



Universidade Federal da Bahia
Instituto de Matemática

Programa de Pós-Graduação em Ciência da Computação

**APLICANDO REDES DE FUNÇÃO DE BASE
RADIAL PARA DETECÇÃO DE MUDANÇAS
DE CONCEITO EM FLUXOS CONTÍNUOS
DE DADOS**

Ruivaldo Azevedo Lobão Neto

QUALIFICAÇÃO DE MESTRADO

Salvador
03 de Abril de 2019

RUIVALDO AZEVEDO LOBÃO NETO

**APLICANDO REDES DE FUNÇÃO DE BASE RADIAL PARA
DETECÇÃO DE MUDANÇAS DE CONCEITO EM FLUXOS
CONTÍNUOS DE DADOS**

Esta Qualificação de Mestrado foi apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal da Bahia, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Ricardo Araújo Rios

Salvador
03 de Abril de 2019

TERMO DE APROVAÇÃO

RUIVALDO AZEVEDO LOBÃO NETO

APLICANDO REDES DE FUNÇÃO DE BASE RADIAL PARA DETECÇÃO DE MUDANÇAS DE CONCEITO EM FLUXOS CONTÍNUOS DE DADOS

Esta Qualificação de Mestrado foi julgada adequada à obtenção do título de Mestre em Ciência da Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação da Universidade Federal da Bahia.

Salvador, 03 de Abril de 2019

Prof. Dr. Ricardo Araújo Rios
UFBA

RESUMO

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Palavras-chave: Aprendizado de Máquina, Fluxos Contínuos de Dados, Mudanças de Conceito, Redes de Função de Base Radial, Não supervisionado

ABSTRACT

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Keywords: Machine Learning, Data Streams, Concept Drift, Radial Basis Function Networks, RBF Network, Unlabeled

SUMÁRIO

Capítulo 1—Introdução	1
1.1 Contexto e Motivação	1
1.2 Hipótese e Objetivo	3
Capítulo 2—Revisão Bibliográfica	5
2.1 Considerações Iniciais	5
2.2 Fluxos Contínuos de Dados e Aprendizado de Máquina	5
2.3 Mudança de Conceito	9
2.3.1 Algoritmos para Detecção de Mudança de Conceito	13
2.3.2 Ferramentas	15
2.3.3 MOA	15
2.3.4 Tornado	18
2.4 Redes de Função de Base Radial	19
2.5 Trabalhos Relacionados	21
2.6 Considerações Finais	21
Capítulo 3—Plano de Pesquisa	23
3.1 Considerações Iniciais	23
3.2 Descrição do Problema	23
3.2.1 Atividades de Pesquisa	23
3.3 Considerações Finais	24
Capítulo 4—Experimentos Iniciais	25
4.1 Considerações Iniciais	25
4.2 Configuração dos Experimentos	25
4.3 Método de Pettitt	25
4.4 Redes de Função de Base Radial	26
4.5 Considerações Finais	26

LISTA DE FIGURAS

2.1	Mudança de Conceito Virtual vs. Mudança de Conceito Real	10
2.2	Padrões de ocorrência de Mudanças de Conceito	11
2.3	MOA - Tela Inicial	16
2.4	MOA - Configuração detector	17
2.5	Framework Tornado. Fonte: (PESARANGHADER, 2018).	18
2.6	Tornado - Resultado para um múltiplos pares (PESARANGHADER, 2018)	19
2.7	Arquitetura RBF	20

LISTA DE TABELAS

2.1	Terminologia - Mudança de Conceito (ZLIOBAITE, 2010)	12
2.2	Sumário - Abordagens de detecção (SETHI; KANTARDZIC, 2017) . . .	15

INTRODUÇÃO

1.1 CONTEXTO E MOTIVAÇÃO

A quantidade de dados produzidos por sistemas computacionais tem crescido exponencialmente. Relatório do IDC (*International Data Corporation*) estimava que em 2014 seriam produzidos 4,4 zettabytes (trilhões de gigabytes) de dados (ZWOLENSKI; WEATHERILL, 2014). O mesmo relatório, prevê que em 2020 essa quantidade será de 44 zettabytes.

Grande parte desses sistemas produz dados através de sequências de eventos. Estas sequências são geradas de forma contínua, ordenada, em alta frequência e potencialmente infinita (FEIGENBAUM et al., 2003). Na literatura, sequências com essas características são denominadas Fluxos Contínuos de Dados (FCDs). Exemplos de sistemas que produzem fluxos dessa natureza, incluem: monitoramento de dados de sensores (Lee; Wang; Ryu, 2007), tráfego TCP/IP, histórico de compra de clientes, filtragem de SPAM em mensagens de e-mail (KATAKIS; TSOUMAKAS; VLAHAVAS, 2010), detecção de intrusos (LANE; BRODLEY, 1998), análise de sentimentos (SMAILOVIĆ et al., 2014), etc.

Nos últimos anos, técnicas de Aprendizado de Máquina (AM) aplicadas a fluxos contínuos de dados têm se tornado um importante tema de pesquisa. Nestes cenários, os algoritmos devem atender a severas restrições de tempo e processamento (BIFET, 2009). Por exemplo, um classificador deve fornecer o rótulo para um evento antes que o próximo ocorra.

Outra dificuldade encontrada nesses ambientes é a mudança na distribuição dos dados. Isto é, o contexto do processo gerador sofre alterações que impactam no fluxo produzido e, por conseguinte, as previsões esperadas. Este problema é conhecido como **mudança de conceito** (GAMA, 2010) e sua ocorrência impacta a acurácia do modelo criado. Podendo, inclusive, torná-lo obsoleto. Estas mudanças são comumente classificadas conforme a velocidade com que ocorrem. Mudanças de conceito **abruptas** identificam transições rápidas entre conceitos. Transições mais lentas, são ditas **graduais** (GAMA et al., 2014).

Como exemplificação, suponhamos que o histórico de transações via cartão de crédito de determinado cliente seja armazenado. Este cliente, ao longo dos anos, tem utilizado

o cartão apenas para comprar alimentos, sempre em uma mesma região da cidade. A partir desse conjunto de dados, um modelo de decisão é criado. Contudo, não é coerente considerar que o comportamento do cliente permanecerá inalterado.

Diante desse contexto, devemos considerar as seguintes hipóteses: 1) Abruptamente, esse cliente compra vários eletrônicos em outro país. Neste caso, compete aos métodos de detecção identificar se houve fraude ou se o cliente está apenas viajando e aproveitando ofertas. 2) Gradualmente, o cliente passa a utilizar o cartão para compras de passagens aéreas e diminui paulatinamente a utilização para compra de alimentos. Após determinado período, o perfil de compra estará completamente renovado. Assim, cabe novamente aos métodos de detecção identificar a mudança de comportamento e emitir um alerta, para que o modelo construído seja atualizado.

Diversos algoritmos para detecção de mudanças de conceito foram propostos na literatura. Cada uma dessas técnicas possui características e parâmetros diferentes que visam aumentar sua eficiência, conforme a natureza dos dados e do tipo de mudança que se deseja otimizar.

Parte dos algoritmos detecta a mudança de conceito através do monitoramento da taxa de erro do modelo em relação a uma determinada métrica. Alguns algoritmos que adotam essa abordagem são: DDM (GAMA et al., 2004), EDDM (BAENA-GARCÍA et al., 2006), ADWIN (BIFET; GAVALDÀ, 2007), ECDD (ROSS et al., 2012), PL (Bach; Maloof, 2008), FCWM (SEBASTIÃO et al., 2010), STEP (NISHIDA; YAMAUCHI, 2007), DOF (SOBHANI; BEIGY, 2011) e SCCDD/CRCDD (COSTA; RIOS; MELLO, 2016).

Outra metodologia bastante comum para lidar com mudanças de conceito é a aplicação de comitês de classificadores (*Ensemble Classifiers*). Esta técnica consiste na aplicação simultânea de múltiplos classificadores. A combinação das predições individuais é utilizada para compor uma predição global (GAMA et al., 2014). Os métodos para atualização do arranjo (criação e exclusão de classificadores) e a forma de integração das predições individuais variam para cada algoritmo.

Exemplos de comitês de classificadores são DWM (KOLTER; MALOOF, 2007), AUE (BRZEZIŃSKI; STEFANOWSKI, 2011), WMA (BLUM, 1997), DDD (MINKU; YAO, 2012), ADOB (SANTOS et al., 2014), dentre outros.

As abordagens mencionadas dependem da disponibilidade do rótulo correto para novos eventos após determinado período de tempo. Contudo, na maioria das aplicações reais, isto não é viável. Assim, foram desenvolvidos algoritmos para detecção de mudanças de conceito que independem da correta rotulação de novos exemplos.

Os métodos independentes de rótulos baseiam-se na identificação de eventos que não se enquadram na atual estrutura dos dados (SPINOSA; CARVALHO; GAMA, 2007). Técnicas de agrupamento e detecção de *outliers* são comumente utilizadas nas implementações. Além destas abordagens, estratégias de sumarização dos dados e aplicação de medidas de dissimilaridade (RYU et al., 2012) também são utilizadas para identificar mudanças.

Entretanto, as técnicas propostas na literatura apresentam limitações. Algoritmos que dependem da correta rotulação são de aplicação limitada e apresentam dificuldade para lidar com mudanças abruptas. Metodologias inspiradas em algoritmos não super-

visionados envolvem maior custo computacional e tempo de resposta. Visando resolver essas limitações, este projeto de mestrado discute um método para detecção de mudanças de conceito baseado em Redes de Função de Base Radial.

1.2 HIPÓTESE E OBJETIVO

Com base nas observações citadas anteriormente, a seguinte hipótese foi formulada:

“A análise de Fluxos Contínuos de Dados através de Redes de Função de Base Radial permite a detecção de mudanças de conceito, em tempo de execução, sem requerer a manutenção de estados prévios, de forma ágil e com precisão satisfatória em relação ao estado da arte.”

O objetivo deste trabalho será o desenvolvimento e comprovação da hipótese. Para atingir este objetivo, será implementado um algoritmo para detecção de mudanças de conceito baseado em Redes de Função de Base Radial. Este algoritmo diferencia-se por realizar a escolha dos centros de forma *online*, conforme novos eventos são recepcionados, e por apresentar um raio dinâmico. A ativação de novos centros é usada como indicador para possíveis mudanças de conceito.

O algoritmo implementado será comparado com o estado da arte. Os fluxos utilizados nos experimentos serão divididos em dois conjuntos. Um conjunto formado por fluxos sintéticos, que permitirão uma análise detalhada da abordagem, uma vez que as características e os comportamentos são conhecidos. O outro conjunto será composto por *datasets* obtidos a partir de sistemas computacionais utilizados na indústria, visando apresentar uma aplicação prática para a solução proposta neste trabalho.

Este trabalho está organizado conforme a seguinte estrutura: O **Capítulo 2** possui uma revisão bibliográfica dos principais conceitos empregados como, por exemplo, Fluxos Contínuos de Dados e Aprendizado de Máquina, Mudança de Conceito e Redes de Função de Base Radial; No **Capítulo 3** o plano de pesquisa definido é detalhado, identificando a metodologia que será aplicada e o cronograma de atividades. Por fim, o **Capítulo 4** apresenta um conjunto de experimentos preliminares e a análise dos resultados obtidos.

REVISÃO BIBLIOGRÁFICA

2.1 CONSIDERAÇÕES INICIAIS

Este capítulo discute conceitos importantes dentro do contexto deste trabalho de pesquisa. Em primeiro momento, a aplicação de técnicas de Aprendizado de Máquina a Fluxos Contínuos de Dados será descrita. Em seguida, será abordada a ideia de Mudança de Conceito, seus tipos e características, principais algoritmos de detecção e ferramentas. Posteriormente, serão detalhadas as Redes de Função de Base Radial. Ao fim do capítulo, elencamos os trabalhos relacionados encontrados na literatura.

2.2 FLUXOS CONTÍNUOS DE DADOS E APRENDIZADO DE MÁQUINA

Os avanços em hardware e software dos últimos anos possibilitaram a aquisição de dados em larga escala (AGGARWAL et al., 2003). Muitos desses dados são adquiridos através de sequências contínuas, potencialmente infinitas e de alta frequência, denominadas **Fluxos Contínuos de Dados** (FCDs) (AGGARWAL et al., 2003; GAMA, 2010; GAMA et al., 2004). Consequentemente, o interesse de pesquisa nesta área tem crescido bastante. Entretanto, lidar com sequências dessa natureza impõe desafios aos pesquisadores, sobretudo limitações de tempo de execução e recursos computacionais.

Conforme (BABCOCK et al., 2002) as principais diferenças de FCDs para os dados convencionais são:

- Possuem tamanho ilimitado;
- Os dados são produzidos de forma contínua e em alta frequência;
- Não há controle sobre a ordem de chegada dos eventos;
- Os eventos processados devem ser descartados.

Diversas aplicações da indústria e academia lidam com dados em fluxos contínuos, por exemplo:

Finanças detecção de fraudes em transações, acompanhamento da bolsa de valores;

Energia monitoramento, predição e planejamento do consumo;

Redes e telecomunicações monitoramento de rede, detecção de intrusos.

A área de Aprendizado de Máquina (AM) tem como objetivo estudar algoritmos que melhoram o seu desempenho na realização de determinada tarefa conforme ganham experiência (MITCHELL, 1997). Atualmente, diversas técnicas de AM estão disponíveis, sendo aplicadas a um amplo conjunto de problemas.

Segundo (GAMA, 2010), a área de AM dedicou-se durante muitos anos em resolver problemas de aprendizado não-incrementais (*batch*). Nestes cenários, a distribuição dos dados é estacionária, ou seja, o conjunto de classes ou grupos possíveis não sofre modificações. Esta estabilidade permite a criação de modelos que se mantêm ao longo do tempo.

Contudo, os principais cenários da indústria e pesquisa que envolvem FCDs são dinâmicos. Nestes cenários, os dados fluem de forma contínua, mas o contexto do processo que os produz pode sofrer alterações com o passar do tempo. Isto é, são fluxos cuja distribuição dos dados não é estacionária. Conforme (GAMA; RODRIGUES, 2009), o principal problema no aprendizado a partir de FCDs é a manutenção de uma boa acurácia (classificadores) e a obtenção de grupos bem formados (*clustering*). Para que esta continuidade seja possível, o algoritmo de aprendizado deve ser atualizado conforme novos dados são analisados. Esta metodologia é denominada Aprendizado Incremental (GAMA et al., 2014).

Conforme (FISHER, 1987), a ideia de aprendizado incremental tem origem na própria observação do mundo real, no qual o conhecimento sofre atualizações constantemente e uma base contínua é mantida para lidar com novos estímulos. O paradigma incremental acredita que o contexto do processo pode mudar ao longo do tempo e, portanto, o conhecimento adquirido deve ser constantemente atualizado.

Segundo (REIMANN; SPADA, 1995), um algoritmo A pode ser dito incremental se A recebe um evento de treinamento por vez, não reprocessa dados do passado e mantém somente uma estrutura de conhecimento em memória. Além destas, o trabalho também elenca as seguintes características desejáveis para os algoritmos incrementais:

- Tempo de processamento constante;
- Dados analisados não são mantidos em memória;
- Possibilidade de suspender o treinamento para realizar predições.

Os algoritmos incrementais apresentam as seguintes vantagens (PINTO; GAMA; PORTO, 2005):

- Otimização do tempo de processamento;
- Menor utilização de memória;

- Aproximam-se da forma de aprendizado dos humanos.

Por suas características, algoritmos incrementais mostram-se bastante interessantes para aplicação em cenários com FCDs. Contudo, (GAMA et al., 2004) ressalta que o aprendizado a partir de FCDs exige que os algoritmos além de incrementais, também tratem mudanças de conceito envolvidas no processo. Nesses casos, é preciso descartar exemplos antigos, que não mais condizem com o cenário atual e adaptar o modelo de decisão aos novos dados.

Algoritmos de aprendizado de máquina clássicos utilizam técnicas como agrupamento (*clustering*) e classificação. A partir destas metodologias, novos algoritmos foram desenvolvidos para trabalhar com FCDs.

A organização de dados em grupos é uma das formas mais fundamentais de compreensão e aprendizagem (JAIN; DUBES, 1988). Agrupamento é a tarefa de associar objetos conforme suas características, objetivando formar grupos com alta similaridade entre seus objetos e baixa similaridade com objetos de outros grupos. Técnicas de agrupamento permitem explorar a estrutura dos dados e não exigem os pressupostos comuns à maioria dos métodos estatísticos. A representação criada permite verificar se os grupos estão de acordo com determinados parâmetros ou ainda sugerir novos experimentos (JAIN; DUBES, 1988).

Os métodos de agrupamento têm sido utilizados em diversas áreas do conhecimento, o que ensejou o desenvolvimento de uma grande variedade de algoritmos. Dentre os principais algoritmos para cenários *batch*, destacam-se: K-Means (LLOYD, 2006), DBSCAN (ESTER et al., 1996), PAM (KAUFMAN; ROUSSEEUW, 1990) e OPTICS (ANKERST et al., 1999).

Segundo (GAMA, 2010), a principal dificuldade na aplicação de técnicas de agrupamentos em FCDs é a manutenção da qualidade e consistência do agrupamento em relação à sequência de dados observados, utilizando pouca memória e tempo de processamento. Estes desafios surgem, pois os dados são recebidos de forma contínua e as respostas precisam ser obtidas em tempo hábil.

Essas características fazem com que os algoritmos de agrupamentos aplicados em FCDs precisem ser incrementais, mantendo estruturas de grupos que evoluam ao longo do tempo. Segundo (BARBARÁ, 2002), as características dos fluxos contínuos implicam três novos requisitos aos algoritmos de agrupamento:

- Produzir representações compactas;
- Processamento rápido e incremental de novos dados;
- Identificação precisa e tempestiva de *outliers*.

Conforme (AGGARWAL et al., 2003), o problema de agrupamentos de dados em cenários com FCD é de difícil resolução, pois a natureza contínua e de alta frequência dos dados tornam os algoritmos tradicionais ineficientes. Assim, foram desenvolvidos algoritmos de agrupamento que realizam uma varredura única nos dados. Todavia, ainda segundo (AGGARWAL et al., 2003), estes algoritmos apesar de apresentarem escalabilidade, não consideram a evolução dos dados. Esta característica faz com que a qualidade

dos grupos obtidos diminua ao longo do tempo. Além disso, por varrerem os dados de forma única, não é possível formar agrupamentos sobre diferentes partes do fluxo. Outro problema para realização de agrupamentos em FCDs é a influência de ruídos e *outliers* (KHALILIAN; MUSTAPHA, 2010).

De acordo com (LING; LING-JUN; LI, 2009), parte dos algoritmos de *clustering* em FCD atua sobre uma projeção contínua do agrupamento de dados, por exemplo (GUHA et al., 2003). Esses algoritmos utilizam a estratégia de divisão e conquista, particionando a FCD em segmentos e formando grupos através da aplicação de algoritmos baseados no K-Means para espaços delimitados (GUHA et al., 2000). Esta técnica apresenta como limitação não conseguir capturar características que evoluem ao longo do fluxo, pois dão o mesmo peso para dados antigos e recentes.

Diversos algoritmos foram propostos para agrupamento em FCD. Destacam-se: CluStream (AGGARWAL et al., 2003), StreamKM++ (ACKERMANN et al., 2012), DenStream (CAO et al., 2006), D-Stream (LING; LING-JUN; LI, 2009) e ClusTree (KRANEN et al., 2011).

De forma similar, as técnicas de classificação continuam sendo ativamente estudadas. Estes métodos pertencem ao conjunto de algoritmos supervisionados, que realizam predições para uma variável (conceito alvo) utilizando um modelo de decisão criado a partir de uma base de treinamento previamente rotulada (KOTSIANTIS, 2007). Se a variável a ser predita é categórica, entende-se como um problema de classificação. Se a predição resulta em um valor numérico, trata-se de um problema de regressão.

Muitos métodos foram propostos para a tarefa de classificação em cenários *batch*: árvores de decisão (BREIMAN et al., 1984), métodos baseados em regras, redes neurais e máquinas de vetores suporte (SVM) (VAPNIK, 1998), dentre outros. Contudo, a maioria dos trabalhos propostos pressupõe ambientes estáticos, onde os modelos de decisão criados não sofrem alterações (AGGARWAL, 2006). Além disso, também assumem que todos os dados estão disponíveis em memória e que é possível realizar múltiplas varreduras sobre eles. Assim, essas técnicas precisam ser repensadas para atenderem as características dos FCDs.

Conforme (LING; LING-JUN; LI, 2009), o problema de classificação em FCD considera que o FCD é uma sequência de exemplos e que cada exemplo está associado a um rótulo de valor discreto. Este rótulo indica a classe a qual o exemplo pertence. O objetivo, portanto, da classificação em FCD é prever com acurácia a classe dos exemplos desconhecidos, que chegam de forma contínua através da FCD.

Nos últimos anos, muitos algoritmos de classificação para FCDs foram propostos (DOMINGOS; HULTEN, 2000; BIFET et al., 2013; WANG et al., 2003; AGGARWAL et al., 2004; GAMA; ROCHA; MEDAS, 2003). Estes trabalhos aplicam técnicas incrementais e adotam medidas para atualização do modelo de decisão. No entanto, a maior parte dos algoritmos propostos requer que o rótulo correto para os exemplos esteja disponível após determinado período de tempo. Este rótulo é utilizado na execução de novos ciclos de aprendizado, que permitem a atualização do modelo de decisão. Estes trabalhos também consideram que todas classes possíveis do problema já são conhecidas.

Segundo (AGGARWAL, 2006), o principal desafio para métodos de classificação aplicados a FCDs é a mudança de conceito, isto é, a evolução dos dados ao longo do tempo.

A Detecção de Mudança de Conceito é uma tarefa de classificação aplicável a diferentes problemas práticos que envolvem FCDs. A técnica consiste em identificar exemplos (ou grupos de exemplos) que não são coerentes com o modelo de decisão vigente. O algoritmo de detecção pode indicar se o conceito alvo aprendido sofreu modificações ou se novos conceitos surgiram. Na próxima seção, a ideia de mudança de conceito será aprofundada.

2.3 MUDANÇA DE CONCEITO

Em ambientes não estacionários, a distribuição subjacente dos dados pode mudar ao longo do tempo, levando ao fenômeno de mudança de conceito (SCHLIMMER; GRANGER, 1986). Por exemplo, podemos considerar como mudança de conceito quando uma pessoa passa a preferir um novo gênero literário. Como consequência desta mudança, modelos de decisão relacionados podem ter sua acurácia reduzida, podendo, inclusive, tornarem-se obsoletos.

A Teoria Bayesiana de Decisão (DUDA; HART; STORK, 2000) é comumente aplicada para descrever o problema de classificação baseado nas probabilidades a priori das classes, $p(y)$, e na função de densidade de probabilidade condicionada às classes, $p(X|y)$, para todo $y \in c_1, \dots, c_k$, onde c_1, \dots, c_k é o conjunto de classes (ZLIOBAITE, 2010; GAMA et al., 2014). Conforme a teoria bayesiana, a classificação de uma instância X é definida com base na máxima probabilidade a posteriori. A máxima probabilidade a posteriori para uma instância X em relação a uma classe c_i pode ser calculada através da fórmula 2.1:

$$p(c_i|X) = \frac{p(c_i) * p(X|c_i)}{p(X)} \quad (2.1)$$

Sendo assim, um **conceito** pode ser definido como um conjunto de probabilidades a priori e condicionais das classes, como mostra a equação 2.2:

$$S = \{(P(c_1), P(X|c_1)), (P(c_2), P(X|c_2)), \dots, (P(c_k), P(X|c_k))\} \quad (2.2)$$

Formalmente, pode-se afirmar que houve mudança de conceito entre os instantes t_0 e t_1 se:

$$\exists X : p_{t_0}(X, y) \neq p_{t_1}(X, y) \quad (2.3)$$

onde, p_{t_0} e p_{t_1} denotam as distribuições de probabilidades conjuntas nos instantes t_0 e t_1 , respectivamente, para X e y (GAMA et al., 2014). Isto é, um conjunto de exemplos possui rótulos de classe legítimos em t_0 , mas passa a ter rótulos diferentes em t_1 (KOLTER; MALOOF, 2007).

A equação 2.1 pode ser utilizada para caracterizar as mudanças na distribuição dos dados (GAO et al., 2007; GAMA et al., 2014), do seguinte modo:

- Mudanças podem ocorrer na probabilidade a priori $P(c)$,
- Mudanças podem ocorrer na função de densidade de probabilidade condicionada às classes $p(X|c)$,

- Mudanças podem ocorrer na probabilidade a posteriori $p(c|X)$, afetando os limites de classificação.

Segundo (WIDMER; KUBAT, 1996), alterações na distribuição dos dados podem ser categorizadas em dois grupos: mudanças de conceito virtuais e mudanças de conceito reais. Estes tipos de mudança foram formalizados por (ZLIOBAITE, 2010; GAMA et al., 2014):

1. Mudanças de conceito virtuais tratam de mudanças em $P(X)$ e, conseqüentemente, em $p(X|y)$, contudo, estas mudanças não têm impacto no conceito alvo (representadas na Figura 2.1 (b)).
2. Mudanças de conceito reais referem-se a mudanças na relação $p(y|X)$ e afetam o conceito alvo (representadas na Figura 2.1 (c)).

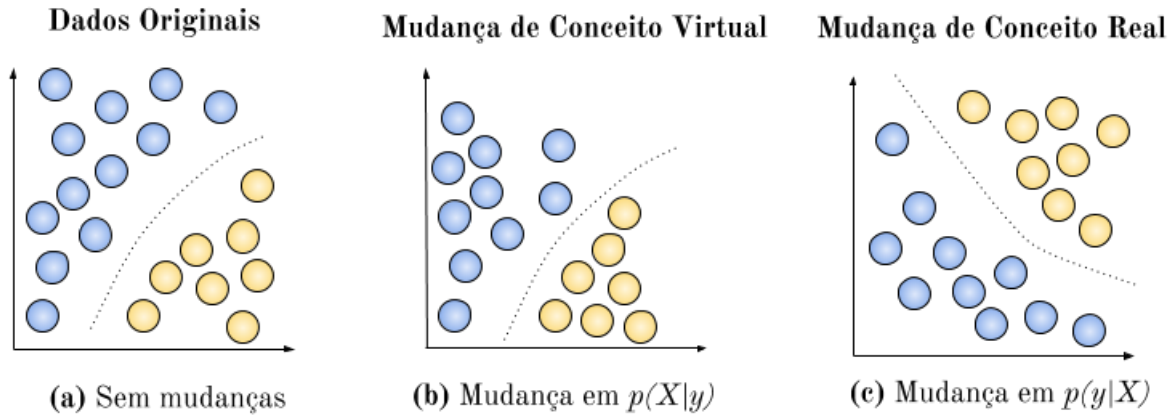


Figura 2.1 Mudança de Conceito Virtual vs. Mudança de Conceito Real

Em (GAMA et al., 2014), o seguinte cenário é utilizado para explicar a diferença entre uma mudança de conceito real e uma virtual. Considere um fluxo *online* de anúncios do setor imobiliário. Um usuário deseja classificar estes anúncios em relevantes e não relevantes. Suponha que, à época, o usuário esteja buscando um novo apartamento. Isto torna anúncios de apartamentos residenciais mais relevantes e os anúncios de casa de férias menos relevantes. Se o autor dos anúncios é alterado, o estilo de escrita sofre mudanças, mas os apartamentos permanecem relevantes para o usuário. Este cenário corresponde a uma mudança de conceito virtual. Considere que devido a uma crise, mais anúncios de apartamentos e menos de casas de férias aparecem, mas o editor dos anúncios permanece o mesmo. Esta situação corresponde a uma mudança na probabilidade das classes. Em outro cenário, assumo que o usuário acabou de comprar um apartamento e está em busca de um local para passar as férias. Neste contexto, apartamentos tornaram-se irrelevantes e casas de férias, relevantes. Apesar do editor dos anúncios e das probabilidades anteriores das classes permanecerem as mesmas, este cenário corresponde a uma mudança de conceito real.

Considerando apenas a perspectiva de predição, adaptações no modelo são necessárias apenas quando mudanças de conceito reais ocorrem, pois os limites de decisão do modelo corrente tornam-se obsoletos em relação aos novos exemplos (GAMA et al., 2014). Adaptação, neste contexto, significa atualizar o modelo de decisão perante a nova distribuição dos dados a fim de manter uma alta acurácia.

Mudanças de conceito podem se apresentar sob diferentes padrões. Conforme ilustrado na Figura 2.2, mudanças de conceito podem ocorrer de forma abrupta, gradual, incremental ou recorrente.

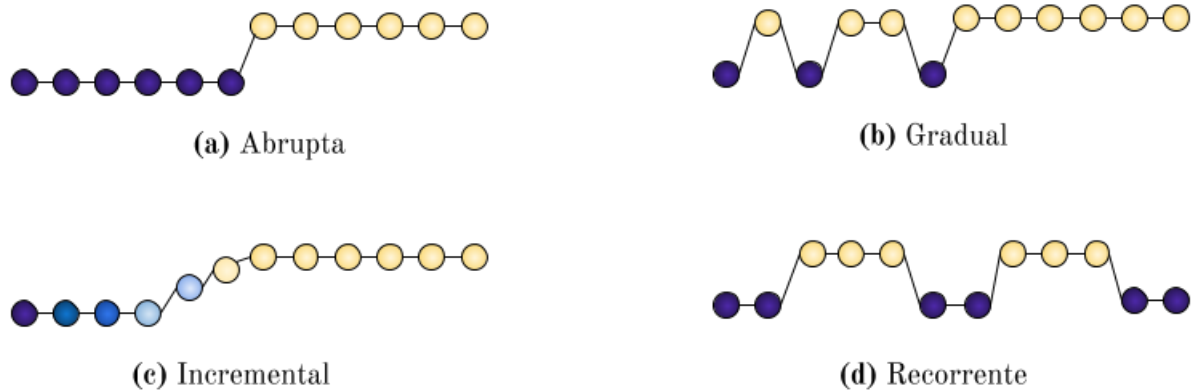


Figura 2.2 Padrões de ocorrência de Mudanças de Conceito

Mudanças de conceito abruptas são alterações súbitas entre conceitos, como descrito na Figura 2.2 (a). Por exemplo, o gênero literário de João pode subitamente mudar de ficção científica para auto-ajuda. Outro exemplo, é a troca de um sensor por outro com diferente calibração, em uma planta industrial química (GAMA et al., 2014).

Em contraste com a mudança de conceito abrupta, a mudança de conceito gradual significa uma transição mais lenta de um conceito para outro. Isto é, durante a transição, são percebidos exemplos de ambos os conceitos. Pode-se considerar que João, ao invés de subitamente passar a preferir livros de auto-ajuda, adquire interesse nestes ao longo do tempo. Isto significa que João ainda tem interesse em livros de ficção científica, mas desenvolve, em paralelo, um crescente interesse por livros de auto-ajuda, que se tornará predominante. A mudança de conceito gradual está representada na Figura 2.2 (b).

Mudanças de conceito incrementais apresentam diversos conceitos intermediários durante a transição de um conceito para outro. No exemplo de João, é como se o seu interesse literário variasse de ficção científica para ficção científica e aventura, de ficção científica e aventura para aventura e suspense, de aventura e suspense para suspense e auto-ajuda, e finalmente de suspense e auto-ajuda para apenas auto-ajuda. Considerando a planta industrial química e os sensores, seria como se um sensor perdesse precisão ao longo do tempo. A Figura 2.2 (c) demonstra a mudança de conceito incremental.

Existe ainda outro padrão, denominado mudança de conceito recorrente (ZLIOBAITE, 2010). Conforme apresentado na Figura 2.2 (d), a mudança de conceito recorrente existe

quando um conceito previamente ativo, reaparece após determinado período de tempo. De volta ao nosso exemplo, João está atualmente interessado em livros de ficção científica. Contudo, ele é presenteado com uma série de livros de auto-ajuda de um renomado autor brasileiro. Após ler toda a série, João é premiado com descontos para livros de ficção científica. Ele, volta, então, a ler livros de ficção científica comprados com os descontos recebidos.

É importante destacar que o fenômeno Mudança de Conceito tem sido estudado em diferentes comunidades de pesquisa, incluindo mineração de dados, aprendizado de máquina, estatística e recuperação de informação (ZLIOBAITE, 2010). Contudo, o mesmo conceito pode ter diferentes nomenclaturas em cada comunidade. Na Tabela 2.3 são listados os termos correspondentes a Mudança de Conceito para cada área de pesquisa.

Área	Termos
Mineração de Dados	Mudança de Conceito
Aprendizado de Máquina	Mudança de Conceito, Mudança de Covariável
Computação Evolucionária	Ambiente Evolutivo, Ambiente em Mudança
IA e Robótica	Ambiente Dinâmico
Estatísticas, Séries Temporais	Não Estacionário
Recuperação de Informação	Evolução Temporal

Tabela 2.1 Terminologia - Mudança de Conceito (ZLIOBAITE, 2010)

Outra possível fonte de equívoco é a utilização dos termos Detecção de *Outliers*, Detecção de Novidade e Detecção de *Change Points*. Estes termos apresentam relação para com Detecção de Mudança de Conceito, pois dizem respeito a encontrar padrões que são diferentes dos padrões conhecidos. Em alguns contextos, esses termos são usados indistintamente, entretanto, neste trabalho eles serão diferenciados.

De acordo com (CHANDOLA; BANERJEE; KUMAR, 2009), Detecção de *Outliers* refere-se a tarefa de encontrar padrões nos dados que não estão de acordo com o comportamento esperado. Esses padrões podem ser denominados como: *outliers*, anomalias, observações discordantes, exceções, peculiaridades ou aberrações.

Para (GAMA, 2010), exemplos esparsos e independentes, cujas características diferem muito daqueles que definem o modelo, devem ser considerados como *outliers*, pois não há garantia de que representem um conceito. Em (AGGARWAL et al., 2003), os autores tipificam os *outliers* como anomalias ou ruídos. As anomalias constituem um tipo especial de *outlier*, que é de interesse dos analistas. Contudo, conforme (GAMA, 2010), é necessário um grupo conciso de exemplos para evidenciar o aparecimento de um novo conceito, ou novidade.

Segundo (CHANDOLA; BANERJEE; KUMAR, 2009), a Detecção de Novidade objetiva detectar padrões não observados (emergentes) nos dados. Entretanto, esse termo se distingue da detecção de *outliers*, pois os novos padrões, geralmente, são incorporados ao

modelo. Apesar de apresentarem definições próximas, a Detecção de Mudança de Conceito engloba a Detecção de Novidade e a extrapola, ao identificar mudanças de conceito reais a partir do *feedback* sobre a performance preditiva (GAMA, 2010).

Técnicas para Detecção de *Change Points* identificam variações abruptas de valor em séries temporais, que podem representar transições entre estados. Diferenciam-se das técnicas para Detecção de Mudança de Conceito pois são aplicadas em séries temporais unidimensionais estacionárias (AMINIKHANGHAHI; COOK, 2017).

Na seção seguinte serão detalhados os principais algoritmos para Detecção de Mudança de Conceito.

2.3.1 Algoritmos para Detecção de Mudança de Conceito

Os algoritmos para Detecção de Mudança de Conceito são os responsáveis por identificar, de forma explícita, as mudanças de conceito nos fluxos de dados (GAMA et al., 2014). Estes métodos caracterizam e quantificam as mudanças de conceito através da delimitação dos instantes ou intervalos de tempo em que as mudanças ocorrem (BASSEVILLE; NIKIFOROV, 1993).

As técnicas propostas na literatura podem ser divididas em duas categorias, baseado na dependência ou não de dados rotulados:

Algoritmos Explícitos/Supervisionados Dependem da rotulação dos dados por um especialista. Estes rótulos são utilizados no cálculo de medidas de performance como taxa de erro e acurácia, que são monitoradas ao longo do tempo. Mudanças de conceito são sinalizadas quando essas medidas atingem um limite previamente definido.

Algoritmos Implícitos/Não Supervisionados Independem da rotulação por especialistas, baseando-se em características dos próprios dados para sinalizar desvios. São mais propensos a alarmes falsos, mas a independência de rótulos os torna interessantes para contextos onde a obtenção desses é dispendiosa, demorada ou inviável.

Em sua pesquisa (GAMA et al., 2014) categorizou os algoritmos supervisionados em três grupos principais:

Métodos Baseados em Análise Sequencial Avaliam, de forma sequenciada, os resultados das previsões conforme tornam-se disponíveis (performance). Indicam a ocorrência de mudança de conceito quando um limite pré-definido é atingido. Os algoritmos *Cumulative Sum (CUSUM)*, *PageHinkley (PH)* (PAGE, 1954) e *Geometric Moving Average (GMA)* (ROBERTS, 2000) são representantes deste grupo.

Abordagens baseadas em Estatística Identificam mudanças de conceito através da análise de parâmetros estatísticos como média e desvio padrão para os resultados das previsões. Os métodos *Drift Detection Method (DDM)* (GAMA et al., 2004), *Early Drift Detection Method (EDDM)* (BAENA-GARCÍA et al., 2006), *Exponentially Weighted Moving Average (EWMA)* (ROSS et al., 2012) e *Reactive Drift Detection Method (RDDM)* (BARROS et al., 2017) são exemplos deste grupo.

Métodos baseados em Janelas Geralmente, utilizam uma janela de tamanho fixo para sumarizar informações passadas e uma janela deslizante para sumarizar dados mais recentes. A mudança de conceito é detectada quando há uma diferença significativa entre as distribuições das janelas. Esta diferença é detectada a partir de testes estatísticos ou desigualdades matemáticas, considerando como hipótese nula a igualdade entre as distribuições. Os algoritmos *Adaptive Windowing (ADWIN)* (BIFET; GAVALDÀ, 2007), *SeqDrift* (PEARS; SAKTHITHASAN; KOH, 2014), *HDDMA* e *HDDMW* (BLANCO et al., 2015) são membros desta família.

Os algoritmos implícitos também foram divididos em três grupos (GONÇALVES et al., 2014):

Detecção de Novidade / Métodos de Agrupamento Utilizam a distância e/ou a densidade dos dados para detectar novos padrões. O algoritmo identifica exemplos suspeitos e os atribui a uma classe *Desconhecido*, indicando a necessidade de uma avaliação adicional. As técnicas de agrupamento e detecção de *outliers* são estratégias de implementação populares para estes algoritmos, pois sintetizam os dados correntes e aplicam medidas de dissimilaridade para identificar novos padrões (RYU et al., 2012). Os métodos *OLINDDA* (SPINOSA; CARVALHO; GAMA, 2007), *MINAS* (FARIA; GAMA; CARVALHO, 2013), *Woo* (RYU et al., 2012), *DETECTNOD* (Hayat; Hashemi, 2010), *ECSMiner* (MASUD et al., 2011) e *GC3* (SETHI; KANTARDZIC; HU, 2016) fazem parte deste grupo.

Monitoramento de distribuição multivariada Monitoram diretamente a distribuição dos dados, considerando cada atributo. Este monitoramento é feito através de subconjuntos dos dados. Um subconjunto de treinamento tem sua distribuição sumariada e armazenada, para ser utilizada como referência. A distribuição referência é comparada à distribuição do subconjunto atual e, havendo diferenças significativas, a mudança de conceito é sinalizada. Os algoritmos *CoC* (Lee; Magoulès, 2012), *HDDDM* (Ditzler; Polikar, 2011), *PCA-detect* (KUNCHEVA, 2008) são representantes deste grupo.

Monitoramento dependente de modelo As técnicas elencadas nos grupos anteriores monitoram desvios na distribuição dos dados para indicar mudanças de conceito. Essencialmente, esses métodos assumem que alterações na distribuição $P(X)$ implicarão mudanças na performance do classificador $p(y|X)$. Apesar de permitir a aplicação de qualquer tipo de classificador, essa presunção leva a uma maior quantidade de falsos positivos. As técnicas dependentes de modelo, por sua vez, restringem-se aos classificadores probabilísticos, o que permite realizar a detecção de mudanças de conceito através do monitoramento das probabilidades a posteriori (ZLIOBAITE, 2010). A utilização das probabilidades a posteriori diminui a incidência de falsos positivos e torna o processo computacionalmente eficiente, pois apenas um único fluxo univariado de valores é observado. Os métodos *A-distance* (DREDZE; OATES; PIATKO, 2010), *CDBD* (LINDSTROM; NAMEE; DELANY, 2013) e *Margin* (DRIES; RÜCKERT, 2009) participam deste grupo.

Por fim, a Tabela 2.2 sumariza as categorias, os grupos e as respectivas técnicas mencionadas nesta seção.

Algoritmos Explícitos/Supervisionados	Métodos Baseados em Análise Sequencial	Cumulative Sum (CUSUM) PageHinkley (PH) (PAGE, 1954) Geometric Moving Average (GMA) (ROBERTS, 2000)
	Abordagens baseadas em Estatística	Drift Detection Method (DDM) (GAMA et al., 2004) Early Drift Detection Method (EDDM) (BAENA-GARCÍA et al., 2006) Exponentially Weighted Moving Average (EWMA) (ROSS et al., 2012) Reactive Drift Detection Method (RDDM) (BARROS et al., 2017)
	Métodos baseados em Janelas	Adaptive Windowing (ADWIN) (BIFET; GAVALDÀ, 2007) SeqDrift (PEARS; SAKTHITHASAN; KOH, 2014) HDDMA/HDDMW (BLANCO et al., 2015)
Algoritmos Implícitos/Não Supervisionados	Deteção de Novidade / Métodos de Agrupamento	OLINDDA (SPINOSA; CARVALHO; GAMA, 2007) MINAS (FARIA; GAMA; CARVALHO, 2013) Woo (RYU et al., 2012) DETECTNOD (Hayat; Hashemi, 2010) ECSSMiner (MASUD et al., 2011) GC3 (SETHI; KANTARDZIC; HU, 2016)
	Monitoramento de distribuição multivariada	CoC (Lee; Magoulès, 2012) HDDDM (Ditzler; Polikar, 2011) PCA-detect (KUNCHEVA, 2008)
	Monitoramento dependente de modelo	A-distance (DREDZE; OATES; PIATKO, 2010) CDBD (LINDSTROM; NAMEE; DELANY, 2013) Margin (DRIES; RÜCKERT, 2009)

Tabela 2.2 Sumário - Abordagens de detecção (SETHI; KANTARDZIC, 2017)

2.3.2 Ferramentas

Nesta seção, os frameworks *Massive Online Analysis* (MOA) e *Tornado* serão apresentados. Estes frameworks possibilitam a implementação e a análise - perante o estado da arte - de algoritmos para detecção de mudança de conceito. Ambos permitem a avaliação entre técnicas e a utilização de diferentes *datasets*.

2.3.3 MOA

O MOA – *Massive Online Analysis*¹ é, atualmente, o principal framework para mineração de dados em fluxos contínuos. O projeto é de código-aberto² e apresenta uma comunidade bastante ativa e crescente (BIFET et al., 2010). A aplicação é composta por uma ampla coleção de algoritmos da área de aprendizado de máquina: classificação, regressão, agrupamento, busca por padrões, detecção de *outliers*, detecção de mudanças de conceito e sistemas de recomendação. Além de conter ferramentas para avaliação destes. Apresenta relação com o projeto WEKA (HALL et al., 2009), sendo também desenvolvido em Java, o que permite a sua execução nos principais sistemas operacionais.

O MOA divide suas funcionalidades em tarefas (*tasks*). As principais tarefas incluem: produção de fluxos de dados, treino de classificadores, aplicação de algoritmos de agrupamento, análise de algoritmos para detecção de *outliers* e de *concept drift*, dentre outras. As tarefas podem ser executadas a partir da interface gráfica (GUI) ou por linha de

¹<https://moa.cms.waikato.ac.nz/>

²<https://github.com/Waikato/moa>

comando. A tela principal da aplicação é demonstrada na Figura 2.3. Através da interface gráfica é possível executar múltiplas tarefas de forma concorrente, controlar suas execuções e visualizar os resultados parciais.

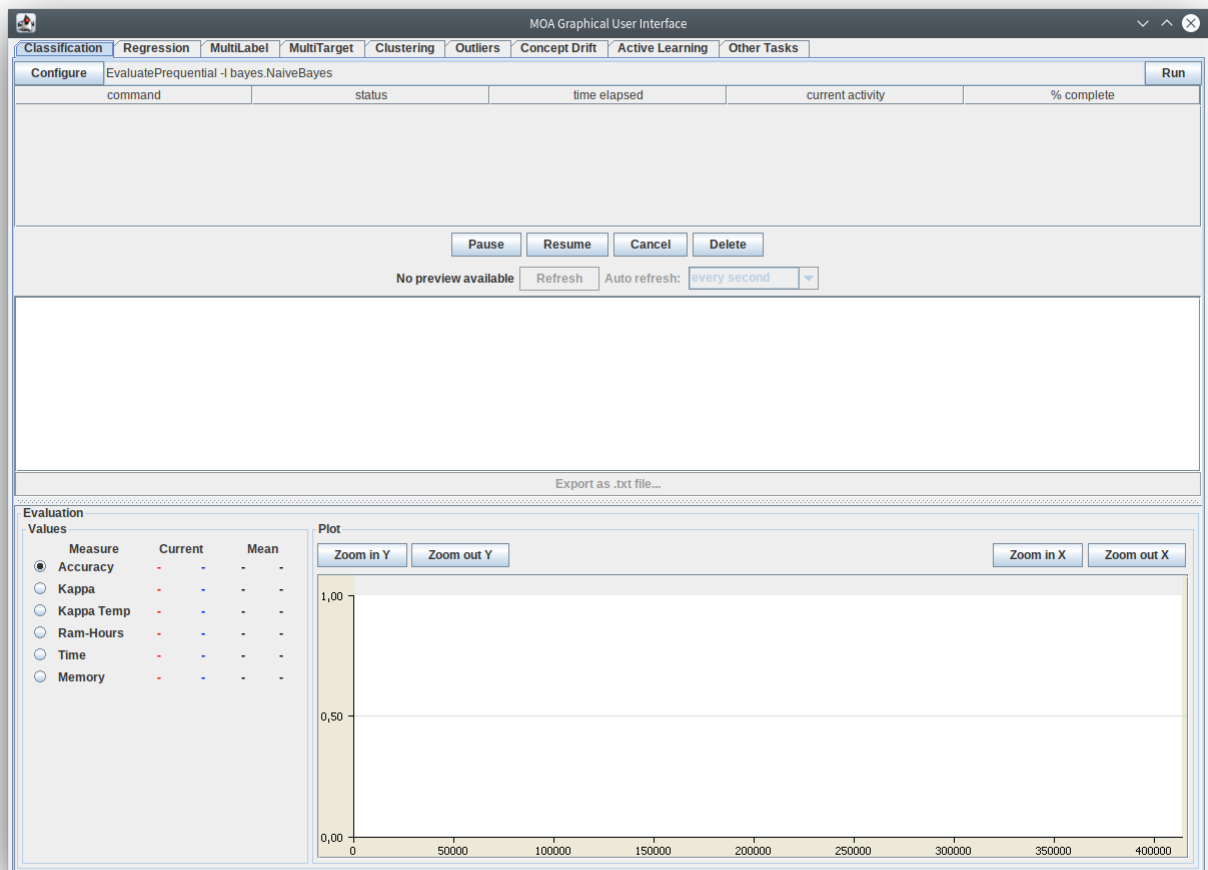


Figura 2.3 MOA - Tela Inicial

A aplicação é capaz de ler arquivos em formato *ARFF*, populares por serem utilizados no projeto WEKA (HALL et al., 2009). A ferramenta também permite a produção de fluxos de dados dinamicamente, através de geradores. Alguns dos geradores de fluxo disponíveis no MOA são: *Random Trees* (DOMINGOS; HULTEN, 2000) *SEA* (STREET; KIM, 2001), *STAGGER* (SCHLIMMER; GRANGER, 1986), *Rotating Hyperplane* (WANG et al., 2003), *Random RBF*, *LED* (GAMA; ROCHA; MEDAS, 2003), *Waveform* (GAMA; ROCHA; MEDAS, 2003), e *Function* (JIN; AGRAWAL, 2003).

Outra característica interessante do framework é a possibilidade de adicionar mudanças de conceito a fluxos estacionários existentes. Isto é feito através de uma função sigmóide, que modela o evento de mudança de conceito como uma combinação balance-

ada de duas distribuições homogêneas, que caracterizam os conceitos alvo antes e depois da mudança (BIFET; KIRKBY, 2009). Além destes conceitos, o usuário também pode definir o momento da mudança e a sua duração (BIFET et al., 2010).

Os principais métodos para detecção de mudança de conceito propostos na literatura estão disponíveis no MOA. O framework também permite a utilização de classificadores do WEKA (HALL et al., 2009) combinados aos detectores. A janela para configuração de um detector é demonstrada na Figura 2.4.

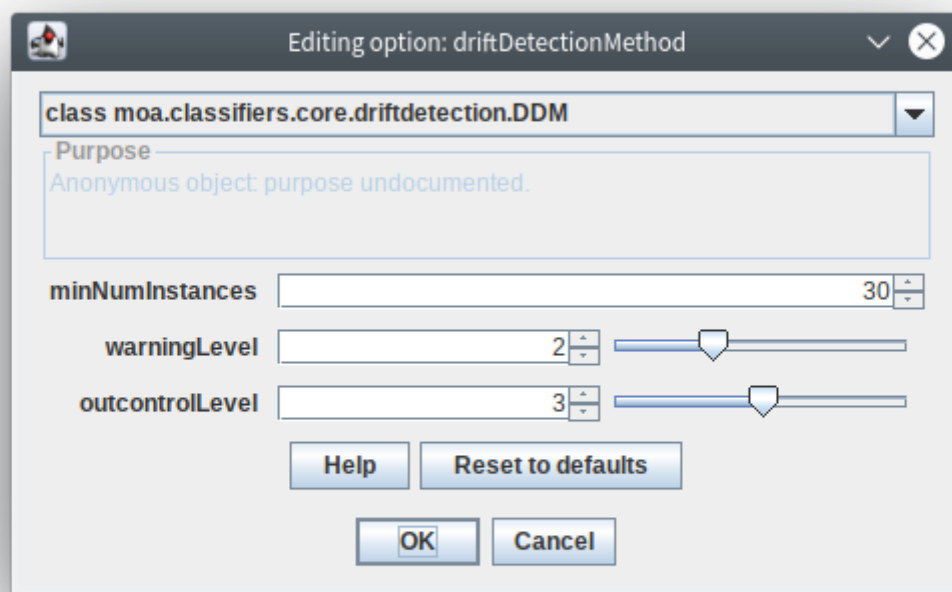


Figura 2.4 MOA - Configuração detector

A arquitetura do framework é modular, o que permite aos usuários implementar novas tarefas com pouco esforço. Por exemplo, para criar um novo detector, basta estender a classe abstrata `moa.classifiers.core.driftdetection.AbstractChangeDetector` e implementar o algoritmo desejado. A janela de configuração para o detector (similar a 2.4) é criada dinamicamente, a partir dos atributos da classe.

O MOA dispõe de diversas classes para avaliação de técnicas de aprendizado de máquina. Para este trabalho, destacam-se as classes `DriftDetectionMethodClassifier` e `BasicConceptDriftPerformanceEvaluator`, que realizam a análise de algoritmos para detecção de mudança de conceito. A classe `DriftDetectionMethodClassifier` permite

avaliar técnicas de detecção que encapsulam um classificador. Por sua vez, a classe `BasicConceptDriftPerformanceEvaluator` avalia a performance das técnicas de detecção diretamente, sem a necessidade de um classificador. Estes avaliadores e os seus indicadores serão detalhados juntamente com os resultados dos experimentos iniciais, na Seção 4.

2.3.4 Tornado

O Tornado é, assim como o MOA, um framework para mineração de dados para fluxos contínuos (PESARANGHADER, 2018). O projeto é desenvolvido na linguagem Python e seu código está disponível³. O framework diferencia-se do MOA por apresentar um cenário de avaliação específico: analisar a execução, em paralelo, de pares (classificador, detector de mudança de conceito), para identificar o par ótimo ao longo do tempo, em relação ao fluxo de dados.

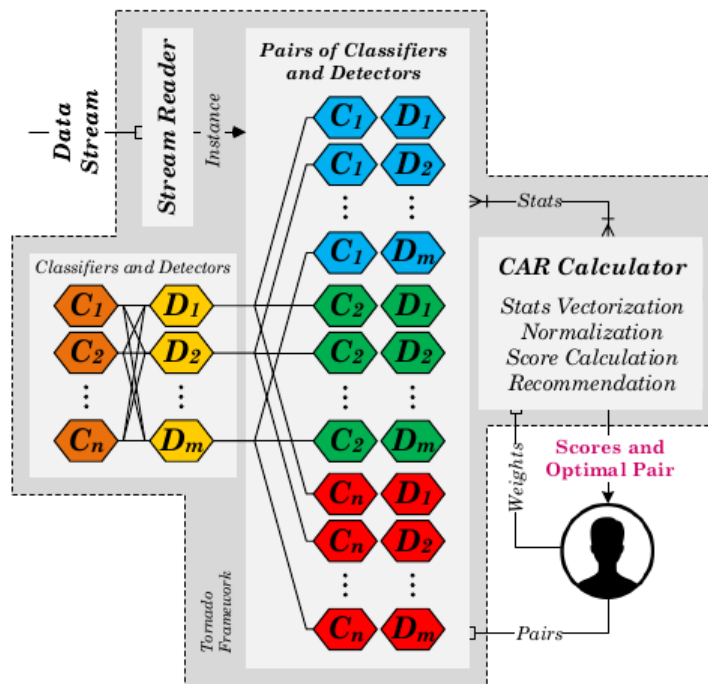


Figura 2.5 Framework Tornado. Fonte: (PESARANGHADER, 2018).

Conforme apresentado na Figura 2.5, os principais componentes do framework são: *Stream Reader*, *Classifiers*, *Detectors*, *Classifier-Detector Pairs* e *CAR Calculator*. A entrada de dados é composta por um fluxo (*Stream*), uma lista de pares (classificador, detector) e um vetor com pesos. O Tornado utiliza a abordagem de validação *prequential*, na qual as instâncias são testadas e depois utilizadas no aprendizado (GAMA et al., 2014).

O componente *Stream Reader* lê instâncias a partir do fluxo e as envia, uma a uma,

³<https://github.com/alipsggh/tornado>

para o par (classificador, detector), para construção do modelo. Os modelos são construídos de forma incremental. Por seguir a abordagem *prequential*, cada instância é primeiramente utilizada para testes e depois como treinamento. Simultaneamente, os classificadores enviam suas estatísticas (taxas de erro ou resultados das predições) aos detectores, para que a mudança de conceito possa ser sinalizada. Por fim, o componente *CAR Calculator* calcula uma pontuação para cada par, considerando taxas de erro, atraso para detecção da mudança de conceito, falsos positivos, falsos negativos, quantidade de memória utilizada e tempo de execução (PESARANGHADER, 2018). O framework apresenta ao usuário o par com maior pontuação. Este par, contudo, pode mudar devido ao aprendizado incremental ou à mudança de conceito.

A Figure 2.6 apresenta um exemplo de resultado obtido através do framework.

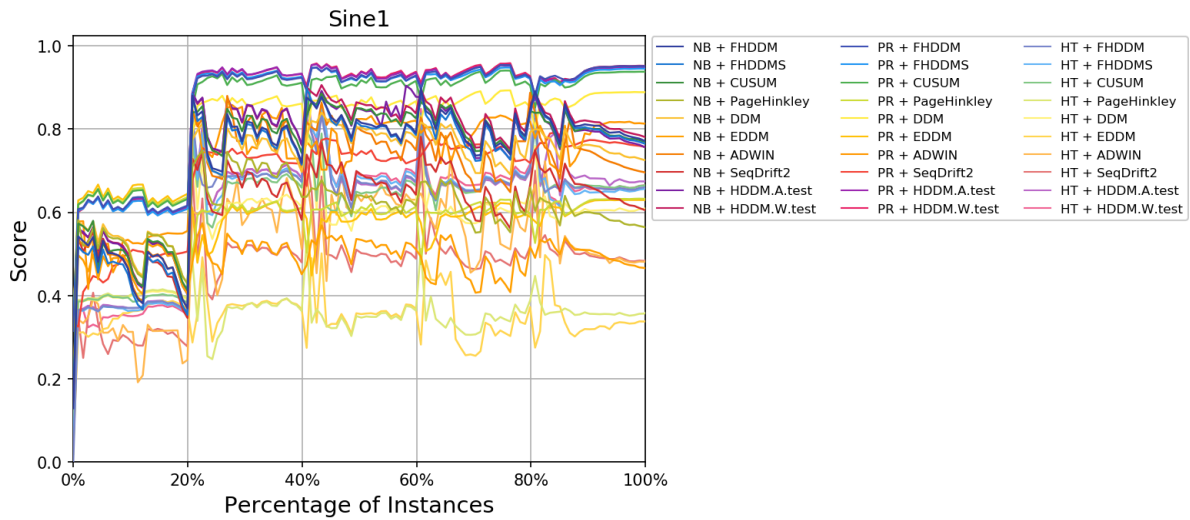


Figura 2.6 Tornado - Resultado para um múltiplos pares (PESARANGHADER, 2018)

2.4 REDES DE FUNÇÃO DE BASE RADIAL

As Redes de Função de Base Radial (RBF *networks*) são aproximadoras universais de funções. As RBFs têm como principal diferença em relação às outras redes neurais, a forma de ativação. Nessas redes, este processo é feito através do cálculo da distância entre os vetores de entrada e os centros estabelecidos (BRAGA; CARVALHO; LUDERMIR, 2007).

Em sua forma básica, a arquitetura de uma rede do tipo RBF é composta por três camadas: uma camada de entrada, uma intermediária e uma de saída (ROJAS, 1996). A camada intermediária (oculta) de uma RBF utiliza funções de base radiais para agrupar os dados de entrada em clusters, transformando padrões de entrada não linearmente separáveis em um conjunto de saídas linearmente separáveis. A camada de saída classifica os padrões recebidos através da combinação linear das saídas das funções (BRAGA; CARVALHO; LUDERMIR, 2007). A Figura 2.7 demonstra essa arquitetura.

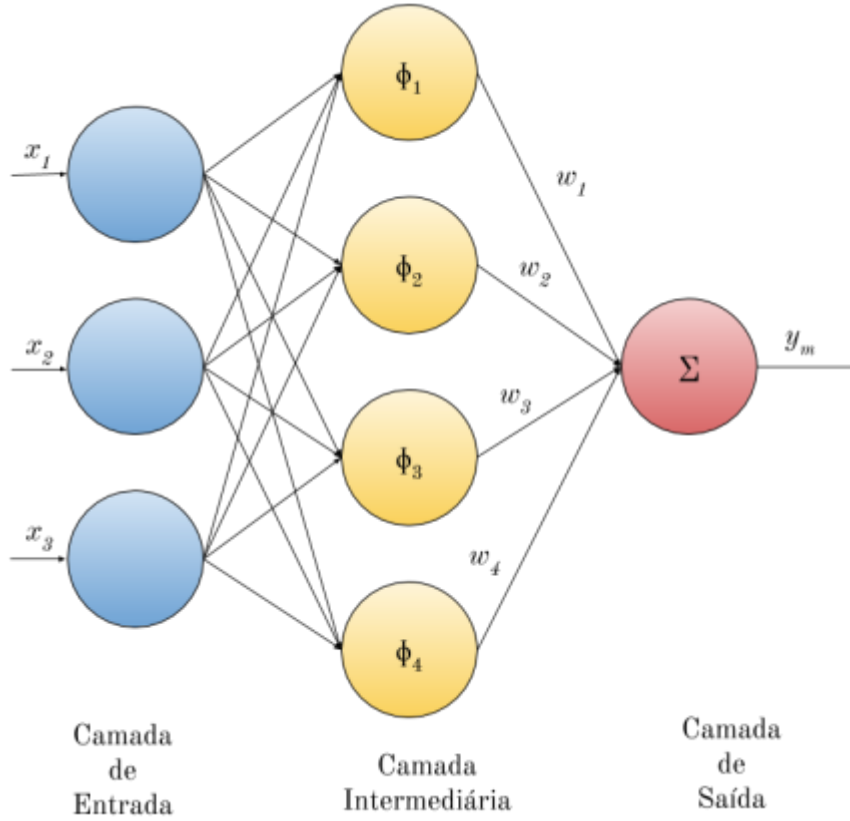


Figura 2.7 Arquitetura RBF

Na literatura, a função de base radial gaussiana (Eq. 2.4) é uma das mais utilizadas para camada intermediária.

$$f(x) = e^{-\frac{v^2}{2\sigma^2}} \quad (2.4)$$

Na equação 2.4, $v = \|x - t_i\|$ é dado pela distância euclidiana, onde x é o valor de entrada da rede, enquanto t_i e σ correspondem respectivamente ao centro e a largura da função radial. Dessa maneira, a resolução de um determinado problema por uma rede do tipo RBF consiste na resolução das funções 2.5 e 2.6 para chegar ao sistema 2.7.

$$f(x) = \sum_{i=1}^N w_{ji} \varphi(\|\mathbf{x} - \mathbf{t}_i\|) \quad (2.5)$$

$$y_i = \sum_{i=1}^N w_{ji} \varphi(\|\mathbf{x} - \mathbf{t}_i\|) + w_{j0} \quad (2.6)$$

Ajustar matriz

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2n} \\ & \dots & \dots & \dots & \dots \\ x_{d1} & x_{d2} & x_{d3} & \dots & x_{dn} \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{d1} & x_{d2} & x_{d3} & \dots & x_{dn} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{d1} & x_{d2} & x_{d3} & \dots & x_{dn} \end{bmatrix} \quad (2.7)$$

2.5 TRABALHOS RELACIONADOS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

2.6 CONSIDERAÇÕES FINAIS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

PLANO DE PESQUISA

3.1 CONSIDERAÇÕES INICIAIS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

3.2 DESCRIÇÃO DO PROBLEMA

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

3.2.1 Atividades de Pesquisa

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut

porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

3.3 CONSIDERAÇÕES FINAIS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

EXPERIMENTOS INICIAIS

4.1 CONSIDERAÇÕES INICIAIS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

4.2 CONFIGURAÇÃO DOS EXPERIMENTOS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

4.3 MÉTODO DE PETTITT

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut

porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

4.4 REDES DE FUNÇÃO DE BASE RADIAL

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

4.5 CONSIDERAÇÕES FINAIS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

REFERÊNCIAS BIBLIOGRÁFICAS

- ACKERMANN, M. R. et al. Streamkm++: A clustering algorithm for data streams. *J. Exp. Algorithmics*, ACM, New York, NY, USA, v. 17, p. 2.4:2.1–2.4:2.30, maio 2012. ISSN 1084-6654. Disponível em: <http://doi.acm.org/10.1145/2133803.2184450>.
- AGGARWAL, C. C. *Data Streams: Models and Algorithms (Advances in Database Systems)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN 0387287590.
- AGGARWAL, C. C. et al. A framework for clustering evolving data streams. In: *Proceedings of the 29th International Conference on Very Large Data Bases - Volume 29*. VLDB Endowment, 2003. (VLDB '03), p. 81–92. ISBN 0-12-722442-4. Disponível em: <http://dl.acm.org/citation.cfm?id=1315451.1315460>.
- AGGARWAL, C. C. et al. On demand classification of data streams. In: *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2004. (KDD '04), p. 503–508. ISBN 1-58113-888-1. Disponível em: <http://doi.acm.org/10.1145/1014052.1014110>.
- AMINIKHANGHAHI, S.; COOK, D. J. A survey of methods for time series change point detection. *Knowl. Inf. Syst.*, Springer-Verlag New York, Inc., New York, NY, USA, v. 51, n. 2, p. 339–367, maio 2017. ISSN 0219-1377. Disponível em: <https://doi.org/10.1007/s10115-016-0987-z>.
- ANKERST, M. et al. Optics: Ordering points to identify the clustering structure. *SIGMOD Rec.*, ACM, New York, NY, USA, v. 28, n. 2, p. 49–60, jun. 1999. ISSN 0163-5808. Disponível em: <http://doi.acm.org/10.1145/304181.304187>.
- BABCOCK, B. et al. Models and issues in data stream systems. In: *Proceedings of the Twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. New York, NY, USA: ACM, 2002. (PODS '02), p. 1–16. ISBN 1-58113-507-6. Disponível em: <http://doi.acm.org/10.1145/543613.543615>.
- Bach, S. H.; Maloof, M. A. Paired learners for concept drift. In: *2008 Eighth IEEE International Conference on Data Mining*. [S.l.: s.n.], 2008. p. 23–32. ISSN 1550-4786.
- BAENA-GARCÍA, M. et al. Early drift detection method. In: *In Fourth International Workshop on Knowledge Discovery from Data Streams*. [S.l.: s.n.], 2006.
- BARBARÁ, D. Requirements for clustering data streams. *SIGKDD Explor. Newsl.*, ACM, New York, NY, USA, v. 3, n. 2, p. 23–27, jan. 2002. ISSN 1931-0145. Disponível em: <http://doi.acm.org/10.1145/507515.507519>.

BARROS, R. S. M. de et al. RDDM: reactive drift detection method. *Expert Syst. Appl.*, v. 90, p. 344–355, 2017.

BASSEVILLE, M.; NIKIFOROV, I. V. *Detection of Abrupt Changes: Theory and Application*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1993. ISBN 0-13-126780-9.

BIFET, A. Adaptive learning and mining for data streams and frequent patterns. *SIGKDD Explor. Newsl.*, ACM, New York, NY, USA, v. 11, n. 1, p. 55–56, nov. 2009. ISSN 1931-0145. Disponível em: <http://doi.acm.org/10.1145/1656274.1656287>.

BIFET, A.; GAVALDÀ, R. Learning from time-changing data with adaptive windowing. In: *SDM*. SIAM, 2007. p. 443–448. ISBN 978-1-61197-277-1. Disponível em: <http://dblp.uni-trier.de/db/conf/sdm/sdm2007.html/#BifetG07>.

BIFET, A. et al. Moa: Massive online analysis. *J. Mach. Learn. Res.*, JMLR.org, v. 11, p. 1601–1604, ago. 2010. ISSN 1532-4435. Disponível em: <http://dl.acm.org/citation.cfm?id=1756006.1859903>.

BIFET, A.; KIRKBY, R. Data stream mining a practical approach. Citeseer, 2009.

BIFET, A. et al. Efficient data stream classification via probabilistic adaptive windows. In: *Proceedings of the 28th Annual ACM Symposium on Applied Computing*. New York, NY, USA: ACM, 2013. (SAC '13), p. 801–806. ISBN 978-1-4503-1656-9. Disponível em: <http://doi.acm.org/10.1145/2480362.2480516>.

BLANCO, I. I. F. et al. Online and non-parametric drift detection methods based on hoeffding's bounds. *IEEE Trans. Knowl. Data Eng.*, v. 27, n. 3, p. 810–823, 2015.

BLUM, A. Empirical support for winnow and weighted-majority algorithms: Results on a calendar scheduling domain. *Machine Learning*, v. 26, n. 1, p. 5–23, Jan 1997. ISSN 1573-0565. Disponível em: <https://doi.org/10.1023/A:1007335615132>.

BRAGA, A.; CARVALHO, A. C.; LUDERMIR, T. B. *Redes Neurais Artificiais: Teoria e aplicações*. LTC Editora, 2007. ISBN 9788521615644. Disponível em: <http://www.worldcat.org/isbn/9788521615644>.

BREIMAN, L. et al. *Classification and Regression Trees*. Monterey, CA: Wadsworth and Brooks, 1984.

BRZEZIŃSKI, D.; STEFANOWSKI, J. Accuracy updated ensemble for data streams with concept drift. In: CORCHADO, E.; KURZYŃSKI, M.; WOŹNIAK, M. (Ed.). *Hybrid Artificial Intelligent Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 155–163. ISBN 978-3-642-21222-2.

CAO, F. et al. Density-based clustering over an evolving data stream with noise. In: GHOSH, J. et al. (Ed.). *SDM*. SIAM, 2006. p. 328–339. ISBN 978-1-61197-276-4. Disponível em: <http://dblp.uni-trier.de/db/conf/sdm/sdm2006.html/#CaoEQZ06>.

CHANDOLA, V.; BANERJEE, A.; KUMAR, V. Anomaly detection: A survey. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 41, n. 3, p. 15:1–15:58, jul. 2009. ISSN 0360-0300. Disponível em: <http://doi.acm.org/10.1145/1541880.1541882>.

COSTA, F. da; RIOS, R.; MELLO, R. de. Using dynamical systems tools to detect concept drift in data streams. *Expert Syst. Appl.*, Pergamon Press, Inc., Tarrytown, NY, USA, v. 60, n. C, p. 39–50, out. 2016. ISSN 0957-4174. Disponível em: <https://doi.org/10.1016/j.eswa.2016.04.026>.

Ditzler, G.; Polikar, R. Hellinger distance based drift detection for nonstationary environments. In: *2011 IEEE Symposium on Computational Intelligence in Dynamic and Uncertain Environments (CIDUE)*. [S.l.: s.n.], 2011. p. 41–48.

DOMINGOS, P.; HULTEN, G. Mining high-speed data streams. In: *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2000. (KDD '00), p. 71–80. ISBN 1-58113-233-6. Disponível em: <http://doi.acm.org/10.1145/347090.347107>.

DREDZE, M.; OATES, T.; PIATKO, C. We're not in kansas anymore: Detecting domain changes in streams. In: . [s.n.], 2010. p. 585–595. Cited By 13. Disponível em: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-80053285630&partnerID=40&md5=2eea89f635e2cbc0920069028e9f7746>.

DRIES, A.; RÜCKERT, U. Adaptive concept drift detection. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, v. 2, n. 5-6, p. 311–327, 2009. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/sam.10054>.

DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern Classification (2Nd Edition)*. New York, NY, USA: Wiley-Interscience, 2000. ISBN 0471056693.

ESTER, M. et al. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. AAAI Press, 1996. (KDD'96), p. 226–231. Disponível em: <http://dl.acm.org/citation.cfm?id=3001460.3001507>.

FARIA, E. R.; GAMA, J. a.; CARVALHO, A. C. P. L. F. Novelty detection algorithm for data streams multi-class problems. In: *Proceedings of the 28th Annual ACM Symposium on Applied Computing*. New York, NY, USA: ACM, 2013. (SAC '13), p. 795–800. ISBN 978-1-4503-1656-9. Disponível em: <http://doi.acm.org/10.1145/2480362.2480515>.

FEIGENBAUM, J. et al. An approximate l1-difference algorithm for massive data streams. *SIAM J. Comput.*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, v. 32, n. 1, p. 131–151, jan. 2003. ISSN 0097-5397. Disponível em: <https://doi.org/10.1137/S0097539799361701>.

FISHER, D. H. Knowledge acquisition via incremental conceptual clustering. *Mach. Learn.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 2, n. 2, p. 139–172, set. 1987. ISSN 0885-6125. Disponível em: <https://doi.org/10.1023/A:1022852608280>.

GAMA, J. *Knowledge Discovery from Data Streams*. 1st. ed. [S.l.]: Chapman & Hall/CRC, 2010. ISBN 1439826110, 9781439826119.

GAMA, J. et al. Learning with drift detection. In: BAZZAN, A. L. C.; LABIDI, S. (Ed.). *SBIA*. Springer, 2004. (Lecture Notes in Computer Science, v. 3171), p. 286–295. ISBN 3-540-23237-0. Disponível em: <http://dblp.uni-trier.de/db/conf/sbia/sbia2004.html/#GamaMCR04>.

GAMA, J.; RODRIGUES, P. An overview on mining data streams. In: _____. [S.l.: s.n.], 2009. v. 206, p. 29–45.

GAMA, J. a.; ROCHA, R.; MEDAS, P. Accurate decision trees for mining high-speed data streams. In: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2003. (KDD '03), p. 523–528. ISBN 1-58113-737-0. Disponível em: <http://doi.acm.org/10.1145/956750.956813>.

GAMA, J. a. et al. A survey on concept drift adaptation. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 46, n. 4, p. 44:1–44:37, mar. 2014. ISSN 0360-0300. Disponível em: <http://doi.acm.org/10.1145/2523813>.

GAO, J. et al. A general framework for mining concept-drifting data streams with skewed distributions. In: *SDM*. [S.l.]: SIAM, 2007. p. 3–14.

GONÇALVES, P. M. et al. A comparative study on concept drift detectors. *Expert Systems with Applications*, v. 41, n. 18, p. 8144 – 8156, 2014. ISSN 0957-4174. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417414004175>.

GUHA, S. et al. Clustering data streams: Theory and practice. *IEEE Trans. on Knowl. and Data Eng.*, IEEE Educational Activities Department, Piscataway, NJ, USA, v. 15, n. 3, p. 515–528, mar. 2003. ISSN 1041-4347. Disponível em: <http://dx.doi.org/10.1109/TKDE.2003.1198387>.

GUHA, S. et al. Clustering data streams. In: *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*. Washington, DC, USA: IEEE Computer Society, 2000. (FOCS '00), p. 359–. ISBN 0-7695-0850-2. Disponível em: <http://dl.acm.org/citation.cfm?id=795666.796588>.

HALL, M. et al. The weka data mining software: An update. *SIGKDD Explor. Newsl.*, ACM, New York, NY, USA, v. 11, n. 1, p. 10–18, nov. 2009. ISSN 1931-0145. Disponível em: <http://doi.acm.org/10.1145/1656274.1656278>.

Hayat, M. Z.; Hashemi, M. R. A dct based approach for detecting novelty and concept drift in data streams. In: *2010 International Conference of Soft Computing and Pattern Recognition*. [S.l.: s.n.], 2010. p. 373–378.

JAIN, A. K.; DUBES, R. C. *Algorithms for Clustering Data*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1988. ISBN 0-13-022278-X.

JIN, R.; AGRAWAL, G. Efficient decision tree construction on streaming data. In: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2003. (KDD '03), p. 571–576. ISBN 1-58113-737-0. Disponível em: <http://doi.acm.org/10.1145/956750.956821>.

KATAKIS, I.; TSOUMAKAS, G.; VLAHAVAS, I. Tracking recurring contexts using ensemble classifiers: An application to email filtering. *Knowl. Inf. Syst.*, Springer-Verlag, Berlin, Heidelberg, v. 22, n. 3, p. 371–391, mar. 2010. ISSN 0219-1377. Disponível em: <https://doi.org/10.1007/s10115-009-0206-2>.

KAUFMAN, L.; ROUSSEEUW, P. *Finding Groups in Data: an introduction to cluster analysis*. [S.l.]: Wiley, 1990.

KHALILIAN, M.; MUSTAPHA, N. Data stream clustering: Challenges and issues. *CoRR*, abs/1006.5261, 2010. Disponível em: <http://arxiv.org/abs/1006.5261>.

KOLTER, J. Z.; MALOOF, M. A. Dynamic weighted majority: An ensemble method for drifting concepts. *J. Mach. Learn. Res.*, JMLR.org, v. 8, p. 2755–2790, dez. 2007. ISSN 1532-4435. Disponível em: <http://dl.acm.org/citation.cfm?id=1314498.1390333>.

KOTSIANTIS, S. B. Supervised machine learning: A review of classification techniques. In: *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2007. p. 3–24. ISBN 978-1-58603-780-2. Disponível em: <http://dl.acm.org/citation.cfm?id=1566770.1566773>.

KRANEN, P. et al. The clustree: Indexing micro-clusters for anytime stream mining. *Knowl. Inf. Syst.*, Springer-Verlag New York, Inc., New York, NY, USA, v. 29, n. 2, p. 249–272, nov. 2011. ISSN 0219-1377. Disponível em: <http://dx.doi.org/10.1007/s10115-010-0342-8>.

KUNCHEVA, L. Classifier ensembles for detecting concept change in streaming data: Overview and perspectives. *Proc. Eur. Conf. Artif. Intell.*, p. 5–10, 2008. Cited By 70.

LANE, T.; BRODLEY, C. E. Approaches to online learning and concept drift for user identification in computer security. In: *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*. AAAI Press, 1998. (KDD'98), p. 259–263. Disponível em: <http://dl.acm.org/citation.cfm?id=3000292.3000339>.

Lee, J.; Magoulès, F. Detection of concept drift for learning from stream data. In: *2012 IEEE 14th International Conference on High Performance Computing and Communication 2012 IEEE 9th International Conference on Embedded Software and Systems*. [S.l.: s.n.], 2012. p. 241–245.

Lee, Y. K.; Wang, L.; Ryu, K. H. A system architecture for monitoring sensor data stream. In: *7th IEEE International Conference on Computer and Information Technology (CIT 2007)*. [S.l.: s.n.], 2007. p. 1026–1031.

LINDSTROM, P.; NAMEE, B. M.; DELANY, S. J. Drift detection using uncertainty distribution divergence. *Evolving Systems*, v. 4, n. 1, p. 13–25, Mar 2013. ISSN 1868-6486. Disponível em: <https://doi.org/10.1007/s12530-012-9061-6>.

LING, C.; LING-JUN, Z.; LI, T. Stream data classification using improved fisher discriminate analysis. *Journal of Computers*, 01 2009.

LLOYD, S. Least squares quantization in pcm. *IEEE Trans. Inf. Theor.*, IEEE Press, Piscataway, NJ, USA, v. 28, n. 2, p. 129–137, set. 2006. ISSN 0018-9448. Disponível em: <http://dx.doi.org/10.1109/TIT.1982.1056489>.

MASUD, M. et al. Classification and novel class detection in concept-drifting data streams under time constraints. *IEEE Trans. on Knowl. and Data Eng.*, IEEE Educational Activities Department, Piscataway, NJ, USA, v. 23, n. 6, p. 859–874, jun. 2011. ISSN 1041-4347. Disponível em: <http://dx.doi.org/10.1109/TKDE.2010.61>.

MINKU, L. L.; YAO, X. Ddd: A new ensemble approach for dealing with concept drift. *IEEE Trans. on Knowl. and Data Eng.*, IEEE Educational Activities Department, Piscataway, NJ, USA, v. 24, n. 4, p. 619–633, abr. 2012. ISSN 1041-4347. Disponível em: <http://dx.doi.org/10.1109/TKDE.2011.58>.

MITCHELL, T. M. *Machine Learning*. 1. ed. New York, NY, USA: McGraw-Hill, Inc., 1997. ISBN 0070428077, 9780070428072.

NISHIDA, K.; YAMAUCHI, K. Detecting concept drift using statistical testing. In: CORRUBLE, V.; TAKEDA, M.; SUZUKI, E. (Ed.). *Discovery Science*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. p. 264–269. ISBN 978-3-540-75488-6.

PAGE, E. S. Continuous Inspection Schemes. *Biometrika*, Biometrika Trust, v. 41, n. 1/2, p. 100–115, 1954. ISSN 00063444. Disponível em: <http://dx.doi.org/10.2307/2333009>.

PEARS, R.; SAKTHITHASAN, S.; KOH, Y. S. Detecting concept change in dynamic data streams - A sequential approach based on reservoir sampling. *Machine Learning*, v. 97, n. 3, p. 259–293, 2014.

PESARANGHADER, A. *A Reservoir of Adaptive Algorithms for Online Learning from Evolving Data Streams*. Université d'Ottawa / University of Ottawa, 2018. Disponível em: <http://ruor.uottawa.ca/handle/10393/38190>.

PINTO, C. M. S.; GAMA, J.; PORTO, U. do. Algoritmos incrementais para aprendizagem bayesiana. *Faculdade de Economia da Universidade do Porto*, 2005.

REIMANN, P.; SPADA, H. *Learning in Humans and Machines: Towards an Interdisciplinary Learning Science*. Emerald Group Publishing Limited, 1995. ISBN 9780080425696. Disponível em: <https://books.google.com.br/books?id=YkdyQgAACAAJ>.

ROBERTS, S. W. Control chart tests based on geometric moving averages. *Technometrics*, American Society for Quality Control and American Statistical Association, Alexandria, Va, USA, v. 42, n. 1, p. 97–101, fev. 2000. ISSN 0040-1706. Disponível em: <http://dx.doi.org/10.2307/1271439>.

ROJAS, R. *Neural Networks: A Systematic Introduction*. Berlin, Heidelberg: Springer-Verlag, 1996. ISBN 3-540-60505-3.

ROSS, G. J. et al. Exponentially weighted moving average charts for detecting concept drift. *Pattern Recogn. Lett.*, Elsevier Science Inc., New York, NY, USA, v. 33, n. 2, p. 191–198, jan. 2012. ISSN 0167-8655. Disponível em: <http://dx.doi.org/10.1016/j.patrec.2011.08.019>.

RYU, J. W. et al. An efficient method of building an ensemble of classifiers in streaming data. In: SRINIVASA, S.; BHATNAGAR, V. (Ed.). *Big Data Analytics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 122–133. ISBN 978-3-642-35542-4.

SANTOS, S. G. T. de C. et al. Speeding up recovery from concept drifts. In: *Proceedings of the 2014th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part III*. Berlin, Heidelberg: Springer-Verlag, 2014. (ECMLPKDD'14), p. 179–194. ISBN 978-3-662-44844-1. Disponível em: https://doi.org/10.1007/978-3-662-44845-8_12.

SCHLIMMER, J. C.; GRANGER, R. H. Incremental learning from noisy data. *Machine Learning*, v. 1, n. 3, p. 317–354, Sep 1986. ISSN 1573-0565. Disponível em: <https://doi.org/10.1007/BF00116895>.

SEBASTIÃO, R. et al. Monitoring incremental histogram distribution for change detection in data streams. In: GABER, M. M. et al. (Ed.). *Knowledge Discovery from Sensor Data*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 25–42. ISBN 978-3-642-12519-5.

SETHI, T. S.; KANTARDZIC, M. On the reliable detection of concept drift from streaming unlabeled data. *Expert Syst. Appl.*, Pergamon Press, Inc., Tarrytown, NY, USA, v. 82, n. C, p. 77–99, out. 2017. ISSN 0957-4174. Disponível em: <https://doi.org/10.1016/j.eswa.2017.04.008>.

SETHI, T. S.; KANTARDZIC, M.; HU, H. A grid density based framework for classifying streaming data in the presence of concept drift. *Journal of Intelligent Information Systems*, v. 46, n. 1, p. 179–211, Feb 2016.

SMAILOVIĆ, J. et al. Stream-based active learning for sentiment analysis in the financial domain. *Inf. Sci.*, Elsevier Science Inc., New York, NY, USA, v. 285, n. C, p. 181–203, nov. 2014. ISSN 0020-0255. Disponível em: <https://doi.org/10.1016/j.ins.2014.04.034>.

SOBHANI, P.; BEIGY, H. New drift detection method for data streams. In: *Proceedings of the Second International Conference on Adaptive and Intelligent Systems*. Berlin, Heidelberg: Springer-Verlag, 2011. (ICAIS'11), p. 88–97. ISBN 978-3-642-23856-7. Disponível em: <http://dl.acm.org/citation.cfm?id=2045295.2045309>.

SPINOSA, E. J.; CARVALHO, A. P. de Leon F. de; GAMA, J. a. Olindda: A cluster-based approach for detecting novelty and concept drift in data streams. In: *Proceedings of the 2007 ACM Symposium on Applied Computing*. New York, NY, USA: ACM, 2007. (SAC '07), p. 448–452. ISBN 1-59593-480-4. Disponível em: <http://doi.acm.org/10.1145/1244002.1244107>.

STREET, W. N.; KIM, Y. A streaming ensemble algorithm (sea) for large-scale classification. In: *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2001. (KDD '01), p. 377–382. ISBN 1-58113-391-X. Disponível em: <http://doi.acm.org/10.1145/502512.502568>.

VAPNIK, V. N. *Statistical Learning Theory*. [S.l.]: Wiley-Interscience, 1998.

WANG, H. et al. Mining concept-drifting data streams using ensemble classifiers. In: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2003. (KDD '03), p. 226–235. ISBN 1-58113-737-0. Disponível em: <http://doi.acm.org/10.1145/956750.956778>.

WIDMER, G.; KUBAT, M. Learning in the presence of concept drift and hidden contexts. *Mach. Learn.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 23, n. 1, p. 69–101, abr. 1996. ISSN 0885-6125. Disponível em: <http://dx.doi.org/10.1023/A:1018046501280>.

ZLIOBAITE, I. Learning under concept drift: an overview. *CoRR*, abs/1010.4784, 2010. Disponível em: <http://arxiv.org/abs/1010.4784>.

ZWOLENSKI, M.; WEATHERILL, L. The digital universe rich data and the increasing value of the internet of things. *Australian Journal of Telecommunications and the Digital Economy*, v. 2, 10 2014.