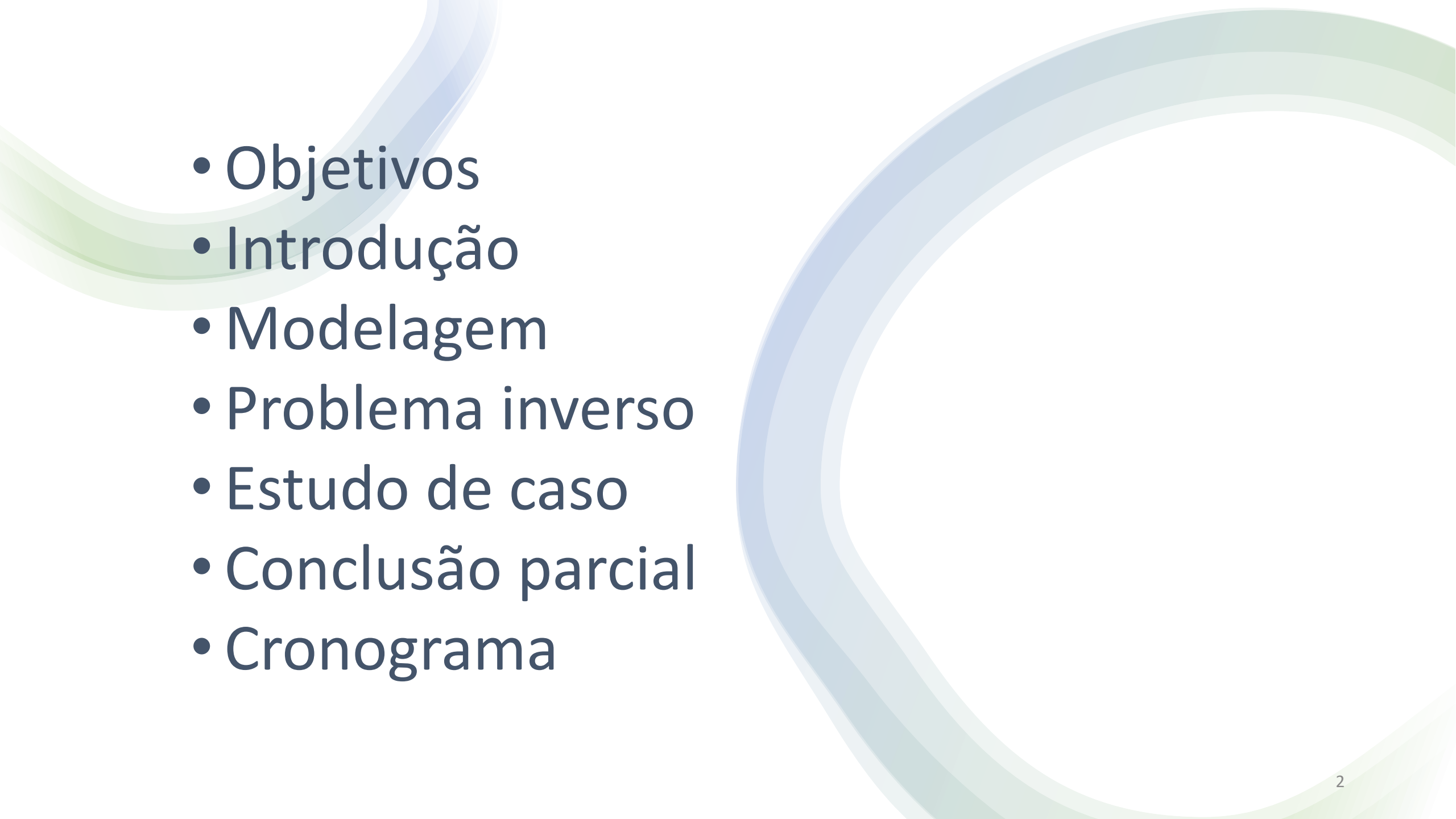


Problema inverso aplicado à modelagem e linearização de transmissores sem fio

Caio Phillipe Mizerkowski

Orientador: Prof. Eduardo Gonçalves de Lima

Parte 1 - Análise comportamental de PAs utilizando o problema inverso.

- 
- Objetivos
 - Introdução
 - Modelagem
 - Problema inverso
 - Estudo de caso
 - Conclusão parcial
 - Cronograma

Objetivos e motivação

Neste trabalho o objetivo geral é a compreensão, a utilização e o desenvolvimento de técnicas que utilizam a resolução de um problema inverso (PI) para a modelagem, a validação e a linearização de amplificadores de potência (PA).

A primeira parte, realizada neste TCC I e detalhada nesta apresentação, é a aplicação de PI na análise comportamental do PA.

A segunda parte, a ser realizada no TCC II, é a aplicação de PI na validação de pré-distorcedores.

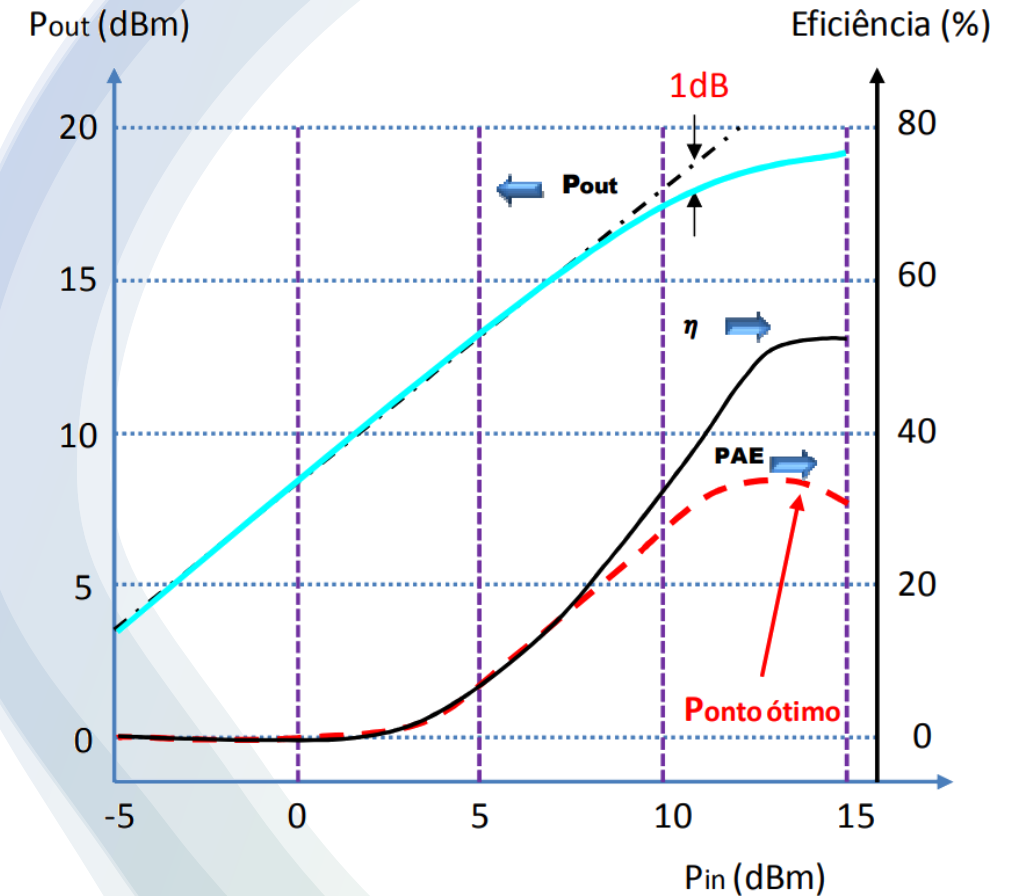
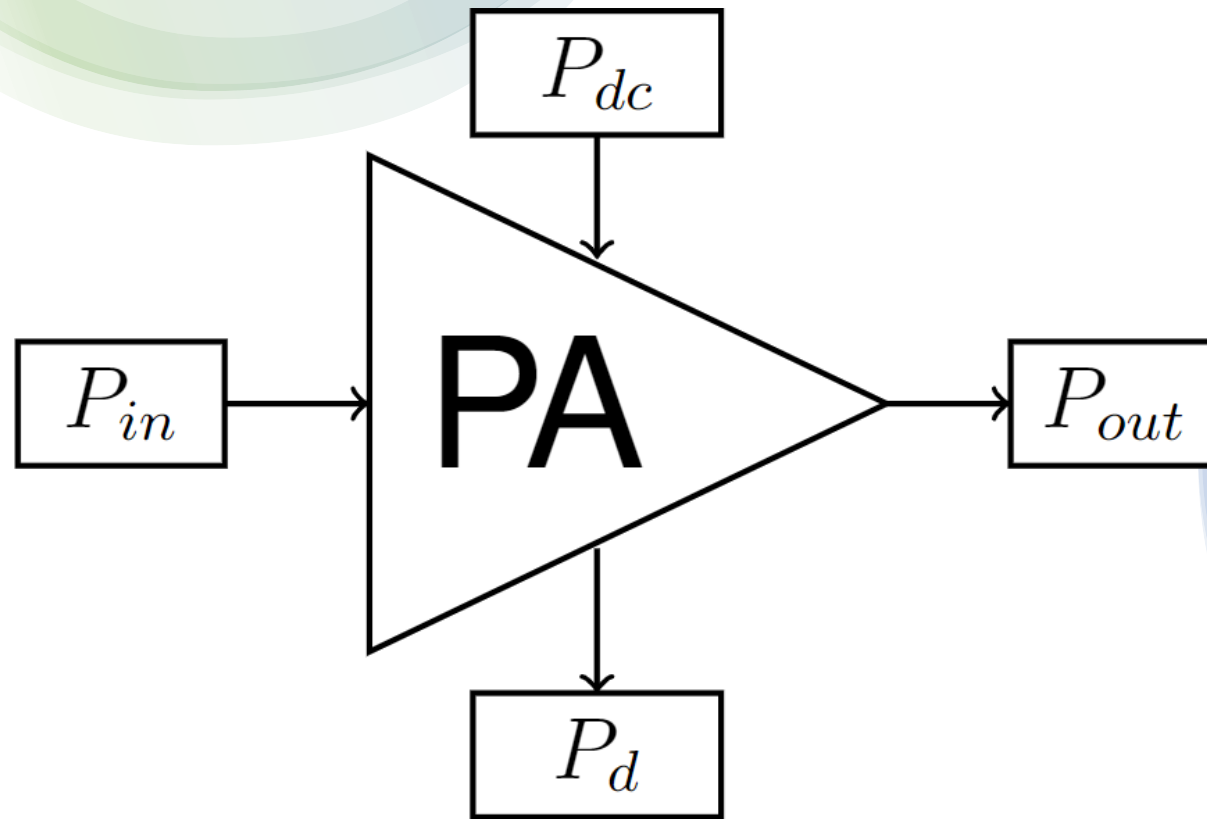
A motivação deste trabalho é explorar o problema inverso para melhorar a eficiência dos amplificadores de potência.

Introdução - Amplificadores de potência

Utilizados na comunicação sem fio.

Não-lineares em razão de efeitos térmicos e saturação.

Dicotomia entre linearidade e eficiência.

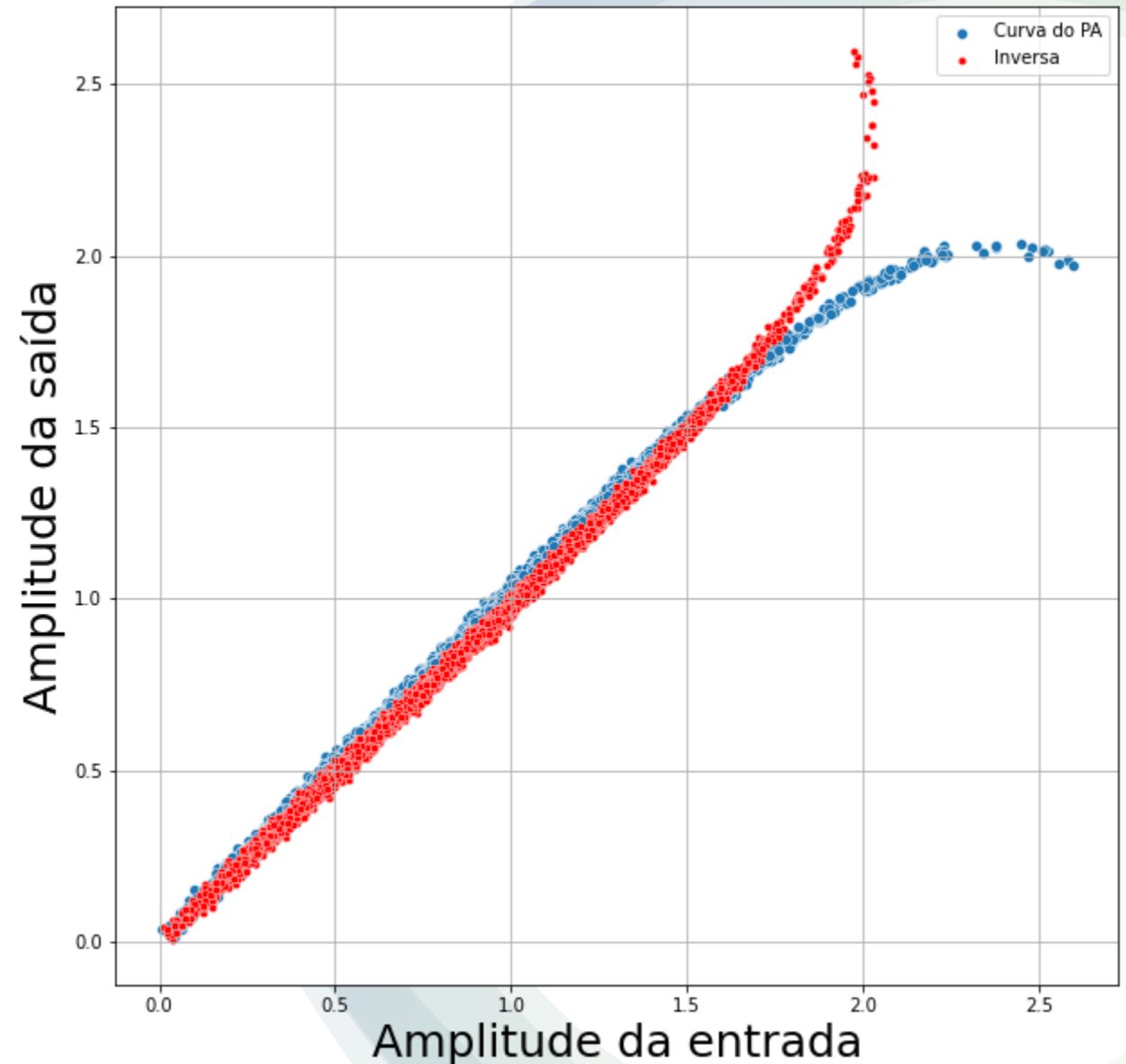


Introdução - Linearização do PA

Para maior eficiência é necessário operar na região não-linear dos PAs.

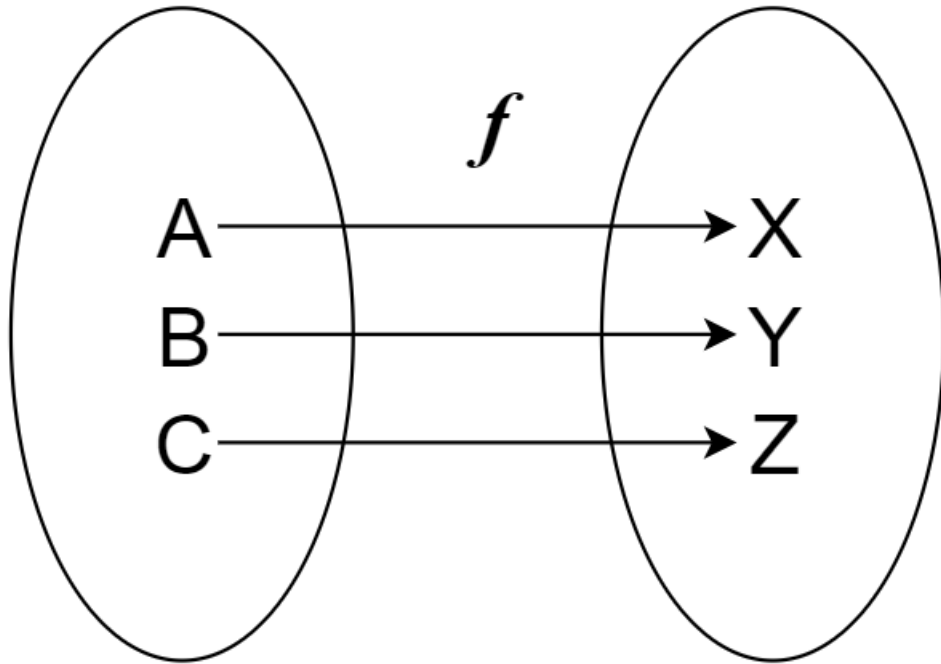
É necessária a linearização da curva através de sua inversa para a operação nesta região.

Quanto menos linear o comportamento nessa região, mais complexo é o processo de modelagem da inversa.

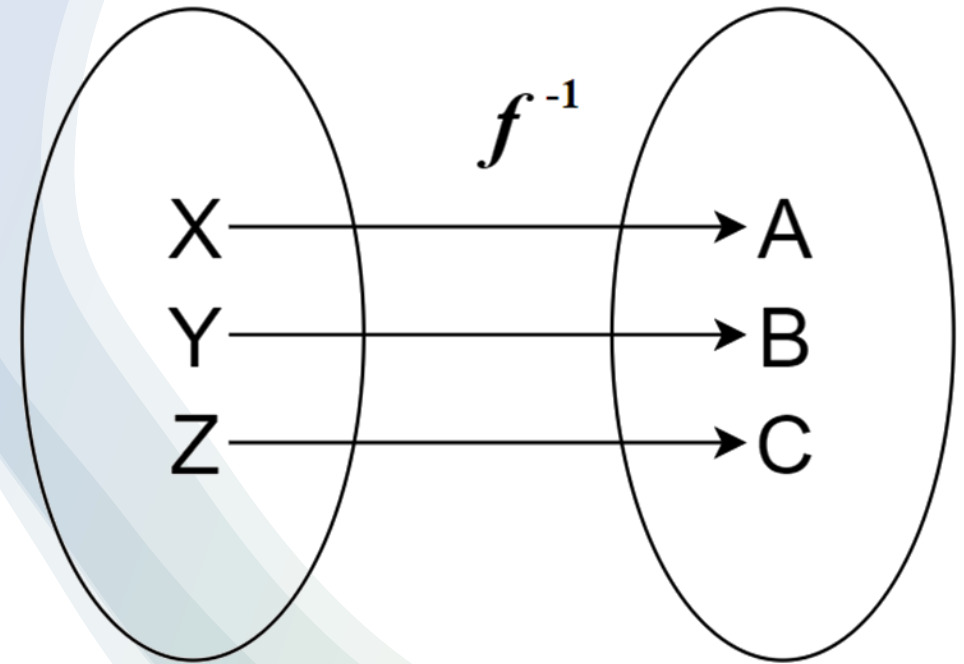


Introdução - Função inversa e bijetividade

Uma característica necessária para a existência de uma inversa única, analítica ou numérica, é a função ser bijetiva. Ou seja, para cada elemento do contradomínio existe um e somente um elemento no domínio.



Os PAs apresentam uma região monotônica (bijetiva) e uma região não monotônica. Em razão desta característica, os modelos de PAs servem como bons estudos de caso na modelagem computacional.

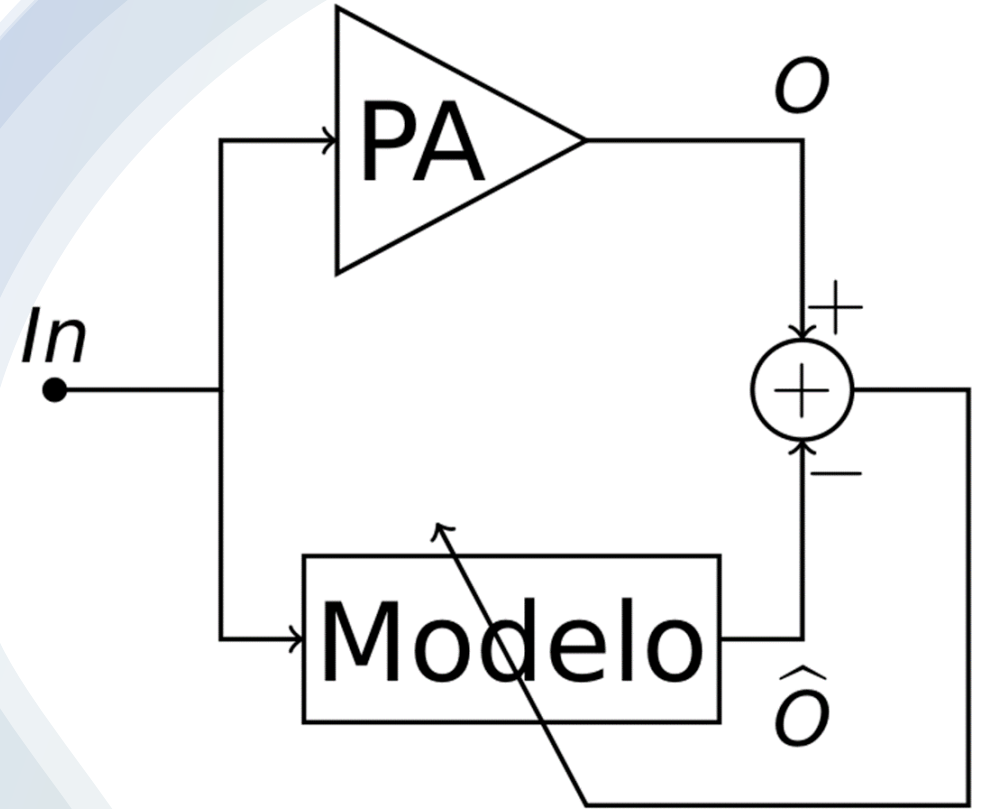


Introdução - Modelagem computacional

A modelagem computacional é o desenvolvimento de modelos matemáticos para o simular o comportamento de componentes, sistemas etc.

Uma de suas funções é prover um substituto às medições físicas. Como as medições físicas oneram em tempo, recursos e espaço, utilizar um modelo permite a realização de análises de forma mais prática e rápida.

Entre as diversas técnicas para a modelagem computacional, como por exemplo as séries polinomiais, foi escolhido para este trabalho o uso de redes neurais artificiais do tipo perceptron de multicamada (MLP) numa topologia específica para valores complexos.

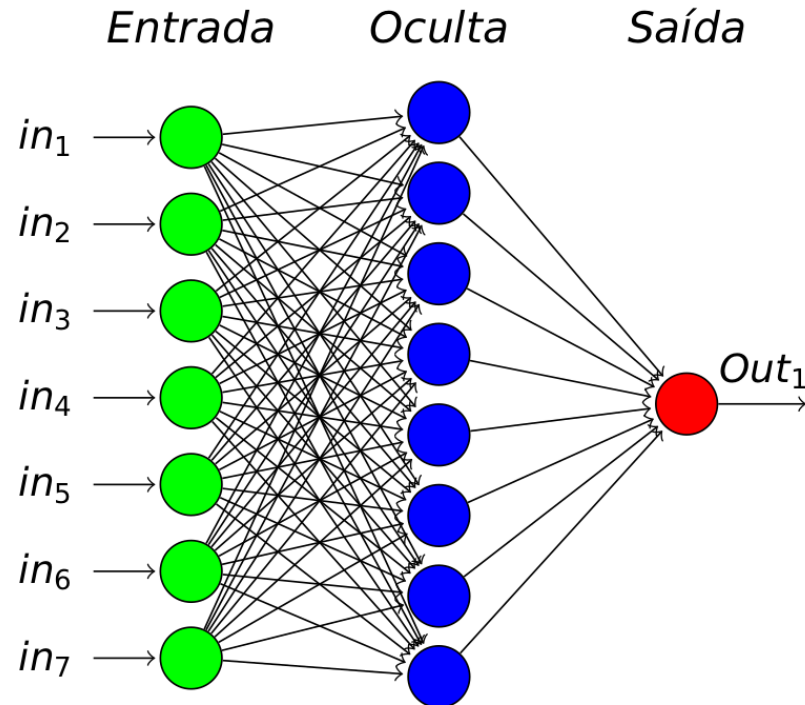


Modelagem com redes neurais - Conceitos básicos

Elemento fundamental das redes neurais artificiais, o perceptron foi inspirado pelo funcionamento dos neurônios.

Sendo capaz de separar linearmente um espaço em duas regiões distintas.

$$\begin{bmatrix} 1 \\ in_1 \\ \vdots \\ in_n \end{bmatrix}^T \begin{bmatrix} b \\ w_1 \\ \vdots \\ w_n \end{bmatrix} = \sum_{i=1}^n in_i \times w_i + b$$



O MLP é composto de várias camadas de perceptrons e de funções de ativação não-lineares (Sigmoides, ReLU etc).

Capaz de aproximar qualquer função continua após o processo de treinamento, sendo o MLP capaz de modelar diversos PAs.

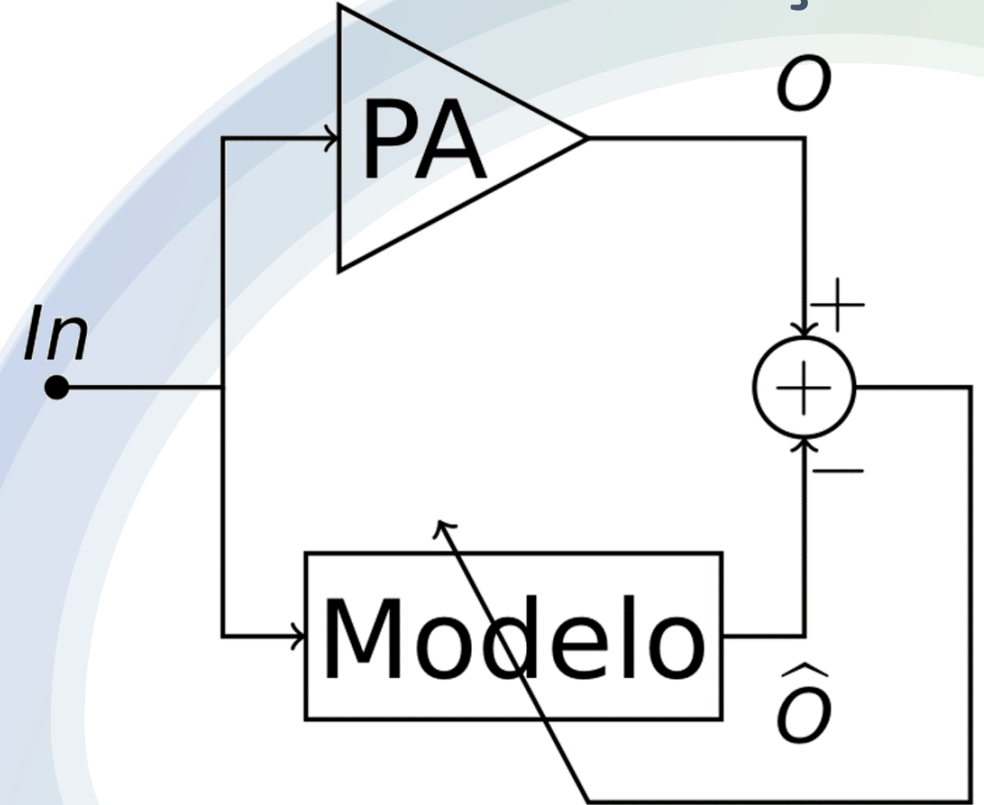
Modelagem com redes neurais - Treinamento e validação

Para garantir a generalização do modelo, as amostras são separadas em conjuntos de treinamento e de validação.

O treinamento é um processo iterativo de ajuste dos coeficientes do modelo para reduzir uma função de perda.

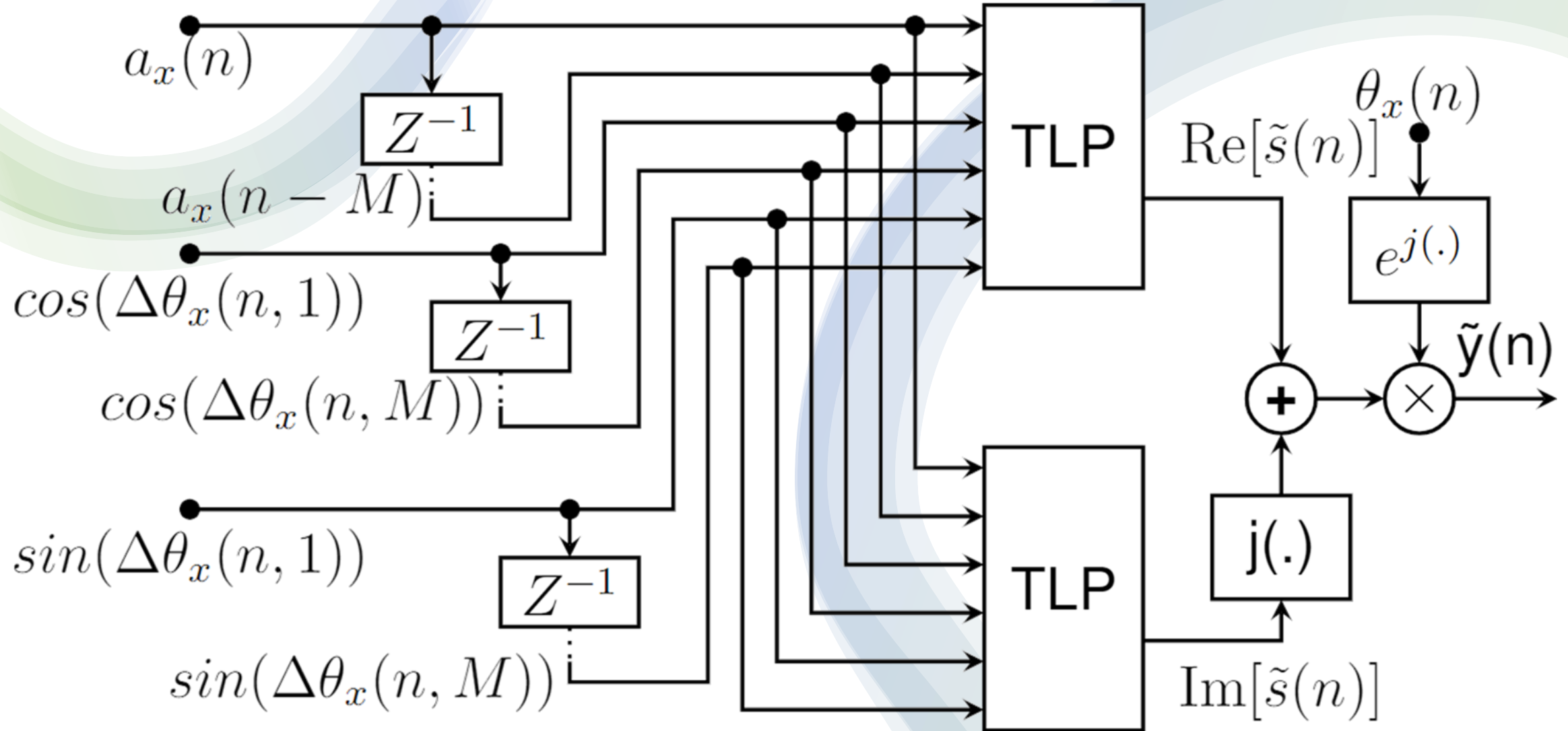
A validação aplica o segundo conjunto de dados no modelo e afere a função de perda para estes dados.

$$NMSE = 10 \log_{10} \left(\frac{\sum_{i=1}^N |o_i - \hat{o}_i|^2}{\sum_{i=1}^N |o_i|^2} \right)$$



O erro quadrático médio normalizado é uma dentre as métricas usadas para se analisar o resultado do treinamento e da validação de um modelo.

Modelagem com redes neurais - Topologia utilizada



Amplitude

Diferença de fase

$$a_x$$

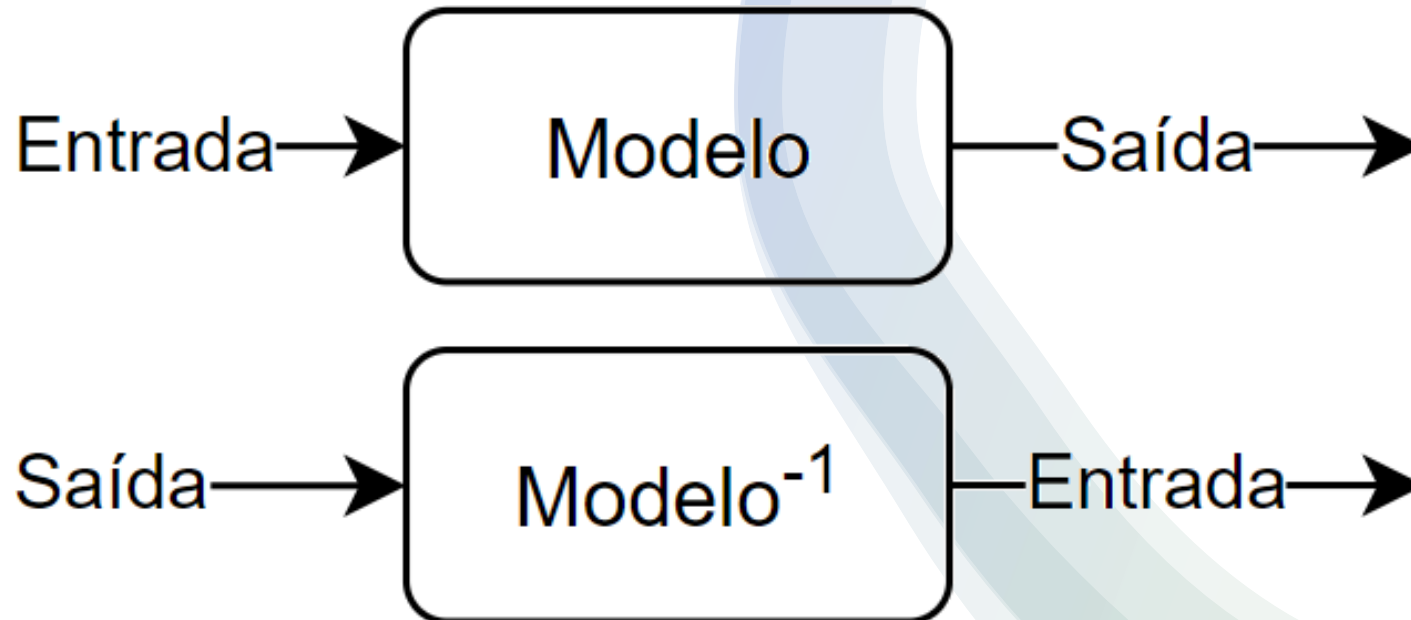
$$\Delta\theta_x(n, m) = \theta_x(n - m + 1) - \theta_x(n - m)$$

Problema inverso - Problemas bem-postos e mal-postos

Um problema é bem-posto caso:

- Uma solução exista
- Esta solução seja única
- O comportamento desta solução seja contínuo

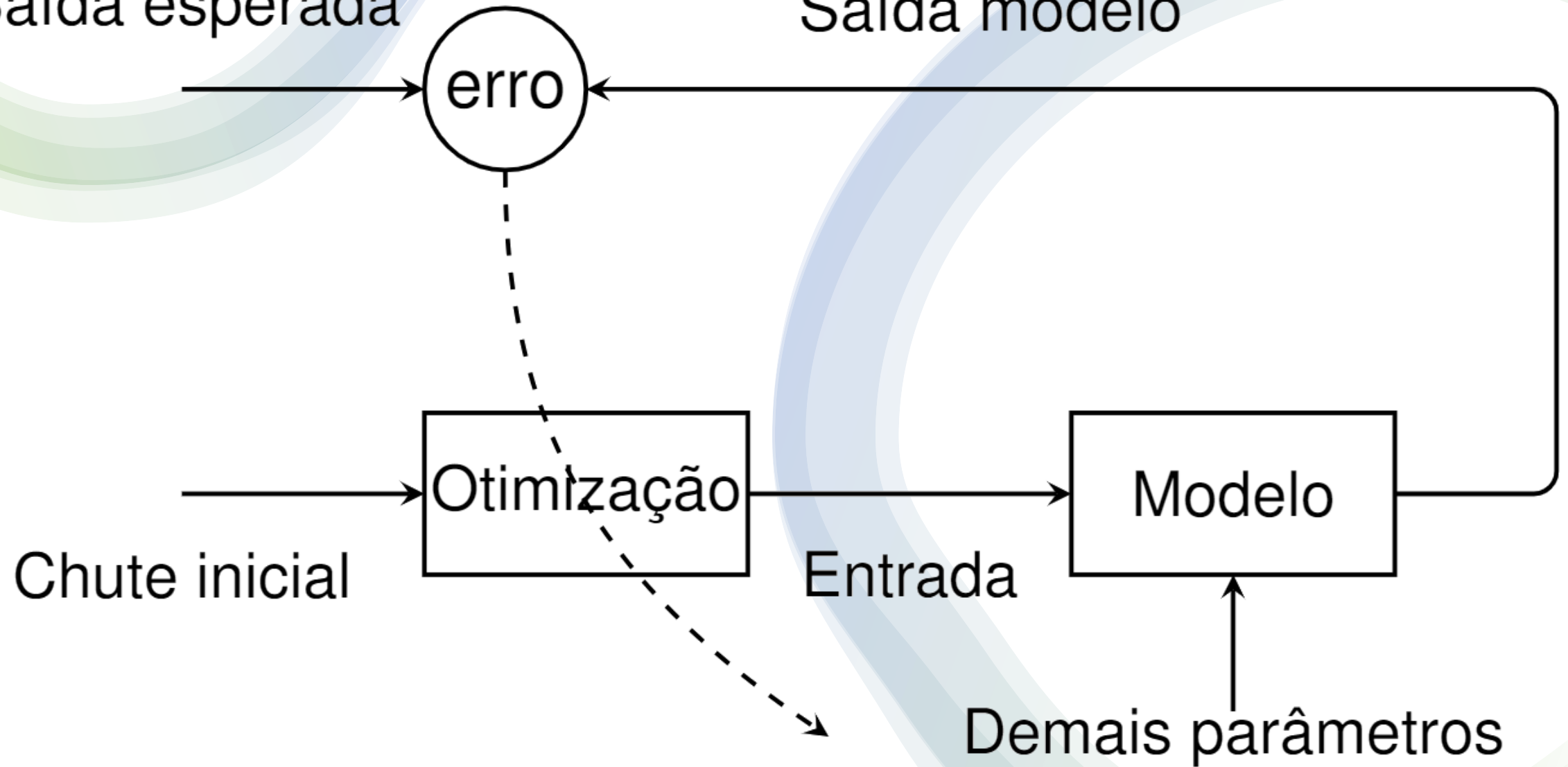
O problema inverso consiste em encontrar as entradas de um modelo conhecendo-se as saídas. Este problema pode ser bem-posto, mas geralmente é mal-posto.



Problema inverso - Aplicação do problema inverso

Saída esperada

Saída modelo



Estudo de caso - PA analisado

No estudo de caso apresentando foi analisado o seguinte modelo de PA

- Advanced Memory Polynomial (AMP)
- PA multimodo e sinal LTE
- Tecnologia CMOS 130 nm
- Largura de banda: 20 MHz
- Frequência de amostragem: 153,6 MHz
- Frequência central em 2,4 GHz

SCHUARTZ, L. ; SANTOS, E. L. ; LEITE, B. R. B. A. ; MARIANO, A. A. ; LIMA, E. G. . **Reduced-Complexity Polynomials with Memory Applied to the Linearization of Power Amplifiers with Real-Time Discrete Gain Control**. CIRCUITS SYSTEMS AND SIGNAL PROCESSING, v. 38, p. 3901-3930, 2019.

DOS SANTOS, E. L. ; RIOS, M. A. ; SCHUARTZ, L. ; LEITE, B. ; LOLIS, L. ; LIMA, E. G. ; MARIANO, A. A. . **A fully integrated CMOS power amplifier with discrete gain control for efficiency enhancement**. MICROELECTRONICS JOURNAL, v. 70, p. 34-42, 2017.

Estudo de caso - Ferramentas

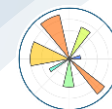
Trabalho realizado com a linguagem de programação Python, com as seguintes bibliotecas:



TF é uma biblioteca para a criação e treinamento de projetos de ML.



Numpy é uma biblioteca para computação científica, com o uso de matrizes.



Matplotlib é uma biblioteca para a criação de gráficos e visualizações.



Keras é uma API para o acesso ao TF de forma consistente e simples.



Scipy é uma biblioteca de computação científica baseada em numpy.

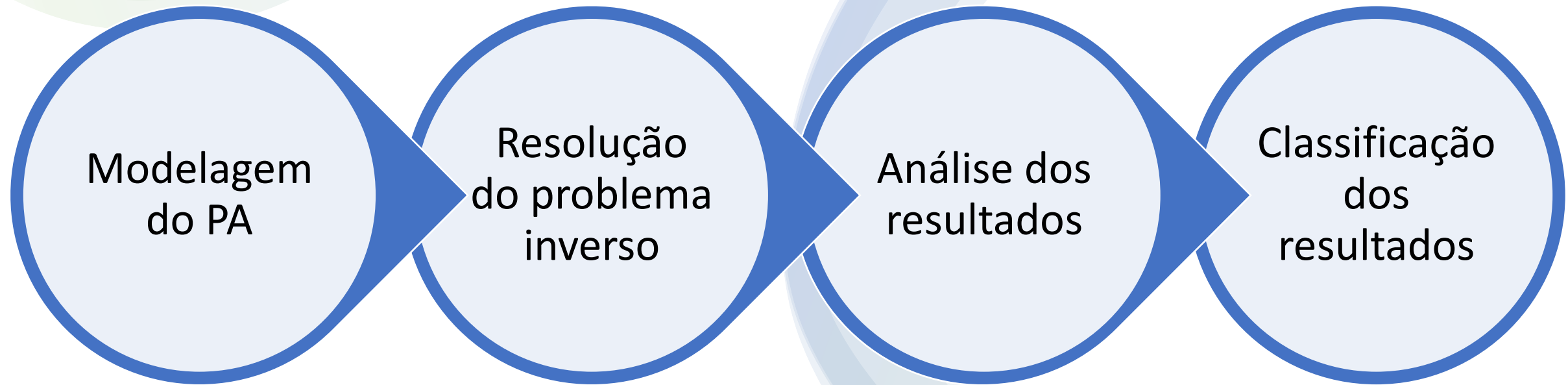


Seaborn é uma biblioteca de alto nível baseada no matplotlib para gráficos.



Polars é uma biblioteca para a representação de dados tabulares.

Estudo de caso - Metodologia



Estudo de caso - Modelagem do PA

Tensorflow, Keras e Numpy

Profundidade de memória $M=3$

7 perceptrons na camada oculta

85 parâmetros por TLP

170 parâmetros no total

Model: "MLP parte real"

Layer (type)	Output Shape	Param #
parte_real (Functional)	(None, 1)	85

Total params: 85

Trainable params: 85

Non-trainable params: 0

Model: "MLP parte imaginária"

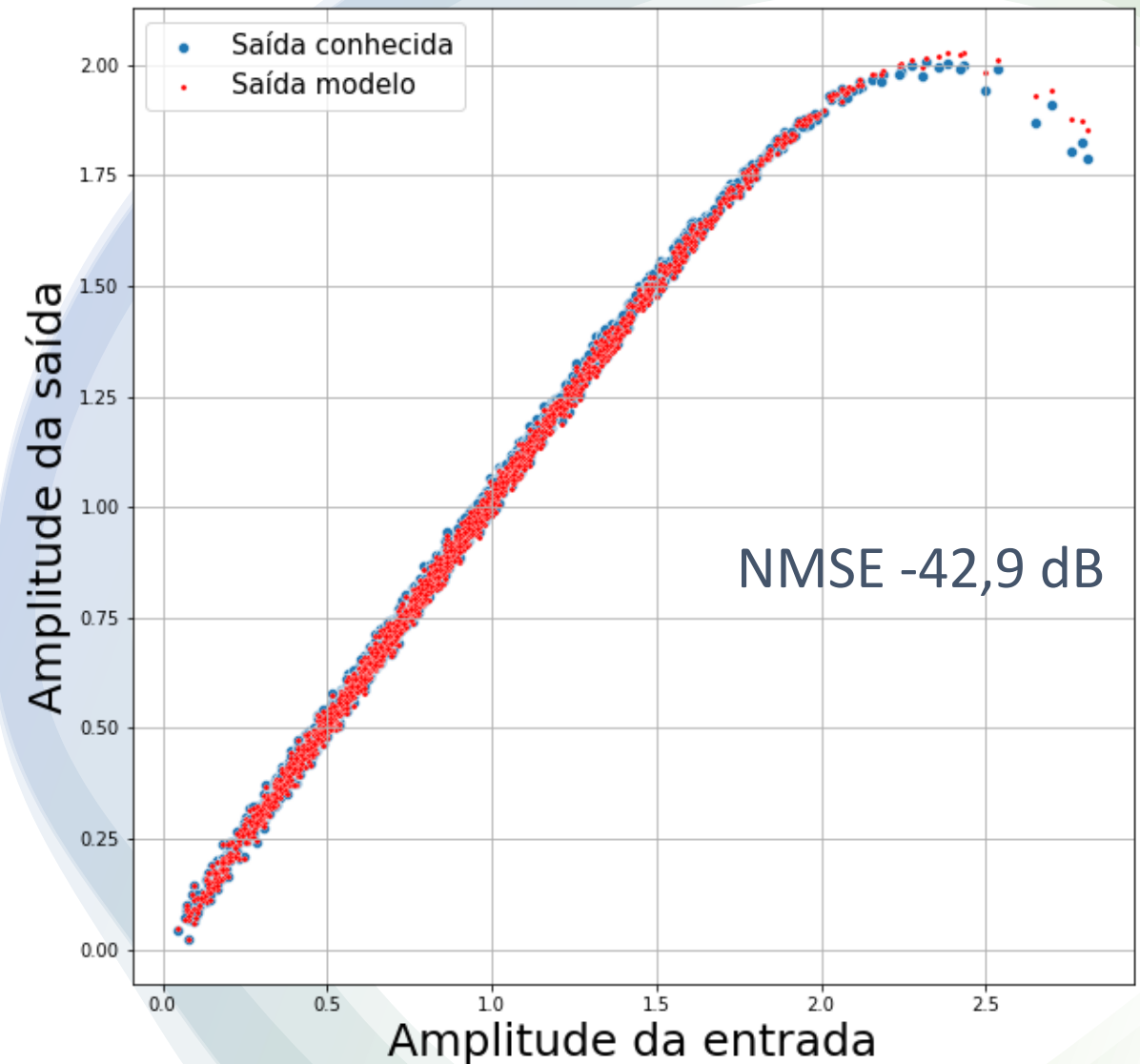
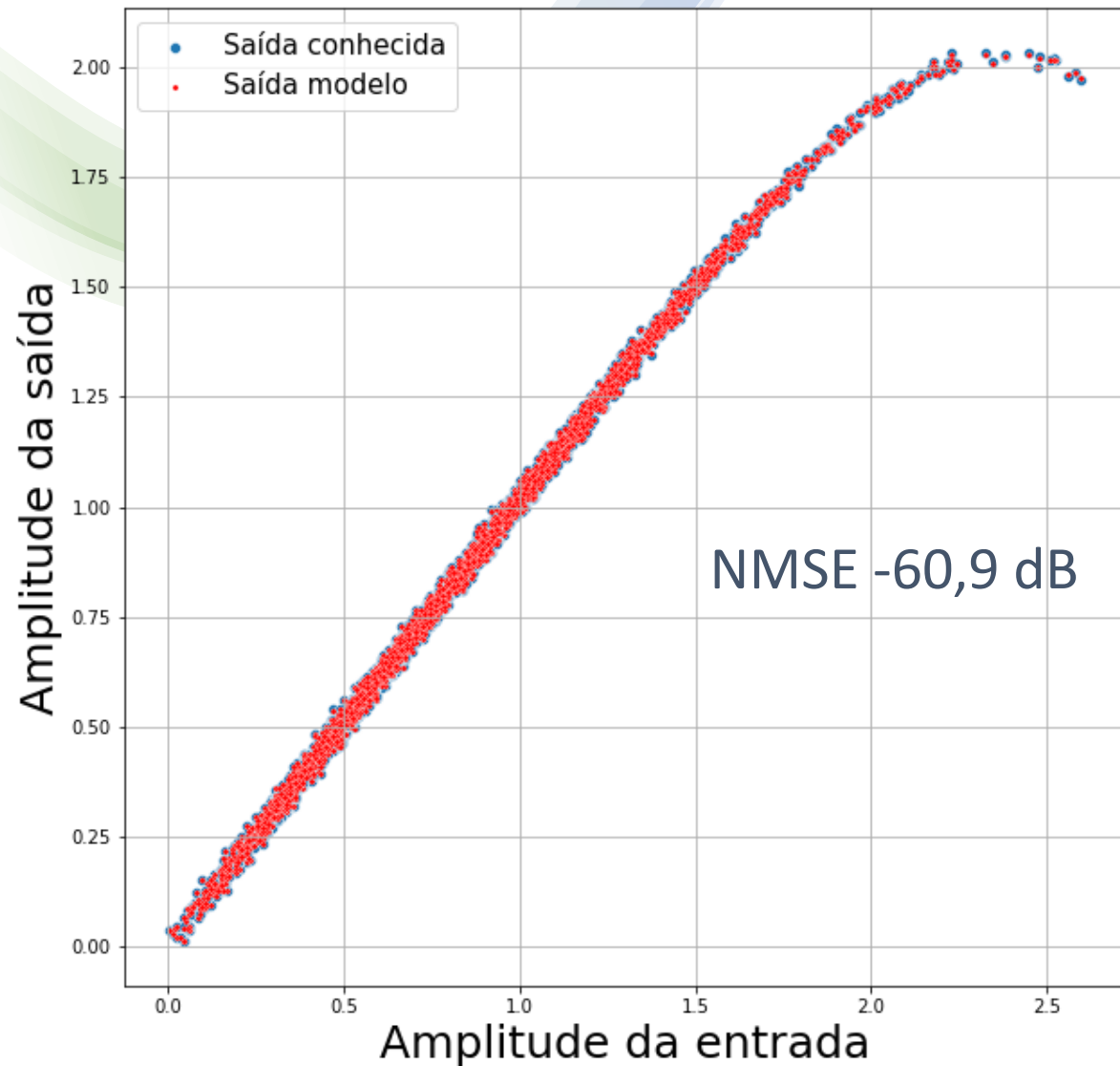
Layer (type)	Output Shape	Param #
parte_imag (Functional)	(None, 1)	85

Total params: 85

Trainable params: 85

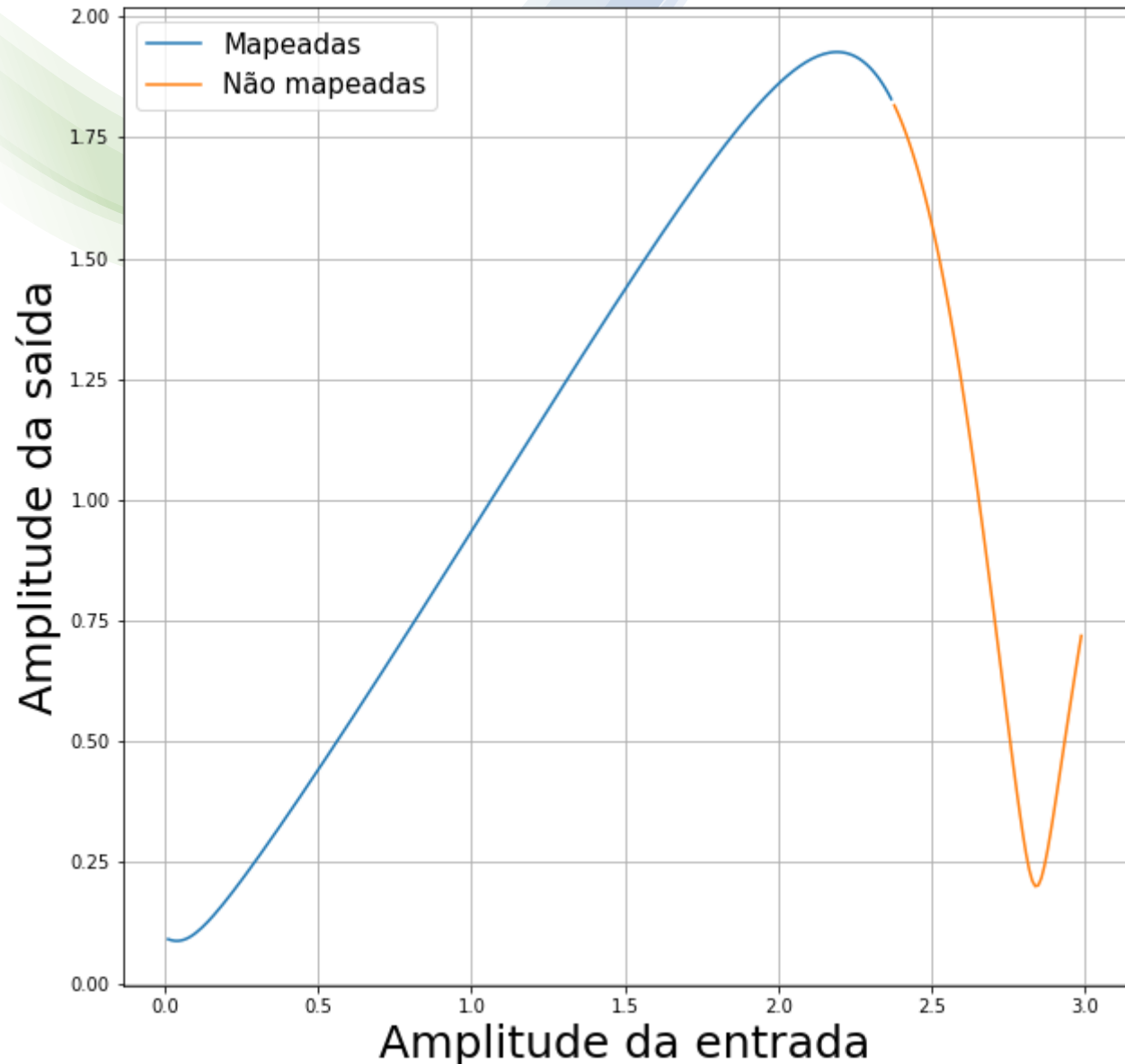
Non-trainable params: 0

Estudo de caso - Modelagem do PA



Estudo de caso - Análise dos resultados

Efeitos da continuidade da NN - Memória fixa e entrada variável



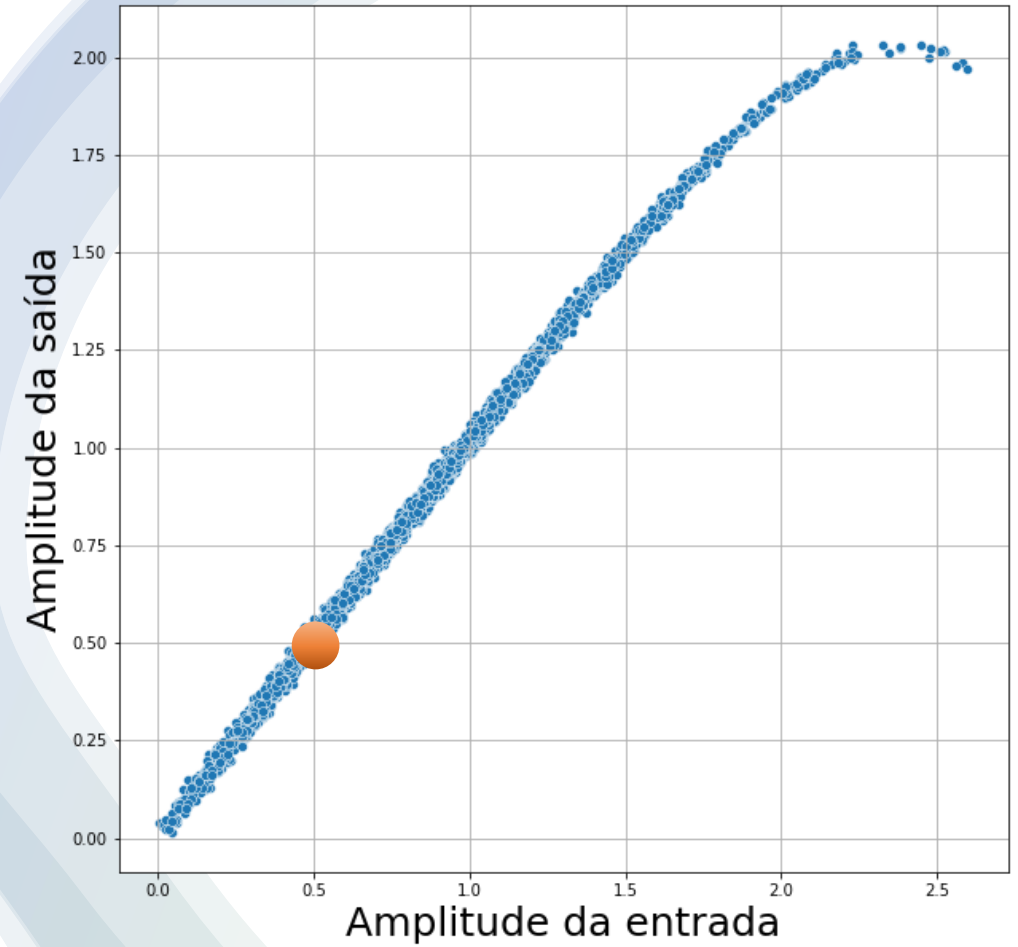
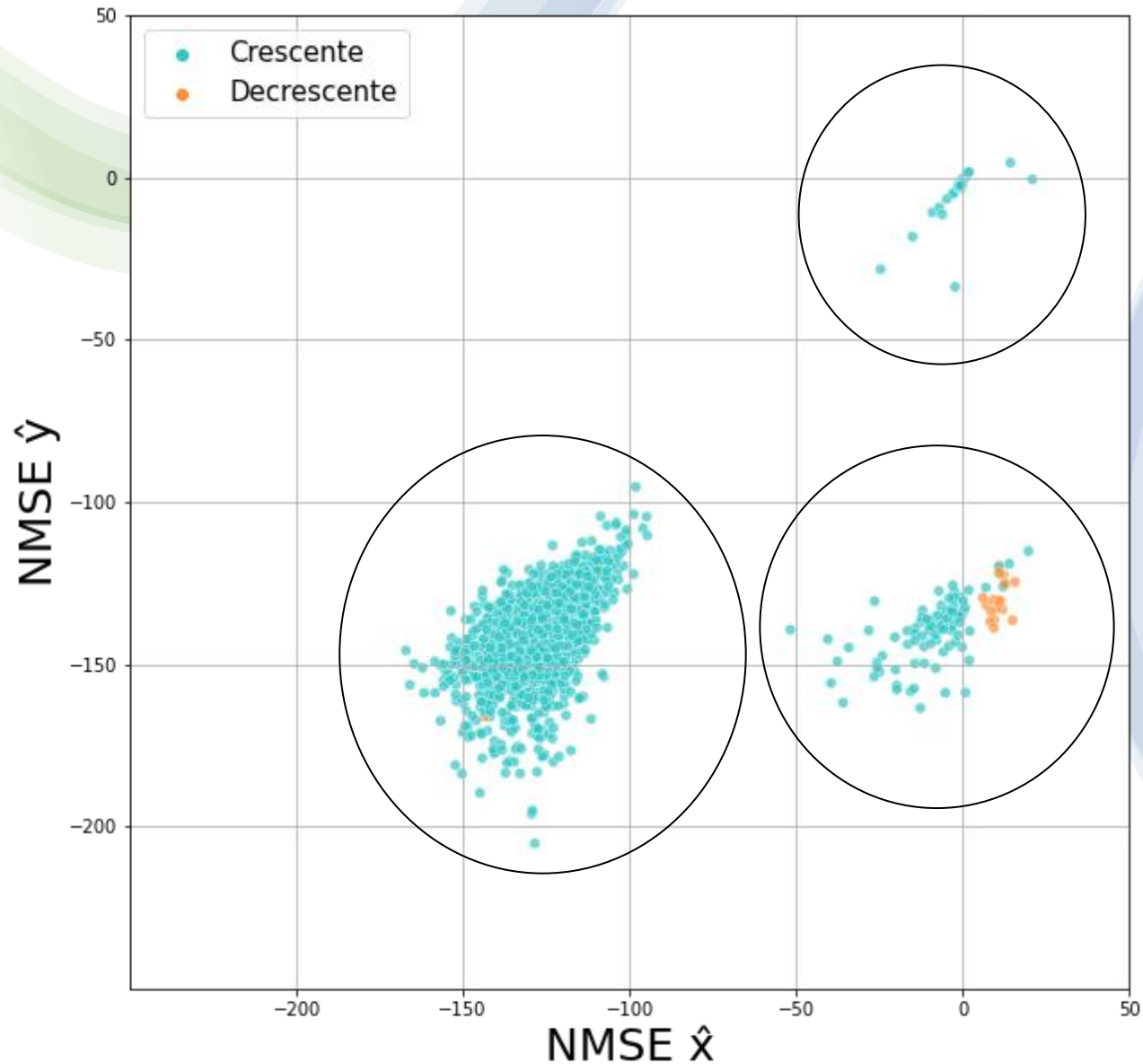
As ANN generalizam o comportamento da função para uma região que não mapeada.

E mesmo para entradas de menor uma região decrescente e não mapeada pode ser formada.

Foi penalizada essa região durante o processo de otimização.

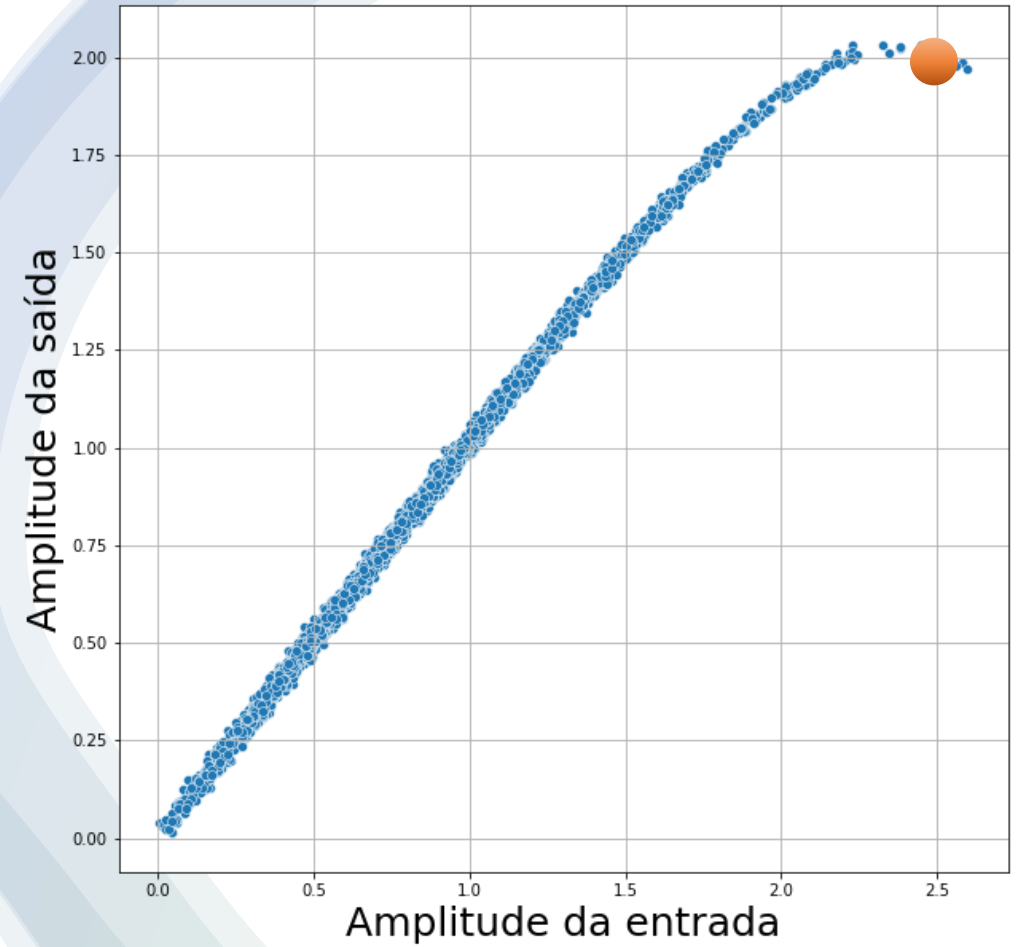
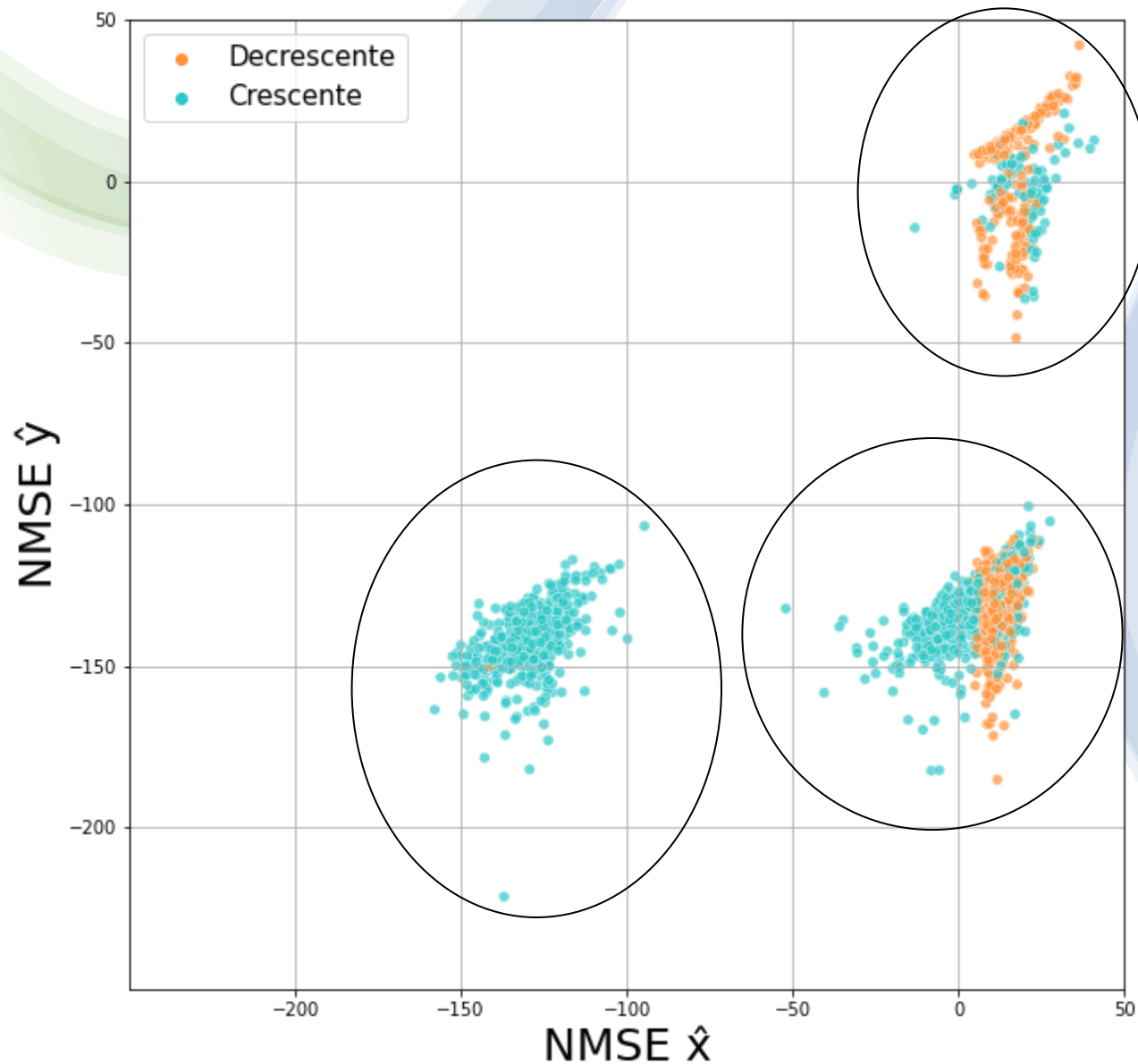
Estudo de caso - Análise dos resultados

Efeitos dos valores iniciais na solução do PI - Amplitude 0,5



Estudo de caso - Análise dos resultados

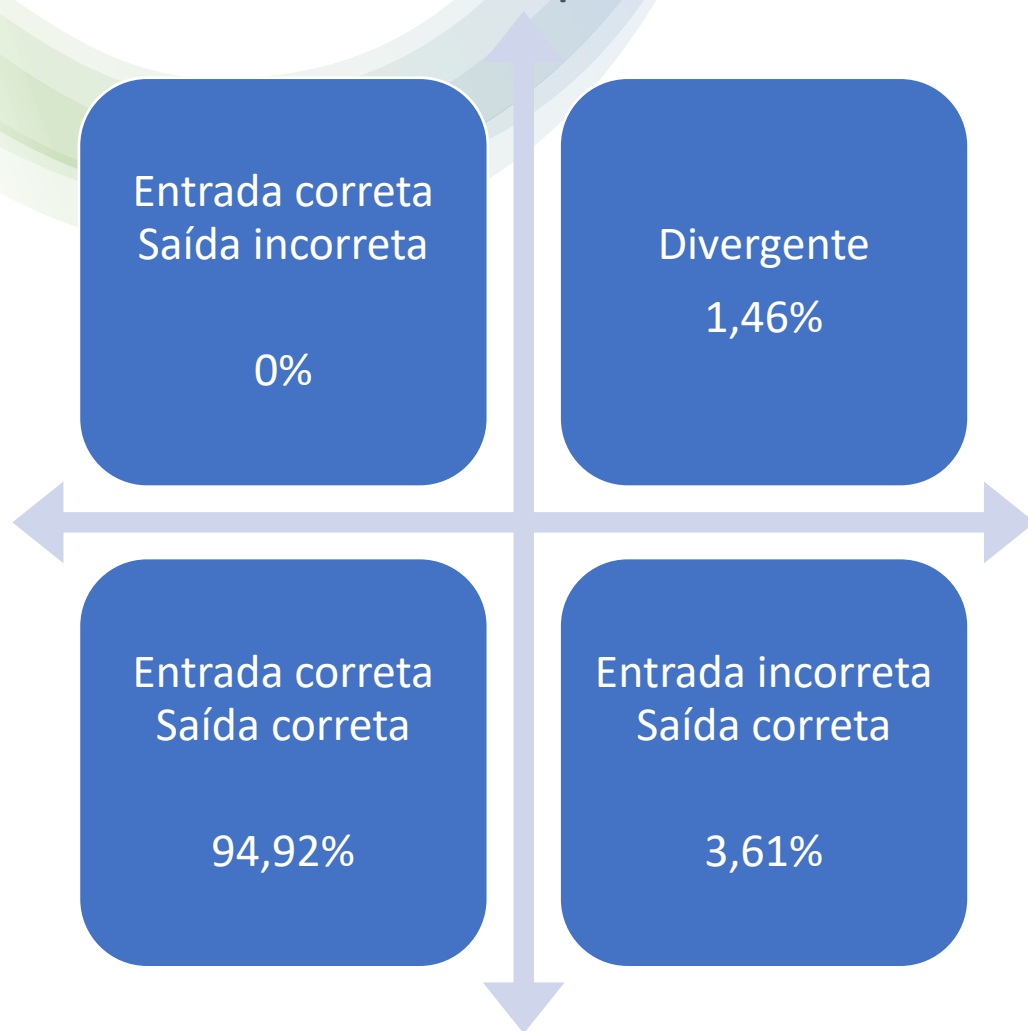
Efeitos dos valores iniciais na solução do PI - Amplitude 2,5



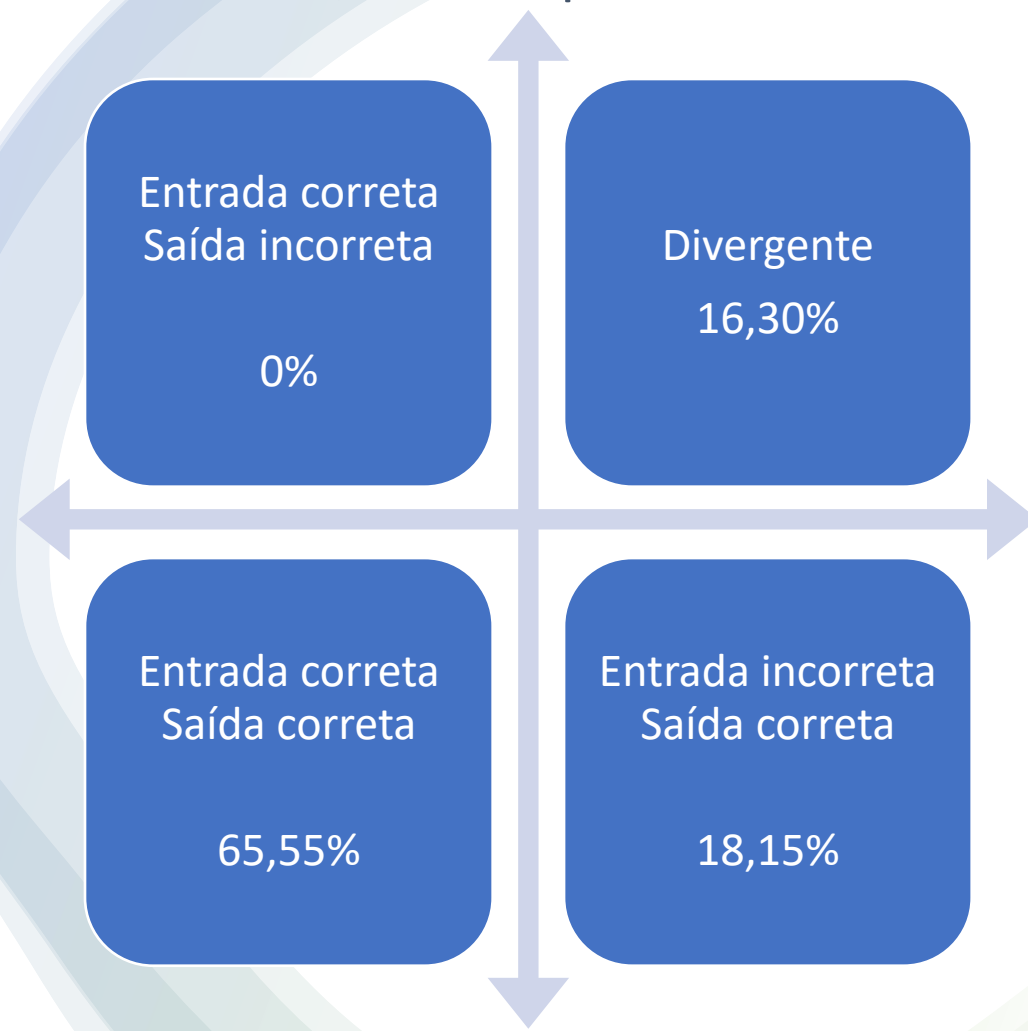
Estudo de caso - Classificação dos resultados

Distribuição por amplitudes

0,5 de amplitude



2,5 de amplitude



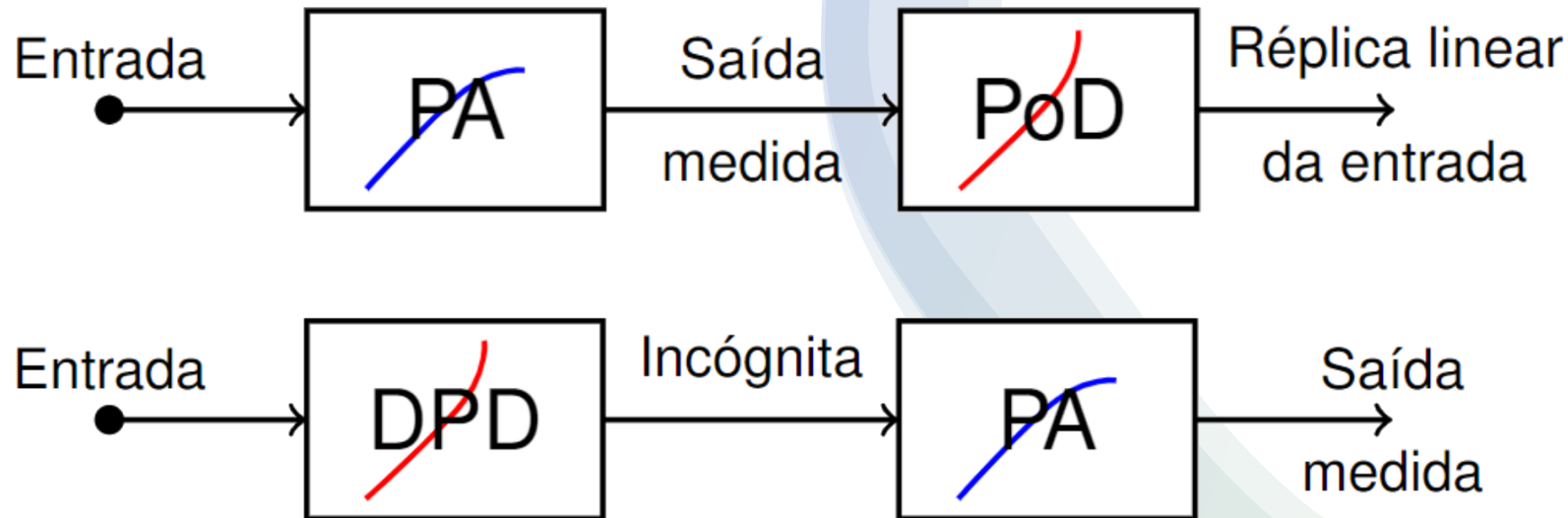
Conclusão parcial

O problema inverso se mostra uma ferramenta promissora para a análise comportamental dos modelos de PAs.

Servindo como ferramenta para a extração de métricas que informam sobre o comportamento do modelo.

Na segunda parte desse trabalho será realizada a validação da pré-inversa do PA através da utilização do problema inverso.

E futuramente conectar a complexidade da inversa do PA a não-linearidade do PA por meio do problema inverso.



Cronograma

