

Estudo Comparativo de Desempenho Entre SVM e MLP no Reconhecimento de Imagens

Arthur Vitor Matias de Oliveira¹, Maykon Moraes da Silva¹, Angélica Félix de Castro¹

¹ Departamento de Ciência da computação – Universidade Federal Rural do Semi-Árido (UFERSA)

Caixa Postal 137 – 59.625-900 – Mossoró – RN – Brazil

{arthuroliveira909.av, maykon.estudos}@gmail.com,
angelica@ufersa.edu.br

***Abstract.** This article complains the results of comparative study of two supervised predictive models: Support Vector Machine (SVM) and Multilayer Perceptron (MLP) that were once used for the classification of images of different magnitudes in order to observe the implications of variations in quantity and dimensions of the data under the prediction models. Respectively, for each variation, an experiment was carried out using the jupyter notebook platform and coded in python. The results point to a better MLP efficiency for large amounts of data based on the training time and prediction time, however for almost all tests the SVM obtained a better accuracy having only higher training and prediction times than MLP with small amounts of data.*

***Resumo.** Este artigo contempla os resultados de um estudo comparativo de dois modelos preditivos supervisionados: Support Vector Machine (SVM) e Multilayer Perceptron (MLP) que foram outrora utilizados para a classificação de imagens de diferentes magnitudes tendo como objetivo observar as implicações das variações de quantidade e dimensões dos dados sob os modelos de predição. Respectivamente, para cada variação foi realizado um experimento utilizando a plataforma jupyter notebook e codificado em python. Os resultados apontam para uma melhor eficiência do MLP para grandes quantidades de dados tomando como base o tempo de treinamento e tempo de predição, entretanto para quase todos os testes o SVM obteve uma melhor acurácia tendo apenas tempos de treino e predição superiores ao MLP com pequenas quantidades de dados.*

1. Introdução

Os sistemas inteligentes se mostram presentes desde pequenos comércios a grandes empresas de tecnologia. A inferência sob base de dados se tornou algo cotidiano e de tendência de mercado, seja de uma perspectiva estatística ou comercial, desta forma a presença e o peso dos sistemas inteligentes são visíveis, pois a capacidade de reconhecer padrões invisíveis para os seres humanos tornam as Redes Neurais Artificiais (RNA) ferramentas extremamente poderosas.

O poder computacional de uma RNA é determinado pelos cálculos matemáticos utilizados para determinar os pesos internos bem como a função de limiar de ativação

[Silva, Ivan Nunes, 2010], isso significa que a rede que possui o melhor treino de pesos sinápticos e a melhor estratégia para determinar o limiar de ativação obterá os melhores resultados de acurácia, entretanto como pode ser visto em [Bonifácio, FERNANDO NUNES, 2010], a base de dados utilizada na fase de inicial afeta diretamente o tempo de treinamento, tempo de predição e acurácia, tornando notório a relação da base de dados com a eficiência dos modelos preditivos.

Uma base de dados composta por imagens vetoriais é compreendida como um conjunto de registros que armazenam vetores de dimensão “ $n \times m$ ” considerando imagens em escala de cinza que possui valores de pixel que varia de 0 a 255. Deste modo, uma imagem 7×7 por exemplo seria disposta nessa base como sendo um registro de 49 variáveis assim como realizado no trabalho de [Horewicz, M. C., et. al]. O real problema em trabalhar com bases de dados desta forma é que a complexidade desses dados são extremamente altas e tem alto custo computacional, podendo demorar horas ou dias para treinar modelos como pode ser visto em [Bonifácio, FERNANDO NUNES, 2010]. Deste modo, quando agregados se faz necessário ter uma ideia de qual preditor utilizar para se obter uma inferência no menor tempo possível e com uma acurácia que supra as necessidades dos problemas, e que mesmo que haja variações nos aspectos dos dados a implicação sob a perda de eficiência dos modelos preditos não seja drástica.

Desta forma, a fim de efetivamente avaliar o desempenho de duas redes que por mais que similares possuem aspectos completamente diferentes, o presente trabalho tem o intuito de determinar uma relação de eficiência em tempo de treinamento, tempo de predição e acurácia, para dois modelos de predição: *Support Vector Machine* (SVM) e *Multilayer Perceptron* (MLP), a partir de dados vetoriais de imagens que terão suas quantidades variadas.

2. Trabalhos Relacionados

A comparação entre os classificadores SVM e MLP na identificação de padrão de imagens pode ser vista em vários trabalhos das mais diversas áreas de atuação, por exemplo [Horewicz, M. C., et. al] utiliza dessas redes para prever caracteres manuscritos dos dígitos de (A - F) e (1 - 9) alterando a base de dados adicionando um ruído do tipo flip-flop, a comparação deste, no entanto, não compara os aspectos como tempo de predição e tempo de treino se limitando a porcentagem de acerto dos modelos.

Outro trabalho que também aborda a utilização dessas redes é o executado por [Marana, A. N., et. al, 2010] que neste aspecto também utiliza as redes para a predição de manuscritos, contudo diferentemente do anterior apresenta como resultado também o tempo de treino e de teste.

Entretanto, o trabalho realizado por [Bonifácio, FERNANDO NUNES, 2010] seja o que mais se relaciona às ideias do presente trabalho, por mais que este não compare diretamente o MLP com o SVM, ele traz um pouco da ideia de aplicar a bases de dados com as mais diferentes características variando nos aspectos de magnitude de dados e dimensão.

3. Fundamentação teórica

3.1. Aspectos sobre SVM e RNA

Encontrar padrões em meio a tantos dados pode se tornar uma tarefa árdua. A utilização de agentes classificadores é o caminho mais correto atualmente para realizar tal tarefa. Basicamente, os agentes classificadores são métodos computacionais (algoritmos) que têm como objetivo prever a classe de um determinado conjunto de dados pré-

estabelecidos [Serasiya, S. D., Chaudhary N, 2012]. Resumidamente, tenta-se extrair determinados padrões fornecidos pelo nosso conjunto de dados da amostra.

Do ponto de vista de análise, pode-se focar em dois classificadores bastante utilizados: SVM e MLP. Tendo em vista sua alta utilização como também a eficiência envolvida no reconhecimento de imagens, como por exemplo, utilizada no reconhecimento de caracteres em [Horewicz, et. al].

3.1.1 SVM

As Máquinas de Vetores de Suporte (*Support Vector Machines*) têm ganhado bastante atenção na área de aprendizado de máquina. Esse modelo foi criado pelo matemático Vladimir Vapnik, cujo o objetivo é classificar um conjunto de pontos que são mapeados para um espaço de característica multidimensional usando uma função kernel, abordagem utilizada para classificar problemas. Nela, o limite de decisão no espaço de entrada é representado por um hiperplano em dimensão superior no espaço [Ana Carolina Lorena, et. al].

O SVM, em suma, realiza uma divisão de um conjunto de objetos com diferentes classes, diretamente usando do conceito de planos de decisão. A separação de objetos pode ser feita traçando uma linha em que melhor se aproxima da classificação dos objetos.

Diante da complexidade envolvida em separar objetos originais fielmente. O SVM para mapeá-los utiliza de um conjunto de funções matemáticas, conhecidas como *kernels*. Aplicado às funções matemáticas, nos aproximamos dos níveis de aceitação para a organização dos objetos. Ou seja, chegamos a uma separação mais simples para a classificação.

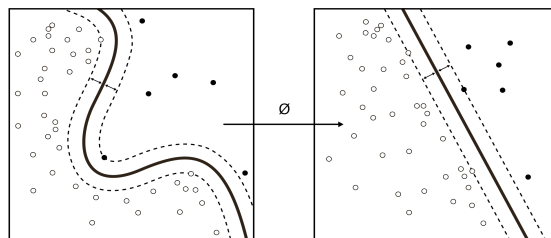


Figura 1. Aplicação de função matemática para separação de objetos

Fonte: Alisneaky, Wikimedia

3.1.2 RNA

Rede Neural Artificial pode ser conceituada como sendo uma estrutura de processamento que é passível de implementação em dispositivos eletrônicos, da qual é composta de unidades interconectadas (neurônios artificiais), tendo cada unidade um comportamento de entrada e de saída, que é determinado pela função de transferência, pelas conexões com outras unidades como também pelas entradas externas [DE CASTRO, et al].

Uma RNA exhibe, segundo [DE CASTRO, et al], algumas características bastante importantes e que servem de complemento para sua definição:

- I. Capacidade de operação em paralelismo.
- II. Não há necessidade de conhecer os modelos matemáticos que descrevem a maneira como uma aplicação funciona.
- III. Sistemas inspirados no cérebro humano.
- IV. Habilidade de aprender conforme a experiência.

Basicamente, os neurônios estão conectando entre si formando grandes redes, pelas quais a informação é passada através das conexões existentes. Daí, temos que a eficiência atrelada a essa conexão é expressa por meio de um peso que corresponde a informação armazenada em seu neurônio.

Todo o conhecimento adquirido é resultado de um processo de aprendizagem, que é basicamente responsável por adaptar os pesos das conexões aos estímulos recebidos do ambiente.

O comportamento da rede neural é influenciada totalmente pelo conhecimento adquirido. Dessa forma, todo processamento da rede (entrada e saída) está atrelado ao conhecimento ao qual está em nossa rede.

3.1.2.1 MLP

O Perceptron Multicamadas é uma rede neural com alta similaridade ao *perceptron*, da qual utiliza-se de um algoritmo de retropropagação do erro [DE CASTRO, F., DE CASTRO, M]. A diferença essencial é que se tem várias camadas de neurônios. Em situações em que a classificação não se dá por meio de uma única reta, a utilização do MLP é apropriada, pois há geração de mais de uma reta classificadora.

A propagação dos sinais de entrada de nossa rede é feita em um único sentido, indo da camada de entrada até uma camada de neural de saída. Tendo esse comportamento, os estímulos ou sinais são apresentados em uma camada de entrada. As camadas intermediárias são responsáveis por extrair a maioria das informações referentes ao comportamento e as codificam por meio dos pesos sinápticos e limiares dos seus neurônios, formando assim uma representação própria do ambiente em que está inserido. Por fim, os neurônios da camada de saída recebem os estímulos da última camada de intermediária, produzindo um padrão de resposta que será disponibilizada pela rede [Silva, Ivan Nunes, 2010].

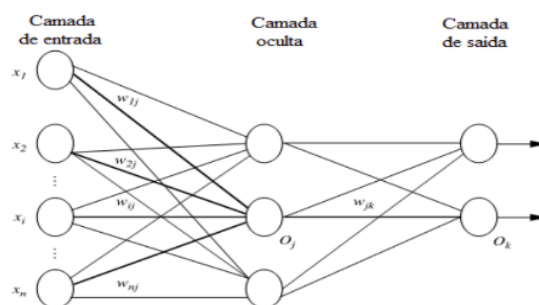


Figura 2. Perceptron Multicamadas

Fonte : HAN, J., KAMBER, M. Data Mining: Concepts and Techniques.

4 . Metodologia

No estudo de caso foi analisado o desempenho dos classificadores SVM e o RNA, o código foi escrito inteiramente em python, utilizando as bibliotecas do pacote Anaconda, como o *Jupyter Notebook*, *Pandas* e *Matplotlib*. Os classificadores utilizados no presente trabalho foram obtidos a partir da biblioteca *sklearn* e foram treinadas com dados obtidos a partir do site *Kaggle*. As bases de dados eram imagens em formato vetorial que tinham como conteúdo caracteres manuscritos representados por 748 variáveis por registro e imagens de células representadas por 8192 variáveis por registro. Respectivamente, ambas as bases estavam em escala de cinza, sendo representadas por variáveis com variação de 0 a 255 para cada pixel da imagem. O

treinamento dos classificadores foi realizada de acordo com o processo de treinamento indicado por [Silva, Ivan Nunes, 2010], onde 30% dos dados seriam destinados ao grupo de teste e 70% dos dados seriam destinados ao conjunto de treino, essa repartição foi feita de forma aleatória.

A relação dos dados implicou dois testes para definir como a alteração da dimensão dos dados e da quantidade de dados inferia no desempenho dos classificadores. No primeiro experimento foi visto o comportamento dos classificadores com relação ao aumento da dimensão dos dados e o segundo experimento foi avaliado o comportamento dos classificadores com o aumento do conjunto de registros.

4.1 Experimento Número 1

Para o treinamento dos classificadores do primeiro experimento foi utilizado as respectivas bases descritas acima com uma quantidade de dados de 2300 registros para cada. Portanto, as bases representaram matrizes de magnitude de 748 x 2300 e 8192 x 2300 respectivamente. Os classificadores foram então treinados com 70% do total dos dados e tiveram como teste os outros 30% dos dados assim como descrito anteriormente.

Tabela 1. de comparação entre os dois classificadores quando se varia a dimensão dos dados. Fonte: Própria do autor

Comparação de desempenho relacionado a dimensão dos dados						
Classificador	Dimensão dos dados					
	2300 x 784			2300 x 8192		
	Tempo de Treino	Tempo de predição	% de acerto	Tempo de Treino	Tempo de predição	% de acerto
MLP	2.385s	0.010s	84,19%	90.259 s	0.078s	91,56%
SVM	1.430s	0.631s	93,39%	14.162 s	3.008s	92,81%

A tabela acima contém os resultados do experimento e a dela é possível inferir as seguintes conclusões. O MLP tem um desempenho de tempo de treino extremamente inferior ao tempo de treinamento do SVM, para o conjunto de maior dimensão o MLP é 6 vezes mais lento que SVM e para o de menor dimensão ele é 1.6 vezes mais lento. É visto também que em termo de porcentagem de acerto, o SVM se saiu melhor que MLP, na primeira base de dados a diferença é razoavelmente grande, pois o SVM acerta 9.2 % a mais que MLP, isso significa que no primeiro conjunto o SVM acertou 43 linhas a mais que MLP. Em termos de porcentagem de acerto e treinamento, o SVM é melhor que MLP, contudo, o tempo de predição é um grande problema. A variação do tempo de predição dos dois conjuntos no classificador SVM indica que, como mostrado a seguir, conforme a dimensão dos dados forem aumentando o tempo de predição também aumentará, do contrário o MLP basicamente se mantém estável como pode ser visto no gráfico abaixo.

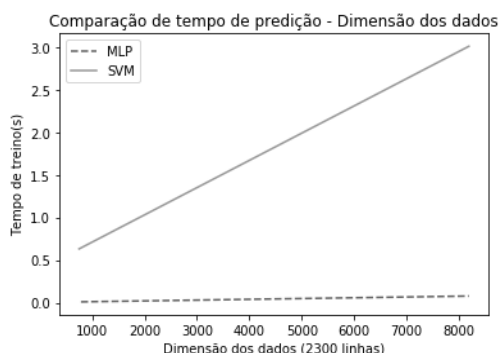


Figura 3. Comparação do tempo que os algoritmos levaram para predizer um conjunto de dados de tamanho igual.

Fonte: Própria do autor

No trabalho realizado por [Marana, A. N., et. al, 2010] também é possível ver essa relação do tempo de desempenho desproporcional do SVM no caso no o MLP nos testes realizados representou um desempenho superior ao SVM-RBF, com uma superioridade de pelo menos 5 vezes mais rápido.

4.2 Experimento Número 2

Para o treinamento da segunda implementação foi utilizado somente a primeira base de dados de 748 x 42000, no entanto foram determinados valores diferentes para a quantidade de registros de cada teste, sendo que para cada mudança de tamanho era necessário retreinar os modelos preditivos. Após todos os treinamentos e predições, o resultado obtido foi o seguinte:

Tabela 2. de comparação entre os dois classificadores quando se varia a quantidade dos dados.

Fonte: Própria do autor

Comparação de desempenho relacionado a quantidade de dados						
Quantidade de dados	Classificador					
	MLP			SVM		
	Tempo de Treino	Tempo de predição	% de acerto	Tempo de Treino	Tempo de predição	% de acerto
2300 x 784	2.385s	0.010s	84,19%	1.430s	0.631s	93,39%
5000 x 784	6.665s	0.026s	87,60%	4.643s	2.347s	94,26%
6000 x 784	6.144s	0.033s	87,38%	6.343s	3.153s	95,11%
7000 x 784	6.667s	0.040s	88,88%	7.925s	4.074s	95,04%
10000 x 784	9.264s	0.059s	90,80%	13.711s	7.352s	96,03%
20000 x 784	29.093s	0.073s	94,26%	38.506s	22.884s	96,55%
40000 x 784	53.1481s	0.186s	95,07%	111.835s	73.170s	97,36%

Diferentemente do que foi notado na primeira implementação, o MLP teve um desempenho melhor em tempo de treino e em tempo de predição, o que comparado com primeira implementação mostra o quão superior a superioridade do MLP sobre o SVM. Na verdade, o SVM representa um crescimento exponencial em tempo de predição enquanto o MLP se mantém em um crescimento pequeno e constante. Pode-se inferir, então, que independente do tamanho da base de dados o preditor do MLP mantém um tempo de predição constante, por mais que o MLP seja menos eficiente em porcentagem de acerto, como pode ser na tabela acima.

Outro ponto que se pode observar é que o tempo de treinamento tem um ponto

de intersecção em torno de 6000 linhas de dados, o que acontece é que após esse ponto o crescimento do tempo de treino do SVM crescem exponencialmente e com uma magnitude de crescimento superior ao do MLP, como pode ser visto abaixo:

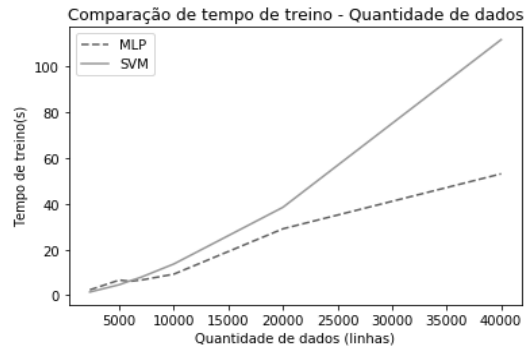


Figura 5. Comparação do tempo que os algoritmos levaram para o treinamento com 80% de um conjunto de dados de tamanho variável.

Fonte: Própria do autor

Por fim, o único parâmetro que o SVM se sai melhor que o MLP é em sua precisão, pois ele funciona muito bem com conjuntos de dados pequenos com várias dimensões e, por isso, diferentemente do MLP, já inicia com uma precisão muito boa. Não obstante, quando se aumenta a quantidade dos dados, a diferença da porcentagem de acerto dos classificadores tende a diminuir, como é visto a seguir:

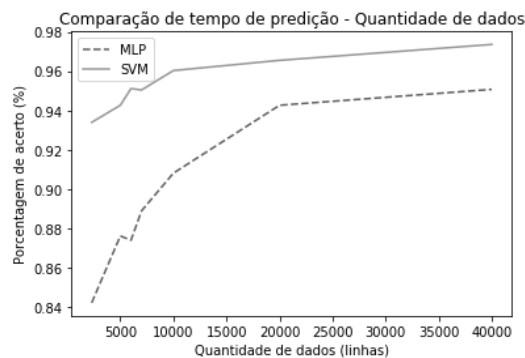


Figura 6. Comparação da porcentagem de acerto entre os algoritmos a partir de um conjunto de dados de tamanho variável.

Fonte: Própria do autor

5. Considerações Finais

Como visto nas duas implementações do presente trabalho, o desempenho do classificador SVM se mostra inferior ao do MLP. Essencialmente, o SVM tem o desempenho superior somente quando existe pouca quantidade de registros e a base é composta por várias dimensões. Em termos de aplicações de banco de dados dificilmente a quantidade de dimensão do registro muda, isso significa que o crescimento dos dados está na quantidade, e como foi visto, quando o SVM é treinado por um conjunto de dados extremamente grande, ele possui um desempenho muito inferior ao do MLP. De fato o SVM em todos os testes obteve uma porcentagem de acerto extremamente alta e superior quando comparada ao MLP, entretanto também é visto que quando a base de dados toma uma certa magnitude essa diferença tende a se

tornar insignificante.

Desta forma, é possível concluir que se uma aplicação depender de precisão e somente disso, o SVM é de fato o classificador recomendado. Mas isso levando em conta que esta aplicação terá um preditor mais lento do que MLP. Por outro lado, se a aplicação puder tolerar um erro percentual um pouco maior ou se a aplicação tiver uma base de dados muito grande, o MLP é a melhor opção visto que ele além de mais rápido para treinar e predizer também possui uma precisão de acerto considerável quando se comparado a outras redes neurais.

6. Referências

Silva, Ivan Nunes, “Redes Neurais artificiais: para engenharia e ciências aplicadas” / Ivan Nunes da Silva; Danilo Hernane spatti; Rogerio Andrade Flauzino. - São Paulo: Artliber, 2010.

Serasiya, S. D., Chaudhary, N. (2012). Simulation of Various Classifications Results using WEKA. International Journal of Recent Technology and Engineering (IJRTE), SARADHI, V., KAMIK, H., MITRA, P. A Decomposition Method for Support Vector Clustering. In Proc. of the 2nd International Conference on Intelligent Sensing and Information Processing (ICISIP), p. 268-271, 2005.

Ana Carolina Lorena, André C. P. L. F. de Carvalho, “Uma Introdução às Support Vector Machines”, 2007.

Horewicz, M. C., Nascimento Jr, C. L., & Silva, T. M. Técnicas de Redes Neurais para o Problema de Reconhecimento de Caracteres: Um Estudo Comparativo.

Marana, A. N., Papa, J. P., & Chiachia, G. (2014). Análise de Desempenho de Classificadores Baseados em Redes Neurais, Máquinas de Vetores de Suporte e Florestas de Caminhos Ótimos para o Reconhecimento de Dígitos Manuscritos. *UNESP, Faculdade de Ciências, Departamento de Computação. Bauru, SP.*

Bonifácio, FERNANDO NUNES. "Comparação entre as Redes Neurais Artificiais MLP, RBF e LVQ na Classificação de Dados." *Paraná: Universidade Estadual do Oeste do Paraná* (2010).

DE CASTRO, F., DE CASTRO, M. Redes neurais artificiais artificiais. Porto Alegre, RS: Pontificia Universidade Católica do Rio Grande do Sul, 2001.