

# Trabalho Computacional 1

## Inteligência Computacional Aplicada (TIP7077)

Igor Carneiro (569029)

### Introdução

Foi abordado nesse trabalho a implementação e a avaliação de dois tipos de problemas utilizando métodos de classificação e regressão. No primeiro problema, foi utilizado o conjunto de dados *MNIST*, famoso pelo reconhecimento de dígitos manuscritos, e no segundo, aplicamos o conjunto de dados de avaliação imobiliária (*Real Estate Valuation Data Set*). Para cada conjunto de dados, são aplicados diferentes modelos, e os resultados são comparados com artigos conhecidos na literatura.

### Parte 1 - Classificação com *Dataset MNIST*

O objetivo desta parte é implementar e comparar diferentes classificadores para o problema de reconhecimento de dígitos manuscritos. O conjunto de dados utilizado foi o MNIST, disponibilizado pelo prof. Guilherme Barreto, em dois arquivos \*.csv.

Foi utilizado no problema a normalização dos dados, juntamente com a técnica denominada de PCA (*Principal Component Analysis* - Análise dos Componentes Principais).

### Modelos Implementados

Foram implementados e analisados os seguintes classificadores, esses com os dados normalizados e aplicando o PCA. Para esse último foram escolhido 180 componentes, representando uma variância de 96.03%

- Classificador Linear de Mínimos Quadrados
- Adaline com Função de Ativação Sigmoidal
- MLP com 2-layers, 300 neurônios

## Resultados

Os resultados obtidos para os classificadores foram comparados com os dados apresentados no site de *Yann LeCun*, em [1], que mantém uma tabela de desempenho de diversos classificadores aplicados ao problema do MNIST.

Classificador	Tempo (seg)	Erro (%)	<i>Yann LeCun</i>
Classificador Linear MMQ	6.67	13,97	12
Classificador Linear MMQ - [PCA]	0,33	13,91	NA
Adaline com Ativação Sigmoidal	14,05	16,92	NA
Adaline com Ativação Sigmoidal - [PCA]	2,93	9,99	NA
MLP com 2-layers, 300 neurônios	488s	100epc - 5,3	4,7
MLP com 2-layers, 300 neurônios - [PCA]	313s	100epc - 4,6	NA

Table 1: Taxa de Erro dos classificadores para o MNIST

Foi realizado busca exaustiva no modelo MLP (2 layers, 300 neurônios), usando camadas com quantidade de neurônios múltiplos (32-64 / 64-128 / 128-256) e 300-300, conforme [1], além de ficar alternando entre 30, 50 e 100 épocas.

Durante a busca, a opção de 100 épocas, com 300 neurônios em cada camada oculta, foi a de melhor desempenho.

## Discussão

Os resultados mostram que o MLP, com dupla camada oculta, obteve um desempenho significativamente superior em relação aos classificadores lineares. A flexibilidade da função de ativação e o uso de camadas ocultas permitiram ao MLP aprender padrões não lineares com maior eficiência.

Além disso, a utilização do PCA (Análise de Componentes Principais) contribuiu para uma melhora geral no rendimento dos modelos, pois houve uma redução no tempo de processamento e uma diminuição na taxa de erro, otimizando tanto a eficiência computacional quanto a precisão dos modelos.

## Parte 2 - Regressão com *Dataset Real Estate Valuation*

### Introdução

A segunda parte tem como objetivo comparar dois modelos de regressão — a Regressão Linear Múltipla de Mínimos Quadrados e o Perceptron Multica-

madras (MLP) com uma camada oculta. Os resultados obtidos são comparados com os resultados do artigo "Building real estate valuation models with comparative approach through case-based reasoning" [2].

O conjunto de dados utilizado foi o *Real Estate Valuation Data Set*, disponível no UCI Machine Learning Repository [3]. Este conjunto de dados contém informações sobre transações imobiliárias, com atributos como:

- Data da transação;
- Idade da casa;
- Distância até a estação de metrô mais próximo;
- Número de lojas de conveniência nas proximidades;
- Latitude e longitude.

O *target* do conjunto é o preço por unidade de área.

## Modelos de Regressão Implementados

Foram implementados dois modelos de regressão:

- **Regressão Linear Múltipla de Mínimos Quadrados:** Este modelo ajusta uma reta (ou hiperplano) que melhor se adapta aos dados, minimizando a soma dos quadrados dos erros.
- **Perceptron Multicamadas (MLP):** Uma rede neural *feedforward* com uma camada oculta de 5 neurônios e uma função de ativação ReLU foi usada. O MLP foi treinado com 1500 épocas, utilizando a função de perda de Erro Quadrático Médio (MSE).

## Métricas de Avaliação

As métricas usadas para avaliar os modelos de regressão foram:

- **Coefficiente de Determinação ( $R^2$ ):** Mede a proporção da variabilidade da variável dependente que é explicada pelo modelo. Um valor mais próximo de 1 indica um bom ajuste.
- **Raiz do Erro Quadrático Médio (RMSE):** Mede a magnitude média dos erros de previsão. Um RMSE mais baixo indica maior precisão.

## Resultados Obtidos

Os resultados dos modelos acima citados e os resultados do artigo [2], são apresentados na Tabela 2. A comparação se dá em termos de  $R^2$  e  $RMSE$ .

Modelo	$R^2$	RMSE
MLP	0,6365	8,19
Regressão Linear	0,5777	8,83
MLP [2]	0,541	7,12
Regressão Linear [2]	0,392	8,04

Table 2: Comparando os resultados obtidos com o artigo [2]

## Discussão

A análise comparativa entre os resultados obtidos e o artigo revela o seguinte:

- O MLP 1-Layer apresentou um  $R^2$  de 0,6365, superior ao valor 0,541 reportado no artigo. Indicando que conseguiu explicar em uma maior proporção a variabilidade nos dados. Contudo, o RMSE (8,19) foi superior ao valor 7,12 do artigo, sugerindo que os erros de previsão ainda são ligeiramente maiores.
- Para a Regressão Linear, que obteve um  $R^2$  de 0,5777, o que é significativamente melhor do que o  $R^2$  de 0,392 reportado no artigo. No entanto, o RMSE (8,83) foi um pouco maior que o reportado no artigo (8,04), indicando que a precisão das previsões pode ser ligeiramente inferior.

Os resultados mostram que, em termos de  $R^2$ , o MLP superou a regressão linear em ambos os cenários. Isso era esperado, já que uma rede de *perceptrons* é capaz de capturar padrões não lineares com mais eficácia. No entanto, em termos de RMSE, o MLP apresentou erros elevados em relação ao trabalho de [2], possivelmente devido a alguma diferença na arquitetura da rede.

## References

- [1] Yann LeCun, The MNIST Database, <http://yann.lecun.com/exdb/mnist/>

- [2] Yeh, I. C., & Hsu, T. K. (2018). Building real estate valuation models with comparative approach through case-based reasoning. *Applied Soft Computing*, 65, 260-271.
- [3] Real Estate Valuation Data Set,  
<https://archive.ics.uci.edu/ml/datasets/Real+estate+valuation+data+set>