Naïve Bayes e *Support Vector Machines*: Uma Análise Comparativa

Gustavo Zanoni Felipe, Mariana Soder

Introdução

- Este trabalho visou realizar uma análise comparativa entre dois algoritmos classificadores, sendo estes:
  - *Naïve Bayes*, em uma versão aqui implementada
  - *Support Vector Machines* (SVM), em uma versão da literatura/biblioteca *sckit-learn*
- Para que isto fosse realizado
  - três bases de dados retiradas do *UCI Machine Learning Repository* (Dheeru and Karra Taniskidou 2017) foram utilizadas em um esquema de classificação
  - realizando *cross-validation* utilizando-se 10 *folds*
  - ao final, os resultados foram analisados tendo em base as matrizes de confusão geradas e utilizando métricas de análise de classificadores
    - *Accuracy*
    - *Recall*
    - *Precision*
    - *F1-Score / F-score / F-measure*

Fundamentação Teórica

1. *Classificação*
   a. *Naïve Bayes*
   b. *Support Vector Machines (SVM)*
2. *Métricas de Avaliação*

1. Classificação

- Parte do aprendizado supervisionado
  - dado um conjunto de treino de ***n*** exemplos de pares de entrada/saída

$$(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$$

  - descobrir uma hipótese (função *h*) que se aproxima de uma função verdadeira *f* que gera cada um dos valores y_i, i. e. **y = f(x)**.
  - Para avaliar o desempenho da hipótese, é dado um conjunto de teste de exemplos distintos daqueles utilizados no conjunto de treino.

- A classificação se caracteriza como a qual possibilidades de valores de uma saída *y* estejam presentes em um conjunto finito de valores

a. *Naïve Bayes*

- Este trata cada atributo de uma amostra como sendo independente dos demais.
- Com base na frequência dos valores dos atributos em relação às classes apresentadas, é possível realizar predições de forma eficiente e rápida.

$$pred = \arg\max P(c_j) \prod_{i=1}^{n} P(X_i = x_i | c_j)$$

b. *Support Vector Machines (SVM)*

- Divide os padrões de amostras presentes em uma base de dados por meio de uma reta chamada de hiperplano.
- Em problemas onde os padrões possuem difícil divisão em seu espaço dimensional original, é possível utilizar do *Kernel Trick*
  - uma dimensão é adicionada ao problema. Assim, facilitando a divisão de tais padrões.

2. *Métricas de Avaliação*

- ***Accuracy***: define no geral, o quão frequente o classificador está correto
- ***Precision***: dentre as predições corretas, quantas efetivamente eram de tal natureza
- ***Recall*:** calcula a frequência em que o classificador encontra os exemplos de uma determinada classe
- **F1-Score:** indica a qualidade geral do sistema de classificação desenvolvido, utilizando da combinação da precisão e recall.

Materiais e Métodos

1. Bases de Dados
    a. Car Evaluation Database
    b. Mushroom Database
    c. Nursery Database
2. Metodologia Abordada

1. Base de dados

- As bases de dados utilizada neste trabalho são algumas das várias bases de dados presentes no repositório UCI (Dheeru and Karra Taniskidou 2017);
- Três bases de dado foram utilizadas, sendo elas: ***Car Evaluation Database***, ***Mushroom Database*** e ***Nursery Database***.

a. Car Evaluation Database

- Esta base de dados possui como principal objetivo, realizar a avaliação de carros.
- São dados seis atributos por amostra, sendo eles: valor de compra, valor de manutenção, número de portas, número de lugares, tamanho do porta-malas e nível de segurança.
- A partir destes, deve-se avaliar a qual classe o carro pertence, sendo as possibilidades: inaceitável (unacc), aceitável(acc), bom (good) e muito bom (v-good).

a. *Car Evaluation Database*

Classe	# de Amostras	Proporção
unacc	1210	0.70023
acc	384	0.22222
good	69	0.03993
v-good	65	0.03762

Tabela 1. Quantidade de amostras por classe da base de dados ”Car Evaluation Database”.

b. Mushroom Database

- Nesta base as amostras representam cogumelos pertencentes às famílias *Agaricus* e *Lepiota*.
- Cada amostra possui 22 atributos que representam características de um cogumelo como: a cor do chapéu, odor, tipo do véu, número do anel, população, habitat e etc.
- Ao final, deve-se decidir entre uma das duas amostras presentes, sendo elas: comestível(edible) ou venenosa (poisonous).

b. Mushroom Database

Classe	# de Amostras	Proporção
edible	4208	0.518
poisonous	3916	0.482

Tabela 2. Quantidade de amostras por classe da base de dados ”Mushroom Database”.

c. Nursery Database

- O objetivo desta base de dados é de classificar aplicações para escolas de enfermagem.
- Dado um conjunto de atributos (que informam a condição social, financeira, de saúde, formação e etc.) de um determinado aplicante, é retornado se o mesmo é não-recomendado (notrecom), recomendado (recommend), muito recomendado (veryrecom), prioridade (priority) e prioridade especial (specprior) à entrar na instituição de ensino de enfermagem.

c. Nursery Database

Classe	# de Amostras	Proporção
not_recom	4320	0.33333
recommend	2	0.00015
very_recom	328	0.02531
priority	4266	0.32917
spec_prior	4044	0.31204

Tabela 3. Quantidade de amostras por classe da base de dados "Nursery Database".

Visão Geral

Base de Dados	# de Amostras	# de Atributos	# de Classes
Cars	1728	6	4
Mushrooms	8124	22	2
Nursery	12958*	8	4*

Tabela 4. Quantidade de amostras, atributos e classes apresentadas para cada uma das bases de dados utilizadas neste trabalho.

2. Metodologia Abordada

Preparação

Implementar o algoritmo de *Naïve Bayes*

Montar um classificador SVM utilizando da biblioteca *scikit learn*

Classificação

Dividir as três bases de dados em 10 *folds cada*

Classificar as três bases de dados utilizando de ambos classificadores

Avaliação

Montar as matrizes de confusão

Calcular:

- *accuracy*
- *precision*
- *recall*
- *f1-score*.

Resultados Encontrados

Resultados Encontrados

Base de Dados	*precision*	*recall*	*f1-score*	*Accuracy*
Cars	0.8152	0.8194	0.8164	0.8194
Mushrooms	0.9866	0.9866	0.9866	0.9865
Nursery	0.8404	0.8430	0.8362	0.8429

Tabela 5. Valores da análise das classificações realizadas, para as diferentes bases, utilizando o algoritmo *Support Vector Machines*. Os valores aqui apresentados são uma média ponderada dentre as classes de cada uma das bases.

Resultados Encontrados

Base de Dados	*precision*	*recall*	*f1-score*	*Accuracy*
Cars	0.8691	0.8715	0.8660	0.8715
Mushrooms	1.0000	1.0000	1.0000	1.0000
Nursery	0.9083	0.9033	0.8945	0.9033

Tabela 6. Valores da análise das classificações realizadas, para as diferentes bases, utilizando o algoritmo *Naïve Bayes*. Os valores aqui apresentados são uma média ponderada dentre as classes de cada uma das bases.

Conclusão

Conclusão

Para o contexto trabalhado e para os problemas abordadas, o algoritmo de classificação **Naïve Bayes** obteve um melhor desempenho.

Quando observadas as acurácias encontradas, a implementação aqui desenvolvida de Naïve Bayes apresentou em média 4% de acertos a mais que o algoritmo do SVM. Onde no maior caso a diferença encontrada foi de 6,04% utilizando-se da base de dados *Nursery* e a menor diferença foi de 1,35% para a base *Mushrooms*.

Destaca-se que o melhor valor de acurácia encontrado, foi de **100%** utilizando-se de Naïve Bayes com a base de dados *Mushrooms*.

Referências

Castillo, G. (2011). Bayesian network classifiers. University Lecture. Disponível em: https://goo.gl/L25Qvk [Online; Acessado em 27 de Setembro de 2018].

Dheeru, D. and Karra Taniskidou, E. (2017). UCI machine learning repository.

Norvig, S. R. P. (2014).Inteligência Artificial. Campus, 3rd edition