



UNIVERSIDADE  
ESTADUAL DO CEARÁ

# Produto Educacional ART - 2

## Disciplina de Redes Neurais Artificiais

### ART 2: self-organization of stable category recognition codes for analog input patterns

Gail A. Carpenter and Stephen Grossberg

Adaptive resonance architectures are neural networks that self-organize stable pattern recognition codes in real-time in response to arbitrary sequences of input patterns. This article introduces ART 2, a class of adaptive resonance architectures which rapidly self-organize pattern recognition categories in response to arbitrary sequences of either analog or binary input patterns. In order to cope with arbitrary sequences of analog input patterns, ART 2 architectures embody solutions to a number of design principles, such as the stability-plasticity tradeoff, the search-direct access tradeoff, and the match-reset tradeoff. In these architectures, top-down learned expectation and matching mechanisms are critical in self-stabilizing the code learning process. A parallel search scheme updates itself adaptively as the learning process unfolds, and realizes a form of real-time hypothesis discovery, testing, learning, and recognition. After learning self-stabilizes, the search process is automatically disengaged. Thereafter input patterns directly access their recognition codes without any search. Thus recognition time for familiar inputs does not increase with the complexity of the learned code. A novel input pattern can directly access a category if it shares invariant properties with the set of familiar exemplars of that category. A parameter called the attentional vigilance parameter determines how fine the categories will be. If vigilance increases (decreases) due to environmental feedback, then the system automatically searches for and learns finer (coarser) recognition categories. Gain control parameters enable the architecture to suppress noise up to a prescribed level. The architecture's global design enables it to learn effectively despite the high degree of nonlinearity of such mechanisms.

#### I. Adaptive Resonance Architectures

Adaptive resonance architectures are neural networks that self-organize stable recognition codes in real time in response to arbitrary sequences of input patterns. The basic principles of adaptive resonance theory (ART) were introduced by Grossberg.<sup>1</sup> A class of adaptive resonance architectures, called ART 1, has since been characterized as a system of ordinary differential equations by Carpenter and Grossberg.<sup>2,3</sup> Theorems have been proved that trace the real-time dynamics of ART 1 networks in response to arbitrary sequences of binary input patterns. These theorems predict both the order of search, as a function of the learning history of the network, and the asymptotic category structure self-organized by arbitrary sequences of binary input patterns. They also prove the self-stabilization property and show that the system's adaptive weights oscillate at most once, yet do not get trapped in spurious memory states or local minima.

This paper describes a new class of adaptive resonance architectures, called ART 2. ART 2 networks self-organize stable recognition categories in response to arbitrary sequences of analog (gray-scale, continuous-valued) input patterns, as well as binary input patterns. Computer simulations are used to illustrate system dynamics. One such simulation is summarized in Fig. 1, which shows how a typical ART 2 architecture has quickly learned to group fifty inputs into thirty-four stable recognition categories after a single presentation of each input. The plots below each number show all those input patterns ART 2 has grouped into the corresponding category. Equations for the system used in the simulation are given in Secs. V-VIII.

ART networks encode new input patterns, in part, by changing the weights, or long-term memory (LTM) traces, of a bottom-up adaptive filter (Fig. 2). This filter is contained in pathways leading from a feature representation field ( $F_1$ ) to a category representation field ( $F_2$ ) whose nodes undergo cooperative and competitive interactions. Such a combination of adaptive filtering and competition, sometimes called competitive learning, is shared by many other models of adaptive pattern recognition and associative learning. See Grossberg<sup>4</sup> for a review of the development of competitive learning models. In an ART network, however, it

The authors are with Boston University, Center for Adaptive Systems, 111 Cummington Street, Boston, Massachusetts 02215.

Received 11 June 1987.

0003-6935/87/234919-12\$02.00/0.



# Introdução ao Produto Educacional

- **Objetivo Geral:** Explorar a teoria e a prática associadas a ART-2, sua estrutura, princípio de ressonância adaptativa, aprendizado e algoritmo. O produto final incluirá pílulas de vídeo didáticas, infográficos e um projeto prático.



# Introdução ao Produto Educacional

- **Formato:**
  - Pílulas de vídeo curtas e didáticas.
  - Slides explicativos.
  - Projeto prático com código comentado em Python.

# Introdução às Redes ART-2 e Comparação com ART-1

Uma Jornada pelas Redes Neurais Adaptativas

# O que são Redes ART?

As Redes de **Ressonância Adaptativa** são um tipo de rede neural artificial não supervisionada desenvolvida por **Stephen Grossberg e Gail Carpenter**. Elas se destacam por sua capacidade de aprendizado incremental, estabilidade-plasticidade e agrupamento de padrões sem a necessidade de definir o número de clusters a priori.

Existem diversas variantes de redes ART, cada uma projetada para lidar com diferentes tipos de dados de entrada.

## ART 2: self-organization of stable category recognition codes for analog input patterns

Gail A. Carpenter and Stephen Grossberg

Adaptive resonance architectures are neural networks that self-organize stable pattern recognition codes in real-time in response to arbitrary sequences of input patterns. This article introduces ART 2, a class of adaptive resonance architectures which rapidly self-organize pattern recognition categories in response to arbitrary sequences of either analog or binary input patterns. In order to cope with arbitrary sequences of analog input patterns, ART 2 architectures embody solutions to a number of design principles, such as the stability-plasticity tradeoff, the search-direct access tradeoff, and the match-reset tradeoff. In these architectures, top-down learned expectation and matching mechanisms are critical in self-stabilizing the code learning process. A parallel search scheme updates itself adaptively as the learning process unfolds, and realizes a form of real-time hypothesis discovery, testing, learning, and recognition. After learning self-stabilizes, the search process is automatically disengaged. Thereafter input patterns directly access their recognition codes without any search. Thus recognition time for familiar inputs does not increase with the complexity of the learned code. A novel input pattern can directly access a category if it shares invariant properties with the set of familiar exemplars of that category. A parameter called the attentional vigilance parameter determines how fine the categories will be. If vigilance increases (decreases) due to environmental feedback, then the system automatically searches for and learns finer (coarser) recognition categories. Gain control parameters enable the architecture to suppress noise up to a prescribed level. The architecture's global design enables it to learn effectively despite the high degree of nonlinearity of such mechanisms.

### I. Adaptive Resonance Architectures

Adaptive resonance architectures are neural networks that self-organize stable recognition codes in real time in response to arbitrary sequences of input patterns. The basic principles of adaptive resonance theory (ART) were introduced by Grossberg.<sup>1</sup> A class of adaptive resonance architectures, called ART 1, has since been characterized as a system of ordinary differential equations by Carpenter and Grossberg.<sup>2,3</sup> Theorems have been proved that trace the real-time dynamics of ART 1 networks in response to arbitrary sequences of binary input patterns. These theorems predict both the order of search, as a function of the learning history of the network, and the asymptotic category structure self-organized by arbitrary sequences of binary input patterns. They also prove the self-stabilization property and show that the system's adaptive weights oscillate at most once, yet do not get trapped in spurious memory states or local minima.

This paper describes a new class of adaptive resonance architectures, called ART 2. ART 2 networks self-organize stable recognition categories in response to arbitrary sequences of analog (gray-scale, continuous-valued) input patterns, as well as binary input patterns. Computer simulations are used to illustrate system dynamics. One such simulation is summarized in Fig. 1, which shows how a typical ART 2 architecture has quickly learned to group fifty inputs into thirty-four stable recognition categories after a single presentation of each input. The plots below each number show all those input patterns ART 2 has grouped into the corresponding category. Equations for the system used in the simulation are given in Secs. V-VIII.

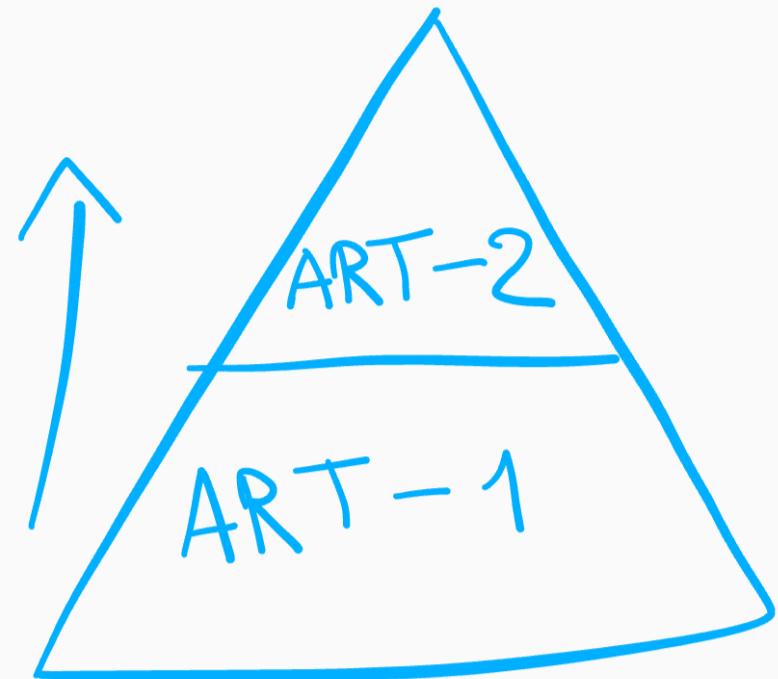
ART networks encode new input patterns, in part, by changing the weights, or long-term memory (LTM) traces, of a bottom-up adaptive filter (Fig. 2). This filter is contained in pathways leading from a feature representation field ( $F_1$ ) to a category representation field ( $F_2$ ) whose nodes undergo cooperative and competitive interactions. Such a combination of adaptive filtering and competition, sometimes called competitive learning, is shared by many other models of adaptive pattern recognition and associative learning. See Grossberg<sup>4</sup> for a review of the development of competitive learning models. In an ART network, however, it is a second, top-down adaptive filter that leads to the

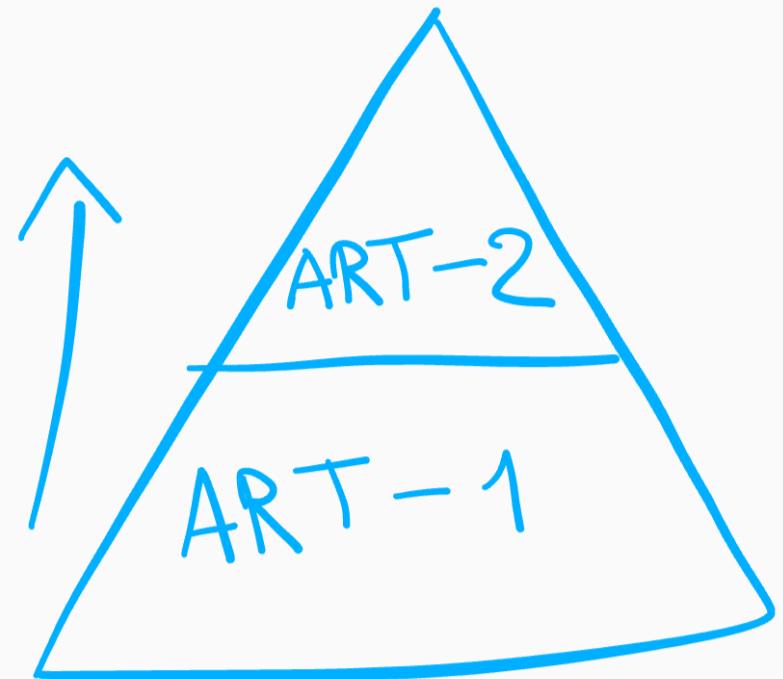
The authors are with Boston University, Center for Adaptive Systems, 111 Cummington Street, Boston, Massachusetts 02215.  
Received 11 June 1987.  
0003-6935/87/234919-12\$02.00/0.  
© 1987 Optical Society of America.

# ART-1: A Base para Dados Binários

A ART-1, a primeira variante da família ART, foi projetada para processar entradas binárias (0 ou 1). Ela opera com base em dois subsistemas principais:

- **Camada de Comparação (F1):** Recebe a entrada e a compara com os protótipos existentes na camada de reconhecimento.
- **Camada de Reconhecimento (F2):** Armazena os protótipos dos clusters aprendidos.





## ART-1: A Base para Dados Binários

A ART-1 utiliza um parâmetro crucial chamado **vigilância**, que controla a similaridade mínima necessária para um padrão de entrada ser associado a um cluster existente. Se a similaridade for inferior à vigilância, um novo cluster é criado.

## Limitações da ART-1 e a Necessidade da ART-2

Apesar de suas vantagens, a ART-1 apresenta uma limitação significativa: ela só pode processar dados binários. Em muitos cenários do mundo real, os dados são contínuos, o que motivou o desenvolvimento da ART-2.

```
<iframe id="dede" src="https://www.desmos.com/calculator/gugok9tmrq?embed"  
width="100%" height="500" frameborder=0></iframe>
```

## ART-2: Expandindo para Dados Contínuos

A ART-2 foi projetada para superar a limitação da ART-1, permitindo o processamento de entradas contínuas. Para isso, ela incorpora mecanismos de normalização e pré-processamento dos dados antes da comparação com os protótipos. A arquitetura da ART-2 é mais complexa que a da ART-1, incluindo etapas adicionais para lidar com ruído e garantir a estabilidade do aprendizado.

# Comparação ART-1 vs. ART-2

Característica	ART-1	ART-2
<b>Tipo de Entrada</b>	Binária	Contínua
<b>Complexidade</b>	Menor	Maior
<b>Pré-processamento</b>	Não necessário	Necessário (normalização, supressão de ruído)
<b>Parâmetros</b>	Vigilância	Vigilância, outros parâmetros de pré-processamento
<b>Aplicações</b>	Dados binários, reconhecimento de imagens simples	Dados contínuos, processamento de sinais, reconhecimento de padrões complexos

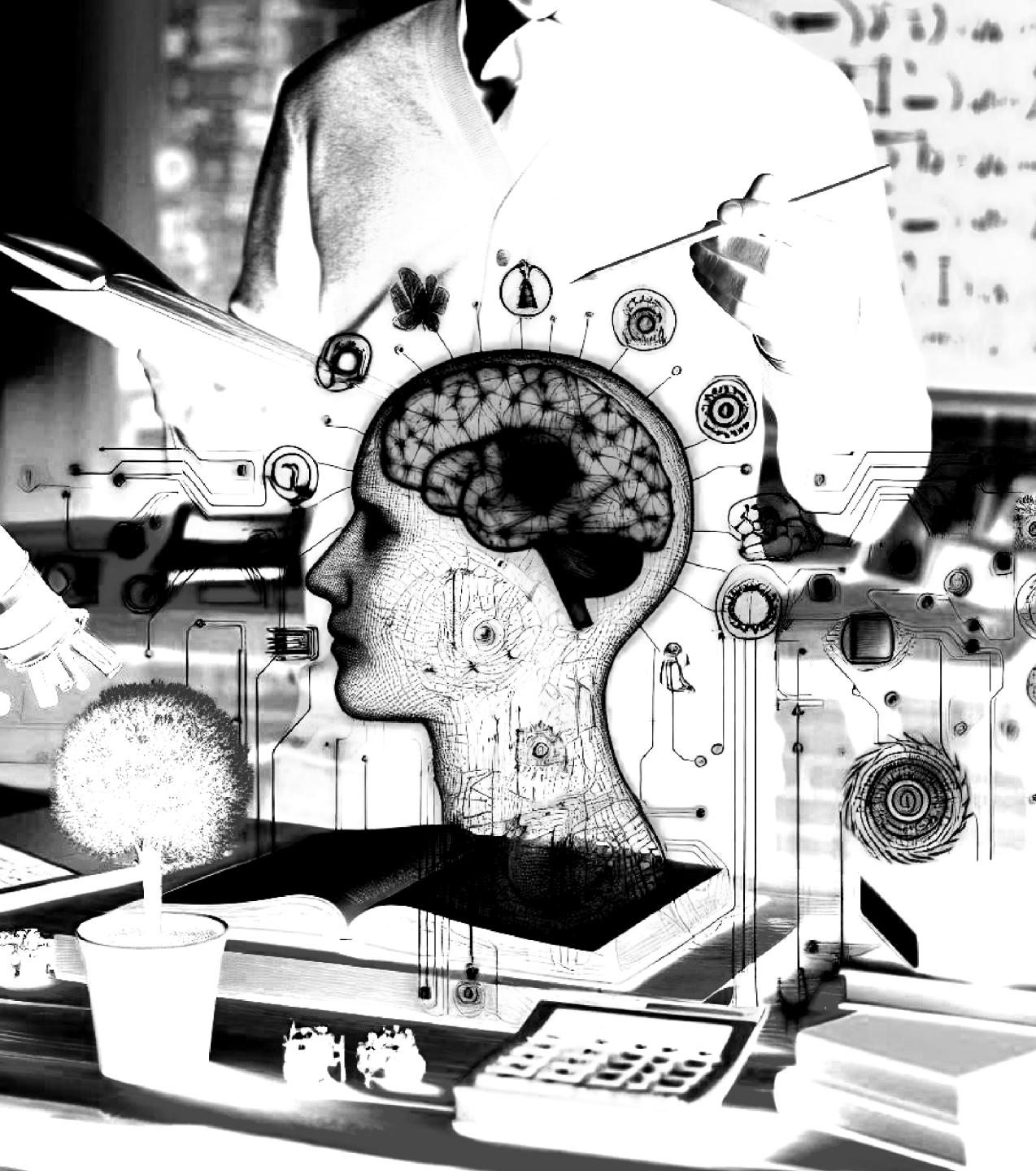
# ART-2-A: Uma Simplificação Eficiente

A ART-2-A é uma versão simplificada da ART-2 que mantém a capacidade de lidar com entradas contínuas, mas com uma arquitetura menos complexa e menor custo computacional. A ART-2-A se torna uma alternativa interessante em cenários onde a performance é crucial.



# Conclusão

As redes ART, em suas diferentes variantes, oferecem uma abordagem interessante para o aprendizado não supervisionado. A ART-1, com sua simplicidade, é adequada para dados binários, enquanto a ART-2 e a ART-2-A estendem essa capacidade para dados contínuos, embora com maior complexidade. A escolha da variante adequada depende das características dos dados e dos requisitos da aplicação. É importante lembrar que, apesar de suas vantagens, as redes ART também possuem limitações, como a sensibilidade aos parâmetros e a dificuldade de lidar com datasets muito grandes, como explorado no documento anterior.



# Estrutura Topológica, Princípios e Algoritmo de Treinamento da Rede ART-2

**Desvendando a  
Arquitetura da ART-2**

# Topologia da ART-2

Reset

R

S<sub>1</sub>

S<sub>i</sub>

S<sub>n</sub>

X<sub>1</sub>

X<sub>i</sub>

X<sub>n</sub>

Y<sub>1</sub>

Y<sub>i</sub>

Y<sub>n</sub>

b<sub>ij</sub>

t<sub>ij</sub>

Input  
F1(a)

Interface  
F1(a)

Cluster  
F2<sup>14</sup>



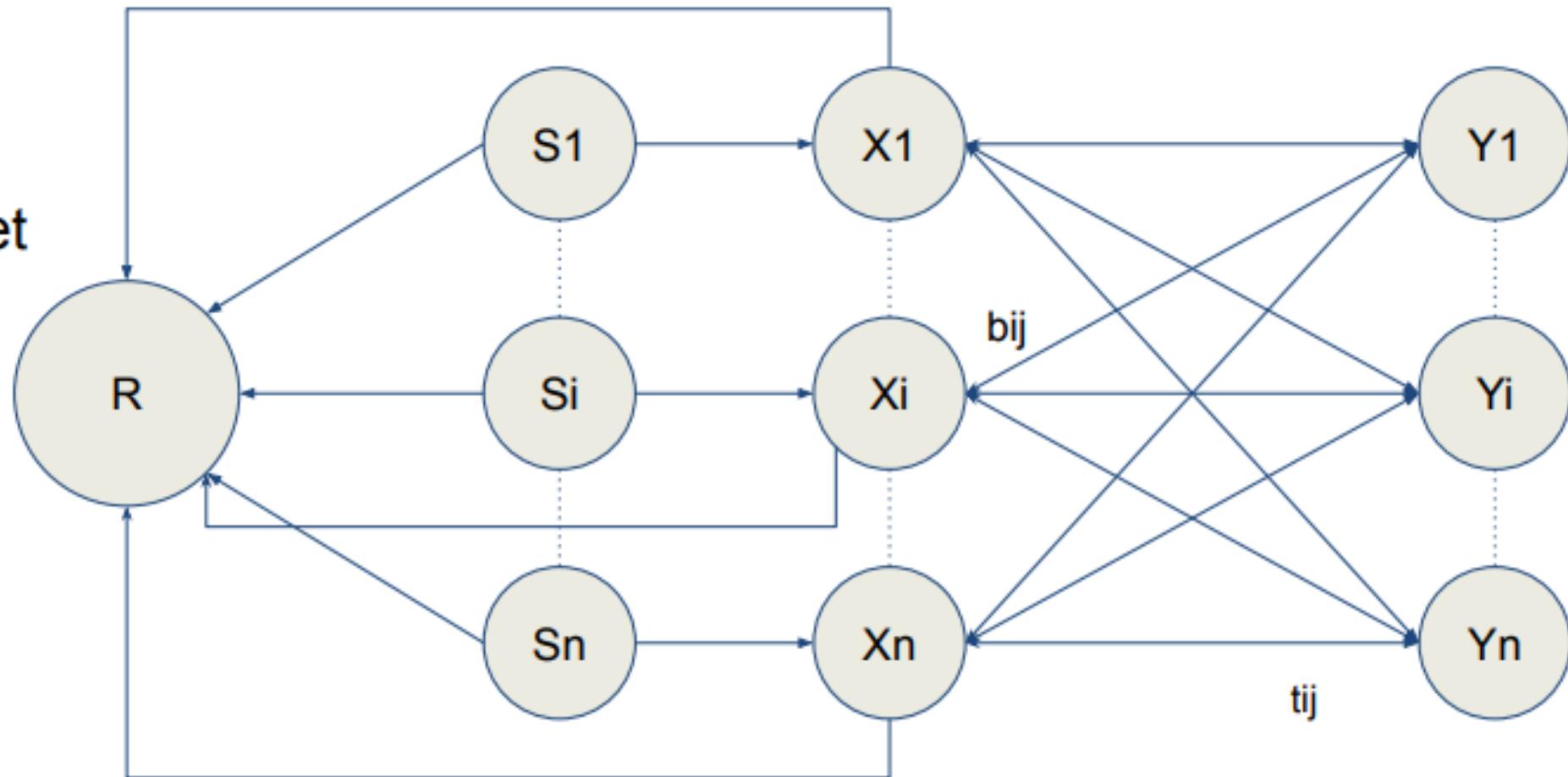
# Topologia da ART-2

A ART-2 possui uma estrutura mais complexa que a ART-1, projetada para processar entradas contínuas. Ela é composta pelos seguintes componentes principais:

- **Camada de Entrada (F0):** Recebe o vetor de entrada  $I$  e o normaliza.
- **Camada de Comparação (F1):** Compara a entrada normalizada com os protótipos existentes na camada de reconhecimento (F2).
- **Camada de Reconhecimento (F2):** Armazena os protótipos dos clusters. Cada nó em F2 representa um cluster.
- **Mecanismo de Reset:** Responsável por controlar a criação de novos clusters quando a similaridade entre a entrada e os protótipos existentes é insuficiente.

Além dessas camadas principais, a ART-2 inclui diversos componentes intermediários que realizam operações de normalização, supressão de ruído e comparação.

Reset



Input  
F1(a)

Interface  
F1(a)

Cluster  
F2

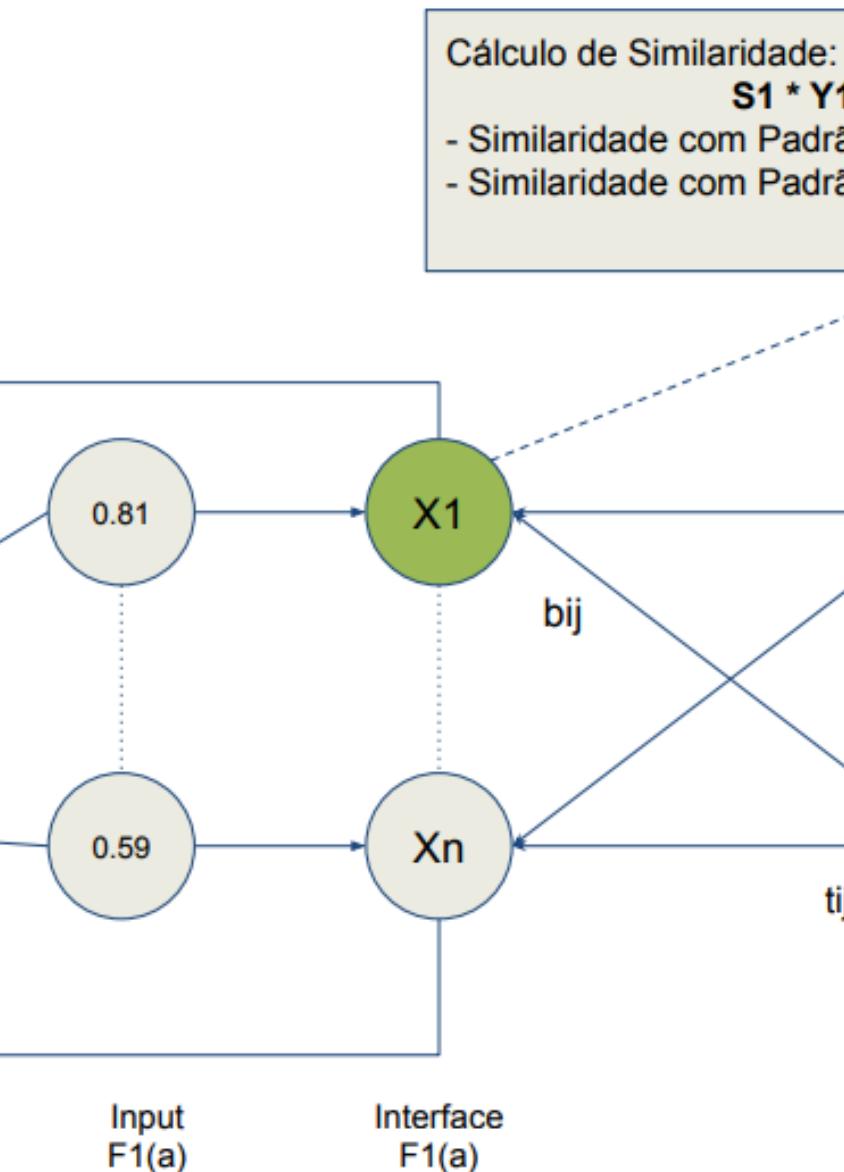
# Princípios de Funcionamento

AART-2 opera com base nos seguintes princípios:

- **Ressonância:** Quando um padrão de entrada é apresentado, a rede busca um protótipo em F2 que seja suficientemente similar. Se a similaridade for alta o suficiente (controlada pelo parâmetro de vigilância), ocorre a ressonância, e o protótipo é atualizado para se aproximar ainda mais da entrada.
- **Reset:** Se nenhum protótipo existente for suficientemente similar à entrada, o mecanismo de reset é ativado, e um novo cluster é criado com a entrada como protótipo.
- **Aprendizado Incremental:** A ART-2 aprende de forma incremental, ajustando os protótipos a cada nova entrada sem esquecer os padrões aprendidos anteriormente.
- **Estabilidade-Plasticidade:** O parâmetro de vigilância controla o equilíbrio entre estabilidade (manutenção dos clusters existentes) e plasticidade (capacidade de criar novos clusters).

# Algoritmo de Treinamento

O algoritmo de treinamento da ART-2 pode ser resumido nos seguintes passos:



- 1. Inicialização:** Inicializar os pesos da rede e definir o parâmetro de vigilância ( $\rho$ ).
- 2. Apresentação da Entrada:** Apresentar um vetor de entrada  $I$ .
- 3. Normalização:** Normalizar o vetor de entrada em  $F_0$ .
- 4. Comparação:** Calcular a similaridade entre a entrada normalizada e os protótipos em  $F_2$ .

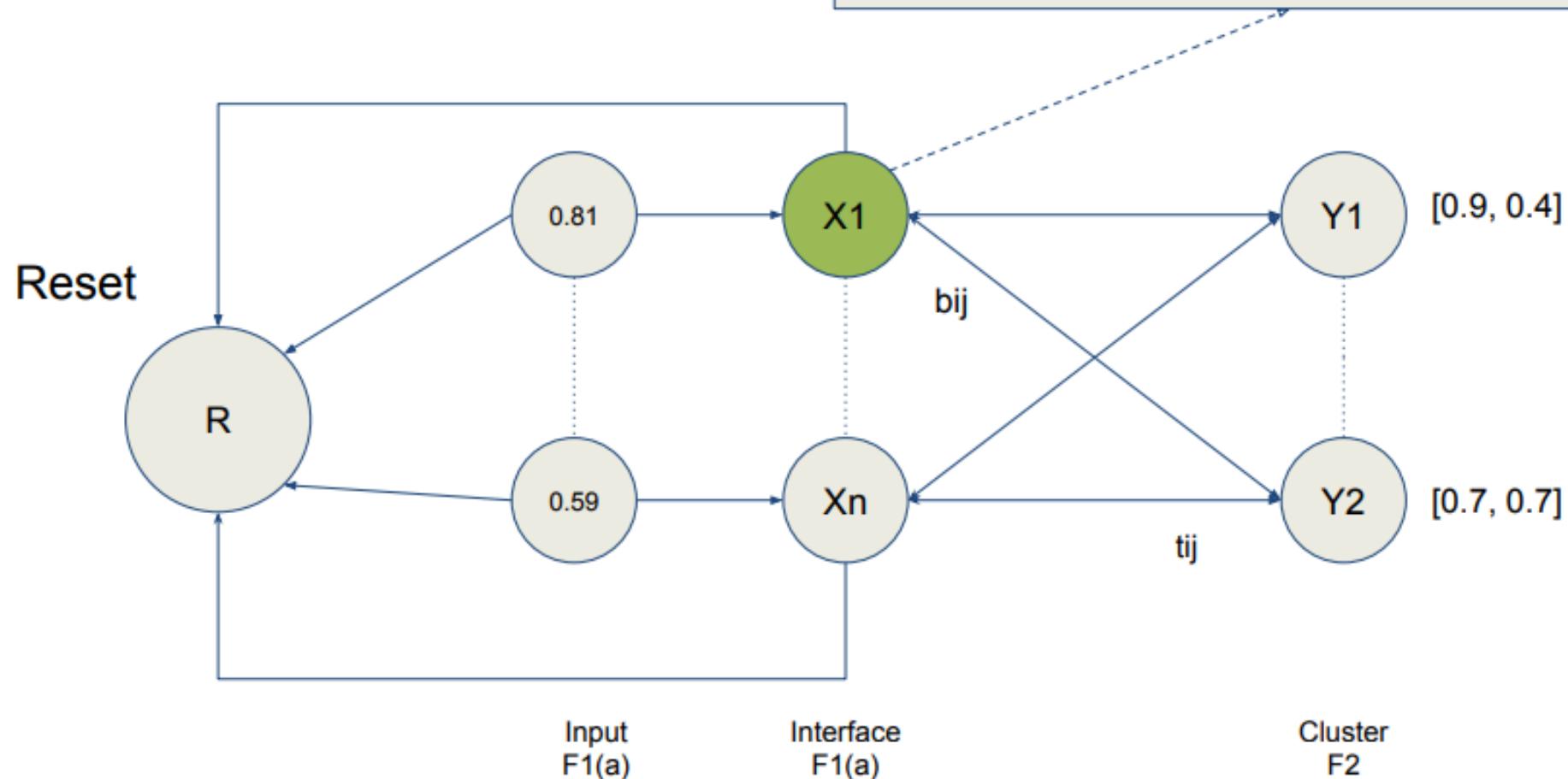
Padrões armazenados:  $Y_1 = [0.9, 0.4]$ ,  $Y_2 = [0.7, 0.7]$

Parâmetro de vigilância:  $p = 0.95$

Cálculo de Similaridade:

$$S_1 * Y_1 + S_2 * Y_2 + \dots + S_n * Y_n$$

- Similaridade com Padrão 1:  $0.81 * 0.9 + 0.59 * 0.4 = 0.93$
- Similaridade com Padrão 2:  $0.81 * 0.7 + 0.59 * 0.7 = 0.98$

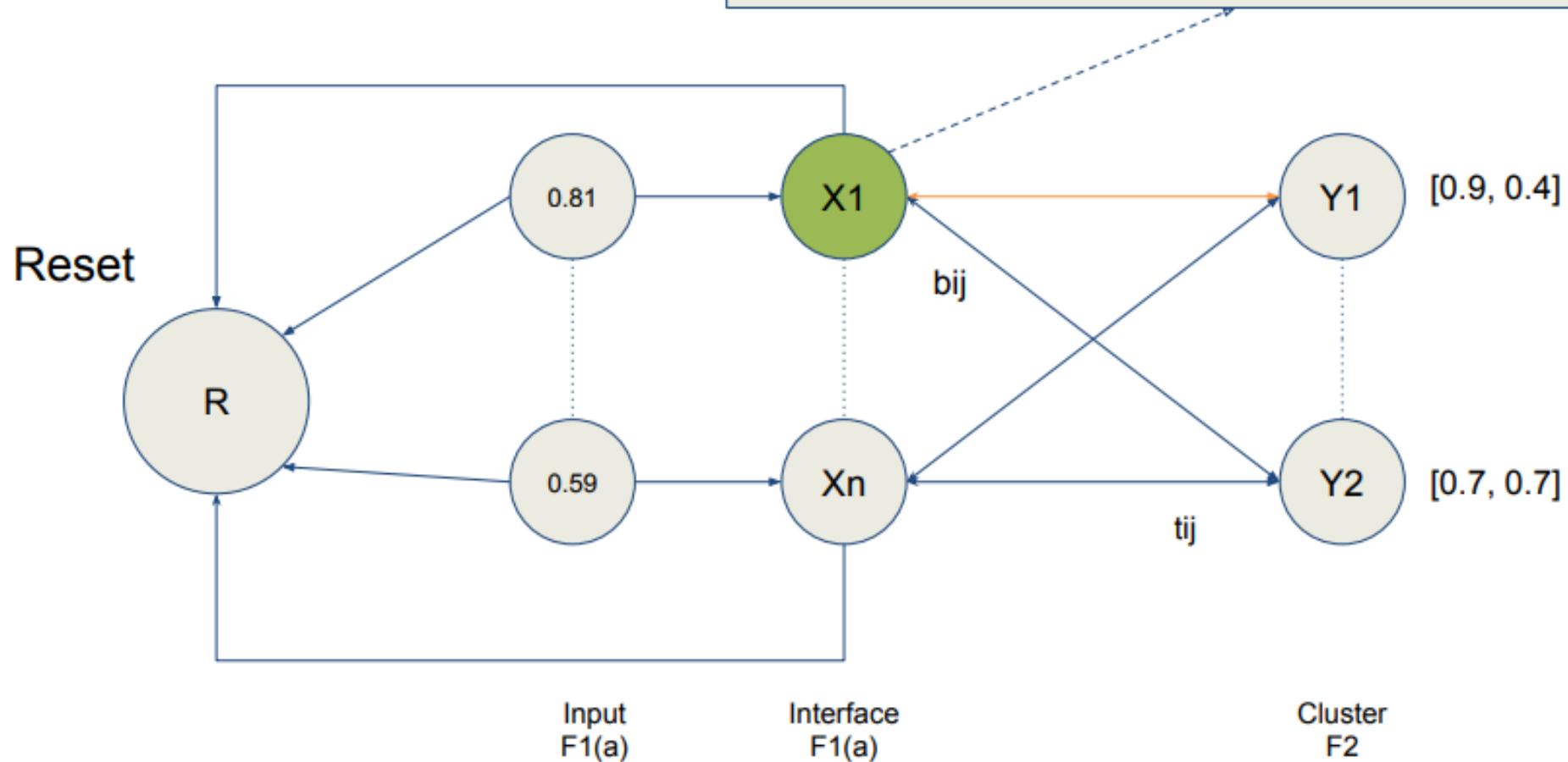


Padrões armazenados:  $Y_1 = [0.9, 0.4]$ ,  $Y_2 = [0.7, 0.7]$

Parâmetro de vigilância:  $p = 0.95$

Cálculo de Similaridade:

- Similaridade com Padrão 1:  $0.81 * 0.9 + 0.59 * 0.4 = 0.93$



## 5. Ressonância ou Reset:

- **Ressonância:** Se a similaridade for maior ou igual à vigilância ( $\rho$ ), atualizar o protótipo vencedor em F2 para se aproximar da entrada.
- **Reset:** Se a similaridade for menor que  $\rho$ , inibir o protótipo vencedor e repetir o processo de comparação com os demais protótipos. Se nenhum protótipo atender ao critério de vigilância, criar um novo cluster com a entrada como protótipo.

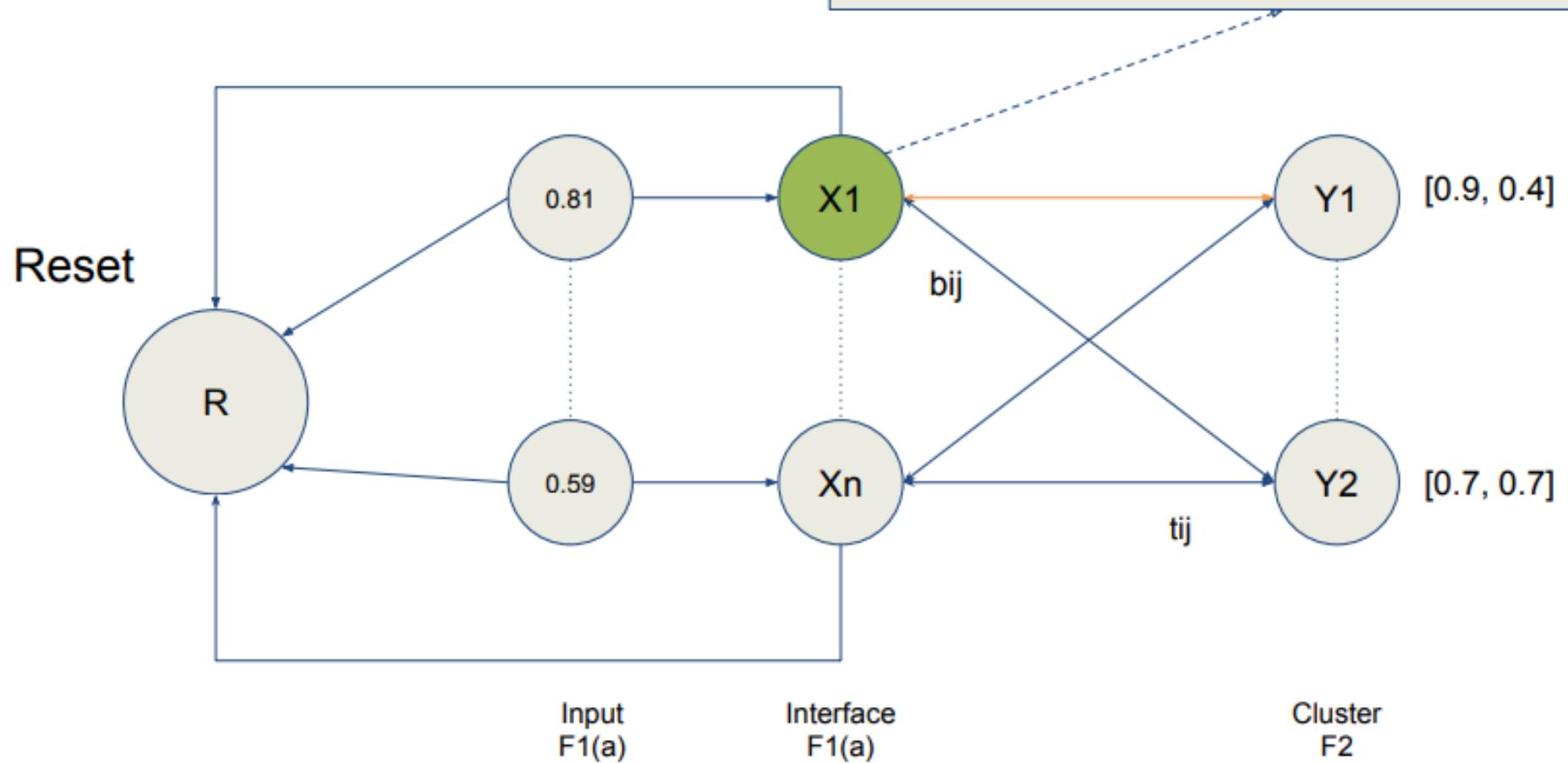
## 6. Repetir: Repetir os passos 2 a 5 para todos os vetores de entrada.

Padrões armazenados:  $Y_1 = [0.9, 0.4]$ ,  $Y_2 = [0.7, 0.7]$

Parâmetro de vigilância:  $\rho = 0.95$

0.93 é menor que o parâmetro de vigilância quando aplicado ao padrão  $Y_1$

$$0.93 < 0.95$$

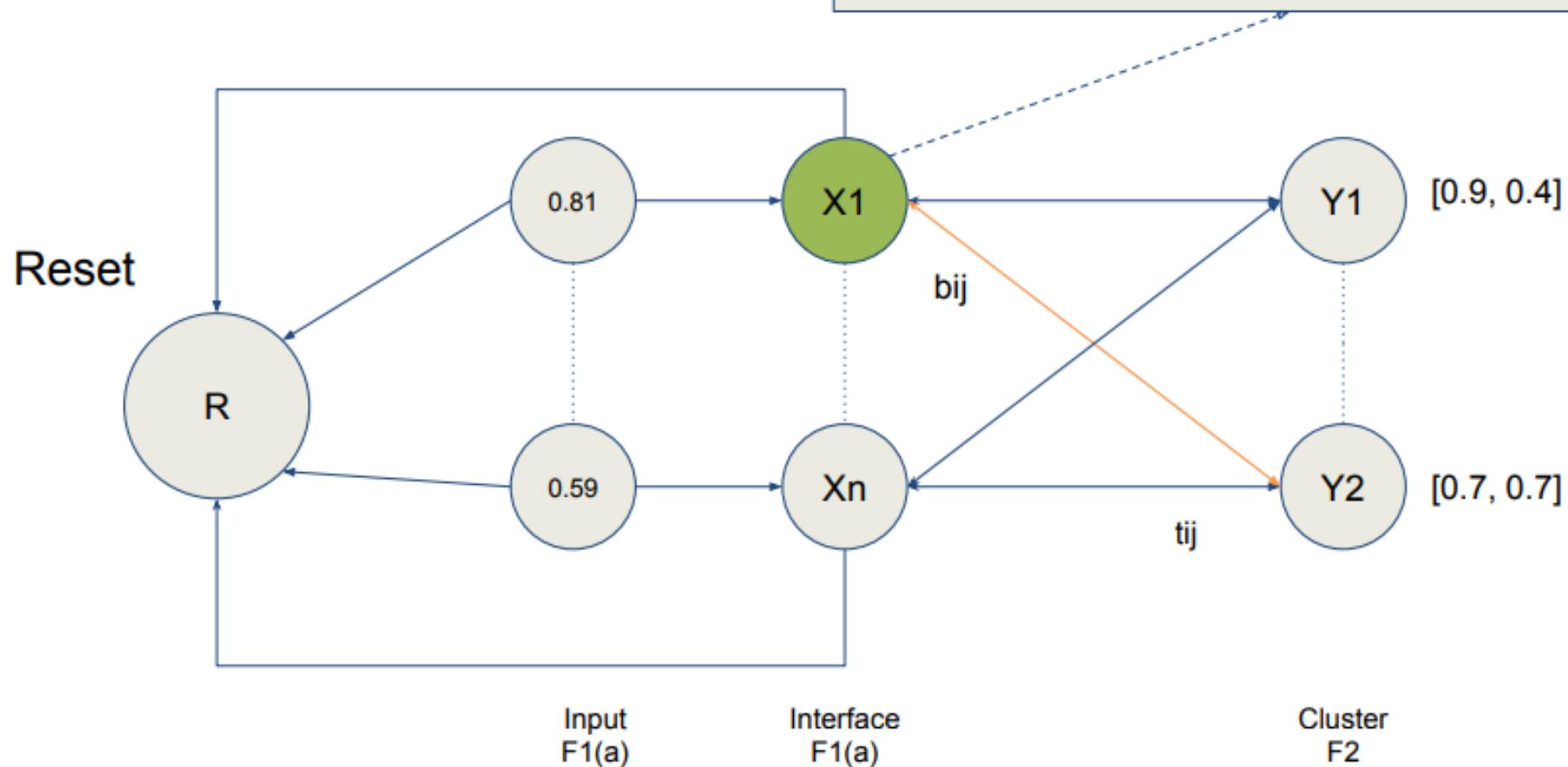


Padrões armazenados:  $Y_1 = [0.9, 0.4]$ ,  $Y_2 = [0.7, 0.7]$

Parâmetro de vigilância:  $p = 0.95$

0.98 é maior que o parâmetro de vigilância quando aplicado ao padrão  $Y_2$

$$0.98 > 0.95$$

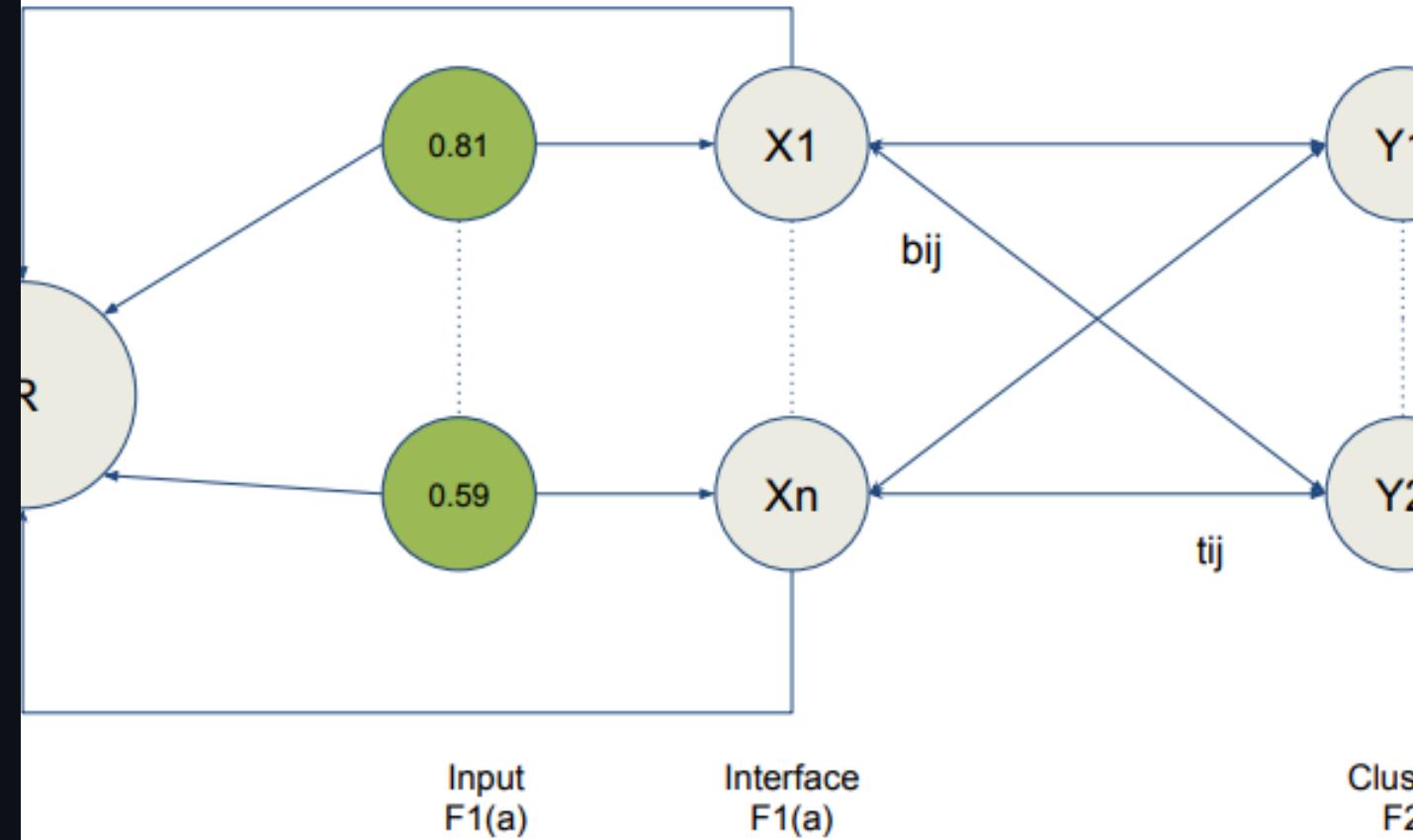


# Detalhes do Pré- processamento e Comparação (F1)

Camada F1 realiza pré-  
processamento e comparação,  
incluindo normalização,  
supressão de ruído e cálculo da  
similaridade entre a entrada e  
os protótipos. Estas operações,  
essenciais para o  
funcionamento da ART-2 com  
dados contínuos.

Protótipos:  $Y_1 = [0.9, 0.4]$ ,  $Y_2 = [0.7, 0.7]$   
Similaridade:  $\rho = 0.95$

## Valores normalizados



# Conclusão

A ART-2 é uma rede neural complexa, mas poderosa, para aprendizado não supervisionado com dados contínuos. Sua arquitetura, princípios de funcionamento e algoritmo de treinamento são fundamentais para entender seu comportamento e suas aplicações. A compreensão desses aspectos permite uma melhor utilização e ajuste da rede para diferentes problemas. A próxima seção explorará as vantagens e desvantagens da ART-2, bem como suas aplicações práticas.

# **Aspectos de Implementação Prática e Início do Projeto Prático**

**Colocando a ART-2 em Ação**

# Desafios da Implementação

Implementar a ART-2 na prática apresenta alguns desafios:

- **Complexidade do Algoritmo:** A ART-2 possui um algoritmo mais intrincado que outras redes neurais, exigindo atenção aos detalhes de normalização, supressão de ruído e o mecanismo de reset.
- **Escassez de Bibliotecas:** Diferente de redes mais populares como MLPs, a ART-2 não possui implementações robustas e amplamente disponíveis em bibliotecas populares como TensorFlow ou PyTorch.
- **Ajuste de Parâmetros:** A performance da ART-2 é sensível aos seus parâmetros, especialmente à vigilância ( $\rho$ ). Encontrar os valores ideais para um determinado problema pode exigir experimentação e ajustes finos.

# Abordagens de Implementação

Apesar dos desafios, existem algumas abordagens para implementar a ART-2:

- **Implementação do Zero:** Construir a ART-2 a partir do zero, utilizando linguagens como Python com bibliotecas numéricas como NumPy. Essa abordagem oferece maior controle sobre o algoritmo, mas requer um esforço significativo de desenvolvimento e depuração.
- **Adaptação de Código Existente:** Utilizar implementações disponíveis online como ponto de partida e adaptá-las às necessidades do projeto. Essa abordagem pode economizar tempo, mas requer cuidado para garantir a correção e a eficiência do código.
- **Bibliotecas Especializadas (se disponíveis):** Pesquisar por bibliotecas menos conhecidas que possam oferecer implementações da ART-2 ou de variantes dela.

# Início do Projeto Prático: Clusterização com ART-2

O projeto prático visa implementar a ART-2 para clusterização de um conjunto de dados. Os passos iniciais incluem:

- 1. Escolha do Conjunto de Dados:** Selecionar um conjunto de dados apropriado para clusterização, com dados contínuos e um número razoável de amostras e dimensões. Considerar conjuntos de dados de benchmark ou dados relevantes para a área de interesse.
- 2. Pré-processamento dos Dados:** Normalizar os dados para garantir que estejam na faixa adequada para a ART-2. Remover ruídos ou outliers se necessário.
- 3. Implementação da ART-2:** Escolher uma das abordagens de implementação mencionadas anteriormente e iniciar o desenvolvimento do código.
- 4. Definição dos Parâmetros:** Definir valores iniciais para os parâmetros da ART-2, incluindo a vigilância ( $\rho$ ).
- 5. Treinamento e Avaliação:** Treinar a ART-2 com o conjunto de dados pré-processado e avaliar a qualidade da clusterização utilizando métricas apropriadas, como o índice de Rand ajustado ou a silhueta.

**Durante o desenvolvimento e experimentação com ART-2, diversos problemas foram identificados:**

- **Ineficácia com Grandes Conjuntos de Dados:**
  - A rede ART-2 não se comporta bem em bases de dados extensas.
- **Desempenho Inferior a Outras Arquiteturas:**
  - Comparada a um MLP padrão, os resultados foram consistentemente piores.
- **Dificuldade de Implementação:**
  - Não há frameworks populares (como PyTorch) que suportem ART-2 diretamente.
  - Implementações existentes são escassas e muitas vezes subótimas.
- **Sensibilidade a Ruídos e Parâmetros:**
  - A calibração de parâmetros (como vigilância) é complexa e impacta diretamente os resultados.

**Nota:** Até mesmo autores da ART-2 reconhecem essas limitações, sugerindo que sua aplicabilidade é restrita.

# **ART2: Uma Análise Crítica no Contexto do Aprendizado de Máquina**

Este documento apresenta uma análise da rede neural ART2 considerando suas vantagens, limitações, e sua adequação em diferentes contextos de aprendizado de máquina, contrastando-a com experiências práticas e comparando-a com outras arquiteturas.

**Aplicações:** A ART2-A encontra aplicações em diversos domínios, incluindo:

- **Processamento de Sinais:** Análise de áudio, vídeo e sinais biomédicos.
- **Controle de robótica:** Tomada de decisão em ambientes dinâmicos e imprevisíveis.
- **Entre outros**

# Conclusão: Escolhendo a Arquitetura Adequada

A ART2 oferece uma abordagem única para determinados problemas de aprendizado de máquina. No entanto, sua eficácia depende muito das características do dataset, da ordem de aprendizado dos dados, da sua dimensionalidade, e de seus parâmetros sendo uma dependencia bem complicada segundo o testado. É essencial avaliar cuidadosamente as suas limitações, especialmente em relação à escalabilidade e à sensibilidade a parâmetros, antes de optar por sua utilização. Em muitos casos, outras técnicas podem ser mais adequadas e eficientes, e o desenvolvimento de aplicações com ART2 costuma exigir mais tempo e esforço do que com arquiteturas mais populares.