PEL208 — Relatório Atividade 5
Implementação do *Naive Bayes*

Miller Horvath
Mestrando em Engenharia Elétrica (Processamento de Sinais e Imagens)
Centro Universitário FEI, São Bernardo do Campo, SP, Brasil

19 de novembro de 2018

1 Introdução

Este trabalho apresenta o relatório do desenvolvimento da quinta atividade avaliativa referente à disciplina PEL208, intitulada Tópicos Especiais em Aprendizagem, apresentada pelo Prof. Dr. Reinaldo Augusto da Costa Bianchi.

O objetivo desta atividade é implementar o algoritmo de classificação *Naive Bayes*, conforme abordado em sala de aula. Para isso, a linguagem de programação *Python* foi adotada.

2 Conceitos Fundamentais

2.1 Naive Bayes

O classificador *Naive Bayes* é baseado no teorema de Bayes, que defina a probabilidade de um determinado evento ocorrer com base em conhecimentos *a priori* relacionados ao mesmo (KENDALL, 1943).

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{1}$$

A Equação 1 mostra o teorema de Bayes, sendo $P(A|B)$ a probabilidade condicional do evento A ocorrer dado que o evento B ocorreu, similarmente, $P(B|A)$ é a probabilidade condicional do evento B ocorrer dado que o evento A ocorreu e $P(A)$ e $P(B)$ são as probabilidades dos eventos A e B ocorrerem.

Sendo assim, o método *Naive Bayes* assume que os atributos são todos linearmente independentes entre si, de modo a simplificar os cálculos de probabilidades quando há um encadeamento de eventos a serem considerados (KIRK, 2014). Desta forma, a probabilidade

de um evento A ocorrer, dado que um conjunto X de eventos ocorreram é definida pela seguinte equação:

$$P(C|X) = P(C) \prod_{x \in X} P(x|C) \tag{2}$$

Por fim, para classificação de dados, calcula-se a probabilidade de uma observação fazer parte de todas as possíveis classes $c \in C$, sendo que a classe com maior probabilidade é escolhida. Este método é conhecido como Probabilidade Máxima *a Posteriori* (MAP), dado pela seguinte equação:

$$\hat{y} = \underset{c \in C}{\operatorname{argmax}} P(c) \prod_{x \in X} P(x|c) \tag{3}$$

A Tabela 1 apresenta uma relação de pessoas e a marca dos dispositivos celulares e computadores que elas possuem.

Nome	Computador	Celular
Kate	PC	Android
Tom	PC	Android
Harry	PC	Android
Annika	Mac	iPhone
Naomi	Mac	Android
Joe	Mac	iPhone
Chakotay	Mac	iPhone
Neelix	Mac	Android
Kes	PC	iPhone
B'Elanna	Mac	iPhone

Tabela 1 – Relação das marcas dos computadores e celulares de um grupo de pessoa.

O classificador *Nayve Bayes* pode ser aplicado neste conjunto de dados para tentar prever, dado a marca do computador de um indivíduo, qual a marca do seu celular. Neste exemplo, aplicando o *Nayve Bayes*, caso o indivíduo possua um computador "Mac", a probabilidade dele possuir um celular "Android" é: $P(Android|Mac) = 0.2$; e a probabilidade dele possuir um celular "iPhone" é: $P(iPhone|Mac) = 0.4$. Portanto, se um indivíduo possui um computador "Mac", a tendencia é que o seu aparelho celular seja um "iPhone".

3 Trabalhos Relacionados

Em (ZHOU et al., 2015), foi proposta uma metodologia para diagnostico automático de doenças cerebrais, através da análise de imagens obtidas através de ressonância magnética nuclear, utilizando *Wavelets* e o classificados *Naive Bayes*.

Em (JIANG et al., 2016), foi apresentada uma metodologia de ponderamento de atributos para estimar as probabilidades condicionais do modelo *Naive Bayes*.

Em (PHAM et al., 2016), foi avaliada a capacidade preditiva dos modelos de Máquina de Vetores de Suporte e *Naive Bayes* ao prever risco de desmoronamentos em um estado da Índia.

Em (TANG; KAY; HE, 2016), foi proposta uma metodologia de seleção de atributos para otimizar o modelo *Nayve Bayes* para classificação de textos.

Em (NG; XING; TSUI, 2014), o modelo *Nayve Bayes* foi aplicado no contexto de gestão de economia de bateria, utilizado para prever a vida útil remanescente de baterias de íon-lítio de operação constante sob diferentes condições de temperatura ambiente.

4 Metodologia

A implementação da atividade foi desenvolvida na linguagem *Python*. Para apoiar o desenvolvimento da atividade, foram utilizadas as bibliotecas *pandas*, para manipulação dos dados através da estrutura de dados chamada *DataFrame*, e *numpy*, para resolução de cálculos básicos.

Foi desenvolvida uma classe chamada *NaiveBayes*, que recebe um *DataFrame* X e uma *string* s como parâmetros de sua função construtora. O parâmetro X armazena o conjunto de observações, definidos por uma série de atributos, utilizados para calcular as probabilidades condicionadas do modelo. O parâmetro s é utilizado para indicar qual dos atributos de X é o atributo alvo, ou seja, a atributo que representa a classe das observações. A classe *NaiveBayes* armazena os seguintes atributos:

- O *DataFrame* X;
- O atributo alvo de X (deve ser um atributo categórico);
- O conjunto de probabilidades independentes de cada um dos possíveis eventos pertencentes aos atributos categóricos de X;
- O conjunto de probabilidades condicionadas dos eventos pertencentes aos atributos categóricos de X (condicionados pelo atributo alvo).
- A média dos atributos contínuos agrupados pelo atributo alvo; e
- O desvio padrão dos atributos contínuos agrupados pelo atributo alvo.

Além disso, a classe também oferece os seguintes métodos:

- *compute_probabilities* — Calcula as probabilidades independentes e condicionadas;
- *print_probabilities* — Escreve na saída padrão as probabilidades independentes e condicionadas previamente calculadas;

- *fit* — Permite alterar os atributos X e/ou s e recalcula as probabilidades independentes e condicionadas.
- *gauss_probability* — Método estático utilizado para calcular a probabilidade condicional para atributos contínuos utilizando uma distribuição Gaussiana.
- *predict_probabilities* — Recebe uma lista de vetores de atributos como parâmetro e retorna a probabilidade de cada vetor ser classificado como sendo pertencente a cada uma das possíveis classes (definidas pelo atributo alvo de X);
- *predict* — Recebe uma lista de vetores de atributos como parâmetro e retorna a classificação de cada vetor em relação as classes definidas pelo atributo alvo de X.

5 Experimentos

Para testar e analisar a implementação desenvolvida, foram utilizadas três bases de dados de classificação, pertencentes ao repositório de aprendizado de máquina da University of California, Irvine (UCI) (DHEERU; TANISKIDOU, 2017), que serão descritas na sequência. Foi utilizado o classificador *Naive Bayes*, descrito na Seção 2.1, para classificar os dados as observações das bases de dados adotadas.

Para avaliar o desempenho do *Naive Bayes*, foi calculada a matriz de confusão para avaliar a taxa de verdadeiros positivos, falsos positivos, verdadeiros negativos e falsos negativos para cada uma das classes.

5.1 Bases de Dados

5.1.1 Iris Data Set

Está é uma das bases de dados mais tradicionais em classificação. Esta base possuí informações sobre três diferentes classes de flores, tendo um total de 150 observações, sendo 50 de cada classe, e 5 atributos. Mais informações sobre esta base de dados podem ser encontradas em (FISHER, 1936).

5.1.2 UCI Wine Data Set

Esta base possuí informações sobre análises químicas de vinhos de três cultivares diferentes, todos de uma mesma região da Itália, possuindo um total de 178 observações e 13 atributos. Mais informações sobre esta base de dados podem ser encontradas em (FORINA, 1991).

5.1.3 UCI Seeds Data Set

Esta base possuí informações sobre três tipos diferentes de trigo, possuindo um total de 210 observações, sendo 70 de cada classe, e 7 atributos. Mais informações sobre esta base de dados podem ser encontradas em (CHARYTANOWICZ et al., 2010).

6 Resultados

Nesta seção, são apresentados os resultados obtidos através do desenvolvimento experimental descrito na Seção 5.

6.1 Iris Data Set

A Tabela 2 apresenta a matriz de confusão resultante da aplicação do algoritmo *Naive Bayes*, descrito na Seção 2.1, na base de dados Iris Data Set, apresentada na Seção 5.1.1, seguindo a metodologia descrita na Seção 4.

		Prevista		
		Iris-setosa	Iris-versicolor	Iris-virginica
Observada	Iris-setosa	50	0	0
	Iris-versicolor	0	47	3
	Iris-virginica	0	3	47

Tabela 2 – Matriz de confusão resultante do Naive Bayes aplicado a base de dados Iris Data Set, descrita na Seção 5.1.1.

A classe *Iris-setosa* é bem discriminada em relação às demais, pois todas as observações foram corretamente classificadas e nenhuma observação de outra classe foi mal classificada como *Iris-setosa*. Em contrapartida, algumas observações das classes *Iris-versicolor* e *Iris-virginica* se confundem no *Naive Bayes*, pois 94% das observações foram corretamente classificadas para ambas as classes. No total, o *Naive Bayes* classificou 96% das observações corretamente.

6.2 Wine Data Set

A Tabela 3 apresenta a matriz de confusão resultante da aplicação do algoritmo *Naive Bayes*, descrito na Seção 2.1, na base de dados Wine Data Set, apresentada na Seção 5.1.2, seguindo a metodologia descrita na Seção 4.

		Prevista		
		Classe 1	Classe 2	Classe 3
Observada	Classe 1	56	3	0
	Classe 2	2	68	1
	Classe 3	0	0	48

Tabela 3 – Matriz de confusão resultante do Naive Bayes aplicado a base de dados Wine Data Set, descrita na Seção 5.1.2.

Nesta base de dados, 96.6% das observações foram corretamente classificados pelo *Naive Bayes*. Para a classe 1, 94.9% (56 de 59) das observações foram classificadas corretamente, sendo que todas observações restantes foram mal-classificadas como pertencentes a classe 2. Já para a classe 2, 95.8% (68 de 71) foram classificados corretamente, sendo que

duas observações (2.8%) foram confundidas com a classe 1 e uma observação (1.4%) foi confundida com a classe 3. Por fim, para a classe 3, todas as quarenta e oito observações foram classificadas corretamente.

6.3 Seeds Data Set

A Tabela 4 apresenta a matriz de confusão resultante da aplicação do algoritmo *Naive Bayes*, descrito na Seção 2.1, na base de dados Seeds Data Set, apresentada na Seção 5.1.3, seguindo a metodologia descrita na Seção 4.

		Prevista		
		Classe 1	Classe 2	Classe 3
Observada	Classe 1	59	3	8
	Classe 2	5	65	0
	Classe 3	3	0	67

Tabela 4 – Matriz de confusão resultante do Naive Bayes aplicado a base de dados Seeds Data Set, descrita na Seção 5.1.3.

Para a classe 1, 84.3% (59 de 70) das observações foram corretamente classificadas, sendo que 4.3% (3 de 70) foram confundidos com a classe 2 e 11.4% foram confundidos com a classe 3. 92.9% (65 de 70) das observações pertencentes à classe 2 foram corretamente classificadas, sendo que as demais foram confundidas com a classe 1. Por fim, 95.7% (67 de 70) observações pertencentes a classe 3 foram corretamente classificadas, as demais foram confundidas com a classe 1. No total, 91% (187 de 210) das observações foram corretamente classificadas.

7 Conclusão

Este trabalho visa relatar a implementação do algoritmo *Naive Bayes*, utilizando *Python* como linguagem de programação, como tarefa do curso PEL208 do programa de pós-graduação em engenharia elétrica do Centro Universitário FEI. O *Naive Bayes* é um dos algoritmos de clusterização mais populares devido a sua simplicidade e facilidade de implementação.

O fato dos resultados, apresentados na Seção 6, serem condizentes com os apresentados em aula sugere que a implementação foi adequada.

Ademais, o classificador *Nayve Bayes* apresentou ótimos resultados com as bases de dados adotadas, pois mais de 90% de assertividade foi obtida para todas as bases, sendo que para as bases de dados Iris 5.1.1 e Wine 5.1.2, foi alcançada uma assertividade em torno de 96%.

Referências

CHARYTANOWICZ, M. et al. *seeds Data Set*. 2010. Acessado em: 2018-11-17. Disponível em: <https://archive.ics.uci.edu/ml/datasets/seeds>. Citado na página 4.

DHEERU, D.; TANISKIDOU, E. K. *UCI Machine Learning Repository*. 2017. Disponível em: <http://archive.ics.uci.edu/ml>. Citado na página 4.

FISHER, R. *Iris Data Set*. 1936. Acessado em: 2018-11-17. Disponível em: <https://archive.ics.uci.edu/ml/datasets/Iris>. Citado na página 4.

FORINA, M. e. a. *Wine Data Set*. 1991. Acessado em: 2018-11-17. Disponível em: <https://archive.ics.uci.edu/ml/datasets/Wine>. Citado na página 4.

JIANG, L. et al. Deep feature weighting for naive bayes and its application to text classification. *Engineering Applications of Artificial Intelligence*, Elsevier, v. 52, p. 26–39, 2016. Citado na página 2.

KENDALL, M. G. *Advanced Theory Of Statistics Vol-I*. [S.l.]: Charles Griffin: London, 1943. Citado na página 1.

KIRK, M. *Thoughtful machine learning: A test-driven approach*. [S.l.]: "O'Reilly Media, Inc.", 2014. Citado na página 1.

NG, S. S.; XING, Y.; TSUI, K. L. A naive bayes model for robust remaining useful life prediction of lithium-ion battery. *Applied Energy*, Elsevier, v. 118, p. 114–123, 2014. Citado na página 3.

PHAM, B. T. et al. Evaluation of predictive ability of support vector machines and naive bayes trees methods for spatial prediction of landslides in uttarakhand state (india) using gis. *J Geomatics*, v. 10, p. 71–79, 2016. Citado na página 3.

TANG, B.; KAY, S.; HE, H. Toward optimal feature selection in naive bayes for text categorization. *arXiv preprint arXiv:1602.02850*, 2016. Citado na página 3.

ZHOU, X. et al. Detection of pathological brain in mri scanning based on wavelet-entropy and naive bayes classifier. In: SPRINGER. *International conference on bioinformatics and biomedical engineering*. [S.l.], 2015. p. 201–209. Citado na página 2.