,     85, 85, false, no
sunny,     80, 90, true,  no
overcast,  83, 86, false, yes
rainy,     70, 96, false, yes
rainy,     68, 80, false, yes
rainy,     65, 70, true,  no
overcast,  64, 65, true,  yes
sunny,     72, 95, false, no
sunny,     69, 70, false, yes
rainy,     75, 80, false, yes
sunny,     75, 70, true,  yes
overcast,  72, 90, true,  yes
overcast,  81, 75, false, yes
rainy,     71, 91, true,  no
```

Figura 9 – Arquivo ARFF para a relação *golf*.

3.3. Funcionamento Interno do Sistema

A compilação do arquivo ARFF é feita com uma gramática especificada com funções do *pyparsing*, um módulo do Python que oferece recursos para compilação robusta de dados com qualquer formatação.

Na primeira fase da compilação, o sistema usa a gramática para gerar duas listas: uma com a descrição dos atributos declarados no arquivo, e outra com os registros de dados devidamente consistidos e convertidos para os tipos declarados.

Na segunda fase da compilação, o sistema usa as duas listas geradas na primeira fase para criar as tabelas de probabilidades de atributos qualitativos (i.e., nominais e categóricos) e quantitativos discretos, bem como as funções de distribuição de probabilidade normal para os atributos quantitativos contínuos. Internamente, tanto as tabelas quanto as funções de probabilidades são representadas por dicionários (i.e., tabelas de *hashing*), que permitem acesso eficiente em tempo constante.

A probabilidade dos atributos qualitativos e quantitativos discretos é obtida por contagem e *ajuste laplaciano*

[2]. A probabilidade dos atributos quantitativos contínuos é obtida a partir da média e do desvio padrão, assumindo-se que tais atributos possuem distribuição normal.

Quando uma nova instância precisa ser classificada, as probabilidades obtidas durante a compilação dos dados são usadas para o cálculo da Equação (7), para cada um dos possíveis valores do atributo de classificação.

4. Discussão dos Resultados Empíricos

Todos os experimentos com o classificador bayesiano desenvolvido foram realizados com conjuntos de dados previamente classificados por especialistas dos domínios e disponíveis no repositório de aprendizado de máquina da UCI (*University of California*, Irvine) [10]. Alguns desses conjuntos precisaram ser formatados de acordo com o padrão ARFF, pois originalmente não se encontravam nesse formato. Os conjuntos de dados selecionados para os experimentos cobrem diversos domínios de aplicação. Por exemplo, o conjunto *Bankruptcy* permite sintetizar um modelo de predição de falência de empresas; o conjunto *Heart* permite sintetizar um modelo de predição de doenças cardíacas; o conjunto *Mushroom* permite sintetizar um modelo de predição de espécie de cogumelos; e o conjunto *Lenses* permite sintetizar um modelo de predição de indicação para uso de lentes de contato.

Cada conjunto de dados usado nos experimentos foi dividido em duas partes: uma delas foi usada para a síntese do modelo preditivo (*conjunto de treinamento*) e a outra foi reservada para avaliação da acurácia do modelo sintetizado (*conjunto de validação*). A Figura 10 ilustra o processo de síntese e avaliação dos modelos preditivos.

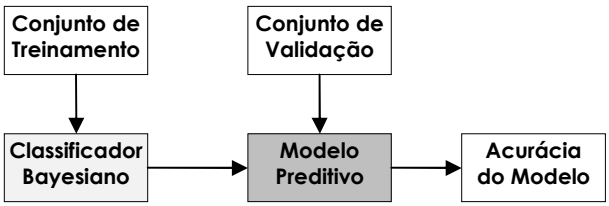

Figura 10 – Síntese a avaliação dos modelos preditivos.

Esse método de avaliação, denominado *holdout* [3], especifica que uma boa proporção para a divisão do conjunto de dados é usar $2/3$ dos dados para treinamento e o $1/3$ restante para validação. Porém, para evitar que a ordenação original dos dados nos conjuntos interferisse nos resultados (por exemplo, em alguns conjuntos de dados, as classes ocorrem agrupadas e a mera divisão desses conjuntos em $2/3$ e $1/3$ pode resultar em conjuntos de treinamento ou validação em que todos os exemplos sejam da mesma classe), antes de serem divididos, os conjuntos de dados foram devidamente embaralhados. Além disso, para manter a proporção da distribuição de classes observada no conjunto de dados original, os dados para compor os conjuntos de treinamento e validação foram criteriosamente selecionados. Por exemplo, para um conjunto de dados original contendo 60% de exemplos de uma classe e 40% de outra, foram selecionados 2/3 dos dados, de modo que 60% fossem de uma classe e os demais 40% fossem da outra classe.

O método *holdout* é indicado para conjuntos de dados grandes. Para conjuntos de dados pequenos, a acurácia calculada pode sofrer muita variação (especialmente, devido ao embaralhamento dos dados). Mesmo assim, os resultados obtidos são úteis para avaliar a capacidade de generalização dos modelos preditivos sintetizados.

Nos experimentos deste trabalho, foram feitas duas medições: *acurácia* e *g-measure* (Tabelas II e III, respectivamente). A acurácia representa a porcentagem dos acertos de predição, que pode ser elevada mesmo quando o modelo erra todas as predições relativas a uma das classes (por exemplo, se o conjunto original de dados tem 99 exemplos da classe A e apenas 1 exemplo da classe B, um modelo preditivo que "chuta" a predição A para toda instância tem uma acurácia de 99%). Para evitar esse problema, usamos a *g-measure*, que é a média geométrica entre taxas de acerto para cada possibilidade de classificação [9]. Assim, quando as instâncias de uma das classes são frequentemente erroneamente classificadas, o valor da *g-measure* indica a real precisão da classificação.

Tabela II – Medidas de acurácia.

Conjunto de dados	Acurácia	
	Média	Desvio Padrão
Bankruptcy	0.9873015873	0.0149716720
Car	0.9103448276	0.0276900542
Chess	0.8544600939	0.0193769256
Flare	0.9457865169	0.0082193857
Heart	**0.6887640449**	0.0439793849
House votes	0.9125000000	0.0582614500
Kr-vs-k	0.9869146006	0.0040361199
Lenses	**0.7791666667**	0.0848638192
Mushroom	0.9473916888	0.0172196717
Nursery	1.0000000000	0.0000000000
Tic-tac-toe	**0.6645833333**	0.0498741232
Iris	0.9575163399	0.0230761695

Tabela III – Medidas *g-measure*.

Conjunto de dados	G-Measure	
	Média	Desvio Padrão
Bankruptcy	0.9860197483	0.0020104407
Car	0.8859770858	0.0141982403
Chess	0.8561947257	0.0056985925
Flare	**0.5039453463**	0.0291906847
Heart	0.7006423324	0.0198258911
Housevotes	0.9071918226	0.0201970488
Kr-vs-k	0.7383559707	0.0206526145
Lenses	0.7735491466	0.0308711535
Mushroom	0.9280810181	0.0034354662
Nursery	1.0000000000	0.0000000000
Tic-tac-toe	**0.6059691723**	0.0137568654
Iris	0.9438368418	0.0096374854

Os resultados apresentados na Tabela II mostram que, para a maioria dos conjuntos de dados usados nos experimentos (75%), a acurácia média dos modelos preditivos sintetizados pelo classificador bayesiano desenvolvido é alta (acima de 85%). Esses resultados foram obtidos a partir de 30 execuções para cada conjunto de dados (para reduzir o impacto da variação resultante do embaralhamento dos conjuntos de dados) e, como os desvios padrão são baixos, essas acurácias são confiáveis. Ademais, os resultados na Tabela III também mostram que, para a maioria dos casos (83,3%), a *g-mesure* dos modelos é superior a 70% (também com desvios padrão baixos), o que aumenta a confiabilidade dos resultados.

Além dos experimentos sumarizados nas Tabelas II e III, também foi feito um experimento preliminar (que será estendido em trabalho futuro) para investigar a influência da discretização de atributos contínuos na acurácia dos modelos sintetizados. Esse experimento foi feito com dois conjuntos de dados descritos em [2]. O primeiro deles é aquele na Figura 9, o segundo é um conjunto equivalente em que os atributos contínuos foram manualmente discretizados (ou seja, `temperature` passou a assumir valores do conjunto `{hot,mild,cool}` e `humidity` passou a assumir valores do conjunto `{high,normal,low}`). Embora a acurácia obtida com ambos os conjuntos de dados tenha sido baixa (devido ao fato de eles serem muito pequenos para possibilitar generalização indutiva), os resultados apresentados na Tabela IV indicam que a acurácia pode aumentar com a discretização de atributos. Isso pode ser explicado pelo fato de que nem todo atributo contínuo possui uma distribuição normal, como é assumido pelo algoritmo *Naïve-Bayes*. Então, uma vez que tais atributos contínuos sejam discretizados, a obtenção de probabilidades por contagem pode ser mais precisa.

Tabela IV – Influência da discretização de atributos.

	Atributos	
	Contínuos	Discretizados
Acurácia	0.52666667	0.64000000
G-Measure	0.33537479	0.36671772

5. Conclusão

Os resultados dos experimentos realizados mostraram que o classificador bayesiano implementado e descrito neste trabalho é capaz de sintetizar modelos preditivos com uma boa capacidade de generalização indutiva, a partir de exemplos. Para alguns conjuntos de dados testados, o classificador implementado chegou a apresentar acurácia e *g-measure* melhores do que aquelas obtidas com o WEKA [11], um sistema de mineração de dados bem conhecido na área. Por exemplo, para os conjuntos de dados *Heart* e *Tic-tac-toe*, as acurácias obtidas com o WEKA foram de 0.647 e 0.60, respectivamente [9].

Os resultados obtidos com o experimento sobre a influência da discretização de atributos contínuos também foram encorajadores e, como trabalho futuro, pretendemos implementar algoritmos de discretização automática de atributos contínuos e investigar mais profundamente este aspecto. Inicialmente, serão considerados os métodos de discretização existentes na literatura. Posteriormente, dependendo dos resultados obtidos, será investigada a possibilidade de criar um método de discretização que possa determinar automaticamente se vale a pena discretizar um ou mais atributos contínuos, a fim de aumentar a acurácia dos modelos preditivos sintetizados.

Como continuidade deste trabalho, será aprimorada a interface de usuário do sistema implementado, visando a apresentação de gráficos que mostrem, por exemplo, a distribuição dos exemplos nas classes dos conjuntos de dados e que exibam os resultados de desempenho obtidos.

Agradecimentos

Ao CNPq pela bolsa de Iniciação Científica[1] (Processo Nº 107647/2017-6).

Referências Bibliográficas

[1] A. Blum. **Machine Learn Theory**. Carnegie Mellon University, Department of Computer Science. Disponível em: www.cs.cmu.edu/afs/cs/user/avrim/www/Talks/mlt.pdf. Acesso em: 26/04/2017.

[2] I. H. Witten; E. Frank; M. A. Hall. **Data Mining: Practical Machine Learning Tools and Techniques**, *3rd edition,* Elsevier, 2011.

[3] D. Michie et al. **Machine Learning, Neural and Statistical Classification**, 1994. Disponível em: www1.maths.leeds.ac.uk/~charles/statlog/whole.pdf. Acesso em: 25/04/2017.

[4] E. Keogh. **Naïve Bayes Classifier**. University of California, Riverside. Disponível em: www.cs.ucr.edu/~eamonn/CE/Bayesian%20Classification%20withInsect_examples.pdf. Acesso em: 26/04/2017.

[5] D. Barber. **Bayesian Reasoning and Machine Learning**, *draft*, University College London, 2010. Disponível em: web4.cs.ucl.ac.uk/staff/D.Barber/textbook/090310.pdf. Acesso em: 26/04/2017.

[6] **Introduction to Probability Theory**, *Lecture 6: Bayes' Theorem*. PennState University. Disponível em: onlinecourses.science.psu.edu/stat414/node/45. Acesso em: 26/04/2017.

[7] V. L. Ceder. **The Quick Python Book**, 2nd edition, Manning Publications Co., USA, 2010.

[8] **ARFF** *(stable version)*. University of Waikato (WEKA). Disponível em: weka.wikispaces.com/ARFF+%28stable+ version%29. Acesso em: 26/04/2017.

[9] R. Morgon; S. L. Pereira. **Evolutionary Learning of Concepts**. Journal of Computer and Communications, vol. 2, pp. 76-86. Disponível em: http://www.scirp.org/journal/PaperInformation.aspx?PaperID=47412. Acesso em: 26/04/2017.

[10] *UCI* **Machine Learning Repository**. Disponível em: archive.ics.uci.edu/ml/datasets.html?sort=nameUp&view=list. Acesso em: 17/04/2017.

[11] **Weka 3**. University of Waikato. Disponível em: www.cs.waikato.ac.nz/ml/weka/. Acesso em: 26/04/2017.