

# Uso de Redes de Função de Base Radial e Cadeias de Markov para detecção online de mudanças de conceito em fluxos contínuos de dados

---

**Discente:** Ruivaldo Neto

**Orientador:** Ricardo Rios

Universidade Federal da Bahia

Departamento de Ciência da Computação

Programa de Pós-Graduação em Ciência da Computação

Contato: [rneto@rneto.dev](mailto:rneto@rneto.dev)

04 de Fevereiro de 2020

1. Concept Drift
2. RBFChain
3. Experimentos
4. Novos experimentos e demonstração

# Concept Drift

---

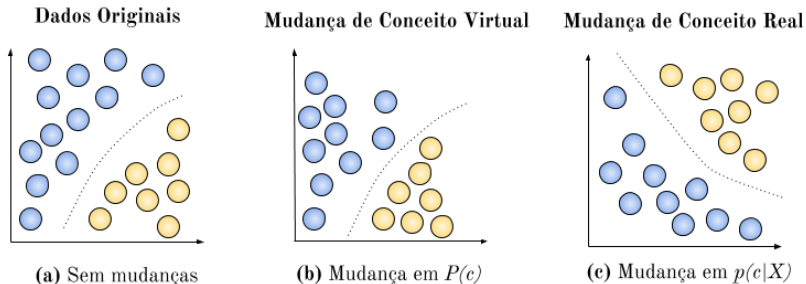
- **Mudanças de conceito** são alterações no contexto do processo e/ou na distribuição dos dados que podem impactar negativamente técnicas de AM.
- Podem ser definidas formalmente através da Teoria Bayesiana de Decisão [?]: sendo  $p_{t_0}$  e  $p_{t_1}$  as distribuições de probabilidades conjuntas nos instantes  $t_0$  e  $t_1$ , há mudança de conceito entre  $t_0$  e  $t_1$  se:

$$\exists X : p_{t_0}(X, c) \neq p_{t_1}(X, c) \quad (1)$$

# Mudança de Conceito

- São categorizadas como **Virtuais** ou **Reais** [?]:
  - **Mudanças Virtuais** são alterações na probabilidade a priori das classes,  $P(c)$ , e **não modificam** os resultados esperados.
  - **Mudanças Reais** são alterações na probabilidade a posteriori,  $p(c|X)$ , e **modificam** os resultados esperados.

# Mudança de Conceito



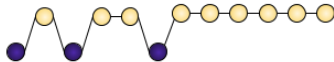
**Figura 1:** Mudança de Conceito Virtual vs. Mudança de Conceito Real.

# Mudança de Conceito

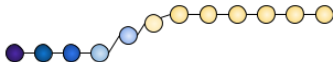
- Ocorrem de forma **abrupta**, **gradual**, **incremental** ou **recorrente** [?].



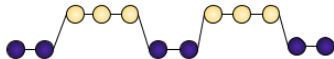
(a) Abrupta



(b) Gradual



(c) Incremental



(d) Recorrente

**Figura 2:** Padrões de ocorrência de Mudanças de Conceito.

# Algoritmos para Detecção de Mudança de Conceito

- Algoritmos de detecção se dividem em dois grupos, conforme a necessidade de rotulação dos dados [?]:
  - **Explícitos/Supervisionados**: **Dependem** da rotulação dos dados. Realizam a detecção a partir do monitoramento de medidas de performance como taxa de erro e acurácia.
  - **Implícitos/Não Supervisionados**: **Independem** da rotulação dos dados. Realizam a detecção através do monitoramento de características dos próprios dados ou de indicadores produzidos pelas técnicas de aprendizado aplicadas.



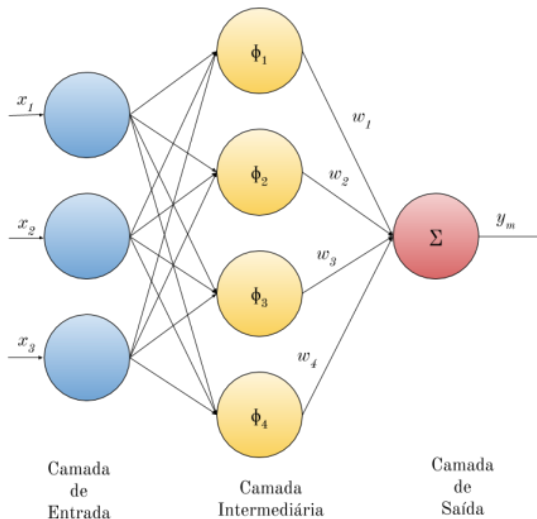
## RBFCChain

---

# Redes de Função de Base Radial

- **Redes de Função de Base Radial** são redes neurais cuja ativação é realizada através do cálculo da distância entre o evento e um centro definido [?].
- A arquitetura mais básica para redes RBF envolve três camadas:
  - **Entrada**: Recepciona os dados e encaminha para camada intermediária.
  - **Intermediária**: Composta por funções de ativação de base radial que atuam como neurônios.
  - **Saída**: Pondera os resultados da camada intermediária, agregando-os linearmente para compor a resposta final da rede.

# Redes de Função de Base Radial



**Figura 3:** Arquitetura RBF.

# Redes de Função de Base Radial

- O RBFCChain utiliza uma rede RBF adaptada, composta apenas pelas camadas inicial e intermediária.
- O processo de ativação da camada intermediária produz, implicitamente, grupos dos eventos recebidos.
- Mudanças de conceito são identificadas quando o grupo ativo desse agrupamento é alterado.

- **Cadeias de Markov** são processos estocásticos no qual a probabilidade do estado em um determinado período de tempo depende apenas do estado no período imediatamente anterior (Equação 2).

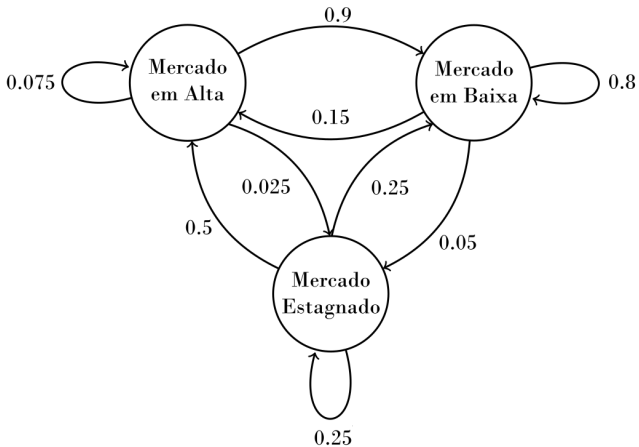
$$\mathbb{P}(X_t = s_j | X_{t-1} = s_i, \dots, X_0 = s_0) = \mathbb{P}(X_t = s_j | X_{t-1} = s_i) = p_{ij} \quad (2)$$

# Cadeias de Markov

- A Cadeia de Markov pode assumir os estados  $a_1, a_2, \dots, a_r$ , de tal modo que a probabilidade de transição de um estado  $a_i$  para um estado  $a_j$  seja  $P_{ij}$  (um valor dependente apenas de  $i$  e  $j$ );
- As probabilidades entre estados podem ser representadas por uma matriz estocástica (Equação 3):

$$P = \begin{bmatrix} P_{11} & P_{12} & \dots & P_{1r} \\ P_{21} & P_{22} & \dots & P_{2r} \\ \vdots & \vdots & \vdots & \vdots \\ P_{r1} & P_{r2} & \dots & P_{rr} \end{bmatrix} \quad (3)$$

# Cadeias de Markov

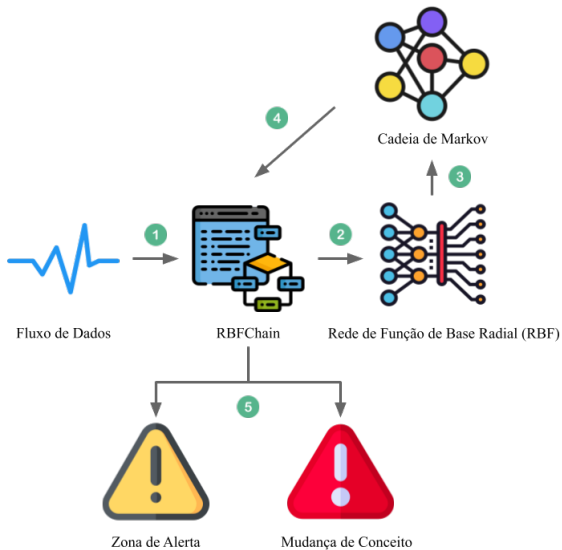


**Figura 4:** Representação Gráfica: Cadeia de Markov com três estados.

- O RBFChain utiliza uma Cadeia de Markov para modelar o agrupamento criado na rede RBF adaptada.
- Os grupos formados representam os estados e as ativações de novos grupos, as transições.
- As transições são refletidas no modelo markoviano através do aumento da probabilidade correspondente e a diminuição proporcional das outras transições, respeitando a condição:  
 $0 \leq P_{ij} \leq 1$ .



- O RBFCChain atua diretamente sobre o fluxo de dados e é composto por dois componentes principais: uma Rede de Função de Base Radial (RBF) adaptada e uma Cadeia de Markov.



**Figura 5:** Arquitetura RBFChain.

# Execução de exemplo

- $S = 0.11, 0.12, 0.13, 0.33, 0.34, 0.45, 0.6, 0.33, 0.25, 0.14, 0.11, 0.15$
- $\sigma = 3, \lambda = 0.8, \alpha = 0.25, \delta = 0.5$

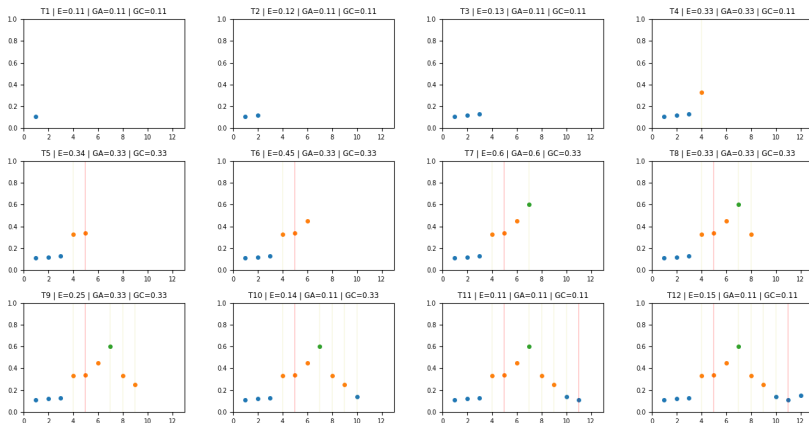


Figura 6: Execução de exemplo do RBFChain.

# Execução de exemplo

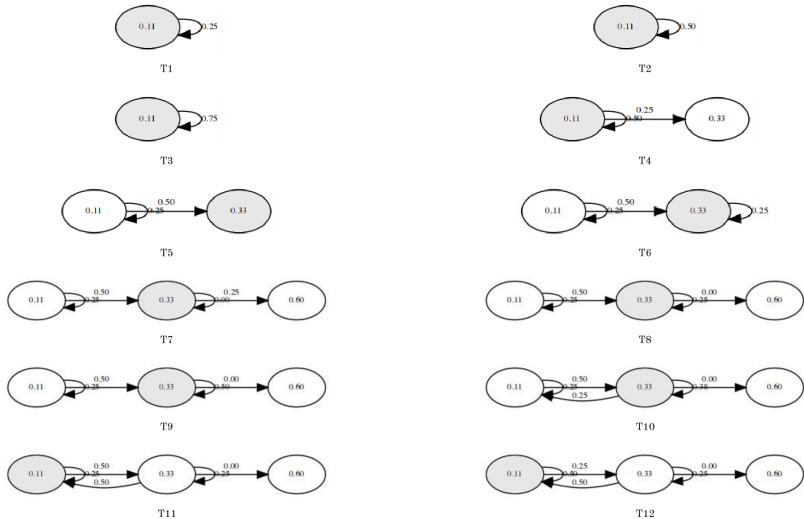
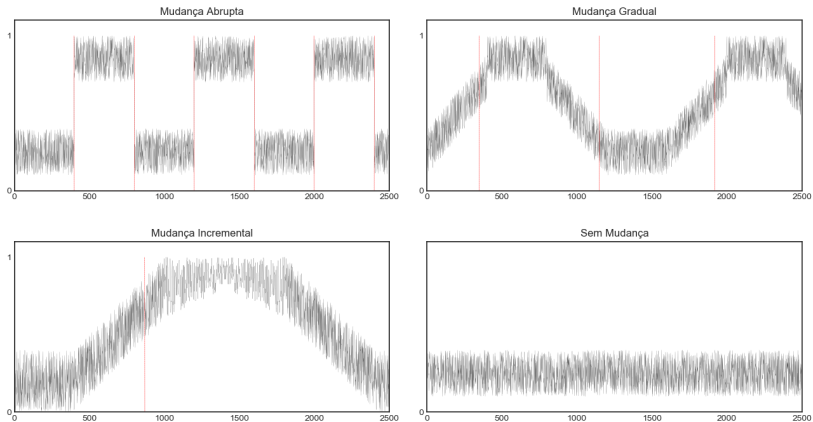


Figura 7: Evolução do modelo markoviano.

# Experimentos

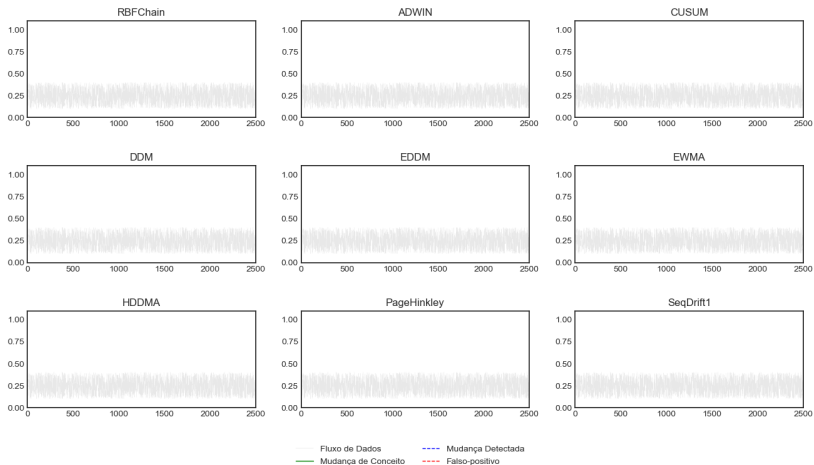
---

# Dados Sintéticos



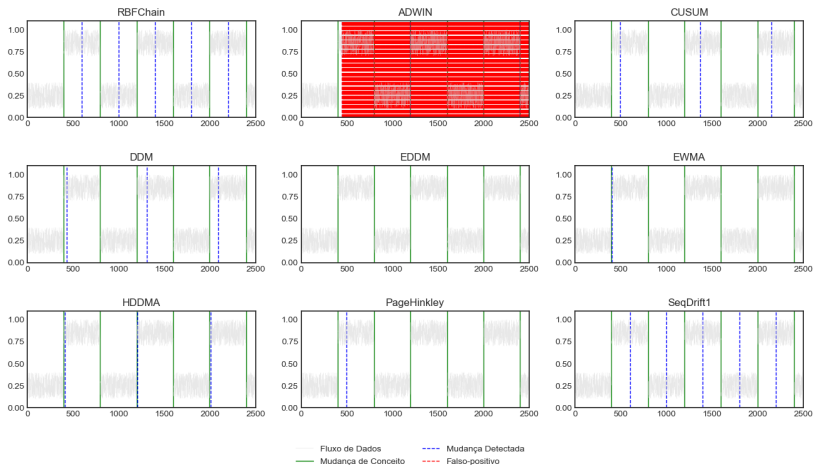
**Figura 8:** Representação gráfica dos conjuntos de dados sintéticos.

# Dados Sintéticos - Sem mudanças de conceito



**Figura 9:** Comportamento dos algoritmos para o conjunto de dados sem mudanças de conceito.

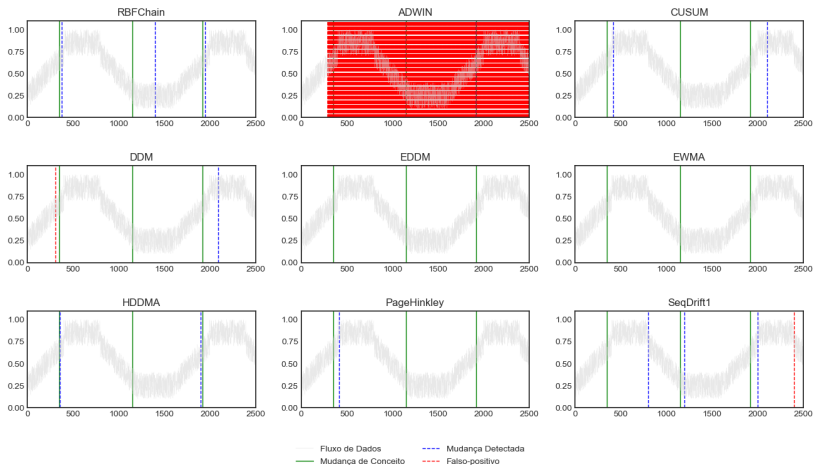
# Dados Sintéticos - Mudanças Abruptas



**Figura 10:** Comportamento dos algoritmos para o conjunto de dados com mudanças de conceito abruptas.

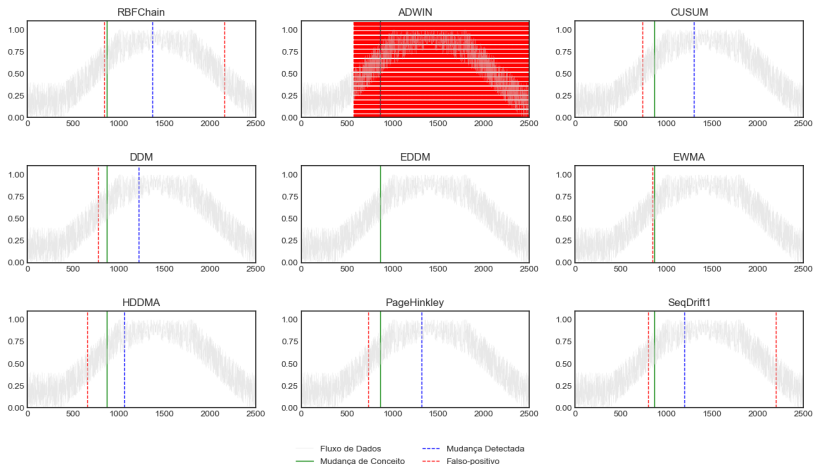


# Dados Sintéticos - Mudanças Graduais



**Figura 11:** Comportamento dos algoritmos para o conjunto de dados com mudanças de conceito graduais.

# Dados Sintéticos - Mudanças Incrementais

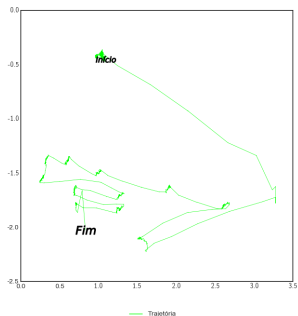


**Figura 12:** Comportamento dos algoritmos para o conjunto de dados com mudanças de conceito incrementais.

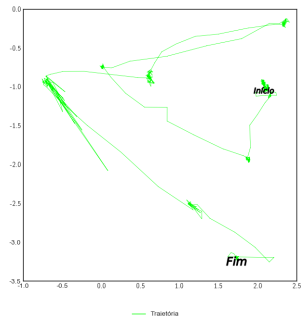
## Dados Reais - Identificação de fixações e sacadas

- O experimento utilizou dados de dois macacos-prego (Dede e Juju) produzidos e cedidos pelo Instituto do Cérebro (UFRN).
- Cada conjunto de dados possui 6.200 eventos, que indicam a localização do olhar ao longo do tempo  $(x, y)$ .
- O RBFChain foi ligeiramente adaptado para analisar a alternância (sacadas) e a continuidade (fixações) dos conceitos.
- Os resultados foram validados através de métricas de classificação, utilizando os resultados do algoritmo ClusterFix [?] como rótulos.

# Dados Reais - Identificação de fixações e sacadas - Trajetórias



**Figura 13:** Trajetória *Dede*.

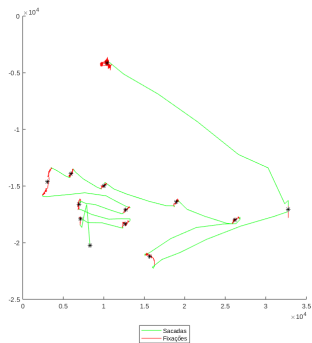


**Figura 14:** Trajetória *Juju*.

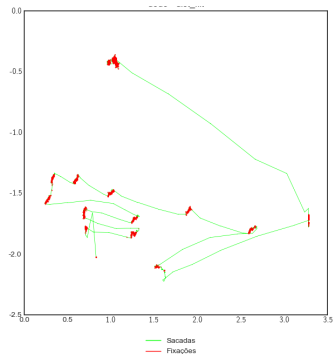
**Tabela 1:** Métricas utilizadas na avaliação com dados reais.

Métrica	Observação
QP	<b>Quantidade de Pontos</b> analisados.
AC	<b>Acurácia.</b> Fração de fixações e sacadas identificadas corretamente.
PR	<b>Precisão.</b> Fração das fixações identificadas pelo algoritmo corretamente.
RE	<b>Recall.</b> Fração das fixações existentes (rotuladas) que também foram identificadas pelo algoritmo.

# Dados Reais - Identificação de fixações e sacadas - Dede



**Figura 15:** ClusterFix - *Dede*.



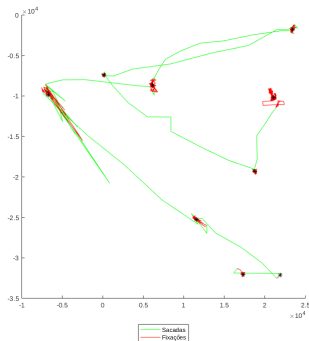
**Figura 16:** RBFChain - *Dede*.

**Tabela 2:** Resultados para o conjunto de dados *Dede*.

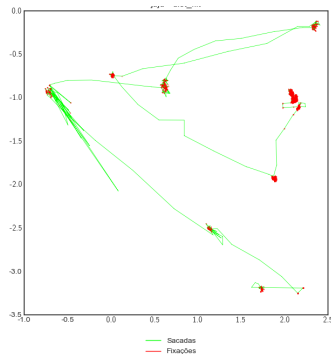
QT	AC	PR	RE
6.200	0.87	0.98	0.88

- 87% das fixações e sacadas identificadas pelo RBFChain tiveram a mesma classificação pelo ClusterFix.
- 98% das fixações identificadas pelo RBFChain tiveram a mesma classificação pelo ClusterFix.
- 88% das fixações identificadas pelo ClusterFix também foram identificadas pelo RBFChain.

# Dados Reais - Identificação de fixações e sacadas - Juju



**Figura 17:** ClusterFix - *Juju*.



**Figura 18:** RBFChain - *Juju*.



**Tabela 3:** Resultados para o conjunto de dados *Juju*.

QT	AC	PR	RE
6.200	0.82	0.98	0.83

- 82% das fixações e sacadas identificadas pelo RBFChain tiveram a mesma classificação pelo ClusterFix.
- 98% das fixações identificadas pelo RBFChain tiveram a mesma classificação pelo ClusterFix.
- 83% das fixações identificadas pelo ClusterFix também foram identificadas pelo RBFChain.

## **Novos experimentos e demonstração**

---