



## Inteligência Artificial Computacional

### Trabalho Computacional 2: Redes Neurais Artificiais.

*Professor: Prof. Msc. Paulo Cirillo Souza Barbosa*

## Introdução.

O presente trabalho é composto por duas etapas nas quais devem-se realizar a implementação de diferentes modelos baseados em redes neurais artificiais, apresentadas em sala de aula. Na primeira etapa do trabalho, utiliza-se dados sintetizados para tentar resolver um problema de classificação binária e com dados possuem apenas duas características. Nesta etapa, os modelos lineares **perceptron** e **ADALINE** devem ser aplicados para solução.

No segundo momento, a tarefa para classificação de padrões é de reconhecimento facial. No conjunto de dados disponibilizados, há um total de 20 pessoas diferentes e os modelos preditivos devem realizar a tarefa de discriminar tais pessoas. Nesta etapa, os modelos utilizados são o **Perceptron de Múltiplas Camadas e a rede RBF**.

Em ambas etapas do trabalho, os modelos a serem implementados, utilizam o paradigma supervisionado para aprender a partir de pares de amostra e valor observado, para resolver um problema de classificação. Contudo, a primeira etapa este problema possui apenas duas classes, ao passo que na segunda existem vinte.

## Problema de Classificação Binária.

Nesta primeira etapa, solicita-se que faça o uso do algoritmo disponibilizado no final deste documento (bem como no AVA), para sintetizar os dados que possuem as dimensões  $\mathbf{X} \in \mathbb{R}^{N \times p}$ . Após a geração do conjunto de dados, faça o que se pede:

1. Faça a implementação dos modelos Perceptron Simples (PS) e Adaline utilizando os critérios descritos no final desta seção .
2. Faça uma visualização inicial dos dados através do gráfico de espalhamento. Nessa etapa, faça uma discussão inicial sobre quais resultados poderão ser obtidos ao utilizar o perceptron simples e o ADALINE.
3. Para utilização dos modelos implementados, faça uma organização no conjunto de dados para que se tenha a nova dimensão,  $\mathbf{X} \in \mathbb{R}^{(p+1) \times N}$ .
4. Faça a definição do  $\eta$  (passo de aprendizagem), conforme as discussões realizadas em sala e escrita nos slides.
5. Para o modelo ADALINE realize a definição do valor de precisão.
6. Inicia-se a etapa de validação dos modelos, assim deve-se realizar o processo de treinamento e teste, em 100 rodadas. Assim, para cada rodada, é necessário realizar a divisão do conjunto de dados para que se tenha 80% da informação em um conjunto de treinamento, e 20% da informação para teste. Considere nesta divisão, que existam variáveis dedicadas para os rótulos de treinamento e teste.
7. Ao final das rodadas, compute os seguintes resultados para o PS e ADALINE:
  - (a) Acurácia Média, com seu desvio padrão, maior e menor valor.
  - (b) Sensibilidade Média, com seu desvio padrão, maior e menor valor.

- (c) Especificidade Média, com seu desvio padrão, maior e menor valor.
- (d) Construa uma matriz de confusão (gráfico) para a rodada em que se teve a melhor acurácia.
- (e) Construa uma matriz de confusão (gráfico) para a rodada em que se teve a pior acurácia.
- (f) Para esses dois casos, construa também um gráfico que mostre o hiperplano de separação dos dois modelos.

8. Com os resultados obtidos, faça discussões!

## Observações PS e ADALINE

A implementação dos modelos PS e ADALINE deve ser realizada com base no pseudocódigo e implementações disponibilizadas em sala de aula.

## Redes Neurais Não Lineares.

A segunda etapa do presente trabalho, trata-se da solução de um problema de classificação de padrões e processamento de imagens. Assim, sua equipe deve realizar o acesso ao conjunto de dados disponibilizado em [CMU Face Images Data Set](#). Além de realizar o download, é importante realizar uma leitura sobre o que se trata o conjunto de dados.

É importante destacar que como se tratam de imagens, é necessário realizar uma biblioteca que faz a aquisição da imagem e a converte para um numpy array. Assim, recomenda-se o uso da biblioteca **opencv** que pode ser instalada no terminal utilizando a seguinte instrução:

- `pip install opencv-python`

Com esta biblioteca, é possível fazer as aquisições e transformações necessárias de modo que as imagens possam ser utilizadas nos algoritmos de aprendizado de máquina vistos em sala de aula. Além disso, é disponibilizado na seção de anexo deste presente documento, uma rotina que realiza a aquisição das imagens e as organizam em um conjunto de dados com seus rótulos associados. Em tarefas que se utilizam imagens, cada amostra pode possuir diversos preditores associados a quantidade de pixels em cada imagem. Assim é interessante realizar uma análise de redimensionamento da imagem (tendo em vista que ainda não foi estudado como extrair as melhores informações de dados). Assim, no algoritmo disponibilizado, há uma opção de redimensionamento das amostras que deve ser escolhida por sua equipe. Neste caso pode-se experimentar alguns redimensionamentos, por exemplo,  $50 \times 50$ ,  $40 \times 40$ ,  $30 \times 30$ ,  $20 \times 20$ . Essa escolha pode ser baseada no melhor compromisso (*tradeoff*) entre acurácia e custo computacional.

Além do descrito anteriormente, faça o que se pede:

1. Realize a implementação dos modelos MLP e rede RBF com base nas discussões realizadas em sala de aula e considerando as observações no final do presente documento.
2. Com base na magnitude dos rótulos presentes no conjunto de dados, faça a normalização dos dados utilizando o método min-max.
3. Para utilização dos modelos implementados, faça uma organização no conjunto de dados para que se tenha a nova dimensão,  $\mathbf{X} \in \mathbb{R}^{(p+1) \times N}$ .
4. Para ambos os modelos, faça a definição do  $\eta$  (passo de aprendizagem) e precisão, conforme as discussões realizadas em sala e escrita nos slides.
5. Para o modelo MLP, faça uma discussão inicial sobre overfitting e underfitting. Assim, realize o projeto de uma rede com poucos neurônios na camada oculta e produza resultados de acurácia e Matriz de Confusão. Em seguida, aumente a quantidade de neurônios e/ou camadas escondidas até que seja identificado o overfitting. Expresse os resultados em duas matrizes de confusão.
6. Após esta análise, faça a validação do modelo escolhendo uma topologia da rede MLP que não produza underfitting bem como overfitting. Como este processo pode ser custoso, faça a definição da topologia com base nas regras discutidas em sala de aula.

7. Para o modelo RBF, escolha também uma quantidade de funções de base, considerando as discussões realizadas em sala de aula.
8. Para compor os resultados desta etapa, sua equipe deve realizar a construção de matrizes de confusão, box-plots e tabelas que expressem a acurácia média, desvio padrão, maior e menor valor.
9. Como o tempo de treinamento pode ser custoso, faça com que se tenham poucas rodadas de validação dos modelos.
10. Exiba nos resultados a quantidade média de épocas que fazem com que os modelos atinjam a convergência.
11. Projete uma rede MLP que seja superdimensionada, contudo, identifique quando o overfitting acontecer. Nesse caso, a parada antecipada deve ser operada. Faça discussões, se essa arquitetura obteve resultados melhores ou piores com relação às topologias anteriores e a rede RBF.

## Observações

- O trabalho pode ser desenvolvido em equipe de no máximo dois alunos.
- A nota do trabalho é dividida da seguinte maneira:
  1. 40% Relatório E implementações.
  2. 60% Arguição.
- O total referente aos 40% (Relatório e implementações) é dividido da seguinte maneira:
  - Título (1%).
  - Resumo (15%).
  - Introdução (20%).
  - Desenvolvimento (27%).
  - Resultados (27%).
  - Referências (5%).
  - **Implementações** (5%).
- Obs1: O envio das implementações é **obrigatório**. Caso a equipe não realize esta entrega, será atribuído nota **zero** para os respectivos alunos.
- Obs2: A data estipulada para entrega do trabalho, também é um critério avaliativo. Assim, caso haja atraso na entrega do trabalho, será aplicada: **de 00:15h até 24h: penalidade de 20% ; 24:15h até 48h: penalidade de 40% ; acima de 48h: penalização máxima (100%)**.
- Ob3: Os trabalhos e implementações serão enviadas a um software anti-plágio. Qualquer caracterização de plágio ocasionará em nota zero para ambas equipes.
- Obs4: Para o presente trabalho, não será permitido o uso de bibliotecas que tenham as implementações prontas dos modelos Perceptron Simples, ADALINE, MLP e rede RBF. Caso sua equipe faça o uso de tais bibliotecas, serão descontadas as pontuações proporcionais.