Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Aprendizado em Fluxos de Dados: Explorando Vantagens da Teoria *Fuzzy* no Contexto de Agrupamento *Online* Semissupervisionado

Priscilla de Abreu Lopes

Orientadora: Profa. Dra. Heloisa de Arruda Camargo

São Carlos, SP, Brasil

Setembro/2016
v-1.0

Lista de Abreviaturas e Siglas

AH	Árvores de Hoeffding
AFD	Aprendizado em Fluxos de Dados
AM	Aprendizado de Máquina
AP	Affinity Propagation
cmc	core-micro-cluster
CVFDT	Concept-adapting Very Fast Decision Tree
EC	Ensemble de Classificadores
FD	Fluxos de Dados
FCM	Fuzzy C-Means
FPT	Fuzzy Pattern Trees
FOO	Framework Online-Offline
PA	Passive-Agressive
pcmc	potential core-micro-cluster
SVM	Support Vector Machine
VFDT	Very Fast Decision Tree

Lista de Símbolos

E	Conjunto de exemplos
e_j	Um exemplo j qualquer do conjunto E
n	Número de exemplos em um conjunto
k	Número de grupos
C	Conjunto de centróides ou protótipos que representam k grupos
c_i	Um centróide ou protótipo i qualquer do conjunto C
U	Matriz de pertinência de cada exemplo e_j a cada grupo c_i
m	Constante de fuzzificação
ξ	Diferença máxima entre dois centróides ou protótipos

Sumário

Introdução

Parte I

Revisão Bibliográfica

1

Aprendizado de Máquina Clássico

O Aprendizado de Máquina (AM) refere-se à investigação de métodos computacionais capazes de adquirir conhecimento de forma automática. Distintas abordagens foram propostas para a realização do processo de aprendizagem desde a formalização do surgimento desta área de pesquisa na década de 80 (LANGLEY, 2011).

Neste capítulo são apresentados conceitos gerais que fundamentam a compreensão do problema tratado neste trabalho. Tais conceitos relacionam-se principalmente a aprendizado não supervisionado e semissupervisionado baseados em técnicas de agrupamento particional.

1.1 Aprendizado Não Supervisionado

No contexto de AM, o aprendizado indutivo é um dos principais mecanismos para derivar conhecimento novo e predizer escritos futuros. Nessa metodologia o aprendizado ocorre por meio de inferência indutiva sobre um conjunto de exemplos, onde um exemplo (também chamado de dado, instância ou objeto) é descrito por um conjunto de atributos (MITCHELL, 1997). O aprendizado indutivo pode ser dividido em três abordagens: supervisionada, não supervisionada e semissupervisionada.

Abordagens supervisionadas são aquelas que realizam a extração de conhecimento pelo desenvolvimento de um modelo geral baseado em um conjunto de exemplos que possui um atributo especial, chamado classe (ou rótulo), que representa o conceito que se deseja aprender. Um exemplo de um conjunto de exemplos é dito rotulado se a classe à qual pertence é conhecida. Métodos conhecidos como de classificação tipicamente utilizam-se de conjuntos totalmente rotulados e, portanto, pertencem à categoria de aprendizado supervisionado. Estes métodos são amplamente utilizados por produzirem bons resultados (WITTEN; FRANK, 2005).

Aplicações de árvores de decisão (QUINLAN, 1986), redes neurais (BISHOP, 1995), métodos estatísticos (DUDA; HART, 1973) e genéticos (GOLDBERG, 1989) fazem

parte do conjunto de paradigmas para a resolução do problema de classificação (MITCHELL, 1997). Existem métodos, como o *K-Nearest Neightbors* (COVER; HART, 1967), que não geram classificadores, mas utilizam a informação de rótulos para classificar novos exemplos, atribuindo classes por meio de métricas de similaridade.

Variações de métodos de classificação baseados na teoria de conjuntos *fuzzy* (ZADEH, 1965) podem realizar a indução de regras que permitem a representação de conhecimento impreciso a partir de um conjunto de exemplos (PEDRYCZ; GOMIDE, 1998). Sistemas neuro-*fuzzy* (KLOSE et al., 2001) se utilizam de algoritmos de aprendizado derivados da teoria de redes neurais para gerar regras *fuzzy*. Outras abordagens são baseadas em árvores de decisão, que podem ser induzidas e, posteriormente, ter regras extraídas da estrutura resultante (QUINLAN, 1993). Propostas para extensões chamadas árvores de decisão *fuzzy* também podem ser encontradas na literatura (JANIKOW, 1998; CINTRA; MONARD; CAMARGO, 2012).

Estratégias evolutivas, como Algoritmos Genéticos, são utilizados na otimização e criação de sistemas *fuzzy*. Inicialmente, os chamados Sistemas *Fuzzy* Genéticos, possuíam grande foco na geração de sistemas com alta acurácia (CORDÓN, 2011). Este paradigma foi modificado e há nas pesquisas mais recentes uma preocupação em aproveitar o potencial de interpretabilidade dos conjuntos *fuzzy* para a geração e otimização de sistemas que, além de alta acurácia, sejam mais claros e interpretáveis para seres humanos (CORDÓN, 2011; FAZZOLARI et al., 2013).

Apesar dos bons resultados produzidos por técnicas supervisionadas, é possível que os rótulos não estejam disponíveis para determinados domínios, impedindo sua aplicação. Nesses cenários costumam ser aplicadas técnicas não supervisionadas de aprendizado.

1.1.1 Agrupamento de Dados

Agrupamento de dados é uma típica técnica não supervisionada, ou seja, um processo capaz de realizar aprendizagem a partir de um conjunto de exemplos não rotulado. A aplicação de agrupamento tem como objetivo definir uma possível partição dos dados em grupos, de forma que exemplos semelhantes pertençam a um mesmo grupo e exemplos distintos pertençam a grupos distintos (JAIN; MURTY; FLYNN, 1999). Essa divisão dos dados é baseada em métricas que determinam a relação de dissimilaridade ou similaridade entre diferentes exemplos.

As diferentes técnicas de agrupamento podem ser divididas nas seguintes categorias (HAN; KAMBER; PEI, 2012):

Hierárquico: cria uma decomposição hierárquica de um conjunto de exemplos de acordo com algum critério (DAY; EDELSBRUNNER, 1984; KAUFMAN; ROUSSEEUW, 1990; ZHANG; RAMAKRISHNAN; LIVNY, 1996);

Particional: constrói uma partição inicial de um conjunto de exemplos e, por meio de um processo iterativo, busca melhorar a partição, mudando exemplos de grupo baseado, geralmente, em uma medida de distância (MACQUEEN, 1967; BEZDEK, 1981; KAUFMAN; ROUSSEEUW, 1990);

Baseado em Densidade: capaz de criar uma partição ou uma decomposição hierárquica de um conjunto de exemplos baseado na ideia geral de que a vizinhança de cada exemplo de um grupo, dentro de um raio determinado, possui um mínimo de pontos, ou seja, a densidade na vizinhança deve exceder um limiar definido. (ESTER et al., 1996; HINNEBURG; KEIM, 1998; ANKERST et al., 1999);

Baseado em Grades: todas as operações de agrupamento são realizadas dentro de uma estrutura de grades (*grid*), que é uma divisão do espaço dos exemplos em um número finito de células (WANG; YANG; MUNTZ, 1997; SHEIKHOLESLAMI; CHATTERJEE; ZHANG, 2000).

Uma transição menos abrupta, comentando algo sobre a estrutura obtida pelo agrupamento hierárquico (desnecessária e pode atrapalhar para alguns domínios) e para as outras categorias mencionadas, talvez? E sobre as categorias baseadas em grades (dimensão do *grid* pode tornar a abordagem inviável)? E sobre baseadas em densidade... É necessário justificar o foco na abordagem particional?

O algoritmo particional *k-means* é um dos mais populares e simples algoritmos de agrupamento, ainda sendo amplamente utilizado e, muitas vezes, servindo de base ao desenvolvimento de novos algoritmos. O objetivo do *k-means* é agrupar os dados em k grupos disjuntos, de maneira que a soma das distâncias entre os exemplos pertencentes a um grupo e seu respectivo centro seja mínima. O centróide de grupo, ou protótipo, representa o ponto médio dos pontos pertencentes a um determinado grupo.

Uma variedade de técnicas agrupamento que utilizam conceitos da teoria de conjuntos *fuzzy* são encontradas na literatura e visam obter melhor resultado da aprendizagem pela generalização do conceito de grupo. O *Fuzzy C-Means* (FCM) (BEZDEK, 1981) é uma proposta pioneira e uma das primeiras extensões *fuzzy* do algoritmo *k-means* (MACQUEEN, 1967).

No FCM a partição dos dados é realizada em grupos que podem ser não disjuntos, ou seja, cada exemplo está relacionado a cada grupo por um grau de pertinência. O Algoritmo 1.1 apresenta o processo geral de agrupamento para o FCM. As variáveis de entrada são um conjunto de n exemplos ($E = \{e_1, e_2, ..., e_N\}$), o número de grupos ($k$) para partição dos exemplos, uma constante de fuzzificação ($m > 1$) e a diferença máxima entre os centróides obtidos nas duas últimas iterações (ξ). Os resultados da aplicação do FCM são um conjunto de centróides finais ($C = \{c_1, c_2, ..., c_k\}$) e uma matriz $k \times N$ com os valores de pertinência para cada exemplo e_j em cada grupo, representada por U.

Algoritmo 1.1: *Fuzzy C-Means* (FCM) (BEZDEK, 1981)

Entrada: E, k, m, ξ
Saída: U, C

```
início
    U = geraMatrizPertinênciaAleatória();
    C = geraCentróidesIniciais(E, U);
    ϵ = ∞;
    enquanto ϵ > ξ faça
        U = atualizarMatrizPertinência(E, C);
        C' = C;
        C = atualizarCentróides(E, U);
        ϵ = max_{1≤i≤k}{‖c_i − c'_i‖²};
    fim
fim
```

A geração inicial e atualização dos centróides é calculada pela Equação 1.1 e as atualizações para a matriz de pertinência seguem a Equação 1.2.

$$c_i = \frac{\sum_{j=1}^{n} u_{ij}^m e_j}{\sum_{j=1}^{n} u_{ij}^m} \tag{1.1}$$

$$u_{ij} = \left[\sum_{l=1}^{k} \left(\frac{\|e_j - c_i\|}{\|e_j - c_l\|} \right)^{\frac{2}{m-1}} \right]^{-1}, \forall i, j \tag{1.2}$$

Bezdek (1981) propõe ainda a versão *Weighted Fuzzy C-Means* (WFCM), que inclui um fator de ponderação w_j para cada exemplo e_j de E para considerar a influência de um exemplo sobre o processo de agrupamento. No FCM o fator de ponderação é igual a 1 (um) para todos os exemplos, ou seja, todos os exemplos são igualmente importantes para o agrupamento. O Algoritmo 1.2 apresenta o processo geral para o WFCM. Com relação ao processo descrito para o FCM, o WFCM conta com uma entrada adicional: um vetor w contendo um conjunto de pesos $\{w_1, w_2, ..., w_n\}$, onde $w_j \geq 0$ determina a influência do exemplo e_j para o processo de agrupamento.

As atualizações para a matriz de pertinência seguem a Equação 1.2, enquanto a geração inicial e a atualização dos centróides é calculada pela Equação 1.3.

$$c_i = \frac{\sum_{j=1}^{n} w_j u_{ij}^m e_j}{\sum_{j=1}^{n} w_j u_{ij}^m} \tag{1.3}$$

Uma transição menos abrupta

Problemas como forte dependência de medidas de distância e normalização dos dados, definição do número correto de grupos para a divisão são observados quando aplicadas técnicas de agrupamento não supervisionadas.

Algoritmo 1.2: *Weighted Fuzzy C-Means* (WFCM) (BEZDEK, 1981)

Entrada: E, k, m, ξ, w
Saída: U, C

```
início
    U = geraMatrizPertinênciaAleatória();
    C = geraCentróidesIniciais(E, U);
    ϵ = ∞;
    enquanto ϵ > ξ faça
        U = atualizarMatrizPertinência(E, C);
        C' = C;
        C = atualizarCentróides(E, U);
        ϵ = max_{1≤i≤k}{‖c_i − c'_i‖²};
    fim
fim
```

O crescimento acelerado de conjuntos de exemplos em muitos domínios torna a rotulação manual e total dos dados onerosa. A aplicação de técnicas supervisionadas pode ser prejudicada por utilizar apenas uma pequena quantidade de dados rotulados. Ao mesmo tempo, a utilização de técnicas não supervisionadas desconsideraria totalmente esse conhecimento prévio disponível no processo de aprendizagem. Nesse contexto, surge a ideia de aprendizado semissupervisionado, apresentada na Seção 1.2.

1.2 Aprendizado Semissupervisionado

A ideia de exploração de informações rotuladas e não rotuladas pelo mesmo processo de aprendizado, chamado aprendizado semissupervisionado, não é atual (PEDRYCZ, 1985; BOARD; PITT, 1989), mas vem sendo mais explorada, principalmente, na última década (CHAPELLE; SCHÖLKOPF; ZIEN, 2006; SCHWENKER; TRENTIN, 2014).

O aprendizado semissupervisionado tem como base técnicas supervisionadas ou não supervisionadas, adaptadas a fim de realizar a aprendizagem utilizando conjuntos parcialmente rotulados e/ou algum outro tipo de informação prévia já disponível.

Um número crescente de publicações e conferências sobre aprendizado semissupervisionado pode ser observado, sendo que as técnicas propostas têm sido aplicadas com sucesso, especialmente, em processamento de imagens (BENSAID et al., 1996; GRIRA; CRUCIANU; BOUJEMAA, 2006; PEDRYCZ et al., 2008) e classificação de textos (LIU; HUANG, 2003; GENG et al., 2009).

As publicações sugerem e analisam modificações de métodos já conhecidos a fim de considerar sua aplicação a um conjunto com maioria de dados não rotulados e uma pequena parte de dados rotulados. A obra de Zhu e Goldberg (2009) apresenta de forma resumida algumas tendências e características para classificação semissupervisionada, como

self-training, *co-training* e *generative models* (CHAPELLE; SCHÖLKOPF; ZIEN, 2006), e apontamentos a respeito de outras formas de aprendizado semisupervisionado, como por agrupamento.

A utilização de métodos de agrupamento em aprendizado semissupervisionado pode ocorrer de duas formas: colaboração na rotulação do conjunto de exemplos ou agrupamento considerando informação prévia. No primeiro caso, algoritmos de agrupamento são aplicados ao conjunto de exemplos não rotulado para gerar grupos que, posteriormente, serão rotulados por algum outro método, com base na porção rotulada do conjunto. No segundo caso, métodos consagrados de agrupamento são modificados a fim de implementar a semissupervisão já no processo de geração de grupos e, em alguns casos, poder definir rótulos para estes grupos.

Chama-se de agrupamento semissupervisionado aquele realizado por métodos que incluem mecanismos para a consideração da informação pré-existente no processo de geração de grupos. Os métodos desta categoria de aprendizado podem ser divididos em duas abordagens para incorporação de semissupervisão, dependendo do conhecimento disponível: abordagem baseada em sementes e abordagem baseada em restrições entre pares.

Técnicas baseadas em sementes (PEDRYCZ; WALETZKY, 1997; BENSAID et al., 1996; BENSAID; BEZDEK, 1998; LABZOUR; BENSAID; BEZDEK, 1998; BASU; BANERJEE; MOONEY, 2002) consideram que uma parte, geralmente pequena, do conjunto de exemplos é rotulada. As sementes, exemplos rotulados do conjunto, podem ser utilizadas de variadas formas, como para estabelecer restrições ao algoritmo, estabelecer restrições entre exemplos/grupos e/ou para definição de rótulos de grupos.

As técnicas baseadas em restrições entre pares (WAGSTAFF et al., 2001; BASU; BANERJEE; MOONEY, 2004; GRIRA; CRUCIANU; BOUJEMAA, 2005; GRIRA; CRUCIANU; BOUJEMAA, 2008) contam com informação prévia na forma de relações entre exemplos que podem ser do tipo *must-link*, indicando que um par de exemplos deve pertencer ao mesmo grupo, ou *cannot-link*, indicando que um par de exemplos deve pertencer a grupos distintos.

O agrupamento *fuzzy* semissupervisionado ocorre quando são incluídos mecanismos de semissupervisão em métodos de agrupamento *fuzzy*. A maior parte das publicações coloca a abordagem de Pedrycz (1985) como o primeiro trabalho na área de agrupamento *fuzzy* semissupervisionado.

A exploração de ————»»»

A proposta de métodos semissupervisionados de aprendizado é crescente, uma vez que questões como o volume de dados e o custo de rotulação manual de exemplos persistem.

Hamasuna e Endo (2011) introduz o conceito de tolerância entre grupos, utili-

zado em conjunto com restrições entre pares de exemplos para a construção de um novo algoritmo de agrupamento semissupervisionado baseado no FCM. Yan e Chen (2011) utilizam um conjunto de exemplos rotulados para inicialização e criação de restrições de pares de exemplos, extraídos a partir dos rótulos, durante o processo de agrupamento explorado dentro do contexto de categorização de documentos. O algoritmo *Data Understanding using Semi-Supervised Clustering* (BHATNAGAR et al., 2012) utiliza uma porção de exemplos rotulados para a identificação de pequenos grupos dentro das classes. Shamshirband et al. (2014) propõem o *D-FICCA*, um algoritmo de agrupamento que integra uma modificação, baseada em densidade e lógica *fuzzy*, para o algoritmo de competição imperialista (ATASHPAZ-GARGARI; LUCAS, 2007). Em (ZHENPENG et al., 2014) é proposta uma técnica de agrupamento semissupervisionado baseada no algoritmo *k-means* e ganho de informação para escolha dos protótipos incias. O trabalho de Schwenker e Trentin (2014) traz uma revisão atual de outros métodos de agrupamento semissupervisionado.

1.3 Considerações Finais

Este capítulo apresenta conceitos gerais relacionados a aprendizado não supervisionado, semissupervisionado e em fluxos de dados, além de particularidades inerentes a estas abordagens. Esta síntese se faz necessária para situar o leitor, facilitando a compreensão do contexto investigado neste trabalho e permitindo entendimento mais claro do conteúdo apresentado nos próximos capítulos.

2

Agrupamento em Fluxos de Dados

As técnicas clássicas de AM consideram particularidades para os dados disponíveis: assume-se que o conjunto de exemplos é finito e que os exemplos seguem uma distribuição estática e estão disponíveis para acesso sempre que necessário durante o processo de aprendizagem.

A evolução e ampliação do acesso a novas tecnologias e a internet tornaram propício o surgimento e desenvolvimento de diferentes e novos domínios para os quais as características assumidas pelas abordagens clássicas de AM não são verdadeiras.

Existe hoje uma variedade de sistemas que produzem grande quantidade de dados em curto espaço de tempo, como monitoração de tráfego de rede (AGGARWAL; YU, 2008; YU et al., 2009; ZHANG et al., 2012; BREVE; ZHAO, 2013), redes de sensores (GAMA; GABER, 2007; PAN; YANG; PAN, 2007; ZHANG et al., 2012; BOUCHACHIA; VANARET, 2014), mineração de *clicks* na *web* (MARIN et al., 2013), medida de consumo de energia (SILVA et al., 2011; ZHANG et al., 2012), fraude de cartão de crédito (WU; LI; HU, 2012), mineração de textos da *web* (FDEZ-RIVEROLA et al., 2007; CHENG et al., 2011; KMIECIAK; STEFANOWSKI, 2011; NAHAR et al., 2014a), rastreamento visual (LIU; ZHOU, 2014), olfação artificial (VITO et al., 2012), pesquisa meteorológica, mercado de ações e registros de supermercados (YOGITA; TOSHNIWAL, 2013).

Sistemas como os citados impulsionaram a pesquisa por técnicas de aprendizado capazes de lidar com as peculiaridades desses novos domínios: tamanho indefinido, potencialmente infinito, e podem gerar exemplos com distribuição estatística mutável de acordo com o tempo (GAMA, 2010). Nesse contexto teve origem uma nova abordagem denominada Aprendizado em Fluxo de Dados (AFD).

No modelo de Fluxo de Dados (FD) alguns ou todos os exemplos de entrada que serão utilizados não estão disponíveis em disco ou memória para acesso a qualquer momento, mas surgem de maneira contínua, em um ou mais fluxos. FDs diferem de conjuntos de exemplos ditos convencionais em diversos aspectos (BABCOCK et al., 2002):

- Os exemplos no fluxo chegam de maneira contínua e constante;
- O sistema não possui controle sobre a ordem na qual os exemplos chegam para serem processados;
- Os fluxos têm tamanho potencialmente infinito;
- Uma vez que um exemplo do FD foi processado, ele é descartado ou arquivado. Estes exemplos não podem ser recuperados de forma simples, pois guardá-los em memória ou disco seria inviável.

Devido às limitações de tempo e espaço que ocorrem por causa das peculiaridades de FDs, as técnicas de AFD devem considerar que encontrar conhecimento válido de maneira rápida é uma prioridade para esses domínios, mesmo que o encontrado seja uma aproximação do obtido caso fosse possível ter o conjunto de exemplos completo para a aplicação de AM clássico.

Aprendizado supervisionado de FD - citar alguma coisa?

A seção a seguir descreve um método para indução de árvores de decisão a partir de dados que chegam de forma contínua, que serve de inspiração para variadas técnicas de aprendizado em FD.

2.1 Árvores de Hoeffding

Um dos métodos mais conhecidos e utilizados para classificação de exemplos é o aprendizado por árvores de decisão. Esses métodos induzem modelos na forma de árvores a partir dos dados disponíveis, onde cada nó contém um teste para um atributo, cada ramo uma possibilidade de valor para o teste e cada folha a predição de uma classe.

Uma árvore de decisão é aprendida pela recursiva troca de folhas por nós de teste. O atributo relacionado ao teste é escolhido pela comparação dos atributos disponíveis, de acordo com alguma métrica.

Métodos clássicos de aprendizado de árvores de decisão (QUINLAN, 1986; QUINLAN, 1993) consideram que todos os exemplos de treinamento podem ser armazenados simultaneamente na memória principal, por isso, são limitados no número de exemplos dos quais podem aprender. Outros métodos consideram que os dados estão disponíveis em disco e realizam o aprendizado acessando sequencial e repetidamente os dados.

Domingos e Hulten (2000) propõem uma árvore de decisão capaz de aprender em domínios *online*, onde o conjunto de exemplos é potencialmente infinito, mas sua distribuição é estática. Esse método é conhecido como Árvore de Hoeffding (AH).

Uma AH requer que cada exemplo seja lido e processado apenas uma vez. A escolha do atributo para um nó da árvore é baseada em um pequeno subconjunto de exemplos treinamento. Dado um fluxo de exemplos, os primeiros serão usados para escolher o atributo da raiz. Escolhido o atributo raiz, os próximos exemplos são passados às folhas correspondentes e usados para escolher os atributos apropriados para a substituição por nós de teste e assim em diante. Em cada nó folha, o rótulo é escolhido de acordo com a maioria de exemplos da mesma classe presentes na folha.

Para definir o momento de criação de um nó teste, o limiar de Hoeffding () é utilizado, sendo desnecessário definir um número fixo de exemplos. O objetivo ao utilizar esse índice é garantir que, com alta probabilidade, o atributo escolhido usando um pequeno conjunto de exemplos é o mesmo que seria escolhido a partir dos infinitos exemplos.

No mesmo trabalho, os autores sugerem uma implementação de um sistema de árvore de decisão baseado em AH, chamado *Very Fast Decision Tree* (VFDT). O sistema VFDT constrói árvores de decisão usando memória e tempo por exemplo constantes, podendo incorporar dezenas de milhares de exemplos por segundo.

Extensões de AH consideram outros métodos para determinar os rótulos nas folhas, podendo ser construído um modelo dentro de cada folha, a partir dos exemplos contidos na folha, para classificação de novos exemplos. Também são implementadas outras formas para determinar o momento de divisão de um nó folha e detecção e adaptação a desvios de conceito. Algumas dessas extensões são apresentadas no Capítulo 3.

O algoritmo de AH original considera que todos os exemplos do conjunto contínuo são rotulados. Essa realidade não é verdadeira para todas as aplicações FD. A próxima seção apresenta algumas questões associadas especificamente a métodos de agrupamento em FD, que consideram exemplos não rotulados no processo de aprendizagem.

Por conta da rápida e contínua chegada dos exemplos de um FD, é natural inferir que grande parte dos domínios onde algoritmos de aprendizado em FD podem ser aplicados encaram a dificuldade da falta de rótulos disponíveis para a utilização de métodos supervisionados. Devido a isso, cresce o interesse por abordagens de agrupamento em FD.

INICIO - sobre AFD baseado em atributos vs baseado em exemplos

No âmbito de AFD podemos identificar duas abordagens distintas: baseado em exemplos e baseado em atributo.

As técnicas de AFD baseado em atributos consideram um conjunto de múltiplos FD e o objetivo do aprendizado é identificar padrões de comportamento entre os diferentes conjuntos. A Figura 2.1 apresenta um *framework* popular para o agrupamento de múltiplos FD. Os FDs de entrada são gerenciados por um mecanismo que mantém uma sumarização estatística dos fluxos e é capaz de realizar uma partição dos múltiplos FDs pela análise de métricas que comparem os atributos presentes no fluxo. Um exemplo seria o agrupamento

do histórico de diferentes ações da bolsa de valores, associando ações com tendências semelhantes a um mesmo grupo.

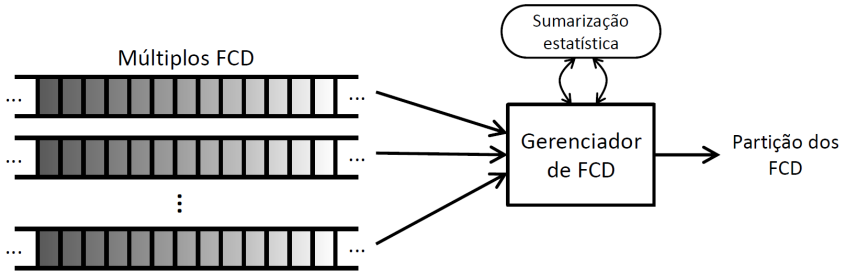


Figura 2.1: modificar a figura, se for mantida*Framework* para agrupamento baseado em atributo (múltiplos FD) (SILVA et al., 2013)

Estratégias de aprendizado supervisionado de múltiplos FD seguem esquema similar, porém considerando a informação de rótulo de cada fluxo como objetivo do aprendizado. O aprendizado semissupervisionado surge como parte de esforços para a proposta de mecanismos capazes de aprender a partir de múltiplos FDs.

Chen, Chen e Sheng (2013) propõem uma estrutura de sumarização para a representação de exemplos de FDs de forma compacta com o objetivo de agrupar múltiplos FDs de forma não supervisionada. O algoritmo *MINETRAC* (CASAS; MAZEL; OWEZARSKI, 2011) combina técnicas de aprendizado não supervisionado e semissupervisionado para identificação e classificação de diferentes classes de fluxos de tráfego de internet, que possuam características similares. Um método semissupervisionado baseado em grafo (LI, 2014) para propagação de rótulos e extensão do conjunto rotulado que realiza o treinamento de um classificador usando *Support Vector Machine* é utilizado para a identificação de padrões em sequências de dados.

As técnicas que seguem a abordagem de aprendizado baseado em exemplos tem como objetivo criar um modelo correspondente aos dados de um FD único.

FIM - sobre AFD baseado em atributos vs baseado em exemplos

2.2 Estruturas de Sumarização de Exemplos

Uma solução para contornar a impossibilidade de armazenamento de todos os exemplos é a criação de sumários ou sinopses da informação encontrada nos dados. Uma grande variedade de técnicas tem sido desenvolvidas para o armazenamento de sumários da informação histórica encontrada em FD. (GAMA; GABER, 2007).

É possível manter estatísticas simples de FD, que podem ser computadas de forma incremental. Para definir a média de um FD, por exemplo, precisamos manter o número de observações (i) e a soma dos valores encontrados até o momento ($\sum x_i$). Assim, com

a chegada de um novo exemplo, a média pode ser calculada de forma incremental, como na Equação 2.1.

$$\bar{x}_i = \frac{(i-1) \times \bar{x}_{i-1} + x_i}{i} \tag{2.1}$$

De maneira semelhante podem ser definidas outras estatísticas, como desvio padrão e coeficiente de correlação entre dois fluxos. O interessante nessas fórmulas é poder manter estatísticas exatas sobre uma sequência de dados potencialmente infinita sem ter que armazenar todos os dados (GAMA; GABER, 2007).

As estruturas de sumarização mais frequentemente utilizadas por técnicas de agrupamento em FD são detalhadas a seguir.

Vetor de Atributos

O uso de vetores de atributos para sumarização de grandes volumes de dados foi introduzido no algoritmo *BIRCH* (ZHANG; RAMAKRISHNAN; LIVNY, 1996). Este vetor, chamado de *Clustering Feature* (CF), conta com três componentes: o número de exemplos (N), a soma linear dos exemplos (LN) e a soma quadrática dos exemplos (SS), sendo que LS e SS são estruturas n-dimensionais, de acordo com o número de atributos do FD. Essas componentes permitem o cálculo de métricas de grupo, como média, raio e diâmetro do grupo.

O vetor CF possui propriedades de incrementais e aditivas, ou seja, é possível inserir um novo exemplo em um CF pela atualização das estatísticas e dois CF podem ser mesclados em um terceiro vetor CF de forma simples.

Algumas abordagens de aprendizado em FD utilizam o vetor CF como descrito nesta descrição, por vezes incluindo pesos para ponderar os grupos (CAO et al., 2006; KRANEN et al., 2011). Entretanto, há outras abordagens que utilizam variações do CF, a fim de produzir outras estatísticas.

A estrutura nomeada microgrupo, usada primeiramente no algoritmo *CluStream* (AGGARWAL et al., 2003), estende o conceito do vetor CF, adicionando mais duas componentes ao CF original: a soma de marcas temporais ou *timestamps* (LST) e a soma quadrática de *timestamps* (SST). As duas novas componentes tem o objetivo de incluir aspecto temporal na descrição de grupos, que pode ser utilizado para identificar *outliers* ou desvios de conceito.

A proposta do algoritmo *SWClustering* (ZHOU et al., 2008) também sugere uma extensão para o vetor CF, chamada de *Temporal CF*, que adiciona uma nova componente ao CF original: a *timestamp* do exemplo mais recente a ser inserido no grupo.

Algoritmos que fazem uso de microgrupos ainda podem manter um histórico dessas estruturas para determinar *snapshots* do FD, i.e., recuperar a situação da partição de

grupos em um determinado momento no tempo. (AGGARWAL et al., 2007)

Arranjos de Protótipos

Alguns algoritmos de agrupamento utilizam uma estrutura simplificada chamada Arranjos de Protótipos, que consiste em um conjunto de protótipos (medóides, centróides, etc) que sumarizam a partição dos dados.

O algoritmo STREAM (GUHA et al., 2000) divide o FD em partes (*chunks*) e, para cada uma das partes, são definidos $2k$ exemplos representantes obtidos por uma variante do algoritmo k-medóides (KAUFMAN; ROUSSEEUW, 1990). Esse processo é repetido até que seja completado um conjunto de m exemplos e, então, o agrupamento é aplicado aos protótipos com o objetivo de reduzir esse conjunto.

Estratégia similar é utilizada para o algoritmo *Stream LSearch* (O'CALLAGHAN et al., 2002), que os protótipos em memória. Quando a memória está cheia, o conjunto de protótipos é agrupado a fim de manter na memória apenas um subconjunto de protótipos.

Grades de Dados

A sumarização dos exemplos de um FD também pode ser feita por meio de grades (CAO et al., 2006; CHEN; TU, 2007; GAMA; RODRIGUES; LOPES, 2011), ou seja, pelo particionamento do espaço n-dimensional de atributos em células grade de densidade.

Uma estratégia (CHEN; TU, 2007) para a utilização de grades é a associação de um coeficiente de densidade que decresce com o tempo. A densidade de uma célula de grade é determinada pela soma das densidades de cada exemplo inserido na grade. Cada célula é representada por uma tupla $< tg, tm, D, label, status >$, onde tg é a última vez que a célula foi atualizada, tm é a última vez que a célula foi removida do conjunto de células válidas (não *outliers*), D é a densidade desde a última atualização, *label* é o rótulo de classe da célula e *status* indica se é uma célula normal ou esporádica (células com poucos objetos, *outliers*).

A manutenção das células de grade é realizada durante a fase *online*. Uma célula pode se tornar esparsa se não receber exemplos por muito tempo e uma célula esparsa pode se tornar densa se receber muitos exemplos. Quando um novo exemplo chega, é verificado a célula a qual pertence e estrutura da célula é atualizada. Células com o *status* esporádico são removidas periodicamente.

Aspecto não estacionário dos conjuntos: Aprendizado baseado em janelas (mais recente = mais importante, menos recente = menos importante) vs detecção de desvios de conceito

2.3 Desvios de Conceito

Na maioria das aplicações do mundo real, os dados são coletados durante um período de tempo. Para longos períodos, é plausível considerar que os exemplos não são independentes ou não possuem mesma distribuição. Em domínios complexos, é provável que a distribuição de classes mude de acordo com o tempo (GAMA et al., 2004). Essas mudanças são conhecidas como desvios de conceito.

Desvios de conceito podem ser graduais, onde há uma transição suave entre as distribuições, ou abruptas, quando a distribuição muda repentinamente.

Abordagens que lidam com desvios de conceitos podem ser classificadas em duas categorias: aquelas que adaptam o modelo em intervalos regulares sem considerar que mudanças ocorreram e aquelas que primeiro detectam desvios de conceitos e, então, adaptam o modelo a essas mudanças. As abordagens da primeira categoria são aquelas que utilizam modelos de janelas temporais.

As estratégias que realizam a detecção de mudanças para posterior adaptação do modelo mantêm um monitoramento, realizado pela definição de indicadores baseados no modelo. Se um desvio é detectado durante o monitoramento, são aplicadas ações para a adaptação do modelo de aprendizado.

Os trabalhos (GAMA et al., 2004; LI; WU; HU, 2012; WU; LI; HU, 2012) trazem maiores informações sobre algumas estratégias de detecção de desvios de conceito.

Janelas Temporais

As janelas temporais são uma abordagem bastante utilizada para resolver a questão de conjuntos abertos (infinitos) como FD. Ao invés do considerar todo o conjunto de exemplos de um fluxo, são considerados subconjuntos de exemplos ao longo do tempo. Neste modelo, uma marca temporal está associada a cada exemplo, a fim de determinar se o exemplo é válido ou não, ou seja, se está dentro ou fora de uma determinada janela temporal.

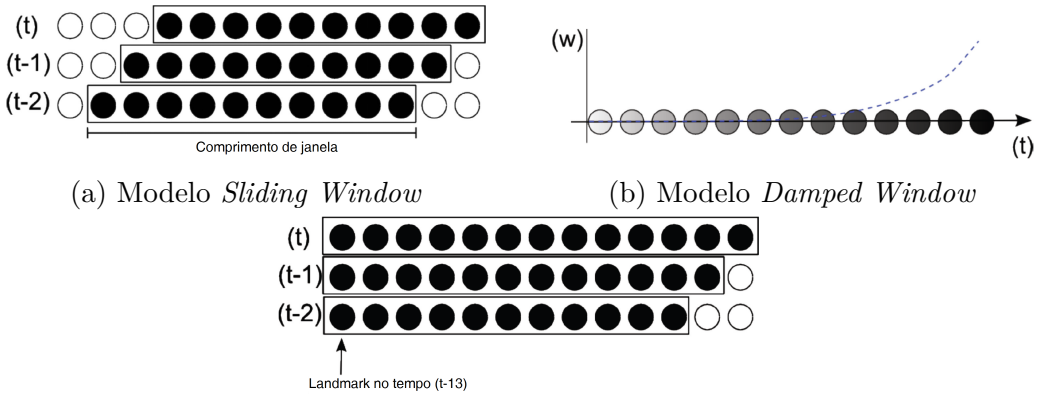
Existem diferentes modelos de janelas que podem ser encontrados na literatura, os mais relevantes (ZHU; SHASHA, 2002) são descritos a seguir:

Modelo *Sliding Window*: neste modelo apenas a informação mais recente do FD é armazenada em uma estrutura de dados (janela) cujo tamanho pode ser variável ou fixo. Esta é uma estrutura tipo *First In, First Out*, que considera os exemplos de um determinado ponto no tempo atual até um ponto no passado. A Figura 2.2a traz um exemplo do modelo *Sliding Window*, com comprimento fixo de janela, em três momentos do tempo ($t-2$, $t-1$ e t) representando o deslizamento da janela para que contenha os exemplos mais recentes no tempo mais atual t. Algoritmos que

utilizam este modelo (REN; MA, 2009) apenas atualizam os sumários estatísticos dos exemplos dentro da janela.

Modelo *Damped Window*: também conhecido como *time-fading*, este modelo considera a informação mais recente pela associação de pesos aos exemplos do FD (JIANG; GRUENWALD, 2006): exemplos mais recentes tem peso maior que exemplos mais antigos e o peso dos exemplos diminui de acordo com o tempo. Um exemplo pode ser visualizado na Figura 2.2b, que mostra o decaimento do peso de acordo com o degradê dos exemplos. Algoritmos (CAO et al., 2006; CHEN; TU, 2007; ISAKSSON; DUNHAM; HAHSLER, 2012) baseados nesse modelo usualmente adotam uma função exponencial de decaimento para o peso dos exemplos.

Modelo *Landmark Window*: o processamento, neste modelo, se faz por porções disjuntas do FD, nomeadas *chunks*, que são separadas de acordo com a ocorrência de *landmarks* (aparecimento de exemplos relevantes). Os *landmarks* podem ser definidos de acordo com o tempo, e.g., diário ou semanal, ou quanto ao número de elementos observados desde o *landmark* anterior. Quando um novo *landmark* é alcançado, todos os exemplos da janela são removidos e novos são adicionados a partir desse momento. Na Figura 2.2c há um exemplo para o modelo. Estratégias possíveis usando este modelo são baseadas na utilização dos modelos obtidos pelos diversos *chunks* em conjunto ou como guias para próximos modelos.



(a) Modelo *Sliding Window* (b) Modelo *Damped Window*

(c) Modelo *Landmark Window*

Figura 2.2: Exemplos ilustrativos de modelos de janela temporal (SILVA et al., 2013)

Além das características mencionadas a respeito do volume de exemplos, há também um componente temporal inerente a aprendizado em FD. Os dados podem evoluir de acordo com o tempo e assim a distribuição do conjunto pode ser alterada. Deste modo, algoritmos que apenas sugerem adaptações para contornar as questões de volume de exemplos podem não ser soluções efetivas neste contexto. Algoritmos de aprendizado em FD devem ter foco claro na evolução dos dados (AGGARWAL, 2007).

O uso de modelos de janela temporal podem auxiliar no tratamento de um dos aspectos da evolução dos exemplos do FD, por permitir uma avaliação de acordo com o tempo. No entanto, existem outros aspectos que podem ser explorados. A próxima seção fala sobre a difícil tarefa de identificação de desvios de conceito.

2.4 Técnicas de Agrupamento em Fluxos de Dados

apresentar de outra forma (tabela?), destacar mais relevantes/importantes?

Abordagens de agrupamento de dados são tipicamente utilizadas no aprendizado não supervisionado. Dentro da área de FD é comum verificar a falta de informação de classe, seja por conta da natureza do domínio (não existem classes definidas) ou pela dificuldade em rotular exemplos de um FD.

Independentemente dos métodos adotados, é desejável que algoritmos de agrupamento em FDs possuam a capacidade de (AMINI; WAH; SABOOHI, 2014): *a*) descobrir grupos de formatos arbitrários; *b*) lidar com ruído; *c*) realizar o agrupamento sem informação prévia sobre o número de grupos.

falar sobre propostas baseada em chunks de exemplos

2.4.1 *Framework Online-Offline* (FOO) para Agrupamento em FD

Algoritmos de agrupamento baseados em exemplos podem ser resumidos em dois passos (CAO et al., 2006; YANG; ZHOU, 2006): abstração dos dados (componente *online*) e agrupamento (componente *offline*), ilustrados na Figura 2.3.

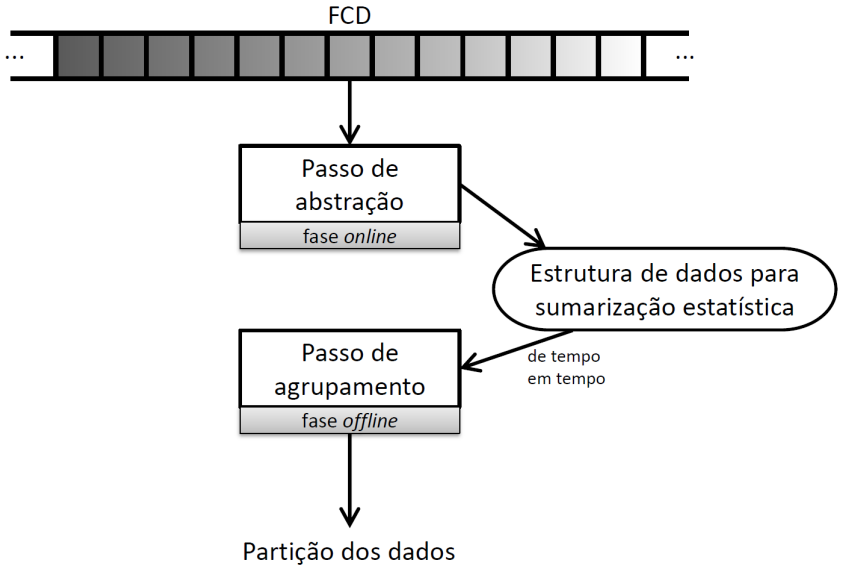


Figura 2.3: *Framework online-offline* (SILVA et al., 2013)

A fase *online*, abstração dos dados, sumariza os dados do FD com o auxílio de estruturas particulares para lidar com restrições de espaço e memória das aplicações FD. Essas estruturas de sumarizam os dados para preservar o significado dos objetos originais sem a necessidade de armazená-los. Estruturas frequentemente utilizadas são vetores de atributos, arranjos de protótipos e grades de dados. Essas estruturas são melhor detalhadas na Seção 2.2.

Para a contínua sumarização dos exemplos que chegam e dar maior importância aos exemplos mais recentes, uma abordagem popular é a definição de janelas temporais, como apresentado na Seção 2.3.

Durante o passo de abstração, algoritmos de agrupamento em FD devem utilizar mecanismos para detecção de *outliers* que sejam capazes de diferenciar verdadeiros *outliers* de evolução de grupos (Seção 2.3), uma vez que a distribuição dos dados pode variar de acordo com o tempo.

Na fase *offline* é possível obter uma partição dos dados pelo passo de agrupamento. Neste momento, pode ser necessária a definição de alguns valores de entrada (número de grupos, por exemplo) para que seja possível ter uma visão geral dos grupos do FD. Algoritmos de agrupamento tradicionais podem ser utilizados considerando o conjunto de estruturas de sumarização para encontrar uma partição dos dados. O formato dos grupos encontrados está ligado ao algoritmo de agrupamento empregado, por exemplo, o *k-means* (MACQUEEN, 1967) gera grupos hiperesféricos enquanto o DBSCAN (ESTER et al., 1996) é capaz de descobrir grupos de formatos aleatórios.

O *framework* apresentado nesta seção é frequentemente utilizado para o desenvolvimento de novas técnicas de agrupamento em FD. Algumas dessas propostas são discutidas no Capítulo 3.

O *ClusTree* (KRANEN et al., 2011), por exemplo, é a proposta de uma estrutura hierárquica compacta e auto-adaptativa para manter sumários de FD, construindo uma hierarquia de microgrupos em diferentes níveis. Outras técnicas que seguem a abordagem hierárquica de agrupamento são o *E-Stream* (UDOMMANETANAKIT; RAKTHANMANON; WAIYAMAI, 2007), que possui suporte para cinco tipos de evolução em grupos (aparecimento, desaparecimento, evolução própria, mescla e divisão), e suas extensões para suporte de incerteza (valores faltantes) em FDs heterogêneos (conjuntos que combinam atributos numérico-contínuos e categóricos) (MEESUKSABAI; KANGKACHIT; WAIYAMAI, 2011) e FDs de alta dimensão (CHAIRUKWATTANA et al., 2014).

As subseções a seguir apresentam técnicas já existentes baseadas em agrupamento particional e baseado em densidade, onde é possível identificar focos de pesquisas mais recentes.

2.4.2 Agrupamento Particional em FD

Os algoritmos de agrupamento particional para FD, em sua maioria, são propostos como extensões de algoritmos de agrupamento particionais conhecidos, como *k-means*, *k*-medóides e *Affinity Propagation*.

O algoritmo *CluStream* (AGGARWAL et al., 2003) é baseado no algoritmo *k-means* e introduz um *framework online-offline* para agrupamento em FD que vem sendo adotado para grande parte dos algoritmos propostos recentemente. Yang e Zhou (2006) propõem uma extensão chamada *HCluStream* para lidar com FDs heterogêneos.

A proposta de Labroche (2014) está baseada no algoritmo *k*-medóides e realiza agrupamento *fuzzy* de forma incremental. O trabalho de Lemos, Caminhas e Gomide (2013) apresenta uma técnica para geração de um classificador *fuzzy* baseado em agrupamento incremental para a geração de regras que descrevem novos estados operacionais de um sistema de detecção e diagnóstico de falhas.

Hore, Hall e Goldgof (2007b) apresentam a proposta de uma abordagem genérica para agrupamento iterativo *fuzzy*/possibilístico em FD, introduzindo equações objetivo transformadas para os algoritmo FCM (BEZDEK, 1981), *possibilitistc c-means* (KRISHNAPURAM; KELLER, 1996) e Gustafson-Kessel (GUSTAFSON; KESSEL, 1978). Outro trabalho (HORE; HALL; GOLDGOF, 2007a) traz uma nova variante do FCM para aprendizado em FD, o *Streaming* FCM, que realiza adaptação à evolução de distribuições pela utilização de uma parte do histórico de protótipos/centróides no agrupamento de *chunks* de dados, conforme sua chegada. Em (HORE et al., 2008) é explorada uma extensão *online* para o FCM que mantém sumarização do agrupamento usando exemplos ponderados. Os exemplos ponderados obtidos pelo agrupamento de cada *chunk* de dados formam um *ensemble* que é transformado em um conjunto de grupos finais.

Alguns trabalhos propõem algoritmos baseados no agrupamento *Affinity Propagation* (AP) (FREY; DUECK, 2007). Usando método de passagem de mensagem, o AP escolhe, entre os exemplos disponíveis, aqueles que melhor representam o conjunto, os chamados *exemplars*, que indicam os diferentes grupos dentro do conjunto de exemplos. Com os *exemplars* não é necessário definir o número de grupos inicialmente.

Uma extensão do AP para aprendizado em FD é o algoritmo *Streaming* AP (ZHANG; FURTLEHNER; SEBAG, 2008). Esta proposta é dividida em dois passos, sendo que o objetivo do primeiro é encontrar os *exemplars* ponderados dentro de um *chunk* de dados por uma extensão do AP (*Weighted Affinity Propagation*), enquanto o segundo visa diminuir a complexidade do modelo pela aplicação do *Weighted Affinity Propagation* para o conjunto de *exemplars*. Em trabalho mais recente (ZHANG et al., 2014), o *Streaming* AP traz melhorias como mecanismo de detecção de mudanças e adaptação do modelo da distribuição dos dados.

2.4.3 Agrupamento Baseado em Densidade em FD

Os algoritmos baseados em densidade também são utilizados como alternativa para a tarefa de agrupamento, sendo duas de suas vantagens a alta tolerância a ruído ou *outliers* e a habilidade em descobrir grupos de formatos arbitrários.

As técnicas de agrupamento baseado em densidade seguem, comumente, duas abordagens que são descritas nas próximas seções e incluem exemplos de algoritmos que se encaixam nessas categorias.

2.4.4 Algoritmos de Microgrupos de Densidade

Em algoritmos de agrupamento de microgrupos de densidade, microgrupos mantêm a informação de sumarização dos exemplos e o agrupamento é realizado usando as estruturas de sinopse.

A proposta de Cao et al. (2006) é o algoritmo de agrupamento em FD baseado em densidade chamado *DenStream*, que utiliza duas estruturas de sumarização para lidar com novas distribuições no FD, diferenciando-as de *outliers*. As estruturas nomeadas *core-micro-cluster* (cmc), referentes ao agrupamento em si, e *pontential core-micro-cluster* (pcmc), distribuição de exemplos que representa regiões menos densas que são mantidas. O aprendizado da estrutura do FD é realizado em duas fases. Na fase *online* do algoritmo, cada novo exemplo pode ser associado a um microgrupo já existente (cmc ou pcmc), de acordo com cálculo de métrica de dissimilaridade (distância Euclidiana) ou será criado um novo pcmc para o novo exemplo. Na fase *offline* é aplicado o algoritmo *DBSCAN* (ESTER et al., 1996) para determinar o grupos finais, de acordo com o conjunto de cmc. De tempos em tempos, um método de poda avalia o conjunto de pcmc para garantir que são *outliers*, de acordo com a densidade, e descartá-los. O *DenStream* não possui mecanismos para a eliminação de microgrupos ou para fundir dois ou mais microgrupos, o que pode ser problemático conforme o crescimento do conjunto de exemplos e limitações de espaço para armazenamento.

O *DenStream* serviu de inspiração para outras técnicas que consideram situações particulares e implementam adaptações para lidar com contextos diversos. Li-xiong et al. (2009) desenvolveram um algoritmo baseado no *DenStream* para aplicações com grande volume de outliers. O algoritmo *SDStream* (REN; MA, 2009) tem a habilidade de descobrir grupos de formatos arbitrários dentro de um modelo de janela deslizante, que permite o esquecimento progressivo dos dados antigos. *HDenStream* (LIN; LIN, 2009) é um algoritmo adaptado para aprendizado em FDs com atributos heterogêneos pela inclusão de um atributo bidimensional para manter a frequência de atributos categóricos. O *HDDStream* (NTOUTSI et al., 2012) traz adaptações ao original *DenStream* para melhorar o agrupamento de FDs de alta dimensão. O *PreDeConStream* (HASSANI et al., 2012) melhora a

eficiência da fase *offline* do *HDDStream*.

Alguns métodos híbridos utilizam conceitos do algoritmo *DenStream* aliado a outras abordagens. O *StreamOptics* (TASOULIS; ROSS; ADAMS, 2007) é um *framework* baseado nos conceitos de cmc e pcmc, que utiliza o algoritmo *OPTICS* (ANKERST et al., 1999) para produzir visualização gráfica da estrutura do FD e sua evolução com o passar do tempo. No entanto, em nenhum momento é gerada uma partição do conjunto, então a análise do agrupamento deve ser realizada manualmente. Isaksson, Dunham e Hahsler (2012) propõem o algoritmo *SOStream*, que detecta estrutura de FDs de rápida evolução pela adaptação automática de limiar para o agrupamento baseado em densidade. O limiar é individual por grupo e é definido automaticamente dentro do processo de agrupamento, baseado na ideia de construir vizinhanças com um mínimo de pontos, como parte da análise para criação, remoção,mescla e divisão de grupos. O algoritmo utiliza aprendizado competitivo como em *Self Organizing Maps* (KOHONEN, 1982), o que pode tornar o processo mais oneroso.

O algoritmo *APDenStream* (ZHANG et al., 2013) baseia-se nos métodos AP e *DenStream* para definição de um modelo geral que representa o FD. O algoritmo AP substitui o *DBSCAN* na fase *offline* do *DenStream*. Baseado em trabalho anterior dos autores (FORESTIERO; PIZZUTI; SPEZZANO, 2009), FlockStream (FORESTIERO; PIZZUTI; SPEZZANO, 2013) utiliza um sistema multi-agente baseado em um modelo de *flocking* (KENNEDY; EBERHART; SHI, 2001). Nesta técnica os agentes são microgrupos que trabalham de forma independente mas formam grupos juntos.

2.4.5 Algoritmos Baseados em Densidade e Grades

Em se tratando de agrupamento não supervisionado de FDs é possível identificar recente tendência no investimento de abordagens baseadas em grades e densidade (AMINI et al., 2011; AMINI; WAH; SABOOHI, 2014).

Uma das primeiras tentativas de associar os dois métodos foi o trabalho de Gao et al. (2005), que propõe um algoritmo de agrupamento incremental de passagem única usando células do espaço de atributos (grades) densas que são consideradas em uma fase de agrupamento caso tenham valor de densidade acima de um limiar pré-definido.

O *D-Stream* (CHEN; TU, 2007; TU; CHEN, 2009) é uma proposta de agurpamento em tempo real baseado em grades, apoiado no fOO. Na fase *online* acontece a leitura de um novo exemplo e seu mapeamento na grade. Na fase *offline* os grupos são criados e ajustados em intervalos. No primeiro ciclo cada grade considerada densa é associada a um grupo distinto, nos intervalos seguintes os grupos são ajustados. O ajuste de grupos acontece por meio da identificação de grades densas e esparsas: grades densas são mescladas a grades vizinhas, formando um grupo; caso contrário a grade é removida do grupo. O método, que

agrupa os exemplos em tempo real, inclui mecanismos para decaimento de densidade com a passagem do tempo, detecção de evolução de comportamento e tratamento de *outliers*.

é aqui que eu comento do problema das grades? Tamanho da grade (como dividir o espaço), número de dimensões/atributos (grades esparsas > grades densas, mais oneroso para manter em memória e fase offline (volume de grade e verificação de grupos)

O D-Stream é o principal baseado em grade e densidade, os outros podem aparecer somente na tabela, se for o caso de mencionar

Pela revisão aqui exposta, pode-se verificar que é crescente o número de novas propostas para agrupamento em FDs, em especial aquelas que utilizam abordagem baseada em densidade e outras abordagens capazes de lidar com o surgimento e desaparecimento de grupos de forma simples.

No entanto, neste tipo de aprendizado, é ignorada qualquer informação prévia que possa existir a respeito da distribuição dos dados. O investimento em novas propostas para aprendizado semissupervisionado em FD também cresceu nos últimos anos, como pode ser inferido pelas técnicas apresentadas na próxima seção.

2.5 Agrupamento Fuzzy em Fluxo de Dados

O método não supervisionado de agrupamento *Single Pass Fuzzy C Means* (SPFCM) (HORE; HALL; GOLDGOF, 2007c) tem como objetivo oferecer uma alternativa ao agrupamento FCM para conjuntos de dados cujo grande tamanho impede seu armazenamento total em memória para a aplicação do processo de agrupamento. A técnica parte do princípio de que os dados do conjunto estão disponíveis em partes, nomeadas *chunks*, às quais são processadas pelo algoritmo WFCM.

O Algoritmo 2.1 apresenta os passos gerais para a proposta. Quando chega o primeiro *chunk* todos os exemplos possuem peso de valor 1 (um). Após a aplicação do WFCM calcula-se o peso para cada centróide obtido, com base na matriz de pertinência. Os centróides serão adicionados ao próximo *chunk* de novos exemplos e o agrupamento WFCM é aplicado ao conjunto resultante dessa união.

O SPFCM contempla a dificuldade de armazenamento de exemplos, processando *chunks* do conjunto e eliminando os exemplos logo após a obtenção dos centróides ponderados. No entanto, é uma técnica que considera que a distribuição dos dados é estática, o que deixa de lado a característica mutável do contexto de FD.

Os autores do SPFCM propõem, então, uma variação do algoritmo(HORE; HALL; GOLDGOF, 2007b) que inclui um mecanismo para lidar com o aspecto não estacionário dos FD. Nesta nova técnica, a partir do segundo *chunk* de dados, os pesos dos centróides são calculados a partir da matriz de pertinência para os exemplos novos apenas, descon-

Algoritmo 2.1: *Single-Pass Fuzzy C-Means* (SPFCM) (HORE; HALL; GOLDGOF, 2007c)

Entrada: E, k, m, n_s
Saída: U, C
início
  U = geraMatrizPertinênciaAleatória();
  C = geraCentróidesIniciais(E, U);
  E' = gerarSubconjuntosExemplos(E, n);
  $w = 1_{n_s}$;
  U, C = WFCM($E'[1]$,k,m,w);
  PAREI AQUI ;
  enquanto $\epsilon > \xi$ **faça**
    atualizarMatrizPertinência(U);
    $C' = C$;
    atualizarCentróides(C);
    $\epsilon = max_{1 \leq i \leq k}\{\|c_i - c'_i\|^2\}$;
  fim
fim

siderando os centróides ponderados, que servem apenas como histórico de agrupamentos anteriores.

A técnica variante do SPFCM é capaz de lidar com quantidades diferentes de histórico, incorporando ao *chunk* de novos exemplos h conjuntos de centróides ponderados obtidos anteriormente. Uma das dificuldades nessa estratégia é determinar o valor de h, uma vez que essa definição depende do domínio ao qual é aplicado o agrupamento.

Outra complicação de estratégias baseadas no SPFCM é a definição prévia de um número fixo de grupos, pois, uma vez que um FD é mutável de acordo com o tempo pode ocorrer o aumento ou diminuição no número de grupos. Outro obstáculo: identificação de grupos de formato arbitrário (?)

número fixo de grupos

sumarização por protótipos - centróides ponderandos (ponderação baseada na matriz de pertinência)

histórico - união de h conjuntos de centróides ao novo *chunk de exemplos*

2.6 Considerações Finais

3

Semissupervisão no Aprendizado em Fluxo de Dados

A busca por melhores resultados no aprendizado em FD impulsionou o desenvolvimento de técnicas semissupervisionadas para trabalhar neste contexto. As abordagens de aprendizado semissupervisionado para conjuntos estáticos, juntamente com as abordagens de aprendizado supervisionado e não supervisionado em FD servem de inspiração para as propostas descritas nas próximas seções.

3.1 Classificação Semissupervisionada em FD

As técnicas de aprendizado semissupervisionado baseadas em classificadores assumem que o conjunto de dados é parcialmente rotulado. No caso desse tipo de aprendizado em FD, a parte rotulada dos exemplos pode ser apenas um pequeno conjunto que irá gerar o modelo inicial de classificação ou é possível encontrar exemplos rotulados no próprio fluxo.

Considerando essas duas situações de disponibilidade de rótulos, podem ser encontradas na literatura propostas baseadas em diferentes métodos de classificação, por exemplo: redes neurais (LEITE; COSTA; GOMIDE, 2010; ASTUDILLO; OOMMEN, 2011; ASTUDILLO; OOMMEN, 2013; BOUGUELIA; BELAID; BELAID, 2013; KASABOV et al., 2013), baseada em grafos (TIWARI; KURHANEWICZ, 2010; BERTINI; LOPES; ZHAO, 2012; BERTINI; ZHAO, 2013), competição de partículas (BREVE; ZHAO, 2012; BREVE; ZHAO, 2013), SVM (FRANDINA et al., 2013), entre outros (PAN; YANG; PAN, 2007; FDEZ-RIVEROLA et al., 2007; SILVA et al., 2011).

Uma proposta (LIANG et al., 2012) baseada em CVFDT considera que o FD possui rótulo positivo e os dados incertos são não rotulados, realizando o aprendizado de forma semissupervisionada.

Técnicas baseadas em *ensemble* de classificadores também são populares no aprendizado semissupervisionado em FD. Kholghi e Keyvanpour (2011) apresentam uma pro-

posta para um *framework* que combina semissupervisão por meio de *Active Learning* (SETTLES, 2010) e a consideração da influência de exemplos não rotulados a fim de melhorar a performance de aprendizagem. Um modelo é construído para predição de rótulos de exemplo futuros com alto valor de acurácia. Esse modelo de predição é baseado em um *ensemble* de classificadores construídos a partir de *chunks* de exemplos do FD. Esta é uma das primeiras tentativas de incorporação de *Active Learning* semissupervisionado em FD.

3.2 Agrupamento Semissupervisionado em FD

No aprendizado semissupervisionado baseado em agrupamento, considera-se conhecimento prévio durante o processo de agrupamento para melhorar o aprendizado. Esta informação pode estar disponível em diferentes formas, por exemplo rótulos para parte do FD, restrições entre pares de exemplos, informações estatísticas sobre a distribuição dos exemplos.

Quando há uma pequena quantidade de exemplos rotulados disponíveis, estes podem ser utilizados como sementes que contribuirão para guiar o algoritmo de agrupamento. O modelo de *flocking* serve de inspiração para a adaptação de um algoritmo de agrupamento para aprendizado em FD (BRUNEAU; PICAROUGNE; GELGON, 2009), utilizando um pequeno conjunto de dados rotulados como informação para um operador de divisão de um grupo de exemplos, que permite a adaptação do agrupamento a mudanças no FD.

Uma técnica de agrupamento semissupervisionado baseada em AP (SHI et al., 2009) utiliza informação prévia na forma de rótulos no ajuste da matriz de similaridade do modelo produzido e promove um estudo para ampliar o conjunto de dados rotulados. Baseado em *Fuzzy Pattern Matching* (CAYROL; FARRENY; PRADE, 1982; DUBOIS; PRADE; TESTEMALE, 1988), o método proposto em (MOUCHAWEH, 2010) tem o objetivo de aprender funções de pertinência com um conjunto de exemplos rotulados inicial limitado. A função de pertinência das classes é aprendida e atualizada, de acordo com a chegada de novos exemplos com rótulos.

O algoritmo *Compound Gaussian Mixture Model* (GAO; LIU; GAO, 2010), baseado no agrupamento *Gaussian Mixture Model*, utiliza amostra de dados rotulados em uma fase que aplica uma extensão do algoritmo *Expectation Maximization* (ZHOU et al., 2007) para melhorar o agrupamento. A técnica *Growing Type-2 Fuzzy Classifier* (BOUCHACHIA; VANARET, 2014) utiliza uma versão *online* do agrupamento *Gaussian Mixture Model* para gerar partições *fuzzy* tipo-2 (MENDEL; JOHN, 2002) para construir regras de classificação, empregando conjunto parcialmente rotulado no aprendizado. Esta técnica possui mecanismos para aprendizado em FD.

Atwa e Li (2014) propõem um algoritmo de agrupamento semissupervisionado que extende o AP para lidar com FD. Um conjunto de instâncias rotuladas é incorporado para detecção de mudança, que requer a atualização do modelo o mais rápido possível.

Em um contexto de FD com multirrótulos, o *Hierarchical Semi-supervised Impurity based Subspace Clustering* (AHMED; KHAN; RAJESWARI, 2010) captura a correlação implícita existente entre cada par de rótulos de classe.

Técnicas que utilizam informação na forma de restrições podem obtê-las a partir de exemplos rotulados, mas também pode ser um conhecimento pré-existente nesse formato. O *C-DenStream* (RUIZ; MENASALVAS; SPILIOPOULOU, 2009) é uma técnica baseada no algoritmo *DenStream* adaptado para a utilização o conceito de restrições entre pares de exemplos estendido para FD. O *C-DenStream* foi uma das primeiras extensões do paradigma de aprendizado por agrupamento semissupervisionado estático para FD e, embora traga ganhos nesse contexto, ainda possui as limitações do *DenStream*.

Halkidi, Spiliopoulou e Pavlou (2012) utiliza, além do FD, um fluxo contínuo de restrições, introduzindo o conceito de multigrupos (regiões densas e sobrepostas) e implementa mecanismo para identificação de *outlier*. Sirampuj, Kangkachit e Waiyamai (2013) propõem um algoritmo para agrupamento em FD também com uso de conhecimento prévio na forma de restrições. A técnica, que é uma extensão do *E-Stream* (UDOMMANETANAKIT; RAKTHANMANON; WAIYAMAI, 2007) possui mecanismos para lidar com restrições que mudam de acordo com o tempo (técnica de esquecimento).

Cheng et al. (2011) desenvolvem um *framework* para análise de agrupamento de textos e desenvolvimento de um novo modelo de agrupamento semissupervisionado, capaz de lidar com informação prévia na forma de restrições entre pares de exemplos e rótulos de maneira simultânea.

Uma proposta de método para agrupamento em FD incertos (domínios onde há ruído e dados incompletos) (AGGARWAL; YU, 2008) utiliza um modelo geral de incerteza, no qual assume-se que algumas estatísticas de incerteza estão disponíveis.

3.3 Aprendizado Semissupervisionado Híbrido em FD

É possível identificar dentro do aprendizado semissupervisionado algumas abordagens híbridas, ou seja, inspiradas em métodos de agrupamento e classificação trabalhando em conjunto para melhorar o aprendizado.

Wu, Ye e Fu (2009) apresentam um método semissupervisionado para a construção de um rastreador de tópicos (*topical crawler*), aplicando um agrupamento *k-means* com restrições entre pares para detectar novas amostras de páginas enviadas a um classificador de páginas e preditor de links para atualização de modelos aprendidos.

A proposta de Borchani, Larranaga e Bielza (2011) é a combinação do método de (DASU; KRISHNAN, 2006) adaptado em um algoritmo de agrupamento para aprendizado semissupervisionado. O algoritmo de agrupamento é utilizado para atualização do modelo quando ocorre desvio de conceito.

Técnicas baseadas em AH podem utilizar métodos de agrupamento para divisão e rotulação em suas folhas. Li, Wu e Hu (2012) estende trabalho anterior (LI; WU; HU, 2010b) e propõem um algoritmo de classificação semissupervisionada em FD, utilizando uma árvore de decisão como modelo de classificação. Para o crescimento da árvore, utiliza-se de um algoritmo de agrupamento baseado no *k-means* para a produção de grupos de conceito e rotulação automática de dados não rotulados. Potenciais desvios de conceito são identificados e conceitos recorrentes são mantidos. Uma técnica semelhante considera informação prévia na forma de rótulos e aplica uma versão semissupervisionada do algoritmo de agrupamento *k-modes* para produzir grupos de conceito (LI; WU; HU, 2010a; WU; LI; HU, 2012).

Clustering Feature Decision Tree (XU; QIN; CHANG, 2011) realiza a construção de uma árvore de decisão a partir de FD parcialmente rotulados, aplicando algoritmo de agrupamento para gerar um vetor de atributos de grupos, sumários estatísticos que serão usados para indução da árvore de decisão. Os vetores de grupos também são empregados na classificação de exemplos nas folhas da árvore.

Zhang, Zhu e Guo (2009) propõem um *framework* para construção de modelos a partir de FD com exemplos rotulados e não rotulados. Para a construção do modelo, os dados do FD são associados a quatro categorias distintas, cada qual correspondendo à situação de desvio de conceito, podendo existir ou não nos exemplos rotulados e não rotulados. Em seguida, é aplicado método de aprendizado SVM semissupervisionado baseado no *k-means*.

A técnica *Concurrent Semi-supervised Learning of Data Streams* (NGUYEN et al., 2011; NGUYEN; NG; WOON, 2013) aplica o potencial de aprendizado semissupervisionado concorrente, onde um modelo de agrupamento e um de classificação são construídos de forma simultânea e colaborativa, fazendo uso de um pequeno conjunto de exemplos rotulados encontrados em um FD.

Outras propostas utilizam aprendizado semissupervisionado com objetivo de apenas estender o conjunto de exemplos rotulados e, então, aplicar método de aprendizado supervisionado com mecanismos disponíveis para lidar com as particularidades de FD. Wu, Yang e Zhou (2006) coloca a proposta de um algoritmo de aprendizado semissupervisionado baseado em treinamento por agrupamento, para seleção de exemplos confiáveis a serem rotulados e utilizados no retreinamento de um classificador. A técnica proposta por (YU et al., 2009) aplica um algoritmo de agrupamento semissupervisionado aos exemplos parcialmente rotulados do FD na tentativa de estender o conjunto de exemplos rotulados,

utilizando-os para atualização de um modelo supervisionado que conta com mecanismos de esquecimento.

O *framework COMPOSE* (DYER; CAPO; POLIKAR, 2014) aprende desvios de conceitos em ambiente de FD onde há apenas um conjunto inicial de dados rotulados e, após a inicialização, apenas dados não rotulados. O *COMPOSE* segue três passos: *1*) combinação dos dados rotulados iniciais aos dados não rotulados atuais para treinar um classificador semissupervisionado e rotular de forma automática o conjunto de dados; *2*) para cada classe, construção de formas que englobam os exemplos, representando a distribuição atual da classe; *3*) compactação das formas e extração de instâncias representantes (*core supports*), que servirão como conjunto rotulado inicial para os próximos novos dados não rotulados.

3.4 *Ensemble* de Modelos para Aprendizado Semissupervisionado em FD

Algumas propostas para aprendizado semissupervisionado em FD tem a intenção de aproveitar da construção de diversos modelos trabalhando em conjunto para melhorar a representação dos exemplos do FD.

O trabalho apresentado em (ZHANG et al., 2012) é uma adaptação do trabalho (ZHANG; ZHU; GUO, 2009), onde para cada categoria de exemplo de treinamento é construído um modelo distinto para classificação, baseado em SVM. Em (ZHANG et al., 2014) os modelos base para o *ensemble* são construídos por método de aprendizado semissupervisionado, utilizando conjunto de exemplos rotulados e não rotulados. Informação histórica é mantida como parte de peso no fator de decisão para classificação de novos exemplos.

O trabalho de Nahar et al. (2014b) propõe um *framework* para detecção de *cyberbullying* utilizando um classificador *ensemble* semissupervisionado. Em outra proposta (NAHAR et al., 2014a), a técnica utiliza inclui a extensão do conjunto de dados rotulados por meio de um classificador *ensemble*, com aplicação de um algoritmo *fuzzy* SVM para ponderar o espaço de atributos do domínio.

Outras técnicas têm a extensão do conjunto de exemplos rotulados como parte do processo de aprendizado semissupervisionado. Cao e He (2008) apresenta um algoritmo iterativo que recupera rótulos de acordo com níveis de confiança para melhorar o sistema aprendido pela geração de vários modelos de classificação. A técnica utilizada por Ahmadi e Beigy (2012) treina classificadores usando os exemplos rotulados e tenta classificar os exemplos não rotulados por meio do *ensemble* para estender o conjunto de treinamento e adaptar os modelos de classificação.

O trabalho (MASUD et al., 2008) descreve uma proposta baseada na construção de microgrupos pela aplicação de método de agrupamento semissupervisionado e construção de classificadores pelo algoritmo *K-Nearest Neighbors* para cada*chunk* de exemplos do FD. Os L melhores modelos (de acordo com acurácia individual) são utilizados em um *ensemble*.

A proposta de Ditzler e Polikar (2011) apresenta um *ensemble* onde classificadores são treinados a partir dos exemplos rotulados do FD. Omodelo de classificação utiliza pesos para determinar a influência de cada classificador na decisão final e esses pesos são determinadospeladistância entre componentes de um *Gaussian Mixture Model* treinado com o conjunto completo de exemplos.

Masud et al. (2012) propõem um *ensemble* onde cada modelo de classificação é construindo como uma coleção de microgrupos, usando agrupamento semissupervisionado, e exemplos não rotulados são classificados de acordo com o conjunto de modelos.

Uma proposta (LIU et al., 2013) mantém um *ensemble* de modelos mistos, baseados em métodos de classificação e agrupamento. Os exemplos rotulados são utilizados para treinamento de classificador supervisionado e novos exemplos rotulados são empregados na atualização desse classificador. Os exemplos não rotulados são utilizados na construção de modelos não supervisionado. O *ensemble* segue um modelo semissupervisionado de classificação de forma a maximizar o consenso entre os diferentes modelos.

3.5 Considerações Finais

Neste capítulo foram colocadas algumas técnicas de aprendizado em FD. A maioria das propostas sugere adaptações para métodos de aprendizado em conjuntos estáticos, a fim de incluir mecanismos que possam lidar com as limitações de aprendizado em FD.

Recentemente, percebe-se uma preocupação em elaborar técnicas que possam realizar o aprendizado de forma semissupervisionada, uma vez que a maior parte dos domínios não possuem uma quantidade grande de rótulos. O modelo supervisionado aprendido apenas pelo pequeno conjunto de exemplos rotulados pode ser ineficiente, enquanto modelos aprendidos de forma não supervisionada perdem a chance de melhorar o resultado pela consideração de informação prévia sobre o FD.

As técnicas apresentadas neste capítulo utilizam métodos variados de aprendizado de máquina e implementam diferentes mecanismos para atacar os problemas característicos de aprendizado em FD. Alguns trabalhos sugerem adaptações e extensões para melhorar um ou outro aspecto de um método. De forma geral, a forma de como lidar com o tempo (mecanismos de esquecimento) e detecção de desvio de conceitos são duas tarefas relevantes que não são consideradas por todas as técnicas de aprendizado em FD. De

qualquer forma, a revisão colocada no capítulo contribui para uma visão geral do estado da arte quanto ao aprendizado em FD.

O Capítulo 4 discute a proposta de trabalho para elaboração da tese de doutorado dentro do tema de aprendizado de máquina semissupervisionado em FD.

Parte II

Proposta, Experimentos e Resultados

4

Proposta

Dificuldade de generalizar técnicas *crisp* de agrupamento em fluxo de dados para técnicas *fuzzy* (HORE; HALL; GOLDGOF, 2007b)

Maior parte das abordagens utiliza protótipos ponderados como estrutura de sumarização. Processa os dados em chunks.

- Uma das dificuldades nessa estratégia é determinar o valor de h, uma vez que essa definição depende do domínio ao qual é aplicado o agrupamento.
- Outra complicação de estratégias baseadas no SPFCM é a definição prévia de um número fixo de grupos, pois, uma vez que um FD é mutável de acordo com o tempo pode ocorrer o aumento ou diminuição no número de grupos. Outro obstáculo: identificação de grupos de formato arbitrário (?)

5

Experimentos

5.1 Conjuntos de Dados

5.1.1 Sintéticos

Estacionários

gerador R: Barras e Gaussianas com 10000 exemplos - A data stream generator which creates the shape of two bars and two Gaussians clusters with different density.

gerador R: Mistura de Gaussianas - A data stream generator that produces a data stream with a mixture of static Gaussians

3 Gaussianas em 2 dimensões com 10000 exemplos 4 Gaussianas em 2 dimensões com 10000 exemplos

Mistura de 3 Gaussianas em 3 dimensões com 10000 exemplos

Não Estacionários

gerador R: Benchmark - A data stream generator that generates several dynamic streams indented to be benchmarks to compare data stream clustering algorithms.

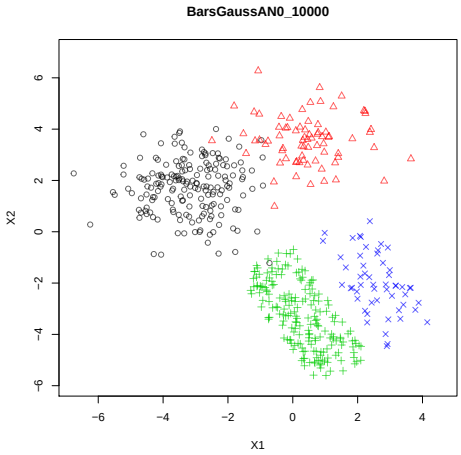
Benchamrk1 com 5500 exemplos (vai)

Benchmark1 com 11000 exemplos (vai e volta)

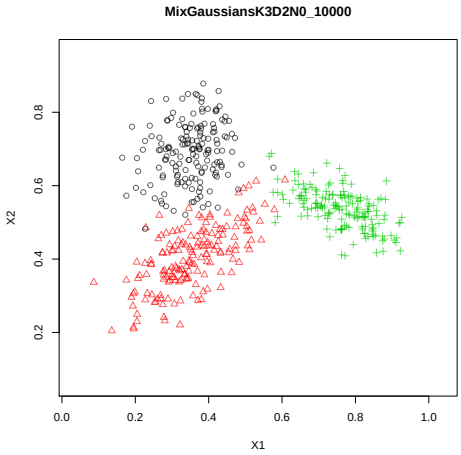
se eu conseguir rodar

5.1.2 Benchmark

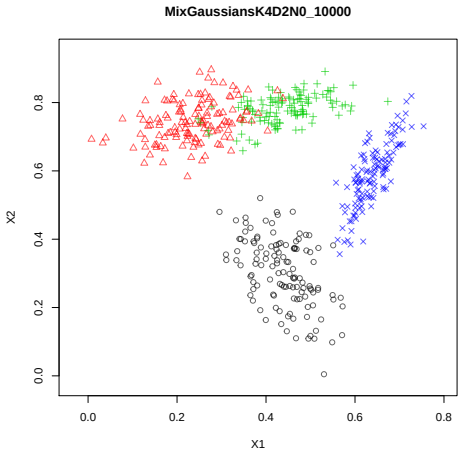
- KDDCup 99
- Forest Covertype



(a) Duas barras e duas gaussianas em duas dimensões



(b) Três gaussianas em duas dimensões



(c) Quatro gaussianas em duas dimensões

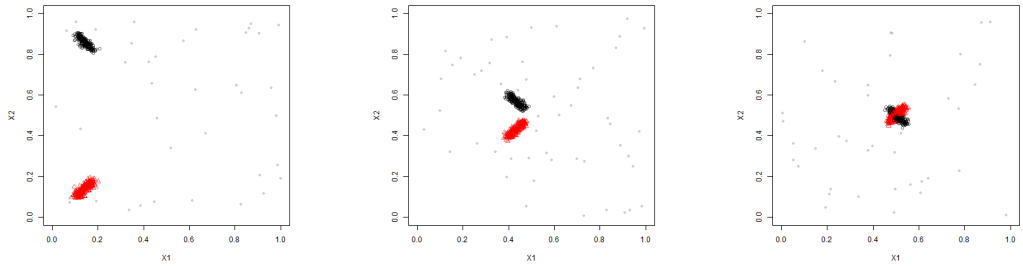
Figura 5.1

5.2 Ferramentas

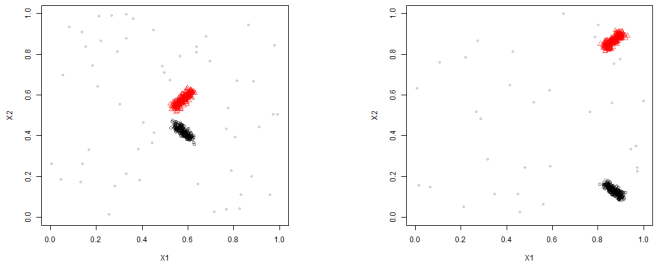
Com o crescimento da pesquisa sobre aprendizado em FD, é interessante o investimento em ferramenta de software para a aplicação das diversas técnicas propostas. Existem hoje ferramentas, disponíveis gratuitamente, como:

- **MOA (*Massive On-line Analysis*** (BIFET et al., 2010) - um *framework* de código aberto que disponibiliza implementação de uma série de algoritmos e métricas para classificação, principalmente, e para agrupamento em FD. A ferramenta também conta com recursos para visualização dos processos de aprendizado.

- **VFML (*Very Fast Machine Learning*)** (HULTEN; DOMINGOS, 2003) - um pacote de implementações para mineração de FD de alta velocidade e conjuntos de exemplos *very large*.



(a) $t = 0$ (b) $t = 4$ (c) $t = 5$



(d) $t = 6$ (e) $t = 10$

Figura 5.2: Plot de 500 exemplos do conjunto Benchmark1 em diferentes tempos t

A linguagem R (R Core Team, 2016) vem sendo muito utilizada para análise de dados e aprendizado de máquina, ... Foi escolhida para implementação da proposta e execução de experimentos.

O pacote **stream** (HAHSLER; BOLANOS; FORREST, 2016) ... R

O pacote **streamMOA** (HAHSLER; BOLANOS; FORREST, 2015) ... interface java moa – R

O pacote **e1071** (MEYER et al., 2015) ... FCM

5.3 Técnicas

- DenStream (micro + macro)
- DenStream (micro) + FCM (macro)
- DenStream (micro) + KMeans (macro)
- DStream (micro + macro)
- Sample (micro) + FCM (macro)
- Sample (micro) + KMeans (macro)
- Window (micro) + FCM (macro)

- Window (micro) + KMeansM (macro)
- FuzzyMicro (micro) + FCM (macro)
- FuzzyMicro (micro) + KMeans (macro)
- Clustream (micro + macro)
- FuzzyMicro2 (micro) + FCM (macro)
- FuzzyMicro2 (micro) + KMeans (macro)
- FuzzyMicro3 (micro) + FCM (macro)
- FuzzyMicro3 (micro) + KMeans (macro)

5.4 Validação

a partir de funções existentes em pacotes R.

- NumMicro
- NumMacro
- numClasses
- SSQ
- silhouette ***
- precision
- recall ***
- F1 ***
- purity
- cRand
- Tempo de Execução (?)

Questão de cluster assignment - validação fuzzy...

Referências

AGGARWAL, C. C. An Introduction to Data Streams. In: AGGARWAL, C. C. (Ed.). *Data Streams*. Boston, MA: Springer US, 2007. p. 1–8.

AGGARWAL, C. C. et al. A Framework for Clustering Evolving Data Streams. In: *Proceedings of the 29th International Conference on Very Large Data bases*. [S.l.: s.n.], 2003. v. 29, p. 81–92.

AGGARWAL, C. C. et al. On Clustering Massive Data Streams: A Summarization Paradigm. In: AGGARWAL, C. C. (Ed.). *Data Streams*. Boston, MA: Springer US, 2007. p. 9–38.

AGGARWAL, C. C.; YU, P. S. A Framework for Clustering Uncertain Data Streams. In: *2008 IEEE 24th International Conference on Data Engineering*. [S.l.]: IEEE, 2008. v. 00, p. 150–159.

AHMADI, Z.; BEIGY, H. Semi-supervised Ensemble Learning of Data Streams in the Presence of Concept Drift. In: *Hybrid Artificial Intelligent Systems*. [S.l.]: Springer Berlin Heidelberg, 2012. p. 526537.

AHMED, M. S.; KHAN, L.; RAJESWARI, M. Using Correlation Based Subspace Clustering for Multi-label Text Data Classification. In: *2010 22nd IEEE International Conference on Tools with Artificial Intelligence*. [S.l.]: IEEE, 2010. p. 296–303.

AMINI, A.; WAH, T. Y.; SABOOHI, H. On Density-Based Data Streams Clustering Algorithms: A Survey. *Journal of Computer Science and Technology*, v. 29, n. 1, p. 116–141, jan 2014.

AMINI, A. et al. A study of density-grid based clustering algorithms on data streams. In: *2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*. [S.l.]: IEEE, 2011. p. 1652–1656.

ANKERST, M. et al. OPTICS. In: *Proceedings of the 1999 ACM SIGMOD international conference on Management of data - SIGMOD '99*. New York, New York, USA: ACM Press, 1999. p. 49–60.

ASTUDILLO, C. A.; OOMMEN, B. J. Imposing tree-based topologies onto self organizing maps. *Information Sciences*, v. 181, n. 18, p. 3798–3815, 2011.

ASTUDILLO, C. A.; OOMMEN, B. J. On achieving semi-supervised pattern recognition by utilizing tree-based SOMs. *Pattern Recognition*, v. 46, n. 1, p. 293–304, 2013.

ATASHPAZ-GARGARI, E.; LUCAS, C. Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition. In: *2007 IEEE Congress on Evolutionary Computation, CEC 2007*. [S.l.: s.n.], 2007. p. 4661–4667.

ATWA, W.; LI, K. Clustering Evolving Data Stream with Affinity Propagation Algorithm. In: *Database and Expert Systems Applications*. [S.l.]: Springer International Publishing, 2014. p. 446–453.

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 - PODS '02*. New York, New York, USA: ACM Press, 2002. p. 1–16.

BASU, S.; BANERJEE, A.; MOONEY, R. Semi-supervised Clustering by Seeding. In: *Proceedings of the 19th International Conference on Machine Learning (ICML-2002)*. [S.l.: s.n.], 2002. p. 27–34.

BASU, S.; BANERJEE, A.; MOONEY, R. J. Active Semi-Supervision for Pairwise Constrained Clustering. In: *Proceedings of the 2004 SIAM International Conference on Data Mining*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2004. p. 333–344.

BENSAID, A. M.; BEZDEK, J. C. Semi-Supervised Point Prototype Clustering. *International Journal of Pattern Recognition and Artificial Intelligence*, v. 12, n. 05, p. 625–643, aug 1998.

BENSAID, A. M. et al. Partially supervised clustering for image segmentation. *Pattern Recognition*, v. 29, n. 5, p. 859–871, may 1996.

BERTINI, J. R.; LOPES, A. D. A.; ZHAO, L. Partially labeled data stream classification with the semi-supervised K-associated graph. *Journal of the Brazilian Computer Society*, v. 18, n. 4, p. 299–310, 2012.

BERTINI, J. R.; ZHAO, L. A Comparison of Two Purity-Based Algorithms When Applied to Semi-supervised Streaming Data Classification. In: *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*. [S.l.]: IEEE, 2013. p. 21–27.

BEZDEK, J. C. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Boston, MA: Springer US, 1981. 256 p.

BHATNAGAR, V. et al. Data Understanding using Semi-Supervised Clustering. In: *2012 Conference on Intelligent Data Understanding*. [S.l.]: IEEE, 2012. p. 118–123.

BIFET, A. et al. MOA: Massive Online Analysis, a Framework for Stream Classification and Clustering. In: *JMLR: Workshop and Conference Proceedings*. [S.l.: s.n.], 2010. v. 11, p. 44–50.

BISHOP, C. M. *Neural Networks for Pattern Recognition*. [S.l.]: Oxford University Press, 1995.

BOARD, R.; PITT, L. Semi-supervised learning. *Machine Learning*, Kluwer Academic Publishers, v. 4, n. 1, p. 41–65, 1989.

BORCHANI, H.; LARRANAGA, P.; BIELZA, C. Classifying evolving data streams with partially labeled data. *Intelligent Data Analysis*, v. 15, n. 5, p. 655–670, 2011.

BOUCHACHIA, A.; VANARET, C. GT2FC: An Online Growing Interval Type-2 Self-Learning Fuzzy Classifier. *IEEE Transactions on Fuzzy Systems*, v. 22, n. 4, p. 999–1018, aug 2014.

BOUGUELIA, M.-R.; BELAID, Y.; BELAID, A. A Stream-Based Semi-supervised Active Learning Approach for Document Classification. In: *2013 12th International Conference on Document Analysis and Recognition*. [S.l.]: IEEE, 2013. p. 611–615.

BREVE, F.; ZHAO, L. Particle competition and cooperation in networks for semi-supervised learning with concept drift. In: *The 2012 International Joint Conference on Neural Networks (IJCNN)*. [S.l.]: IEEE, 2012. p. 1–6.

BREVE, F.; ZHAO, L. Semi-supervised Learning with Concept Drift Using Particle Dynamics Applied to Network Intrusion Detection Data. In: *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*. [S.l.]: IEEE, 2013. p. 335–340.

BRUNEAU, P.; PICAROUGNE, F.; GELGON, M. Incremental semi-supervised clustering in a data stream with a flock of agents. In: *2009 IEEE Congress on Evolutionary Computation*. [S.l.]: IEEE, 2009. p. 3067–3074.

CAO, F. et al. Density-Based Clustering over an Evolving Data Stream with Noise. In: *Proceedings of the 6th SIAM International Conference on Data Mining*. [S.l.: s.n.], 2006. p. 328–339.

CAO, Y.; HE, H. Learning from testing data: A new view of incremental semi-supervised learning. In: *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. [S.l.]: IEEE, 2008. p. 2872–2878.

CASAS, P.; MAZEL, J.; OWEZARSKI, P. MINETRAC: Mining flows for unsupervised analysis & semi-supervised classification. In: *Proceedings of the 23rd International Teletraffic Congress*. [S.l.: s.n.], 2011. p. 87–94.

CAYROL, M.; FARRENY, H.; PRADE, H. Fuzzy Pattern Matching. *Kybernetes*, v. 11, n. 2, p. 103–116, 1982.

CHAIRUKWATTANA, R. et al. SE-Stream: Dimension Projection for Evolution-Based Clustering of High Dimensional Data Streams. In: *Knowledge and Systems Engineering*. [S.l.]: Springer International Publishing, 2014. p. 365–376.

CHAPELLE, O.; SCHÖLKOPF, B.; ZIEN, A. *Semi-Supervised Learning*. [S.l.]: MIT Press, 2006. 523 p.

CHEN, J.; CHEN, P.; SHENG, X. A Sketch-based Clustering Algorithm for Uncertain Data Streams. *Journal of Networks*, v. 8, n. 7, p. 1536–1542, jul 2013.

CHEN, Y.; TU, L. Density-based clustering for real-time stream data. In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '07*. New York, New York, USA: ACM Press, 2007. p. 133–142.

CHENG, Y. et al. Learning to Group Web Text Incorporating Prior Information. In: *2011 IEEE 11th International Conference on Data Mining Workshops*. [S.l.]: IEEE, 2011. p. 212–219.

CINTRA, M. E.; MONARD, M. C.; CAMARGO, H. A. FuzzyDT - A Fuzzy Decision Tree Algorithm Based on C4. 5. In: *CBSF - Brazilian Congress on Fuzzy Systems*. [S.l.: s.n.], 2012. p. 199–211.

CORDÓN, O. A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems. *International Journal of Approximate Reasoning*, v. 52, n. 6, p. 894–913, sep 2011.

COVER, T.; HART, P. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, v. 13, n. 1, p. 21–27, jan 1967.

DASU, T.; KRISHNAN, S. An information-theoretic approach to detecting changes in multi-dimensional data streams. In: *Proceedings of the Symposium on the Interface of Statistics, Computing Science, and Applications*. [S.l.: s.n.], 2006. p. 1–24.

DAY, W. H. E.; EDELSBRUNNER, H. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, v. 1, n. 1, p. 7–24, dec 1984.

DITZLER, G.; POLIKAR, R. Semi-supervised learning in nonstationary environments. In: *The 2011 International Joint Conference on Neural Networks*. [S.l.]: IEEE, 2011. p. 2741–2748.

DOMINGOS, P.; HULTEN, G. Mining high-speed data streams. In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '00*. [S.l.: s.n.], 2000. p. 71–80.

DUBOIS, D.; PRADE, H.; TESTEMALE, C. Weighted fuzzy pattern matching. *Fuzzy Sets and Systems*, v. 28, n. 3, p. 313–331, 1988.

DUDA, R. O.; HART, P. E. *Pattern classification and scene analysis*. [S.l.]: John Wiley and Sons, 1973.

DYER, K. B.; CAPO, R.; POLIKAR, R. COMPOSE: A semisupervised learning framework for initially labeled nonstationary streaming data. *IEEE transactions on neural networks and learning systems*, v. 25, n. 1, p. 12–26, jan 2014.

ESTER, M. et al. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: *Second International Conference on Knowledge Discovery and Data Mining*. [S.l.: s.n.], 1996. p. 226–231.

FAZZOLARI, M. et al. A Review of the Application of Multiobjective Evolutionary Fuzzy Systems : Current Status and Further Directions. *Fuzzy Systems, IEEE Transactions on*, v. 21, n. 1, p. 45–65, 2013.

FDEZ-RIVEROLA, F. et al. Applying lazy learning algorithms to tackle concept drift in spam filtering. *Expert Systems with Applications*, v. 33, n. 1, p. 36–48, jul 2007.

FORESTIERO, A.; PIZZUTI, C.; SPEZZANO, G. FlockStream: A Bio-Inspired Algorithm for Clustering Evolving Data Streams. In: *2009 21st IEEE International Conference on Tools with Artificial Intelligence*. [S.l.]: IEEE, 2009. p. 1–8.

FORESTIERO, A.; PIZZUTI, C.; SPEZZANO, G. A single pass algorithm for clustering evolving data streams based on swarm intelligence. *Data Mining and Knowledge Discovery*, v. 26, p. 1–26, 2013.

FRANDINA, S. et al. On-Line Laplacian One-Class Support Vector Machines. In: *Artificial Neural Networks and Machine Learning (ICANN2013)*. [S.l.]: Springer Berlin Heidelberg, 2013. p. 186–193.

FREY, B. J.; DUECK, D. Clustering by Passing Messages Between Data Points. *Science*, v. 315, n. 5814, p. 947–949, feb 2007.

GAMA, J. *Knowledge Discovery from Data Streams*. [S.l.]: Chapman and Hall, 2010. 255 p.

GAMA, J.; GABER, M. M. (Ed.). *Learning from Data Streams: Processing Techniques in Sensor Networks*. [S.l.]: Springer, 2007.

GAMA, J. et al. Learning with drift detection. In: *Advances in Artificial Intelligence (SBIA2004)*. [S.l.: s.n.], 2004. p. 286–295.

GAMA, J.; RODRIGUES, P. P.; LOPES, L. Clustering Distributed Sensor Data Streams Using Local Processing and Reduced Communication. *Intelligent Data Analysis*, IOS Press, Amsterdam, The Netherlands, The Netherlands, v. 15, n. 1, p. 3–28, 2011.

GAO, J. et al. An incremental data stream clustering algorithm based on dense units detection. *Advances in Knowledge Discovery and Data Mining*, v. 3518, p. 420–425, 2005.

GAO, M.-m.; LIU, J.-z.; GAO, X.-x. Application of Compound Gaussian Mixture Model clustering in the data stream. In: *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*. [S.l.]: IEEE, 2010. p. V7–172–V7–177.

GENG, C. et al. An Algorithm of Semi-supervised Web-Page Classification Based on Fuzzy Clustering. In: *2009 International Forum on Information Technology and Applications*. [S.l.]: IEEE, 2009. v. 1, p. 3–7.

GOLDBERG, D. *Genetic Algorithms in Search, Optimization, and Machine Learning*. [S.l.]: Addison-Wesley, 1989. 432 p.

GRIRA, N.; CRUCIANU, M.; BOUJEMAA, N. Active semi-supervised fuzzy clustering for image database categorization. In: *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval - MIR '05*. [S.l.: s.n.], 2005. p. 9–16.

GRIRA, N.; CRUCIANU, M.; BOUJEMAA, N. Fuzzy clustering with pairwise constraints for knowledge-driven image categorisation. In: *IEE Proceedings - Vision, Image, and Signal Processing*. [S.l.: s.n.], 2006. v. 153, p. 299–304.

GRIRA, N.; CRUCIANU, M.; BOUJEMAA, N. Active semi-supervised fuzzy clustering. *Pattern Recognition*, v. 41, n. 5, p. 1851–1861, 2008.

GUHA, S. et al. Clustering data streams. In: *Proceedings 41st Annual Symposium on Foundations of Computer Science*. [S.l.]: IEEE Comput. Soc, 2000. p. 359–366.

GUSTAFSON, D. E. G. D. E.; KESSEL, W. C. K. W. C. Fuzzy clustering with a fuzzy covariance matrix. *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, v. 17, n. 2, p. 761–766, 1978.

HAHSLER, M.; BOLANOS, M.; FORREST, J. *streamMOA: Interface for MOA Stream Clustering Algorithms*. [S.l.], 2015. R package version 1.1-2. Disponível em: <https://CRAN.R-project.org/package=streamMOA>.

HAHSLER, M.; BOLANOS, M.; FORREST, J. *stream: Infrastructure for Data Stream Mining*. [S.l.], 2016. R package version 1.2-3. Disponível em: <https://CRAN.R-project-.org/package=stream>.

HALKIDI, M.; SPILIOPOULOU, M.; PAVLOU, A. A semi-supervised incremental clustering algorithm for streaming data. *Advances in Knowledge Discovery and Data Mining*, v. 7301, p. 578–590, 2012.

HAMASUNA, Y.; ENDO, Y. On semi-supervised fuzzy c-means clustering with clusterwise tolerance by opposite criteria. In: *2011 IEEE International Conference on Granular Computing*. [S.l.]: IEEE, 2011. p. 225–230.

HAN, J.; KAMBER, M.; PEI, J. *Data Mining: Concepts and Techniques*. [S.l.]: Morgan Kaufmann Publishers, 2012. 744 p. (Data Management Systems Series).

HASSANI, M. et al. Density-Based Projected Clustering of Data Streams. In: *Scalable Uncertainty Management*. [S.l.: s.n.], 2012. p. 311–324.

HINNEBURG, A.; KEIM, D. A. An Efficient Approach to Clustering in Large Multimedia Databases with Noise. In: *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98)*. [S.l.: s.n.], 1998. v. 5865, p. 58–65.

HORE, P. et al. Online fuzzy c means. In: *NAFIPS 2008 - 2008 Annual Meeting of the North American Fuzzy Information Processing Society*. [S.l.]: IEEE, 2008. p. 1–5.

HORE, P.; HALL, L. O.; GOLDGOF, D. B. A fuzzy c means variant for clustering evolving data streams. In: *2007 IEEE International Conference on Systems, Man and Cybernetics*. [S.l.]: IEEE, 2007. p. 360–365.

HORE, P.; HALL, L. O.; GOLDGOF, D. B. Creating Streaming Iterative Soft Clustering Algorithms. In: *NAFIPS 2007 - 2007 Annual Meeting of the North American Fuzzy Information Processing Society*. [S.l.]: IEEE, 2007. p. 484–488.

HORE, P.; HALL, L. O.; GOLDGOF, D. B. Single pass fuzzy c means. *IEEE International Conference on Fuzzy Systems*, 2007.

HULTEN, G.; DOMINGOS, P. *VFML - A toolkit for mining high-speed time-changing data streams*. 2003. Disponível em: <http://www.cs.washington.edu/dm/vfml/>.

ISAKSSON, C.; DUNHAM, M. H.; HAHSLER, M. SOStream: Self Organizing Density-Based Clustering over Data Stream. In: *Machine Learning and Data Mining in Pattern Recognition*. [S.l.: s.n.], 2012. v. 7376, p. 264–278.

JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data clustering: a review. *ACM Computing Surveys*, ACM, New York, NY, USA, v. 31, n. 3, p. 264–323, sep 1999.

JANIKOW, C. Z. Fuzzy decision trees: issues and methods. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, v. 28, n. 1, p. 1–14, 1998.

JIANG, N.; GRUENWALD, L. Research issues in data stream association rule mining. *ACM SIGMOD Record*, v. 35, p. 14–19, 2006.

KASABOV, N. et al. Dynamic evolving spiking neural networks for on-line spatio- and spectro-temporal pattern recognition. *Neural networks : the official journal of the International Neural Network Society*, Elsevier Ltd, v. 41, n. 1995, p. 188–201, may 2013.

KAUFMAN, L.; ROUSSEEUW, P. J. *Finding Groups in Data: An Introduction to Cluster Analysis*. [S.l.]: John Wiley and Sons, 1990. 368 p.

KENNEDY, J.; EBERHART, R. C.; SHI, Y. *Swarm Intelligence*. [S.l.]: Morgan Kaufmann, 2001. 512 p.

KHOLGHI, M.; KEYVANPOUR, M. Active Learning Framework Combining Semi-Supervised Approach for Data Stream Mining. In: *Intelligent Computing and Information Science*. [S.l.]: Springer Berlin Heidelberg, 2011, (Communications in Computer and Information Science, v. 135). p. 238–243.

KLOSE, A. et al. Data mining with neuro-fuzzy models. In: KANDEL, A.; LAST, M.; BUNKE, H. (Ed.). *Data Mining and Computational Intelligence*. Heidelberg, Germany: Physica-Verlag GmbH, 2001. p. 1–35.

KMIECIAK, M. R.; STEFANOWSKI, J. Semi-supervised approach to handle sudden concept drift. *Control and Cybernetics*, v. 40, n. 3, p. 667–695, 2011.

KOHONEN, T. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, v. 43, p. 59–69, 1982.

KRANEN, P. et al. The ClusTree: indexing micro-clusters for anytime stream mining. *Knowledge and Information Systems*, Springer-Verlag, v. 29, n. 2, p. 249–272, 2011.

KRISHNAPURAM, R.; KELLER, J. M. The possibilistic C-means algorithm: Insights and recommendations. *IEEE Transactions on Fuzzy Systems*, v. 4, n. 3, p. 385–393, 1996.

LABROCHE, N. Online fuzzy medoid based clustering algorithms. *Neurocomputing*, Elsevier, v. 126, p. 141–150, feb 2014.

LABZOUR, N.; BENSAID, A.; BEZDEK, J. Improved semi-supervised point-prototype clustering algorithms. In: *1998 IEEE International Conference on Fuzzy Systems Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98CH36228)*. [S.l.]: IEEE, 1998. v. 2, p. 1383–1387.

LANGLEY, P. The changing science of machine learning. *Machine Learning*, v. 82, n. 3, p. 275–279, 2011.

LEITE, D.; COSTA, P.; GOMIDE, F. Evolving granular neural network for semi-supervised data stream classification. In: *The 2010 International Joint Conference on Neural Networks (IJCNN)*. [S.l.]: IEEE, 2010. p. 1–8.

LEMOS, A.; CAMINHAS, W.; GOMIDE, F. Adaptive fault detection and diagnosis using an evolving fuzzy classifier. *Information Sciences*, Elsevier Inc., v. 220, p. 64–85, jan 2013.

LI, F. A Pattern Query Strategy Based on Semi-supervised Machine Learning in Distributed WSNs. *Journal of Information and Computational Science*, v. 11, n. 18, p. 6447–6459, dec 2014.

LI, P.; WU, X.; HU, X. Learning from Concept Drifting Data Streams with Unlabeled Data. In: *Proceedings of the National Conference on Artificial Intelligence*. [S.l.: s.n.], 2010. p. 1945–1946.

LI, P.; WU, X.; HU, X. Mining Recurring Concept Drifts with Limited Labeled Streaming Data. In: *JMLR: Workshop and Conference Proceedings 13*. [S.l.: s.n.], 2010. v. 3, n. 2, p. 241–252.

LI, P.; WU, X.; HU, X. Mining Recurring Concept Drifts with Limited Labeled Streaming Data. *ACM Transactions on Intelligent Systems and Technology*, v. 3, n. 2, p. 1–32, feb 2012.

LI-XIONG, L. L.-x. L. et al. A three-step clustering algorithm over an evolving data stream. In: *2009 IEEE International Conference on Intelligent Computing and Intelligent Systems*. [S.l.: s.n.], 2009. v. 1, p. 160–164.

LIANG, C. et al. Learning very fast decision tree from uncertain data streams with positive and unlabeled samples. *Information Sciences*, Elsevier Inc., v. 213, p. 50–67, dec 2012.

LIN, J.; LIN, H. A density-based clustering over evolving heterogeneous data stream. In: *2009 Second ISECS International Colloquium on Computing, Communication, Control, and Management, CCCM 2009*. [S.l.: s.n.], 2009. v. 4, p. 275–277.

LIU, H.; HUANG, S.-t. A Genetic Semi-supervised Fuzzy Clustering Approach to Text Classification. In: *Advances in Web-Age Information Management*. [S.l.: s.n.], 2003. p. 173–180.

LIU, J. et al. A Semi-supervised Ensemble Approach for Mining Data Streams. *Journal of Computers*, v. 8, n. 11, p. 2873–2879, nov 2013.

LIU, Y.; ZHOU, Y. Online Detection of Concept Drift in Visual Tracking. In: *Neural Information Processing*. [S.l.]: Springer International Publishing, 2014. p. 159–166.

MACQUEEN, J. B. Some Methods for Classification and Analysis of MultiVariate Observations. In: *In Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*. [S.l.]: University of California Press, 1967. v. 1, p. 281–297.

MARIN, L. et al. On-line dynamic adaptation of fuzzy preferences. *Information Sciences*, Elsevier Inc., v. 220, p. 5–21, jan 2013.

MASUD, M. M. et al. A Practical Approach to Classify Evolving Data Streams: Training with Limited Amount of Labeled Data. In: THE UNIVERSITY OF TEXAS AT DALLAS. *2008 Eighth IEEE International Conference on Data Mining*. [S.l.]: IEEE, 2008. p. 929–934.

MASUD, M. M. et al. Facing the reality of data stream classification: coping with scarcity of labeled data. In: *Knowledge and Information Systems*. [S.l.: s.n.], 2012. v. 33, n. 1, p. 213–244.

MEESUKSABAI, W.; KANGKACHIT, T.; WAIYAMAI, K. HUE-Stream: Evolution-Based Clustering Technique for Heterogeneous Data Streams with Uncertainty. In: *Advanced Data Mining and Applications*. [S.l.]: Springer Berlin Heidelberg, 2011. p. 27–40.

MENDEL, J.; JOHN, R. Type-2 fuzzy sets made simple. *IEEE Transactions on Fuzzy Systems*, v. 10, n. 2, p. 117–127, apr 2002.

MEYER, D. et al. *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien*. [S.l.], 2015. R package version 1.6-7. Disponível em: <https://CRAN.R-project.org/package=e1071>.

MITCHELL, T. *Machine Learning*. [S.l.]: McGraw-Hill Education (ISE Editions), 1997.

MOUCHAWEH, M. S. Semi-supervised classification method for dynamic applications. *Fuzzy Sets and Systems*, Elsevier, v. 161, n. 4, p. 544–563, feb 2010.

NAHAR, V. et al. Semi-supervised Learning for Cyberbullying Detection in Social Networks. In: *Databases Theory and Applications*. [S.l.]: Springer International Publishing, 2014. p. 160–171.

NAHAR, V. et al. Detecting cyberbullying in social networks using multi-agent system. *Web Intelligence and Agent Systems*, v. 12, n. 4, p. 375–388, 2014.

NGUYEN, H.; NG, W.; WOON, Y. Concurrent Semi-supervised Learning with Active Learning of Data Streams. In: *Transactions on Large-Scale Data- and Knowledge-Centered Systems VIII*. [S.l.]: Springer Berlin Heidelberg, 2013. v. 7790, p. 113–136.

NGUYEN, H.-l. et al. Concurrent Semi-supervised Learning of Data Streams. In: *Data Warehousing and Knowledge Discovery*. [S.l.]: Springer Berlin Heidelberg, 2011. p. 445–459.

NTOUTSI, I. et al. Density-based Projected Clustering over High Dimensional Data Streams. In: *Proceedings of the 2012 SIAM International Conference on Data Mining*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2012. p. 987–998.

O'CALLAGHAN, L. et al. Streaming-data algorithms for high-quality clustering. In: *Proceedings 18th International Conference on Data Engineering*. [S.l.]: IEEE Comput. Soc, 2002. p. 685–694.

PAN, J.; YANG, Q.; PAN, S. Online co-localization in indoor wireless networks by dimension reduction. In: *Proceedings of the National Conference on Artificial Intelligence*. [S.l.: s.n.], 2007. p. 1102–1107.

PEDRYCZ, W. Algorithms of fuzzy clustering with partial supervision. *Pattern Recognition Letters*, v. 3, n. January, p. 13–20, 1985.

PEDRYCZ, W. et al. Fuzzy clustering with partial supervision in organization and classification of digital images. *IEEE Transactions on Fuzzy Systems*, v. 16, n. 4, p. 1008–1026, 2008.

PEDRYCZ, W.; GOMIDE, F. *An Introduction to Fuzzy Sets: Analysis and Design*. [S.l.]: MIT Press, 1998. (A Bradford book).

PEDRYCZ, W.; WALETZKY, J. Fuzzy clustering with partial supervision. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, v. 27, n. 5, p. 787–795, 1997.

QUINLAN, J. R. Induction of decision trees. *Machine Learning*, v. 1, n. 1, p. 81–106, 1986.

QUINLAN, J. R. *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2016. Disponível em: <https://www.R-project.org/>.

REN, J.; MA, R. Density-Based Data Streams Clustering over Sliding Windows. In: *2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery*. [S.l.]: IEEE, 2009. v. 5, p. 248–252.

RUIZ, C.; MENASALVAS, E.; SPILIOPOULOU, M. C-DenStream: Using domain knowledge on a data stream. In: *Discovery Science*. [S.l.]: Springer Berlin Heidelberg, 2009. p. 287–301.

SCHWENKER, F.; TRENTIN, E. Partially supervised learning for pattern recognition. *Pattern Recognition Letters*, v. 37, p. 1–3, feb 2014.

SETTLES, B. *Active Learning Literature Survey*. [S.l.], 2010. 65 p. Disponível em: <http://burrsettles.com/pub/settles.activelearning.pdf>.

SHAMSHIRBAND, S. et al. D-FICCA: A density-based fuzzy imperialist competitive clustering algorithm for intrusion detection in wireless sensor networks. *Measurement*, Elsevier Ltd, v. 55, p. 212–226, sep 2014.

SHEIKHOLESLAMI, G.; CHATTERJEE, S.; ZHANG, A. WaveCluster: a wavelet-based clustering approach for spatial data in very large databases. *The VLDB Journal The International Journal on Very Large Data Bases*, v. 8, n. 3-4, p. 289–304, feb 2000. ISSN 1066-8888.

SHI, X. et al. An incremental affinity propagation algorithm and its applications for text clustering. In: *2009 International Joint Conference on Neural Networks*. [S.l.]: IEEE, 2009. p. 2914–2919.

SILVA, D. et al. Semi-supervised classification of characterized patterns for demand forecasting using smart electricity meters. In: *2011 International Conference on Electrical Machines and Systems*. [S.l.]: IEEE, 2011. p. 1–6.

SILVA, J. A. et al. Data stream clustering: A survey. *ACM Computing Surveys*, v. 46, n. 1, p. 1–31, oct 2013.

SIRAMPUJ, T.; KANGKACHIT, T.; WAIYAMAI, K. CE-Stream : Evaluation-based technique for stream clustering with constraints. In: *The 2013 10th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. [S.l.]: IEEE, 2013. p. 217–222.

TASOULIS, D. K.; ROSS, G.; ADAMS, N. M. Visualising the Cluster Structure of Data Streams. In: *Advances in Intelligent Data Analysis VII*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. p. 81–92.

TIWARI, P.; KURHANEWICZ, J. Semi supervised multi kernel (SeSMiK) graph embedding: identifying aggressive prostate cancer via magnetic resonance imaging and spectroscopy. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI 2010)*. [S.l.]: Springer Berlin Heidelberg, 2010. p. 666–673.

TU, L.; CHEN, Y. Stream data clustering based on grid density and attraction. *ACM Transactions on Knowledge Discovery from Data*, v. 3, n. 3, p. 12:1–12:27, 2009.

UDOMMANETANAKIT, K.; RAKTHANMANON, T.; WAIYAMAI, K. E-Stream: Evolution-Based Technique for Stream Clustering. In: *Advanced Data Mining and Applications*. [S.l.]: Springer Berlin Heidelberg, 2007. p. 605–615.

VITO, S. et al. Semi-Supervised Learning Techniques in Artificial Olfaction: A Novel Approach to Classification Problems and Drift Counteraction. *IEEE Sensors Journal*, v. 12, n. 11, p. 3215–3224, nov 2012.

WAGSTAFF, K. et al. Constrained K-means Clustering with Background Knowledge. In: *Proceedings of the Eighteenth International Conference on Machine Learning (ICML '01)*. [S.l.: s.n.], 2001. p. 577–584.

WANG, W.; YANG, J.; MUNTZ, R. STING: A statistical information grid approach to spatial data mining. In: *Proceedings of International Conference on Very Large Data*. [S.l.: s.n.], 1997. p. 1–18.

WITTEN, I. H.; FRANK, E. *Data Mining: Practical machine learning tools and techniques*. [S.l.: s.n.], 2005. 560 p.

WU, Q.-Y.; YE, Y.; FU, J. Learnable topical crawler through online semi-supervised clustering. In: *2009 International Conference on Machine Learning and Cybernetics*. [S.l.]: IEEE, 2009. p. 231–236.

WU, S.; YANG, C.; ZHOU, J. Clustering-training for Data Stream Mining. In: *Sixth IEEE International Conference on Data Mining - Workshops (ICDMW'06)*. [S.l.]: IEEE, 2006. p. 653–656.

WU, X.; LI, P.; HU, X. Learning from Concept Drifting Data Streams with Unlabeled Data. *Neurocomputing*, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 92, p. 145–155, 2012.

XU, W.-h.; QIN, Z.; CHANG, Y. Clustering feature decision trees for semi-supervised classification from high-speed data streams. *Journal of Zhejiang University SCIENCE C*, SP Zhejiang University Press, v. 12, n. 8, p. 615–628, 2011.

YAN, Y.; CHEN, L. Label-based semi-supervised fuzzy co-clustering for document categoraization. In: *2011 8th International Conference on Information, Communications & Signal Processing*. [S.l.]: IEEE, 2011. p. 1–5.

YANG, C.; ZHOU, J. HClustream: A Novel Approach for Clustering Evolving Heterogeneous Data Stream. In: *Sixth IEEE International Conference on Data Mining - Workshops (ICDMW'06)*. [S.l.]: IEEE, 2006. p. 682–688.

YOGITA, Y.; TOSHNIWAL, D. Clustering techniques for streaming data - a survey. In: *2013 3rd IEEE International Advance Computing Conference (IACC)*. [S.l.]: IEEE, 2013. p. 951–956.

YU, Y. et al. Anomaly intrusion detection for evolving data stream based on semi-supervised learning. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. [S.l.: s.n.], 2009. v. 5506 LNCS, p. 571–578.

ZADEH, L. A. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338–353, 1965.

ZHANG, B. et al. A New Semi-supervised Learning Based Ensemble Classifier for Recurring Data Stream. In: *Pervasive Computing and the Networked World*. [S.l.]: Springer International Publishing, 2014. v. 8351, p. 759–765.

ZHANG, J.-p. et al. Online stream clustering using density and affinity propagation algorithm. In: *2013 IEEE 4th International Conference on Software Engineering and Service Science*. [S.l.]: IEEE, 2013. p. 828–832.

ZHANG, P. et al. A framework for application-driven classification of data streams. *Neurocomputing*, Elsevier, v. 92, p. 170–182, sep 2012.

ZHANG, P.; ZHU, X.; GUO, L. Mining Data Streams with Labeled and Unlabeled Training Examples. In: *2009 Ninth IEEE International Conference on Data Mining*. [S.l.]: IEEE, 2009. p. 627–636.

ZHANG, T.; RAMAKRISHNAN, R.; LIVNY, M. BIRCH: An Efficient Data Clustering Method for Very Large Databases. In: *Proceedings of the 1996 ACM SIGMOD international conference on Management of data - SIGMOD '96*. New York, New York, USA: ACM Press, 1996. (SIGMOD '96), p. 103–114.

ZHANG, X. et al. Data Stream Clustering With Affinity Propagation. *IEEE Transactions on Knowledge and Data Engineering*, v. 26, n. 7, p. 1644–1656, jul 2014.

ZHANG, X.; FURTLEHNER, C.; SEBAG, M. Data streaming with affinity propagation. In: *Machine Learning and Knowledge Discovery Database*. [S.l.]: Springer Berlin Heidelberg, 2008. p. 628–643.

ZHENPENG, L. et al. An Improved Semi-supervised K-Means Algorithm Based on Information Gain. In: *2014 IEEE 17th International Conference on Computational Science and Engineering*. [S.l.: s.n.], 2014. p. 1960–1963.

ZHOU, A. et al. Tracking clusters in evolving data streams over sliding windows. *Knowledge and Information Systems*, v. 15, n. 2, p. 181–214, may 2008.

ZHOU, A. et al. Distributed Data Stream Clustering: A Fast EM-based Approach. In: *2007 IEEE 23rd International Conference on Data Engineering*. [S.l.]: IEEE, 2007. p. 736–745. ISBN 1-4244-0802-4.

ZHU, X.; GOLDBERG, A. B. *Introduction to Semi-Supervised Learning*. [S.l.: s.n.], 2009. 130 p.

ZHU, Y.; SHASHA, D. StatStream: Statistical Monitoring of Thousands of Data Streams in Real Time. *Proceedings of the 28th international conference on Very Large Data Bases*, v. 54, p. 358–369, 2002.