



## Inteligência Artificial Computacional

### Trabalho Computacional AV2: Redes Neurais Artificiais.

Professor: Prof. Msc. Paulo Cirillo Souza Barbosa

## Observações iniciais.

Solicita-se que realize a leitura por completo do presente documento. Na AV3 as bibliotecas permitidas são:

1. numpy;
2. matplotlib;
3. seaborn;
4. opencv (para leitura de imagens) instalação: pip install opencv-python;

## Introdução.

O presente trabalho é composto por duas etapas em que deve-se utilizar os modelos de IA, que são bioinspirados no funcionamento do cérebro humano. Tais modelos são redes neurais artificiais (RNA) lineares e não-lineares. A primeira etapa do trabalho, trata-se da solução de um problema simples, e serve como um validador das implementações dos modelos solicitados. A segunda etapa, trata-se da resolução de um problema de reconhecimento facial de imagens.

## Primeira etapa - Regressão e Classificação para problemas bidimensionais. [4,0 pts]

### Tarefa de Classificação

Solicita-se que faça o acesso ao conjunto de dados disponibilizado na plataforma AVA, chamado **spiral\_d.csv** que trata-se de dados sintetizados que possuem duas classes (as quais estão rotuladas no próprio arquivo na terceira coluna). Assim, faça o que se pede:

1. Faça a organização do conjunto de dados para serem apresentados às redes neurais.
2. Faça uma visualização inicial dos dados através do gráfico de espalhamento.
3. Os modelos de RNA a serem implementados nessa etapa serão: **Perceptron Simples, ADALINE, Perceptron de Múltiplas Camadas (MLP) e Rede Função de Base Radial (RBF)**. Para cada modelo, deve-se discutir como os hiperparâmetros foram escolhidos.
4. Para os modelos MLP e RBF, pede-se que identifique o *underfitting* e *overfitting*. Neste caso, sua equipe deve explorar diferentes topologias das duas redes, ao modificar os hiperparâmetros associados a quantidade de neurônios e camadas (realizando um subdimensionamento e superdimensionamento). Para os dois casos, faça a composição do resultado ao descrever as topologias encontradas e calcule os desempenhos utilizando matriz de confusão, acurácia, especificidade, sensibilidade e curva de aprendizado para cada um dos casos (modelos MLP e RBF bem como underfitting e overfitting).

5. Para validar os modelos (com suas topologias definidas) utilizados na tarefa, sua equipe deve projetar a validação por amostragem aleatória. Nessa etapa, defina a quantidade de rodadas da simulação igual a  $R = 500$ . Em cada rodada, deve-se realizar o particionamento em 80% dos dados para treinamento e 20% para teste. As medidas de desempenho de cada modelo, devem ser: **acurácia, sensibilidade, especificidade, precisão e f1-score**.
6. Das 500 rodadas do item anterior, selecione aquelas que proporcionaram o maior e o menor valor de cada métrica. Para esses casos, construa uma matriz de confusão. **Obs:** sua matriz de confusão deve ser implementada pela equipe (ou seja, realizar a soma de seus elementos), no entanto, para realizar seu plot, pode-se utilizar a biblioteca *seaborn*. Exemplo: [link heatmap](#). Além disso, para os casos selecionados, pede-se que plote a curva de aprendizado de todos os modelos.
7. Ao final das  $R = 500$  rodadas calcule para cada modelo utilizado, média aritmética, desvio-padrão, valor maior, valor menor, para as seguintes métricas de desempenho: **acurácia, sensibilidade, especificidade, precisão e f1-score**. Coloque os resultados obtidos em uma tabela (UMA PARA CADA MÉTRICA) e discuta os resultados obtidos. **Obs:** O resultado não precisa ser limitado a tabela, como pode ser expresso via gráficos (BOXPLOT, VIOLIN PLOT E OUTROS). Como exemplo, segue uma tabela que serve para uma das três métricas solicitadas. Segue um exemplo para apenas uma das tabelas:

Modelos	Média	Desvio-Padrão	Maior Valor	Menor Valor
<b>Perceptron Simples</b>				
<b>Perceptron de Múltiplas Camadas</b>				
<b>Rede RBF</b>				

## Classificação Multiclasse para Problema Multidimensional [6,0 pts]

A segunda etapa do trabalho, trata-se da identificação de 20 pessoas diferentes em um contexto de reconhecimento facial em imagens. O conjunto de dados possui 640 imagens, e cada imagem sendo uma matriz  $120 \times 128$ . O conjunto de dados está disponibilizado no AVA (pasta com nome **"RecFac"**) e consiste em 640. Os preditores podem ser (e devem) ser redefinidos, ao redimensionar as imagens para uma menor quantidade de atributos. Após esse contexto apresentado, faça o que se pede:

1. Escolha uma quantidade de linhas e colunas de cada imagem entre as seguintes:  $(30 \times 30)$ ,  $(40 \times 40)$ ,  $(50 \times 50)$  ou  $(80 \times 80)$ .
2. De posse do conjunto de dados, organize a massa de dados  $\mathbf{X}$  com as dimensões  $\mathbb{R}^{p+1 \times N}$ .
3. Como existem 20 classes, faça a codificação utilizando a estratégia *one-hot-encoding*. Assim, considere

como exemplo, as cinco primeiras pessoas do conjunto de dados:

$$\begin{array}{c}
 \text{an2i} = \begin{bmatrix} +1 \\ -1 \end{bmatrix} \quad \text{at33} = \begin{bmatrix} -1 \\ +1 \\ -1 \end{bmatrix} \quad \text{boland} = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \quad \text{bpm} = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \quad \text{ch4f} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}
 \end{array}$$

Assim, sua equipe deve organizar a massa de dados  $\mathbf{Y}$  (rótulos), com as dimensões  $\mathbb{R}^{C \times N}$ .

4. Os modelos de RNA a serem implementados nessa etapa serão: **Perceptron Simples, ADaptive LI-Near Element, Perceptron de Múltiplas Camadas (MLP) e Rede RBF**. Para cada modelo, deve-se discutir como os hiperparâmetros foram escolhidos. Neste caso, utiliza a experiência sua adquirida na primeira etapa do trabalho.
5. Para validar os modelos utilizados na tarefa, sua equipe deve projetar a validação utilizando as simulações por Monte Carlo. Nessa etapa, defina a quantidade de rodadas da simulação igual a  $R = 10$ . Em cada rodada, deve-se realizar o particionamento em 80% dos dados para treinamento e 20% para teste. As medidas de desempenho de cada modelo, deve ser a **acurácia**.
6. Das 10 rodadas do item anterior, selecione aquelas que proporcionaram o maior e o menor valor de acurácia. Para esses dois casos, construa as matrizes de confusão. **Obs:** sua matriz de confusão deve ser implementada pela equipe, no entanto, para realizar seu plot, pode-se utilizar a biblioteca *seaborn*. Exemplo: [link heatmap](#). Com essas matrizes de confusão criadas, tente entender para qual categoria os modelos fazem previsões piores e melhores. Tente verificar se há um padrão percebido nos modelos em questão (sempre erram ou sempre acertam as mesmas categorias). Além disso, para os dois casos (maior e menor acurácia), pede-se que plote a curva de aprendizado de todos os modelos.
7. Ao final das  $R = 100$  rodadas, calcule para cada modelo utilizado, média aritmética, desvio-padrão, valor maior, valor menor, para as seguintes métricas de desempenho: **acurácia**. Coloque os resultados obtidos em uma tabela **ediscuta os resultados obtidos**. **Obs:** O resultado não precisa ser limitado a tabela, como pode ser expresso via gráficos (BOXPLOT, VIOLIN PLOT E OUTROS). Como exemplo, segue uma tabela que serve para a métrica solicitada.

Modelos	Média	Desvio-Padrão	Maior Valor	Menor Valor
<b>ADaptive LINear Element.</b>				
<b>Perceptron de Múltiplas Camadas</b>				
<b>Rede RBF</b>				

## Informações Adicionais aos modelos de RNA

De maneira geral, para as duas etapas do trabalho, sua equipe deve considerar que tais modelos possuem hiperparâmetros e procedimentos específicos. Para tais modelos, é necessário que os dados sejam normalizados (caso eles não já estejam). Além disso, todos os modelos precisam de um critério de convergência baseado no número máximo de épocas. Essa última informação é importante, pois, o modelo Perceptron Simples tradicional não possui esse critério. Assim, tal modificação deve ser feita pela sua equipe.

Considere que para os modelos Perceptron Simples e ADALINE, a função de ativação é linear (degrau ou degrau bipolar). A função de ativação utilizada na rede MLP deve ser escolhida entre a tangente hiperbólica e a sigmoide logística. Para simplicidade do projeto, faça com que todos os neurônios possuam a mesma função.

Para o caso da redes MLP e RBF, considere escolha dos hiperparâmetros, conforme as discussões realizadas em sala (ou através do que está descrito nos slides). Os hiperparâmetros são :

1. Quantidade de camadas escondidas (apenas para MLP).
2. Quantidade de neurônios nas camadas escondidas (na rede RBF existe apenas uma camada oculta).
3. Quantidade de neurônios na camada de saída.
4. Valor da precisão (ou critério de parada da convergência).
5. Número máximo de épocas.
6. Taxa de aprendizagem.

## 5) Relatório.

Além das implementações, o presente trabalho deve ser entregue em modelo de relatório. Este deve possuir as características descritas nos slides de apresentação do curso. Desta maneira, deve possuir:

1. Título (2,5%).
2. Resumo (2,5%).
3. Metodologia (42,5%).
4. Resultados (42,5%).
5. Conclusões (10%).

O modelo para trabalho pode ser encontrado neste [LINK](#)

## 6) Observações.

- **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%)**.
- **Obs 3:** As implementações devem ser realizadas utilizando apenas as bibliotecas descritas no início do presente documento. Caso alguma biblioteca diferente seja utilizada, será atribuída a nota **zero** para todos os membros da equipe.
- **Obs 4:** QUAISQUER alterações dos modelos, que não foram discutidas em sala de aula, ou que não estão presentes no conjunto de slides, devem ser explicadas pela equipe. Caso a equipe não saiba o motivo pelo qual as adequações foram realizadas, será atribuída a nota **zero** aos respectivos alunos.
- Caso não sejam utilizados os modelos descritos no presente documento, será considerado que sua equipe não atendeu aos requisitos, e assim, a nota para os membros da equipe será **zero**.
- **Obs 5:** Os trabalhos e implementações serão enviadas a um software anti-plágio. Qualquer caracterização de plágio ocasionará em nota zero para ambas equipes.