ANDERSON JOSÉ DE SOUZA

COMPARAÇÃO ENTRE ABORDAGENS DE *DRIFT DETECTION* BASEADAS EM CONJUNTOS DE CLASSIFICADORES: UM ESTUDO DE CASO PARA PREVISÃO DE CRIMES

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito para obtenção do título de mestre em Informática.

Curitiba

ANDERSON JOSÉ DE SOUZA

COMPARAÇÃO ENTRE ABORDAGENS DE *DRIFT DETECTION* BASEADAS EM CONJUNTOS DE CLASSIFICADORES: UM ESTUDO DE CASO PARA PREVISÃO DE CRIMES

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito para obtenção do título de mestre em Informática.

Área de Concentração: Agentes de Software

Orientador: Prof. Dr. Fabrício Enembreck

Curitiba

Dezembro/2013

Dados da Catalogação na Publicação
Pontifícia Universidade Católica do Paraná
Sistema Integrado de Bibliotecas – SIBI/PUCPR
Biblioteca Central

S729c 2013

Souza, Anderson José de
Comparação entre abordagens de *Drift Detection* baseados em conjuntos de classificadores : um estudo de caso para previsão de crimes / Anderson José de Souza ; orientador, Fabricio Enembreck. – 2013. – 2013.
81 f. : il. ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2013
Bibliografia: f. 64-67

1. Informática. 2. Algoritmos. 3. Agentes inteligentes (Software). 4. Criminalidade urbana. I. Enembreck, Fabrício. II. Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática Aplicada. III. Título.

CDD 20. ed. – 004

Pontifícia Universidade Católica do Paraná
Escola Politécnica
Programa de Pós-Graduação em Informática

PUCPR
GRUPO MARISTA

ATA DE DEFESA DE DISSERTAÇÃO DE MESTRADO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

DEFESA DE DISSERTAÇÃO Nº 08/2013

Aos 02 dias do mês de Dezembro de 2013 realizou-se a sessão pública de Defesa da Dissertação **"Comparação entre Abordagens de *Drift Detection* Baseadas em Conjuntos de Classificadores: Um Estudo de Caso para Previsão de Crimes"** apresentado pelo aluno **Anderson José de Souza,** como requisito parcial para a obtenção do título de Mestre em Informática, perante uma Banca Examinadora composta pelos seguintes membros:

Prof. Dr. Fabrício Enembreck
PUCPR (Orientador) _____ APROVADO
(assinatura) *(Aprov/Reprov.).*

Prof. Dr. Julio Cesar Nievola
PUCPR _____ APROVADO
(assinatura) *(Aprov/Reprov.).*

Profª. Drª. Deborah Ribeiro Carvalho
PUCPR/PPGTS _____ APROVADO
(assinatura) *(Aprov/Reprov.).*

Conforme as normas regimentais do PPGIa e da PUCPR, o trabalho apresentado foi considerado _____APROVADO_____ (*aprovado/reprovado*), segundo avaliação da maioria dos membros desta Banca Examinadora. Este resultado está condicionado ao cumprimento integral das solicitações da Banca Examinadora registradas no Livro de Defesas do programa.

Prof. Dr. Mauro Sério Pereira Fonseca
Diretor do Programa de Pós-Graduação em Informática

Dedicatória

A Deus, a minha amada esposa Pricilla, aos meus pais José Edgard e Cleusa e a todos os meus familiares.

Agradecimentos

A Deus, pela onipresença e por ter me conduzido pela dádiva da vida e por estar sempre ao meu lado em todas as viagens e momentos.

Aos Professores, por não medirem esforços para a realização do Minter e pela dedicação destinada aos seus alunos. Agradeço, com especial consideração, ao professor Dr. Fabrício Enembreck, por suas orientações, pela compreensão durante os problemas passados, mas principalmente por ter me instruído na elaboração deste trabalho.

Aos meus Coordenadores e alunos pelo incentivo e apoio.

A minha amada esposa Pricilla, que compreendeu, apoiou e tomou conta de tudo na minha ausência, para sempre vou te Amar.

Aos meus pais José Edgard e Cleusa e aos meus irmãos por terem me incentivado e acreditado que esse dia chegaria.

Aos amigos que fiz durante esta jornada, especialmente aos colegas do Minter e amigos encontrados no Laboratório de Agentes que muito me auxiliaram. Enfatizo meu muito obrigado de forma especial aos amigos André Pinz Borges, Heitor Murilo Gomes e Jean Paul Barddal por quem desenvolvi uma profunda amizade, sou eternamente grato pelas dicas, orientações, conversas e apoio. Novamente ao Heitor por disponibilizar sua implementação do algoritmo *DWM*, que colaborou muito importante para a conclusão dessa pesquisa. A Cheila, secretária do PPGIA, sempre tão prestativa não medindo esforços para nos auxiliar.

Enfim, a todos que direta ou indiretamente sempre estiveram do meu lado, apoiando, incentivando e torcendo para o meu sucesso.

Sumário

CAPÍTULO 1 ..... **1**

INTRODUÇÃO **1**

1.1. Motivação 2
1.2. Objetivos 2
1.3. Proposta 3
1.4. Contribuições 4
1.5. Estrutura do Documento 4

CAPÍTULO 2 ..... **5**

REFERENCIAL TEÓRICO **5**

2.1. Aprendizagem de Máquina - AM 5
2.2. Concept Drift 7
2.3. Concept Drift Detection 10
2.4. Algoritmos para *Concept Drift Detection* 11
2.4.1 Algoritmo k-NN (*K-Nearest Neighbor*) 12
2.4.2 Algoritmo IB3 (Baseado em Instâncias) 13
2.5. Conjuntos de Classificadores 15
2.6. Algoritmos de Detecção de Mudanças baseados em Conjuntos de Classificadores 15
2.6.1 Algoritmo DWM (*Dynamic Weighted Majority*) 16
2.6.2 Algoritmo ADD Expert (*Additive Expert*) 18
2.7. Meta-Classificadores 19
2.7.1 *Bagging* 19
2.7.2 *Boosting* 20
2.7.3 *Leveraging Bagging* 21
2.7.4 Algoritmo *ASHT (Adaptive-Size Hoeffding Tree)* 21
2.7.5 Algoritmo ADWIN 22
2.8. *Drift Detection* e Ocorrências Policiais 22
2.9. A Plataforma *MUMPS* 26
2.10. A Central Regional de Emergência 190 26
2.11. A Ferramenta MOA (Massive Online Analysis) 27
Considerações Finais 28

CAPÍTULO 3 ..... **29**

METODOLOGIA **29**

3.1. Atividades da Pesquisa 29
3.2.1 Modelagem dos Dados 30
a) Extração dos Dados do EMAPE 30
b) Limpeza e Organização dos Dados 30
c) Importação para planilhas eletrônicas 31
d) Limpeza e Organização dos Dados II 32
e) Leitura das Ocorrências para classificação 32
f) Complementação de novos campos 33
g) Criação da Classe Crime ..... **Error! Bookmark not defined.**
h) Estruturação do arquivo para conversão em ARFF 35
3.2. Métricas de Avaliação 37
3.2.1 Procedimentos 39
3.2.2 Implementação do algoritmo *ADDExp.D* 42
Considerações Finais 45

CAPÍTULO 4 ..... **46**

RESULTADOS E ANÁLISE **46**

4.1. Média da Taxa de Acerto 46

4.2. ANÁLISE DO ALGORITMO 51
4.2.1 RESULTADOS ALGORITMO *ADDEXP.D* 52
CAPÍTULO 5 62
CONSIDERAÇÕES FINAIS 62
REFERÊNCIAS BIBLIOGRÁFICAS 65
ANEXOS 69

Lista de Figuras

FIGURA 1. TIPOS DE CONCEPT DRIFTS (BASEADO EM NALEPA, 2010). 9
FIGURA 2. DRIFTING DE A PARA B (BASEADO EM ENEMBRECK ET AL, 2007). 10
FIGURA 3. MODELO DE CLASSIFICAÇÃO DE CONJUNTOS. FONTE: WANG ET AL (2010) 12
FIGURA 4. MODELO DO ALGORITMO DWM (BASEADO EM NALEPA, 2010) 17
FIGURA 5. FLUXO GERAL DE ATIVIDADES DO ALGORITMO ADDEXP (BASEADO EM NALEPA, 2010). 18
FIGURA 6. PERSPECTIVA HISTÓRICA DO NYPD - PERÍODO DE 2001 A 2008[3]. 25
FIGURA 7. ETAPAS DE AQUISIÇÃO E TRATAMENTO DOS DADOS. 30
FIGURA 8. MODELO DE PLANILHA CRIADA PARA TRATAR OS DADOS. 32
FIGURA 9. EXEMPLO DE ARQUIVO NO FORMATO .ARFF 37
FIGURA 10. MODELO MÉTODO PREQUENTIAL EM JANELA DE 3 DIAS 40
FIGURA 11. MODELO MÉTODO PREQUENTIAL EM JANELA DE 7 DIAS. 40
FIGURA 12. MODELO MÉTODO PREQUENTIAL EM JANELA DE 10 DIAS. 41
FIGURA 13. PSEUDOCÓDIGO *ADDEXP* PARA CLASSES DISCRETAS. (KOLTER E MALOOF, 2005) 43
FIGURA 14. FRAGMENTO DA IMPLEMENTAÇÃO DO *ADDEXP.D*. 44
FIGURA 15. REMOÇÃO DE ESPECIALISTAS (*WEAKEST FIRST E OLDEST FIRST*). 44

Lista de Quadros e Gráficos

Quadro 1. Classificação dos índices de Ocorrências 36
Quadro 2. Organização dos dados para análise 36
Quadro 3. Modelo de Conjunto de dados preparado para criar o arquivo .arff 36
Quadro 4. Modelo de Controle para realização das análises. 42
Quadro 5. Média da Taxa de Acerto dos algoritmos. 47
Quadro 6. Média de acerto do algoritmo com janelas de diferentes tamanhos. 48
Quadro 7. Dias com chuva em Joinville. 51

Gráfico 1. Distribuição de Valores das Classes 33
Gráfico 2. Média da taxa de acerto do algoritmo ADDExp.D com classificador OzaBagAdwin. 48
Gráfico 3. Média de acerto do algoritmo com janelas de diferentes tamanhos 49
Gráfico 4. *ADDExp.D Oldest First* 50
Gráfico 5. Número de dias com chuva em Joinville. 52
Gráfico 6. ADDExp.D - Weakest First – 3 dias 52
Gráfico 7. ADDExp.D - Weakest First - 7 dias 54
Gráfico 8. ADDExp.D - Weakest First - 10 dias 54
Gráfico 9. Média de acertos - ADDExp.D - Weakest First - 3 dias 55
Gráfico 10. Média de acertos - ADDExp.D - 7 dias 55
Gráfico 11. Média de acertos - ADDExp.D - 10 dias 55
Gráfico 12. *ADDExp.D - Oldest First* - 3 dias 56
Gráfico 13. *ADDExp.D - Oldest First* - 7 dias 57
Gráfico 14. *ADDExp.D - Oldest First* - 10 dias 58
Gráfico 15. *ADDExp.D - Oldest First* - 3 dias 58
Gráfico 16. *ADDExp.D - Oldest First* - 7 dias 59
Gráfico 17. ADDExpert.D - Oldest First - 10 dias 60

Lista de Símbolos

BAGGING	Bootstrap Aggregating
BOOSTING	Meta-algoritmo de mineração de dados
K-NN	K-Nearest Neighbor – Vizinho mais próximo
AM	Aprendizagem de Máquina
IA	Inteligência Artificial
NB	Naïve Bayes
DWM	Dynamic Majority Weight
ADDExp	Additive Expert
HT	Hoeffding Tree
ASHT	Adaptive Size Hoeffding Tree
ADWIN	Adaptive Slide Windows
OZA	OzaBag ADWIN
ARFF	Attribute Relation File Format

Resumo

Este trabalho apresenta um estudo comparativo sobre técnicas de resolução de *Drift Detection,* abordando alguns algoritmos utilizados para trabalhar com conjuntos de classificadores. Os classificadores são gerados a partir de bases de dados, as quais podem possuir mudanças comportamentais, chamadas de *Concept Drift.* Esta última pode ser definida como uma área de pesquisa onde algoritmos de aprendizagem devem prever e adaptar modelos de conceitos em função de mudanças que ocorrem em um determinado conjunto de dados. Várias abordagens baseadas em conjuntos de classificadores para *Concept Drift* foram propostas na literatura, mas a carência de estudos comparativos torna a escolha de uma abordagem uma tarefa bastante difícil. Neste trabalho, implementamos algumas destas técnicas para analisar seus comportamentos quando aplicadas à bases de dados reais de ocorrências policiais de uma região do estado de Santa Catarina, buscando assim, uma nova ferramenta que possa auxiliar as autoridades a planejar ações de segurança conforme a migração da criminalidade pelos bairros da cidade.

Palavras-Chave: *Drift Detection,* Conjunto de Classificadores*, Concept Drift*, Criminalidade.

Abstract

This work presents a comparative study on techniques for solving Drift Detection, addressing some algorithms used for working with sets of classifiers. The classifiers are generated from a database, which may have behavioral changes, called Drift Concept. This can be defined as an area of research where learning algorithms must anticipate and adapt concept models based on changes that occur in a given data set. Several approaches based on a set of classifiers for Concept Drift have been proposed in literature, but the lack of comparative studies makes the choice of approach very difficult. In this work, we have implemented some of these techniques to analyze their behavior when applied to real data on police incidents in a region of the state of Santa Catarina, thus seeking a new tool that could assist the authorities and planning actions of security following the migration of crime through the neighborhoods.

Keywords: *Drift Detection, Classifiers sets, Concept Drift, Criminality*.

Capítulo 1

Introdução

A evolução da tecnologia tem trazido grandes desafios, um dos mais importantes é encontrar métodos ágeis que nos possibilitem extrair conhecimento a partir de grandes volumes de dados. Mesmo que essa situação seja desafiadora, o homem é incansável, e a todo momento busca aprimorar os métodos existentes e criar novos métodos suficientemente autônomos no aprendizado para a criação de modelos que sejam capazes de extrair este conhecimento. Técnicas de Mineração de Dados são utilizadas para criação de modelos a partir de dados históricos armazenados pelo sistema, porém, já não tem surtido um grande efeito em determinadas áreas de conhecimento uma vez que novos perfis estão sendo criados de acordo com a evolução de antigas tendências. Na perspectiva da criação de perfis de clientes, dados históricos não estão sendo suficientes pois as pessoas estão mudando seu comportamento a todo momento, ou seja, o que ontem elas consideravam interessante, hoje já não consideram mais.

Técnicas de *Concept Drift Detection* (Detecção de Mudanças de Conceitos), tem apresentado um melhor desempenho em estudos de comportamentos. Esse termo pode ser definido como uma área de pesquisa na qual, algoritmos de aprendizagem de máquina buscam alterações de comportamentos existentes em um determinado conjunto de dados. Alguns algoritmos são utilizados com diferentes estratégias, por exemplo: os baseados em janelamento (*windowing*), ponderação de instâncias, aprendizagem baseada em instâncias ou ainda, baseados em conjuntos de classificadores.

A proposta deste trabalho é apresentar um estudo que envolve algumas técnicas de *Concept Drift Detection,* no qual serão utilizados algoritmos baseados em conjuntos de classificadores para analisar um conjunto de dados reais. Além de fazer uma análise das taxas de acerto obtidas, pretendemos identificar o algoritmo mais indicado para atuar em conjuntos de dados reais relacionadas à Segurança Pública.

Sabemos que a criminalidade a nível nacional vem aumentando consideravelmente e, que os órgãos de Segurança Pública precisam encontrar ferramentas que os coloque sempre um passo à frente dos criminosos, uma destas ferramentas é a tecnolo-

gia especializada. Investimentos para criação e a constante atualização dos dados servem como instrumentos para a manutenção da ordem pública.

As técnicas propostas neste trabalho, servem como modelo para uma ferramenta tecnológica voltada não somente para facilitar a ação dos órgãos da Segurança Pública, almejando coibir, o máximo possível, o aumento da incidência da criminalidade de acordo com os 3 tipos de crimes estudados (roubos a residência, roubos contra pessoa e roubos a estabelecimentos comerciais) e, que posteriormente, possam servir aos demais tipos de crimes existentes. Mas, é adaptável a várias áreas de conhecimento tal como a saúde, educação, empresarial e outras.

1.1. Motivação

A partir de uma breve reflexão sociocultural, no que diz respeito a necessidade de uma tecnologia atualizada para os serviços de Segurança Pública *versus* o registro da criminalidade na atualidade, surgiu a motivação para a realização deste trabalho.

As técnicas de *Concept Drift Detection* são uma possibilidade de ferramenta específica para o auxílio e organização burocrática dos serviços de Segurança Pública. Algumas técnicas envolvendo Inteligência Artificial e Segurança Pública já foram desenvolvidas e serão apresentadas no decorrer deste trabalho. No entanto, não encontramos relatos de técnicas que promovem a análise dos registros iniciais das ocorrências, ou seja, não há de modo formalizado registros que facilitem a análise da frequência dos tipos de crimes que estão sendo levantados nesta pesquisa, para que se possa trabalhar em dados analíticos desde o momento da ligação para o 190 até a probabilidade de ocorrências similares.

Frequentemente vemos em jornais mapas da criminalidade de uma certa região onde crimes vem aumentando e chamam a atenção da imprensa e da sociedade. Para que respostas sejam dadas para a sociedade e ações sejam tomadas, surge a necessidade de monitoração da criminalidade em uma região.

1.2. Objetivos

Objetivo geral:

- Analisar o comportamento de técnicas de *Concept Drift Detection* em um conjunto de dados reais, coletados a partir das ocorrências de crimes que en-

volvem roubo a residência, roubo a estabelecimento comercial e roubo contra pessoa, numa determinada região da cidade de Joinville (SC).

Objetivos específicos:

- Modelar o Conjunto de Dados de crimes;
- Identificar os algoritmos pertinentes ao cumprimento do objetivo geral;
- Avaliar o comportamento do algoritmo quando submetidos a cenários reais de aplicação;
- Apontar a viabilidade da aplicação de técnicas de *Concept Drift Detection* para o desenvolvimento de ferramentas de suporte à Segurança Pública;

Com estes resultados, esperamos contribuir de forma positiva para as áreas científicas que estudam o ambiente Segurança Pública, que estudam algoritmos baseados em conjuntos de classificadores em *stream* de dados reais, organizações que investem na modernização e atualização dos serviços de Segurança Pública e, também, para outros modelos de estruturas organizacionais que buscam meios de analisar seus dados.

1.3. Proposta

A criação de meios mais ágeis e eficazes que possam auxiliar a Segurança Pública no aumento da possibilidade de prevenção dos crimes analisados, e por consequência, a melhor distribuição do policiamento. Para tanto, propomos utilizar algoritmos de classificação on-line submetidos a cenários reais de aplicação.

O intuito é que se possa abrir um caminho de novas pesquisas, tanto para área abordada neste estudo quanto para outros cenários reais de aplicação. Além disso, almejamos que os resultados aqui obtidos possam ser compartilhados entre as Policias Civil[1] e Militar[2] buscando diminuir o desencontro de informações que existe entre estas duas forças.

---

[1] **POLÍCIAS CIVIS** são os órgãos do sistema de segurança pública aos quais competem, ressalvada competência específica da União, as atividades de polícia judiciária e de apuração das infrações penais, exceto as de natureza militar.

[2] **POLÍCIAS MILITARES** são os órgãos do sistema de segurança pública aos quais competem as atividades de polícia ostensiva e preservação da ordem pública.
Fonte:
http://www.policiacivil.sc.gov.br/index.php?option=com_content&view=article&id=48&Itemid=135

Conforme exposto no item 2.8 deste trabalho, pesquisas vem sendo realizadas, em sua grande maioria, com registros da Policia Civil, e possuem objetivos e relatos diferentes com os aqui analisados, tanto em conteúdo, quanto em tipos de crimes.

Além disso, propomos que sejam utilizados somente registros de ocorrências geradas a partir de ligações telefônicas para o 190. Com os resultados obtidos, analisamos o conhecimento descoberto com os algoritmos de *Drift Detection* e avaliamos a viabilidade da aplicação de técnicas de *Concept Drift Detection* para o desenvolvimento de ferramentas de suporte à Segurança Pública.

1.4. Contribuições

As contribuições científicas do presente trabalho são: (i) comparar diferentes abordagens de *Drift Detection* baseados em conjuntos de classificadores, avaliar a viabilidade da aplicação de técnicas de *Concept Drift Detection* em cenários reais de aplicação; e (ii) contribuir para a definição de uma ferramenta de suporte à Segurança Pública, possibilitando uma melhor gestão baseada nos resultados obtidos, podendo ser positivamente utilizados para traçar novos objetivos para a organização analisada.

1.5. Estrutura do Documento

As seções subsequentes estão organizadas da seguinte forma: o capítulo 2 introduz alguns conceitos sobre o tema abordado como: Aprendizagem de Máquina, *Bagging* e *Boosting*, *Concept Drift, Drift Detection,* Algoritmos para Detecção de Mudanças baseados em Conjunto de Classificadores e *Drift Detection* em ocorrências policiais. O capítulo 3 definido como Metodologia, apresenta como o trabalho foi desenvolvido, a origem dos dados analisados e os algoritmos a serem trabalhados e as técnicas utilizadas. O capítulo 4 mostra os resultados obtidos e uma análise comparativa entre os algoritmos, destacando os que apresentaram os melhores resultados. Por fim, o capítulo 5 apresenta as nossas considerações finais sobre o trabalho realizado.

Capítulo 2

REFERENCIAL TEÓRICO

Neste capítulo, será apresentado um breve histórico sobre a Aprendizagem de Máquina (AM), conceitos de *bagging* e *boosting*, técnicas utilizadas para um melhor desempenho dos algoritmos estudados. O capítulo também aborda o problema de *Concept Drifts* (Mudanças de Conceito), e alguns algoritmos existentes para *Concept Drift Detection* (Detecção de Mudanças de Conceitos).

2.1. Aprendizagem de Máquina - AM

Considerada de vital importância para a evolução de muitas áreas dentro da Inteligência Artificial (IA), a Aprendizagem de Máquina (AM) apresenta diversos conceitos agregados. A ideia de aprendizagem, nos remete a forma individual de recebimento, processamento e assimilação de informações, mais ou menos complexas, tanto em humanos quanto em animais. A aprendizagem de máquina, por sua vez, pode implementar alguns conceitos da aprendizagem de humanos e animais, de acordo com MELLIT e KALOGIROU(2008): "A aprendizagem é uma característica inerente dos seres humanos e é através dessa capacidade que, durante a execução de tarefas semelhantes, conseguimos melhorar nosso desempenho." Podemos então, considerar que a aprendizagem do ser humano baseia-se em experiências, individuais ou coletivas, vividas anteriormente. Ao compartilhar uma experiência, o indivíduo acessa informações recentes e remotas do seu complexo sistema de memória, essas informações vão atingir o outro de acordo com suas próprias experiências pessoais, provocando um comportamento essencialmente particular, haja vista que a experiências assimilada é uma condição individual.

Numa sociedade, essencialmente moralista, o par de opostos certo e errado, bom e ruim, vai sendo transmitido de geração em geração. No entanto, não podemos perder de vista que toda sociedade é fundamentada por culturas distintas, desse modo toda família é um núcleo de transmissão cultural. É a partir da estrutura do ambiente familiar que o indivíduo recebe, processa e assimila os dados objetivos da vida, construindo internamente sua própria subjetividade numa verdadeira rede de pensamentos, sentimen-

tos, emoções e sensações, que darão origem a sua atitude perante as vicissitudes da vida em comunidade.

Num recorte mais definido, de acordo com a influência do homem para a evolução tecnológica, é possível detectar no comportamento humano a ampliação de sua capacidade racional na **inter-ação** (grifo meu) entre homem e máquina, pois o aprendizado é mútuo. Uma vez que atua sobre a máquina e os modelos são gerados a partir da aprendizagem dela, consequentemente, estes modelos podem ser submetidos à verificação de um especialista, para garantir a confiabilidade do resultado. Confirmamos, por isso, uma relação dialética.

Voltamos nossa atenção para os algoritmos computacionais que implementam técnicas de aprendizado de máquina com o objetivo de encontrar e descrever padrões de comportamentos a partir dos dados obtidos no ambiente analisado. O aprendizado pode ser realizado de diferentes formas de acordo com um paradigma, por meio um elemento simbólico, estatístico, baseado em instâncias, conexionista e genético. O aprendizado através desses paradigmas consiste na escolha ou na adaptação dos parâmetros de representação do modelo. Mas, independente do paradigma, a principal tarefa dos algoritmos de aprendizado é aprender um modelo a partir do ambiente e, manter esse modelo consistente de modo a atingir os objetivos de sua aplicação. Explicar os dados e fazer predições são objetivos comuns da aplicação desses algoritmos (SENGER, 2004).

"AM é uma área da Inteligência Artificial, cujo objetivo é o desenvolvimento de técnicas computacionais sobre processo de aprendizado" (BISHOP, 2006). As classificações propostas na literatura para a AM são: a **Aprendizagem de Máquina Supervisionada; Aprendizagem de Máquina Não-Supervisionada; Aprendizagem de Máquina por Reforço; Aprendizagem de Máquina baseada em Instâncias** também conhecida como Aprendizagem Preguiçosa e a **Aprendizagem Bayesiana.** O presente trabalho trata apenas da aprendizagem supervisionada, mais especificamente sobre o problema de classificação, onde os exemplos de treinamento são rotulados previamente por um supervisor. Nesse caso, o problema consiste em atribuir rótulos, ou classes, a novos exemplos não utilizados no treinamento, ou seja, exemplos de teste. O leitor interessado em se aprofundar em outros modelos de aprendizagem pode utilizar a referência de Mitchell (1997).

2.2. Concept Drift

A atividade humana é basicamente estruturada numa rotina mais ou menos rígida, as normas e regras pré-estabelecidas servindo como referência para a organização habitual de cada um. Quando essas regras sofrem uma alteração, um *Concept Drift* é identificado, significando que ocorreu uma mudança em um ou mais comportamentos que estavam estabelecidos há bastante tempo.

Segundo BIFET et al (2011), "*Concept Drifts* descrevem uma mudança gradual no conceito." Estas mudanças ocorrem frequentemente no mundo real e quando um classificador para um conceito estático é aprendido, ele não pode ser usado para classificar futuras instâncias indefinidamente, pois este conceito pode mudar e o classificador aprendido pode não ser mais útil para o novo problema de classificação que se apresenta. KLINKENBERG (2004) considera uma mudança na distribuição dos dados, se tiver sofrido alterações após a fase de treinamento, existindo a possibilidade de que novos conceitos tenham surgido e que tenham sido alteradas as características dos dados antigos.

De acordo com ENEMBRECK et al (2007), em domínios onde o ambiente sofre constantes mudanças, ou dados são continuamente produzidos, técnicas de aprendizado dinâmico devem ser usadas, uma vez que o conceito alvo do algoritmo pode mudar no decorrer do tempo. Os autores ainda relatam que estas mudanças podem ser abruptas ou ocorrer lentamente. Sem a implementação de técnicas de *Drift Detection*, a dificuldade para extrair um conhecimento válido desse conjunto de dados, ou então de conseguirmos tomar decisões válidas utilizando aprendizagem de máquina supervisionada seria muito maior. Afinal, estaríamos trabalhando com conjuntos antigos que não refletiriam o padrão atual do ambiente analisado.

ENEMBRECK et al (2007) ainda fazem algumas considerações de *Concept Drift*, apresentando que as técnicas devem incluir três metas principais definidas como:

- **Rápida detecção de *Concept Drift* –** pode ser visto como um dos pontos mais positivos dentre as metas, pois essa rapidez na detecção pode ser muita vantajosa, minimizando o impacto negativo causado no sistema.
- **Distinção entre um *Concept Drift* e um ruído –** busca verificar e diferenciar o que é uma mudança, do que é um ruído no conjunto de dados analisado. Um ruído pode ser caracterizado como dados incompletos ou incorretos, que

foram inseridos no sistema, e não como uma ação que irá afetar todo o processo de aprendizado até então realizado.

- **Reconhecer e trabalhar com os contextos recorrentes –** estes contextos podem ser caracterizados como variações de estados entre válidos e inválidos com um grau de regularidade, como: os ciclos das estações do ano, ou datas específicas como Dia dos Pais, Dia das Mães e Natal. Modelos aprendidos nestes Períodos podem ser armazenados e reutilizados em situações futuras.

Em um sistema de aprendizagem a carga de dados pode ser feita geralmente de duas formas, a primeira em *batch* (lote ou *off-line*), na qual os volumes de dados são fornecidos ao sistema que por sua vez os analisa, comparando-os com os dados antigos, identificando se houve ou não um *Drift* nesse volume fornecido. A segunda forma é *online*, aqui o sistema busca os dados nas bases de dados em períodos pré-configurados pelos administradores, ou em *real time*, onde a detecção das anomalias pode ser mais eficiente, pois o sistema está focado diretamente na entrada dos dados como uma espécie de guardião, tentando identificar algo suspeito que possa causar mudanças em seu aprendizado.

DELANY e CUNNINGHAM (2004), afirmam que a maior dificuldade com o aprendizado em domínios do mundo real é que o conceito de interesse pode depender de algum contexto escondido, não representado explicitamente na forma de modelos preditivos. Exemplos clássicos desse problema são as previsões do tempo e a classificação de tipos de consumidores. Estes dois modelos podem variar devido a inúmeras ações ocorridas no ambiente externo como: mudanças rápidas de clima para as previsões do tempo, taxas de inflação e período do mês para os consumidores. Ainda relatam que existem 3 tipos de *Concept Drift* (ver **Figura 1**), que ocorrem com frequência nestes problemas do mundo real:

- **Abrupto (Instantâneo),** identificado logo após ocorrer, onde um conceito *A* é instantaneamente substituído por um conceito *B*;
- **Gradual,** que mistura os conceitos *A* e *B,* fazendo que ambos fiquem ativos durante um determinado período, mas, com o passar do tempo, o número de ocorrências do conceito *A* reduz à medida que o conceito *B* aumenta;
- **Moderado (Gradual),** também identificado por STANLEY (2003) como moderado e lento, onde as mudanças são identificadas aos poucos. Neste tipo

de *Drift*, as diferenças acontecem aos poucos sendo notadas apenas em longos períodos de tempo.

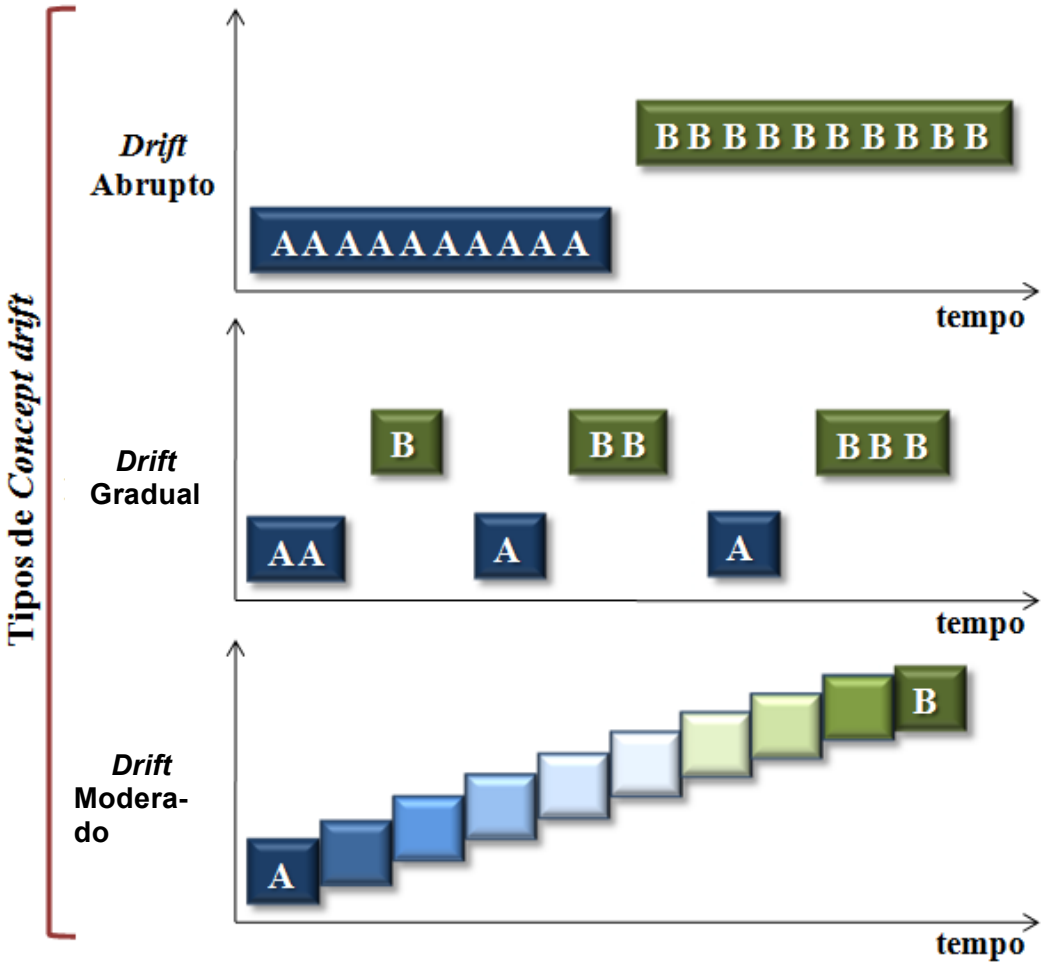

Figura 1. Tipos de Concept Drifts (Baseado em NALEPA, 2010).

Na visão de HELMBOLD E LONG (1994), a taxa de *Drift Moderado* também pode ser associada com o aprendizado do *Concept Drift* em problemas de fácil aprendizagem, por possuir uma quantidade maior de dados rotulados disponíveis, para aprender a mesma função objetivo antes de sofrer as modificações. Mudanças ocultas no contexto, podem não somente ser a causa de uma mudança no conceito alvo, mas podem causar uma mudança sobre a distribuição dos dados. WIDMER e KUBAT (1993) relatam que toda vez que a distribuição de dados dos conceitos alvos sofrem alterações, pode ser necessária a reconstrução de um novo modelo, pois estes erros podem tornar os modelos não aceitáveis. Para ele, existem dois tipos de *Concept Drift,* o primeiro denominado *Virtual Concept Drift*, que não ocorre na realidade, porém, reflete em modelos computacionais e o segundo denominado *Real Concept Drift* que reflete em mudanças do mundo real como: a moda e a música.

2.3. Concept Drift Detection

Drift Detection é definido como a tarefa de detectar mudanças ocorridas em um conjunto de dados corrente, ou seja, uma mudança em um modelo que até então encontrava-se parametrizado. STANLEY (2003) procurou ilustrar este conceito, tomando A e B como um par de conceitos, e um conjunto de instâncias i_1 até i_n. Até a instância i_x o conceito corrente A é estável, após um certo número Δ_x de observações entre i_x e $i_{x+}\Delta_x$, o conceito B passa a ser o conceito corrente. A mudança do conceito A para o conceito B é feita suavemente entre as instâncias i_x e $i_{x+}\Delta_x$ conforme apresenta a Figura 2.

Figura 2. Drifting de A para B (Baseado em ENEMBRECK et al, 2007).

Quando $\Delta_x = 1$, dizemos que ocorreu uma mudança abrupta entre A e B, por outro lado, quando $\Delta_x > 1$ a modificação ocorre gradualmente no espaço denominado Zona de Mudança, que por sua vez, pode ser modelada como uma função de probabilidade, na qual define a dominância do conceito de A sobre B. Então, $p(A) = \alpha$ e $p(B) = 1 - \alpha$. Segundo ENEMBRECK et al (2007), com o intuito de medir a probabilidade do conceito A em um $i_c \in \{i_x, i_{x+1}, \ldots, i_{x+\Delta x}\}$ seria necessário usar $\alpha = (c - x) / \Delta_x$. Esta equação estabelece que a probabilidade de ocorrências de um conceito A é reduzida linearmente, enquanto a probabilidade do conceito B aumenta. *Drift Detection* é um caminho para responder ao *Concept Drift* (NISHIDA e YAMAUCHI, 2007).

Exemplos de problemas reais, onde a detecção de mudanças é relevante incluem modelagem de usuário, monitoração na biomedicina, processos industriais e falha na detecção de diagnósticos. Eles ainda afirmam que alguns métodos de detecção para monitorar os erros de classificação em um classificador *online* durante a aprendizagem foram propostos recentemente por GAMA et al (2004); BAENA-GARCIA et al (2006) e NISHIDA et al (2005), onde relatam ainda, que os métodos não dependiam do tipo de atributo de entrada, mas que eram capazes de detectar *Concept Drift* em um número pequeno de exemplos, além de ter um baixo custo computacional.

NISHIDA e YAMAUCHI (2007) propuseram um método de detecção de mudanças que usa um teste estatístico de proporções igualitárias (*STEPD – Statistical Test*

of Equal Proportions), buscando rapidamente e corretamente detectar vários tipos de *drifts,* no qual demonstraram seu desempenho e precisão utilizando cinco conjuntos de dados que continham *Concept Drift*.

Com a intenção de tornar o processo de *Drift Detection* mais ágil e correto, vários tipos de algoritmos estão sendo criados e profundamente analisados, para um melhor aproveitamento no aprendizado em grandes volumes de dados. A seção a seguir apresentará alguns algoritmos classificadores utilizados em diferentes trabalhos, buscando identificar, aprender e resolver *Concept Drifts*.

2.4. Algoritmos para *Concept Drift Detection*

Muitos esforços foram realizados para minimizar as incertezas com relação à confiabilidade dos resultados apresentados pelos métodos utilizados em *Data Mining* e *Machine Learning*. Estas incertezas se referem ao fato de que os treinamentos são realizados sobre um volume de dados que pode ser menos relevante do que os dados reais. WIDMER e KUBAT (1996) apresentam como exemplo a previsão do tempo, onde as regras podem variar de acordo com a estação.

Já TSYMBAL (2004), relata que, "a maioria dos algoritmos tradicionais de *Machine Learning* está suscetível ao *Concept Drift,* pois a precisão da previsão reduz com o passar do tempo." Algoritmos com estas características são chamados de algoritmos de *batch* – bons no aprendizado a partir de bases de dados armazenadas em *batch*, mas ineficientes quando estes dados crescem dinamicamente, ficando expostos ao *Concept Drift.*

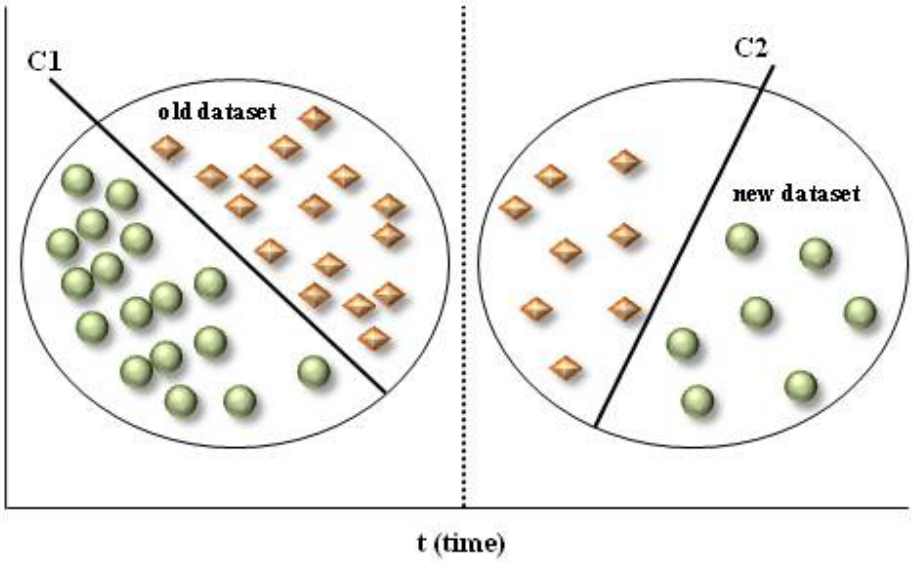

Figura 3. Modelo de Classificação de Conjuntos. Fonte: Wang et al (2010)

Uma visão de WANG et al (2010) sobre os problemas do *Concept Drift* para classificação é apresentada na Figura 3, onde são dados dois conjuntos de dados C1 (Conjunto Antigo) e C2 (Conjunto Novo). Podemos identificar as mudanças na classificação no primeiro ciclo (C1) que classificou corretamente os *data points* e no segundo ciclo (C2), quando aplicado C1, alguns *data points* são reclassificados de outra forma. Para evitar este problema, os referidos autores sugerem que se faça uma combinação dos dois conjuntos de dados (C1 e C2), o que é catastrófico segundo POLIKAR e ELWELL (2009), pois afirmam que existirá um novo conhecimento e que o conhecimento antigo será esquecido, desse modo não será possível fazer uma previsão precisa. Conforme esses autores, *Concept Drift* requer uma solução de AM que não possua apenas uma fase inicial de treinamento, mas muitas fases e, que a partir delas possa existir um treinamento contínuo, também denominado aprendizado incremental.

2.4.1 Algoritmo k-NN (*K-Nearest Neighbor*)

O **algoritmo k-NN** é um exemplo de aprendizado baseado em instância. Este método é definido por NISHIDA (2008) como o mais básico da aprendizagem de máquina, pois ele não tenta generalizar a partir dos dados de treinamento para elaborar uma hipótese que combine com todos os dados de entrada, mas, em vez disso, armazena os dados de treinamento e usa estes dados para determinar uma classificação para cada novo fragmento de dados que for encontrado. Por sua vez, COPPIN (2010), considera

que o k-NN opera em situações na qual cada instância pode ser definida por um vetor de *n* dimensões, onde *n* é o número de atributos usados para descrever cada instância e as classificações sejam valores numéricos discretos. Ele obtém as classificações dos *k* vizinhos mais próximos da instância a ser classificada e atribui a ela a classificação mais comumente retornada por aqueles vizinhos. Para atributos numéricos, a distância é geralmente definida em termos de distância euclidiana padrão.

É possível permitir que cada instância de dados de treinamento contribua para a classificação de uma nova instância (COPPIN, 2010). Ao contrário do aprendizado baseado em árvores de decisão, o algoritmo k-NN apresenta um bom desempenho com dados de entrada com ruído, uma vez que está predisposto a verificar as distâncias entre as instâncias, buscando similaridades baseadas na distância Euclidiana.

Os algoritmos baseados em instâncias como por exemplo o k-NN, por demandarem poucos parâmetros de ajustes, são considerados de simples implementação. Por outro lado, quando aplicados em grandes quantidades de dados, necessitam de mecanismos mais eficientes para encontrar os vizinhos mais próximos, haja visto que o cálculo da distância entre os pontos exige alto tempo de processamento. Apesar disso, as estratégias de gerenciamento de memória, algoritmos baseados em instância são utilizados com frequência nos problemas de aprendizagem *on-line* pelo fato de não necessitar de treinamento.

2.4.2 Algoritmo IB3 (Baseado em Instâncias)

Algoritmos baseados em instâncias são derivados dos modelos de classificadores do vizinho mais próximo sendo bastante similares entre eles, pois também salvam e usam somente as instâncias selecionadas para gerar os classificadores de predição (AHA et al, 1991). Este algoritmo é mais aplicável em fluxos de dados e aprendizagem *online* no qual os conjuntos de treinamento não existem como uma coleção de casos antes que a edição possa ser realizada.

PEREIRA et al (2010) afirma que a essência geral do método baseado em instâncias consiste na representação de instâncias de treinamento como pontos *n*-dimensionais. Tais pontos são definidos pelas *n* características que descrevem estas instâncias. Considera-se então que, esta técnica baseada em instâncias elabora sua hipótese com base nas próprias instancias de treinamento.

O algoritmo IB3 é considerado uma extensão do IB2 que emprega métodos de coleta de provas do tipo "esperar e ver" para então determinar quais instâncias salvas apresentarão um bom desempenho durante a classificação. AHA et al (1991), relata que, no algoritmo IB2, para que um novo exemplo não seja descartado, ele deve possuir sua classificação incorreta e então é adicionado na descrição do conceito.

Apesar do IB2 e do IB3 possuírem funções bastante semelhantes, algumas características adicionais são encontradas no IB3 tais como:

- Manter um registro de classificação (número de tentativas de classificação correta e incorreta). Este registro busca resumir o desempenho de uma instância de classificação sobre as instâncias de formação, apresentando e sugerindo como será seu desempenho no futuro;
- Empregar um teste de significância para determinar quais instâncias serão bons indicadores de classificação. Posteriormente são utilizados para classificações subsequentes e para mostrar quais são os ruídos que serão descartados;
- Atualizar o registro de classificação de todos os casos que poderiam ser vizinhos Embora não colete informações sobre os prováveis danos que certos casos podem causar, esta classificação é identificada como possível ou potencial;
- Utilizar a exatidão da classificação ao invés de erro de classificação. Desse modo é possível indicar casos bem sucedidos e executáveis, assim espera-se que mesmo num caso não classificado corretamente, seja atingido um nível satisfatório antes de removê-lo;
- Normalizar a aceitação de uma instância em relação ao conceito de distribuições de frequência, com o intuito de diminuir sua sensibilidade à distribuição heterogênea;
- Possuir intervalos iniciais de confiança relativamente grandes, sendo encolhidos em sua largura com treinos e tentativas de classificação;
- Utilizar um filtro seletivo que funcione em domínios com grande variedade de ruídos, promovendo uma exatidão relativamente alta de classificação.

NALEPA et al (2010) fazem uma comparação entre os algoritmos da família IB afirmando que, o IB1 e o IB2 selecionam as instâncias mais similares, com a diferença

de que o IB2 foca na redução da necessidade de armazenar e, o IB3 busca selecionar a instância mais similar aceitável do conjunto. Ele mantém a classificação gravada com cada instância, executando um teste para determinar se uma dada instância pode ser relevante em classificações futuras ou se é somente um ruído.

2.5. Conjuntos de Classificadores

Considerando o princípio da criação de um conjunto de classificadores a partir dos dados de treinamento, a ideia central desta abordagem é que a predição da classe desconhecida de uma instância seja realizada combinando as predições feitas pelos classificadores deste conjunto. Gomes (2012, apud Polikar (2006)), afirma que, para a construção de um algoritmo baseado em conjuntos de classificadores, dois componentes devem ser considerados:

- a **estratégia** de construção de conjunto de forma que ele seja o mais diverso possível;
- a **combinação** da saída dos classificadores onde as decisões corretas sejam segregadas e as incorretas descartadas.

Além disso, afirma que na configuração *online*, é importante que seja considerado algum método de adaptação do conjunto tendo em vista que *Concept Drifts* são esperados.

De um modo geral, o principal objetivo em conjuntos de classificadores é que cada classificador possua características únicas e exclusivas diferenciando-se assim dos demais classificadores. Contudo, Ohno (2011) afirma que estes métodos foram desenvolvidos para buscar um melhor desempenho do que classificadores individuais, cuja ideia seja de construir um conjunto de classificadores a partir dos dados de treinamento e, em geral, não há garantias de que esses classificadores sejam diferentes ou complementares. Essa "construção" se dá pela execução repetida do algoritmo de aprendizagem, combinando prováveis resultados em um resultado conjunto.

2.6. Algoritmos de Detecção de Mudanças baseados em Conjuntos de Classificadores

A capacidade de detectar e responder a um comportamento incorreto ou mudanças de condições com pouca ou nenhuma interferência humana é um dos aspectos importantes em um sistema autônomo. Para MARTINS (2003), isso significa que no para-

digma de AM a aquisição de conhecimento em algoritmos classificadores provém de processos indutivos, onde um classificador para uma categoria C_i é construído sob o "aprendizado" de características herdadas de um conjunto de dados previamente rotulado como C_i. O objetivo é acelerar ou melhorar os resultados dos processos de classificação, para isso, técnicas que manipulam conjuntos de classificadores tem sido desenvolvidas como por exemplo os métodos *Bagging* e *Boosting*. Nesta seção, serão apresentados alguns algoritmos de *Drift Detection* baseados em conjuntos de classificadores.

2.6.1 Algoritmo DWM (*Dynamic Weighted Majority*)

DWM consiste em um algoritmo de aprendizagem em tempo real, que busca detectar *Concept Drifts*. Seu critério de classificação é realizado através de uma base de dados de aprendizado formada por exemplos mantidos pelo algoritmo, posteriormente repassados para o aprendizado dos especialistas. Os especialistas possuem conhecimento do ambiente sobre o qual estão operando, sendo criados e destruídos dinamicamente em resposta às suas mudanças de desempenho. Estes especialistas agem através de um processo de votação, onde a maioria é escolhida através das opiniões expressadas, formando assim um modelo de predição do algoritmo (PEREIRA et al, 2010).

Caso um especialista classifique erroneamente uma instância em relação à predição majoritária, o algoritmo diminui seu peso por uma constante de multiplicação β, ou caso ache necessário, criará um novo especialista com peso 1. Por outro lado, remove especialistas com pesos inferiores ao limiar definido θ, onde o parâmetro ρ rege a frequência com que o *DWM* cria, remove e atualiza seus especialistas (ENEMBRECK et al, 2009). O algoritmo normaliza o peso dos especialistas de tal forma que o maior peso será igual a 1, tentando impedir que qualquer especialista recém criado domine a tomada de decisão conforme apresenta a Figura 4.

Quando inicia seu processo, o *DWM* cria um grupo contendo apenas um especialista e a sua predição será considerada a predição global. Quando este grupo contém diversos especialistas, ele mantém a classificação local e o peso de cada um, utilizando-os para definir a predição global.

1: $\boldsymbol{Parameters}$: $\beta \in [0,1]$: $fator\ de\ pesos\ decrescentes$,
p: $periodo\ entre\ a\ remoçao, criaçao\ e\ atualizaçao\ de\ peso\ do\ classificador$
2: $\boldsymbol{Initialize}$ $numero\ de\ especialistas\ \boldsymbol{J} = 1, peso\ do\ especialista\ w_1 = 1.$
3: $\boldsymbol{while}$ $mais\ pontos\ de\ dados\ (x_t, y_t) estiverem\ disponíveis\ \boldsymbol{do}$ // $x_t \in X, y_t \in Y$
4: $obtenha\ os\ classificadores\ de\ saída\ \{\boldsymbol{H_j}\ (\boldsymbol{x_t})\}_{j=1}^{\boldsymbol{J}} \in Y.$
5: $\boldsymbol{Output}\ \boldsymbol{H}_f(x_t) = \arg max_{y \in Y} \sum_{j=1}^{J} w_j[[H_j(x_t) = y]]$
6: $\boldsymbol{if}\ n\ mod\ p = 0\ \boldsymbol{then}$
7: $atualize\ \forall w_j \leftarrow w_j \beta^{[[H_j(x_t) \neq y_t]]}.$
8: $normalize\ \forall w_j = w_j\ /\ max_i[w_i]$
9: $\boldsymbol{if}\ H_f(x_t) \neq y_t\ \boldsymbol{then}$
10: $adicione\ um\ novo\ classificador$: $w_{j+1} = 1, J \leftarrow$ J + 1
11: $\boldsymbol{end\ if}$
12: $\boldsymbol{end\ if}$
13: $treine\ todos\ os\ classificadores\ \forall H_j : X \rightarrow Y\ com\ (x_t, y_t).$
14: $\boldsymbol{end\ while}$

Figura 4. Modelo do algoritmo DWM (Baseado em NALEPA, 2010)

Conforme estudos realizados por NALEPA (2010), este é um método genérico que permite o uso de qualquer algoritmo de aprendizagem por um especialista e que utiliza os seguintes mecanismos para lidar com *Concept Drift:*

a) realiza o treinamento dos especialistas do grupo de forma *online;*

b) atribui pesos aos especialistas de acordo com o seu desempenho;

c) remove especialistas do grupo de acordo com o seu desempenho;

d) cria especialistas de acordo com o desempenho do grupo.

"*DWM* é um algoritmo de ponderação que combina as decisões de especialistas, retornando a maioria dos votos sobre uma previsão." (KOLTER e MALOOF, 2003).

2.6.2 Algoritmo ADD Expert (*Additive Expert*)

Apresentado por KOLTER e MALOOF (2005), esse é um método genérico capaz de utilizar qualquer especialista de forma *online* para a detecção de *Concept Drifts*. Comparando-o com o algoritmo *DWM*, vemos que eles possuem muitas similaridades tendo em vista que também mantém um conjunto de especialistas com pesos associados. Da mesma forma, o especialista que realizar classificação incorreta tem seu peso reduzido e se a predição global classificar incorretamente, um novo especialista é criado e adicionado ao grupo. Finalizando, cada instância também é enviada para o treinamento dos especialistas.

KOLTER e MALOOF (2005) ainda propõem duas versões para o algoritmo *DWM* que são definidas como **ADDExp.D** utilizado para classes discretas e o **ADDExp.C** para as classes contínuas. Ambos apresentam o mesmo funcionamento geral, diferenciando-se no quesito predição de classes, onde a versão "D" trabalha com um grupo discreto de classes, enquanto a versão "C" trabalha com classes contínuas [0,1], não executando nenhuma atividade para remover especialistas ruins (ver Figura 5).

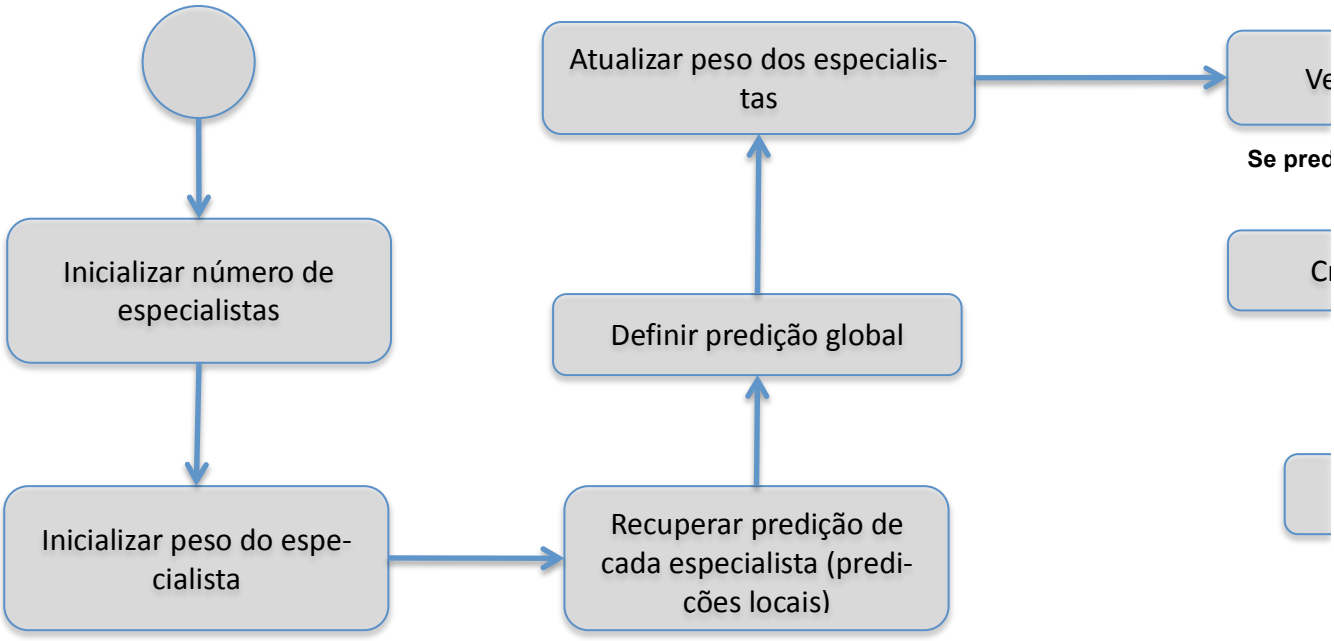

Figura 5. Fluxo geral de atividades do algoritmo ADDExp (Baseado em NALEPA, 2010).

Junto com estas duas versões, os autores também propuseram duas técnicas para remover especialistas ruins, com o intuito de limitar o número de especialistas existentes no conjunto. A primeira técnica é chamada ***Oldest First***, significando que quando

um novo especialista é adicionado ao conjunto, e se o número de especialistas for maior que uma constante *K* definida, o especialista mais antigo é então removido antes que um novo membro seja adicionado ao grupo. A segunda técnica proposta por KOLTER e MALOOF (2005), denominada ***Weakest First*** trabalha da seguinte forma: quando um novo especialista é adicionado ao conjunto, se o número de especialistas for maior que uma constante *K* definida, opta-se pela exclusão do especialista com menor peso antes de adicionar um novo membro ao grupo.

2.7. Meta-Classificadores

Meta-Classificadores, ou algoritmos de nível um, combinam as saídas geradas dos classificadores base em uma predição final. Vários algoritmos de Aprendizagem de Máquina são utilizados para desempenhar o papel de meta-classificador, permitindo ainda, utilizar diferentes configurações para os algoritmos base (PARADELA, 2007). Nesta seção, serão apresentados alguns algoritmos meta-classificadores utilizados na análise de um conjunto de dados reais. Alguns desses algoritmos também foram adaptados para aprendizagem *on-line* e para detecção de mudanças de conceito.

2.7.1 *Bagging*

Determinados problemas não são suficientemente descritos através dos conjuntos de treinamento disponíveis e seus dados representam uma visão parcial do problema todo. DE PAULA et al (2007) esclarece que muitas vezes um algoritmo de aprendizado não será capaz de gerar um classificador bom porque é baseado em um modelo local. Os métodos *Bagging e Boosting* são considerados os mais populares quando falamos de aprendizagem por conjuntos. Estes métodos criam diferentes classificadores, a partir desse ponto podem ser combinados em um classificador composto eleito através do voto majoritário, ou seja, a obtenção de uma classificação global é resultado da votação dos classificadores.

Bagging pretende resolver este problema de modelos locais, através de um processo de seleção de amostras, que seja igual em tamanho de subconjuntos de treinamento, gerando assim, classificadores para subconjuntos compostos de casos selecionados de forma aleatória, mas com substituição, baseados em um algoritmo de aprendizagem. Pode ocorrer então que uma determinada instância apareça ou não repetidamente em

todos ou em alguns subconjuntos de treinamento, sendo que a classificação de uma instância teste é dada por uma estratégia de votação.

BREYMAN (1996) afirma que o *bagging* é uma maneira relativamente fácil de melhorar um método existente, pois é necessário apenas adicionar um laço inicial que faz a seleção dos modelos de *bootstrap* e os envia para um procedimento em *back-end* que faz a agregação, ganhando desta forma, maior precisão. Na opinião de BIFFET et al (2011) "*bagging* funciona melhor com algoritmos de aprendizagem instáveis como por exemplo, árvores de decisão e redes neurais, pela diversidade do conjunto." Nesses casos, ele produz pequenas alterações no conjunto de treinamento gerando classificadores muito diferentes e que podem ser complementares entre si, mesmo para as entradas semelhantes. Pelo fato de que os casos são selecionados aleatoriamente, este método não consegue garantir modelos totalmente complementares. Dessa forma, o *bagging* não se torna tão eficiente quando trabalha com algoritmos estáveis, como por exemplo o *k-NN*.

2.7.2 *Boosting*

Outro procedimento comumente utilizado para explorar o problema da falta de garantia da criação de modelos complementares é o *boosting*. Essa abordagem utiliza um algoritmo chamado *Hill-climbing-like* que faz com que os modelos sejam tão complementares quanto possível (DE PAULA et al, 2007). Quando há orientação para a criação dos modelos, os exemplos são criados com um peso inicial. Estes pesos são utilizados para estimativa de erros de classificação, calculados pela soma dos pesos das instâncias classificadas incorretamente e dividida pela soma dos pesos de todas as instâncias. A estratégia de ponderação do algoritmo foca uma maior atenção aos casos classificados incorretamente com pesos muito altos.

Este processo consiste nos seguintes passos:

1 - um peso uniforme é atribuído a todas as instâncias de formação;

2 - o algoritmo de aprendizagem gera o classificador e os pesos em todas as instâncias são atualizados. Desse modo, os pesos das instâncias classificadas incorretamente são aumentados e os pesos dos casos bem classificados são diminuídos;

3 - um erro global da classificação é calculado e armazenado;

4 - o processo repete de forma interativa até que um pequeno erro seja gerado.

DE PAULA et al (2007) afirmam que este procedimento gera os classificadores e atribui os pesos para as instâncias de treinamento. Estes pesos representam a frequência das instâncias que foram classificadas incorretamente pelos classificadores. Para classificar uma nova instância, a decisão de um classificador é tomada através da soma dos pesos para cada classe, retornando a classe com maior peso.

Pesquisas de FREUND e SCHAPIRE (1996), relatam que a proposta do método *boosting* é melhorar a precisão dos algoritmos de aprendizagem, reduzindo a taxa de erro por meio de uma técnica que busca combinar classificadores, explorando modelos que são complementares.

2.7.3 *Leveraging Bagging*

É uma forma de melhorar o aproveitamento do método *bagging*, com duas melhorias de randomização definidas. A primeira é o aumento de re-amostragem com reposição utilizando a distribuição de Poisson, para modelar o número de eventos que ocorreram dado num determinado intervalo de tempo. A segunda melhoria é a utilização de códigos de detecção de erros de saída que utilizam apenas um classificador binário, no qual as classes atribuídas a cada exemplo são modificadas para criar um novo classificador induzido por um mapeamento a partir do conjunto de classes (BIFET et al, 2011). Essa técnica tem sido utilizada com sucesso em problemas de detecção de mudanças e classificação *on-line*.

2.7.4 Algoritmo *ASHT (Adaptive-Size Hoeffding Tree)*

Este algoritmo é derivado do algoritmo *Hoeffding Tree*. O *ASHT* é um dos algoritmos utilizados com o método *Bagging* com as seguintes diferenças:

- possui um número máximo de nós de divisão ou tamanho;
- após uma divisão dos nós, se o número de nós da árvore ASHT é maior que o seu volume máximo, sua tendência é reduzir seu tamanho, deletando alguns nós.

Segundo BIFET et al (2011), a intenção através desse método é de que árvores pequenas rapidamente se adaptem melhor a mudanças e árvores maiores são melhores durante períodos com pouca ou nenhuma mudança apresentada, simplesmente porque

foram construídas com mais dados. Desta forma, podemos dizer que como estes modelos não apresentam variações em seu conceito existe um maior conhecimento agregado a esta árvore pelo volume de dados que recebeu.

Existem duas diferentes opções de remoção caso o tamanho da árvore ultrapasse seu tamanho limite. A primeira opção diz respeito à remoção do nó mais antigo, a raiz e todos os seus filhos, com exceção de onde foi realizada a divisão do nó que, por sua vez, passa a ser a nova raiz. A segunda opção é apagar todos os nós da árvore, ou seja, criar uma nova árvore com base em uma nova raiz. Podemos ainda dizer que este algoritmo procura melhorar a performance do método *bagging* para *streams* de dados que apresentam *Concept Drifts*.

2.7.5 Algoritmo ADWIN

O algoritmo ADWIN é adequado para problemas cujas instâncias possuam atributos contínuos, "este algoritmo mantém uma janela de tamanho variável com as últimas instâncias na forma de um histograma" (GOMES, 2012). A mudança de conceito nesse caso é identificada de acordo com a variação do tamanho da janela, na qual se descarta um fragmento da janela caso não existam evidências de que o valor médio deste seja diferente do valor médio do restante da janela. Nesse caso, GOMES (2012) afirma que existe uma indicação de que houve uma mudança de conceito caso a janela apresente diminuição de seu tamanho.

2.8. *Drift Detection* e Ocorrências Policiais

Existem poucos trabalhos que relatam a utilização de ferramentas de gestão do conhecimento que atuem sobre bases de dados da Segurança Pública, mais especificamente, com dados gerados pela Polícia Militar. Um dos motivos, é a falta de profissionais com um saber específico nestas ferramentas. Outro motivo, que é viável considerar, é o baixo número de pesquisas que envolvem especificamente dados oriundos da base de dados da Polícia Militar. Pelo conteúdo existente nestes dados, o acesso a eles muitas vezes são disponibilizados apenas para integrantes de determinadas funções da corporação militar.

Grande parte dos estudos encontrados se limitam aos dados gerados pela Polícia Civil, de várias regiões do Brasil e convém ressaltar que estes dados não representam a realidade dos dados mantidos pelas Polícias Militares destas regiões. A explicação para este fato é a inexistência de um sistema integrado que mantenha estes dados reunidos em um só local. Em determinadas ocorrências, uma pessoa liga para o 190 gerando uma ocorrência mas não faz contato com a Polícia Civil para fazer o registro e vice-versa. Quando isto ocorre, ou Polícia Militar ou Polícia Civil ficam sem as informações.

MIRANDA et al (2008) em parceria com o Instituto de Segurança Pública, reuniram em um livro intitulado: A Análise Criminal e o Planejamento Operacional. Nesta obra, é possível encontrar uma série de análises realizadas sobre dados inerentes à Segurança Pública. Nestas análises, a ênfase foi dada sobre formas de trabalho das Instituições Policiais Militares, sobre a importância da coleta de dados e sobre georeferenciamento de ocorrências.

GONÇALVES (2002) trabalhou com a Geocodificação e Análise do mapeamento da criminalidade na cidade de Ipatinga – SP. Machado (2008) apresentou a importância do uso da informação na gestão inteligente da Segurança Pública. Este trabalho focou os sistemas existentes na Segurança Pública e a integração e utilização por diversas instituições policiais militares. Esses estudos nos ajudam a pensar sobre os dados registrados pelas Polícias e a necessidade de uma ferramenta que auxilie na resolução dos crimes.

Na França, SIKLÓSSY e AYEL (1997) realizaram um trabalho com dados de ocorrências registradas pela Polícia Internacional – INTERPOL, foi utilizada a Gestão do Conhecimento para mapear e estruturar toda uma organização criminal, por exemplo, uma quadrilha que rouba carros. O analista de ocorrências da INTERPOL, foca seu trabalho no mapeamento de todos os integrantes da quadrilha, *modus operandi*, recursos que eles disponibilizam, possíveis locais para onde os produtos de roubo são encaminhados, enfim, toda a estrutura organizacional do crime. Este grupo trabalha com ocorrências como: narcotráfico, roubo de peças de arte e roubos de carros de luxo.

POELMANS et al (2010), apresenta num trabalho realizado, a utilização de regras de classificação e *FCA – Formal Concept Analysis* (Análise Formal de Conceito) com dados de crimes envolvendo violência doméstica em Amsterdã – Holanda. Outro trabalho encontrado foi desenvolvido por CASTELA (2003) que busca implementar uma aplicação envolvendo Inteligência Artificial e análise de Boletins de Ocorrência,

propondo soluções baseadas em *KMAI* – Gestão do Conhecimento em Inteligência Artificial; o foco deste trabalho foi a Investigação Criminal na era do Governo Eletrônico.

NATH (2006), fez análise criando modelos de crimes usando *Data Mining* e algoritmos de agrupamento, disponibilizando os resultados em um ambiente geoespacial. A região que ela realizou os estudos foi com números da criminalidade de *New Orleans.* O trabalho mais recente encontrado foi desenvolvido por (WANG et al, 2013), no qual tinham como meta detectar automaticamente modelos de crimes, prevendo a série em que eles ocorreriam, e se eram cometidos pela mesma pessoa ou grupo. Para conseguir identificar os indivíduos, existe uma captura de aspectos importantes como *modus operandi* do indivíduo ou grupo. Para este trabalho, eles utilizaram o que definiram como *Series Finder for Pattern Detection* (método de aprendizagem supervisionada para detectar modelos de crime). Os dados que eles trabalharam nesta pesquisa eram da região de Cambridge e Portland entre os anos 1997 e 2012 num total de 4855 registros.

Um importante passo frente ao desenvolvimento de ferramentas que tornem a Segurança Pública mais inteligente está sendo dado pela IBM[3] que, em parceria com a Policia de Nova York (*NYPD – New York Police Department*) criaram várias tecnologias em prol da redução da criminalidade (ver Figura 6). Uma delas facilita no cadastramento de indivíduos suspeitos e suas características em tempo real, uma vez que os policiais possuem dispositivos portáteis para este procedimento. Também faz parte desta inovação tecnológica a implantação de câmeras de vigilância em regiões na qual a criminalidade possui um alto índice. O Professor W. Richard Janikowski[4], Diretor do Centro de Pesquisa e Criminologia Comunitária da Universidade de Memphis – EUA, afirma que em 19 anos de trabalho com as tecnologias, a redução foi de mais de 75%.

No que se refere à vigilância por câmeras, os resultados parecem ser positivos em todo lugar que é instalada. Em um trabalho que envolveu Mineração de Dados de ocorrências policiais relacionadas a furtos de veículos na região central da cidade de Joinville/SC, SOUZA (2005) afirma que uma das causas que favoreceu para a redução de furtos de veículos foi justamente a implantação de câmeras de monitoramento no local onde a mineração apontou como existiria um alto índice de furtos de veículos. Estas câmeras estão conectadas diretamente com a CRE190 onde policiais militares acompanham as imagens 24 horas por dia. Porém, cogita-se o fato de não existir uma

[3] http://www.ibm.com/smarterplanet/br/pt/public_safety/ideas/index.html?re=sph
[4] http://www.youtube.com/watch?v=eMipfJcYSk0#t=43

redução, mas sim uma migração para regiões onde não possuem monitoramento por câmeras.

Figura 6. Perspectiva Histórica do NYPD - Período de 2001 a 2008[3].

A necessidade de uma ferramenta para apoiar a Gestão na Segurança Pública torna-se ainda mais importante, porque atualmente as operações realizadas são baseadas no conhecimento empírico dos profissionais de Segurança Pública, ou seja, a relação de existir um período com baixos índices de ocorrências e repentinamente sair do normal e estes índices se elevarem de uma forma abrupta gera uma atenção especial para que rapidamente estes índices voltem a baixar. De uma certa forma, sem ferramenta específica a atividade de identificar uma oscilação gradual nos índices de ocorrência se torna muito difícil de acontecer. Estas mudanças podem estar relacionadas a vários fatores como por exemplo: períodos de mês, vésperas de feriados, dias de pagamento, onde existe uma maior movimentação de dinheiro na região.

Com base nestes fatores, estes profissionais já tendem a ficar em locais estratégicos onde supostamente poderia acontecer um dos 3 tipos de crimes analisados neste trabalho. Surge então a necessidade de uma ferramenta que, através dos algoritmos baseados em conjuntos de classificadores, rapidamente detectem esta mudança nos padrões da criminalidade e que tão rapidamente se possa planejar ações para que o estado volte a normalidade.

Os estudos e os resultados apresentados neste trabalho podem abrir caminhos para que outros trabalhos sejam desenvolvidos, criando assim, um vasto leque de op-

ções disponíveis para o Suporte à Decisão em Segurança Pública, tendo em vista que os tipos de crimes analisados não possuem uma característica fixa para acontecer.

O *Drift Detection* torna-se importante para a área da Segurança Pública, podendo transformar estas informações subliminares das ocorrências diárias, numa informação precisa e importante e que nem sempre são conhecidas pelas autoridades em função dos contextos escondidos e mudanças de comportamento que podem ocorrer.

2.9. A Plataforma *MUMPS*

A Secretaria de Segurança Pública do Estado de Santa Catarina utiliza o sistema EMAPE (Estação Multitarefa de Atendimento Policial e Emergências), desenvolvido na plataforma *Mumps,* sendo utilizado por todas as Centrais 190 do Estado de Santa Catarina para registro de ocorrências recebidas via telefone 190. A plataforma *Mumps* (*Massachussetts General Hospital Utility Multi-Programming System*), possui uma linguagem de programação procedimental, normalmente interpretada, tendo sido criada por Neil Pappalardo no ano de 1969. Esta linguagem oferece uma ampla gama de recursos com baixo custo e dela derivou-se a *Super Mumps* e o Banco de Dados *Caché*.

Ele foi durante algum tempo uma boa linguagem para criar sistemas administrativos multiusuários. Seu início ocorreu devido à necessidade do Hospital Geral de *Massachussetts*[5], controlar uma grande epidemia de "caxumba" em seus pacientes e eles necessitarem de um sistema que pudesse ajudá-los a controlar tal situação. Iniciaram então o desenvolvimento de uma aplicação na referida plataforma.

O sistema utilizado em Santa Catarina apenas armazena as ocorrências e até então não realizava nenhuma análise através de uma ferramenta que pudesse explorar o conhecimento ali escondido. Isto poderia auxiliar a Policia Militar de Santa Catarina em um melhor planejamento para a distribuição de policiamento e consequentemente uma redução na criminalidade.

2.10 A Central Regional de Emergência 190

A Central Regional de Emergência 190 (CRE 190), atua com Policiais Militares e Agentes temporários no atendimento do 190 onde recebem as ligações e fazem o re-

[5] Considerado o terceiro hospital mais antigo daquele país, no ano de 2007 foi eleito o melhor hospital norte americano segundo pesquisa realizada pela *U.S News* e pela *World Report.*

gistro das ocorrências de uma cidade do estado de Santa Catarina. Aproximadamente 2000 ligações por dia são recebidas pela CRE 190, gerando as ocorrências pelos Agentes temporários que são repassadas para os Policiais Militares que atuam na rua para o atendimento da ocorrência. No sistema EMAPE, existem cerca de 1030 tipos de ocorrências divididas em 09 subgrupos como por exemplo Auxílios a Comunidade, Crimes e Contravenções, Ocorrências Diversas, Emergências, Crimes contra o Meio Ambiente, Incêndios, Serviços/Atividades de Policiamento, Serviços/Atividades de Manutenção e ocorrências de Transito que, depois de atendidas pelos policiais militares que estão trabalhando na rua, entram em contato novamente com a CRE 190 para passar os dados para o fechamento destas.

Esta Central atua 24 horas por dia, 7 dias por semana dividindo seus horários de trabalho em equipes com aproximadamente 10 pessoas por grupo que são responsáveis pela inserção correta dos dados no sistema para que se mantenha a integridade nas informações ali registradas.

2.11 A Ferramenta MOA (Massive Online Analysis)

Desenvolvida em Java na Universidade de Waikato, é um framework semelhante ao WEKA desenvolvido para mineração de dados por HALL et al (2009). No framework MOA, BIFET et al (2011) incorporou ferramentas para o problema de análise de *data streams*. Ele pode ser executado tanto em modo de linha de comando como através de uma interface gráfica.

Seguindo alguns padrões do próprio framework, é possível desenvolver novos algoritmos para serem incorporados na ferramenta, podendo assim ser utilizados para análises de dados. Este framework possui algoritmos de classificação *online*, geradores de dados, métodos de avaliação, agrupamento *online* e estatísticas. Os algoritmos *DWM* (GOMES, 2012) e o *ADDExpert* implementado para esta pesquisa, foram adicionados ao framework MOA para que pudessem ser realizadas comparações com outros algoritmos semelhantes e verificar qual o melhor algoritmo para analisar conjuntos de dados reais, em específico, dados relacionados a ocorrências policiais de uma região.

Considerações Finais

Através da revisão apresentada neste capítulo, podemos classificar as ferramentas que utilizaremos para a análise dos dados relativos às ocorrências registradas na base de dados utilizada neste trabalho. Vemos que trata-se de um desafio, pois esta base ainda não sofreu nenhum tipo de análise através de outra ferramenta e dela também não houve um conhecimento extraído.

Capítulo 3

METODOLOGIA

Este capítulo aborda o desenvolvimento metodológico da pesquisa. O ponto inicial se refere a modelagem dos dados, em seguida veremos o processo de remoção dos ruídos, a criação de novos atributos e a conversão do conjunto de dados em valores, na grande maioria quantitativos uma vez que as ocorrências diárias foram somadas para a definição das classes e em determinadas ocorrências pode existir mais de uma vítima.

No segundo momento, apresentamos o desenvolvimento do algoritmo *ADDExpert.D* proposto por KOLTER e MALOOF (2005) o qual foi utilizado como referência em nossas análises. Finalizando o capítulo, apresentaremos as métricas de avaliação propostas para a pesquisa e seus procedimentos usuais para a avaliação de classificadores.

3.1. Atividades da Pesquisa

Para que a pesquisa pudesse ser realizada, definimos etapas (Figura 7) para a organização dos dados que até então se encontravam na base de dados do Sistema EMAPE no CIEMER190.

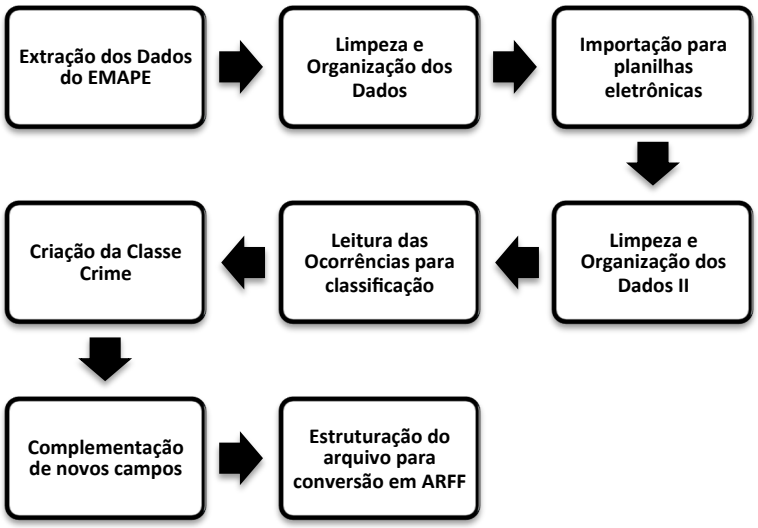

Figura 7. Etapas de aquisição e tratamento dos dados.

Dessa forma, pudemos definir mais claramente o que seria executado em cada uma das etapas, nomeamos este conjunto de etapas como modelagem dos dados.

3.2.1 Modelagem dos Dados

a) Extração dos Dados do EMAPE

A extração de dados do sistema EMAPE é um tanto quanto ultrapassada onde, por incapacidade do próprio sistema, vários tipos de relatórios solicitados para o planejamento da divisão do policiamento não é possível criar pela ferramenta. O usuário precisa ter um conhecimento mais avançado da ferramenta, pois vários itens precisam ser realizados. Esta é uma das dificuldades dos usuários, pois em sua grande maioria, utilizam o sistema inserindo dados de ocorrências e muito pouco extraem relatórios dela.

Esta tarefa de extrair relatórios é de competência de outro setor dentro da corporação e que precisa customizar outros modelos em outras ferramentas para se ter o resultado solicitado pelo comandante do batalhão. O arquivo extraído do sistema, precisa ser enviado para um endereço eletrônico funcional através da própria ferramenta e então o usuário pode iniciar a criação do seu relatório, fazendo uma leitura nesse arquivo filtrando somente o que lhe é interessante para aquele momento.

Depois de lido este arquivo que está inicialmente em formato de texto, o usuário começa a estruturação de seu relatório em planilha eletrônica assim como também o preenchimento de seus campos. Esta atividade é demorada e cansativa e pode não apresentar dados coerentes com a situação que tenha ocorrido.

b) Limpeza e Organização dos Dados

Como o sistema em que as ocorrências policiais são geradas possui muitas limitações para gerar análises e extrair relatórios, o arquivo extraído em formato de texto apresentou um total de 65479 ocorrências geradas no período de 01 de janeiro de 2010 a 30 de julho de 2010 na cidade de Joinville, Santa Catarina. A escolha por este espaço de tempo tem como principal motivo um problema ocorrido com a base de dados da aplicação onde nos meses de agosto, setembro e outubro de 2010 existiram muitos problemas, deixando o sistema inativo por mais de 15 dias, causando vários espaços de tempo sem ocorrências inseridas. Quando a aplicação voltou ao normal, todas estas ocorrências que até então haviam sido geradas em papeis, precisaram ser inseridas no sistema porém não com a data em que ela existiu, mas sim com a data em que ela foi inserida, ou seja,

comprometeu todo o conjunto de dados para extração de relatórios confiáveis a partir de agosto de 2010.

O volume de ocorrências envolve aproximadamente 1030 tipos de ocorrências (auxílios diversos a comunidade, acidentes de transito, averiguações, roubos, furtos, homicídios, suicídios e crimes ambientais). Os dados foram obtidos com autorização formal do Tenente Coronel PM Dirceu Neundorf, oficial comandante da CRE190 (Anexo 1).

De todas as ocorrências extraídas do sistema somente as que estão inseridas no Grupo de Crimes e Contravenções, mais especificamente os crimes de roubo ou assalto contra pessoa, estabelecimento comercial e residências foram utilizadas, ou seja, dos aproximadamente 1030 tipos de ocorrências, trabalhamos com 3 tipos, totalizando 211 ocorrências no período analisado. A escolha por estes 3 tipos de ocorrências está relacionada diretamente ao tipo de crime, tendo em vista que, o principal objetivo da Segurança Pública hoje, em qualquer região do país é a redução dos índices de criminalidade, porém, ainda dispõem de poucos artifícios para que possa antever as possibilidades dos crimes, o que faz com que grande parte das análises ocorram depois que os crimes foram cometidos.

c) Importação para planilhas eletrônicas

O filtro destas ocorrências foi realizado em planilha eletrônica para que todas as ocorrências fossem analisadas, separadas e complementadas com as informações adicionais importantes para um melhor resultado na análise (ver Figura 8). Para a criação do arquivo que foi submetido a análise dos algoritmos, os dados preparados foram classificados de acordo com os códigos utilizados para cada registro, significando o tipo de ocorrência que aquele fato corresponde, seguido pelos demais campos como o bairro do fato, logradouro (nome da rua), dia, mês e ano do fato, dia da semana, hora, minuto, sexo da vítima, o tipo de estabelecimento e o índice de crimes conforme foi classificado com base nos valores estabelecidos no **Error! Reference source not found.**.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	COD	BAIRRO	LOGRADOURO	DIA	MÊS	ANO	DIA_SEM	HORA	MIN	SEXO	IDADE	TP_ESTAB	INDICE CRIMES
2	C221	JD IRIRIU	SENADOR RODRIGO LOBO	1	1	10	6	4	4	M	23	PESSOA	2
3	C221	JD IRIRIU	HERMANN HUHN	1	1	10	6	5	30	M	27	PESSOA	2
4	C221	CENTRO	SAO PAULO	1	1	10	6	13	15	F	21	PESSOA	2
5	C221	VILA NOVA	DOS SUICOS	2	1	10	7	4	52	M	27	PESSOA	1
6	C222	BUCAREIN	GETULIO VARGAS	3	1	10	1	8	36	F	24	FARMACIA	1
7	C221	CENTRO	ORESTES GUIMARAES	4	1	10	2	16	47	M	20	PESSOA	1
8	C222	COMASA	RORAIMA	4	1	10	2	19	17	F	21	PANIFICAI	1
9	C221	JD IRIRIU	LAURO FAGUNDES DOS REIS	6	1	10	4	4	50	F	32	PESSOA	1
10	C221	GLORIA	BENJAMIN CONSTANT	6	1	10	4	9	18	M	49	PESSOA	1
11	C222	BOA VISTA	ALBANO SCHMIDT	7	1	10	5	17	40	F	31	FERRAGEN	2
12	C221	JD IRIRIU	TELEMACO BORBA	7	1	10	5	20	40	F	23	PESSOA	2
13	C218	IRIRIU	WILLY SCHOSSLAND	7	1	10	5	22	58	F	42	RESIDENC	2
14	C218	AMERICA	ALBRECHT SCHMALZ	9	1	10	7	2	23	M	55	RESIDENC	1
15	C221	VILA NOVA	PAULINO DE JESUS	10	1	10	1	0	52	M	56	PESSOA	1
16	C221	COMASA	PONTE SERRADA	10	1	10	1	14	28	F	24	PESSOA	1
17	C221	BOA VISTA	DESEMBARGADOR TAVARES SOBRINHO	11	1	10	2	21	40	M	19	PESSOA	1
18	C221	COSTA E SILVA	VICE PREFEITO LUIZ CARLOS GARCIA	12	1	10	3	13	48	M	13	PESSOA	1
19	C221	AMERICA	ARARANGUA	13	1	10	4	6	12	F	47	PESSOA	1

Figura 8. Modelo de planilha criada para tratar os dados.

d) Limpeza e Organização dos Dados II

A cidade de Joinville possui dois Batalhões de Polícia Militar, o 8º Batalhão de Polícia Militar (8º BPM), que é o foco deste estudo, é responsável pelos atendimentos de ocorrências em toda a Região Norte da Cidade e o 17º Batalhão de Polícia Militar (17º BPM) é responsável pelos atendimentos de ocorrências em toda a Região Sul da Cidade. No entanto, Joinville possui apenas uma Central Regional de Emergência 190 (CRE190), isso significa que todas as ligações de ambas as regiões são recebidas em um único lugar. Deste ponto, é realizada a distribuição para as viaturas via radiocomunicação. Por este motivo, todas as ocorrências são mantidas no mesmo banco de dados, haja visto que possuem um número da ocorrência que é sequencial.

e) Leitura das Ocorrências para classificação

Como nos relatórios extraídos, não existe a possibilidade de escolher qual região deseja obter as ocorrências, este trabalho teve que ser feito manualmente depois que o relatório de todas as ocorrências relacionadas ao estudo foi extraído do sistema. Com este processo de segregação das ocorrências da região desejada onde ocorrências que estavam cadastradas em outros bairros que não pertencessem a área do 8º Batalhão foram removidas e a planilha ficou pronta para ser então convertida em um modelo que pudesse ser trabalhado no *MOA*, ferramenta que foi apresentada no item 2.11.

Como o sistema apresenta poucas informações a respeito do tipo de estabelecimento alvo dos marginais e outras informações que seriam interessantes para a pesquisa, foi necessário fazer uma leitura de todas as 211 ocorrências selecionadas. Desse modo conseguimos identificar o dia do mês, se o crime aconteceu em véspera de feriado, durante um feriado, o período do dia, o sexo das vítimas, o tipo de estabelecimento e o período da semana em que aconteceu o crime.

O Gráfico 1 apresenta a distribuição dos valores das classes, criamos um gráfico em forma de histograma que fosse possível visualizar a distribuição onde todos os valores foram separadas a cada 10 instâncias. Podemos ver que se trata de um conjunto de dados com algumas variações entre uma quantidade de instâncias e outra e em algumas vezes encontramos até valores nulos.

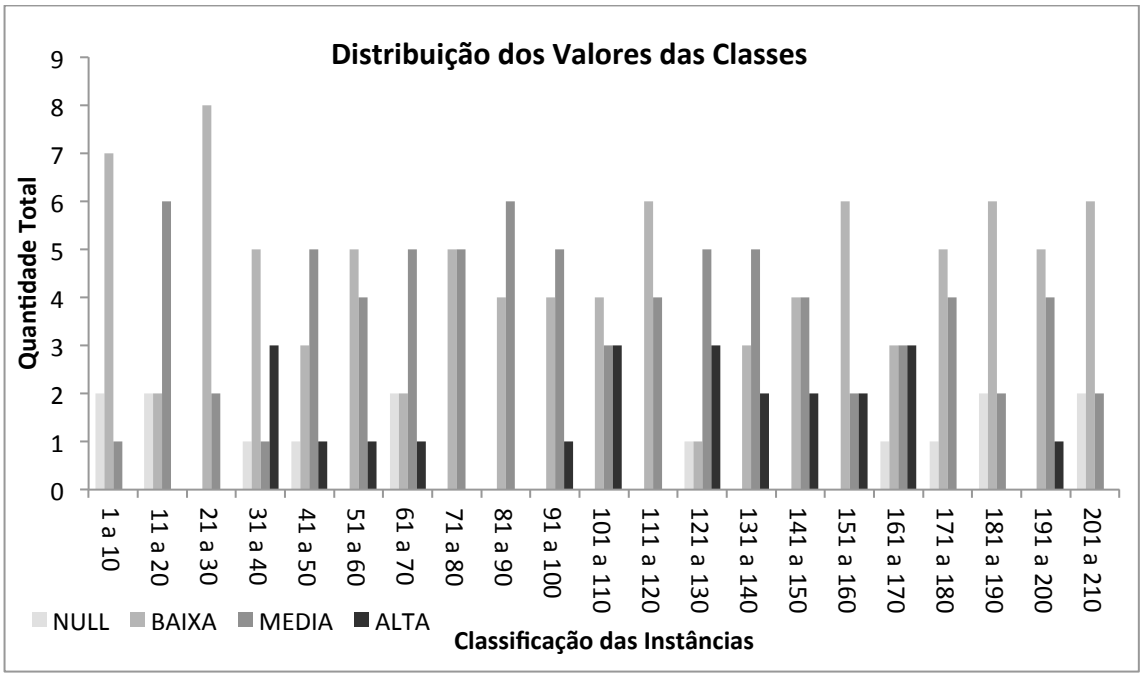

Gráfico 1. Distribuição de Valores das Classes

f) Criação da Classe Crime

O próximo passo foi fazer uma classificação das ocorrências (**Error! Reference source not found.**) por quantidade de crimes apresentada diariamente. Contamos com o apoio de profissionais da área de Segurança Pública onde definimos quatro faixas de crime iniciando da Faixa 1 para períodos com índice de ocorrências nulo, que denominamos Null, até 4 para períodos com altos índices de ocorrências. Foi realizada então uma contagem dos tipos de ocorrências registradas por dia e a classe correspondente a faixa de ocorrências foi inserida na tabela (Quadro 1).

Quadro 1. Classificação dos índices de Ocorrências.

Faixa_Crime	Quantidade de Ocorrências	*CLASS*
Faixa 1	0 ocorrências	Null
Faixa 2	De 1 a 3 ocorrências	Baixa
Faixa 3	De 4 a 6 ocorrências	Média
Faixa 4	Acima de 7 ocorrências	Alta

g) Complementação de novos campos

A planilha criada para fazer a limpeza e a organização dos dados (Figura 8) não foi a escolhida para a análise por não ser o objetivo, neste momento, de realizarmos uma análise que apresentasse somente locais de roubo e horários que eles ocorreram. Optamos por importar os dados para uma planilha mais completa onde dados até então desconhecidos por quem olhava somente o registro inicial da ocorrência foram inseridos. Estes dados são inerentes a quantidade de pessoas envolvidas na ocorrência, sexo e idade.

Todas as datas das ocorrências foram verificadas em um calendário do ano de 2010 buscando identificar as vésperas de feriados e os feriados, início, meio e fins de semana. Fizemos isso buscando trazer para a pesquisa, fatos e formas bem próximas da realidade. Estes itens são utilizados para realizar o mapeamento da criminalidade de uma região quando se trabalha com algum tipo de consulta de ocorrências para se estruturar uma operação policial com o intuito de coibir a criminalidade em determinado período do dia ou da semana.

A definição dos campos Madrugada, Manhã, Tarde e Noite, recebem valores quantitativos de acordo com a quantidade de vezes que ocorreu (Quadro 1), sempre seguindo a seguinte configuração:

- 00:01 e 05:59 – Madrugada
- 06:00 e 11:59 – Manhã
- 12:00 e 17:59 – Tarde e
- 18:00 e 00:00 Noite

O Período da Semana foi estabelecido da seguinte forma:

- Segunda e Terça-feira valor *Inicio*
- Quarta e Quinta-feira valor *Meio* e
- Sexta-feira, Sábado e Domingo valor *Fim*.

Os campos Sexo (M/F) e tipo de roubo (Pessoa, Estabelecimento ou Residência) também recebem valores quantitativos pois em um dia pode ocorrer nenhuma ou várias ocorrências do tipo analisado e estas ocorrências podem ter de uma a várias vitimas do mesmo sexo ou de sexo distinto.

Um novo arquivo foi criado e nele foram atribuídos campos numéricos, verdadeiros e falsos e com as definições de faixas criadas para uma melhor visualização dos resultados (**Error! Reference source not found.**). Este arquivo foi utilizado neste tra-

balho, utilizamos a denominação *TRUE* para a ocorrência em véspera ou em feriados e *FALSE* quando fora destes períodos.

Podemos ver que o campo INDICE CRIMES foi renomeado para *CLASS* e campos como COD, BAIRRO, LOGRADOURO, DIA, MÊS, ANO, DIA_SEM, HORA, MIN, IDADE, TP_ESTAB foram alterados de acordo com a necessidade de inserção dos dados e então toda a reclassificação dos dados foi realizada.

Estes campos recebem a soma de crimes registrados em um determinado período ou quando esta ocorrência envolveu mais de uma pessoa como vítima e que foram de sexos diferentes, como por exemplo: se uma ocorrência fosse registrada no período da manhã em uma residência e que no local estivesse um casal, o campo SEXO_F e SEXO_M receberiam valor igual a 1, pois foram duas vítimas envolvidas na ocorrência. Por este motivo foram convertidos para tipo numérico tendo em vista que receberiam somente valores inteiros.

Quadro 2. Organização dos dados para a Análise.

Atributo	Tipo
DIA_MES	NUMERIC
VESP_FERIADO	TRUE/FALSE
FERIADO	TRUE/FALSE
MADRUGADA	NUMERIC
MANHÃ	NUMERIC
TARDE	NUMERIC
NOITE	NUMERIC
SEXO_F	NUMERIC
SEXO_M	NUMERIC
PESSOA	NUMERIC
ESTAB	NUMERIC
RESID	NUMERIC
PERSEMANA	INICIO, MEIO, FIM
CLASS	NULL, BAIXA, MEDIA, ALTA

h) Estruturação do arquivo para conversão em ARFF

O Quadro 1 apresenta o modelo final do arquivo que foi submetido a análise nesta pesquisa. Podemos ver que os valores numéricos representam a quantidade de

ocorrências registradas no dia, a quantidade de pessoas envolvidas e os alvos dos roubos.

Quadro 1. Modelo de Conjunto de dados preparado para criar o arquivo .arff

DIAMES	VESP_FERIADO	FERIADO	MADRUG.	MANHA	TARDE	NOITE	SEXO_F	SEXO_M	PESSOA	ESTAB	RESID	PER_SEM	CLASSE
3	FALSE	FALSE	0	1	0	0	1	0	0	1	0	FIM	BAIXA
4	FALSE	FALSE	0	0	1	1	1	1	1	1	0	INICIO	BAIXA
5	FALSE	FALSE	0	0	0	0	0	0	0	0	0	INICIO	NULL
6	FALSE	FALSE	1	1	0	0	1	1	2	0	0	MEIO	BAIXA
7	FALSE	FALSE	0	0	1	2	3	0	1	1	1	MEIO	MEDIA
8	FALSE	FALSE	0	0	0	0	0	0	0	0	0	FIM	NULL
9	FALSE	FALSE	1	0	0	0	0	1	1	0	0	FIM	BAIXA
10	FALSE	FALSE	2	0	0	0	2	0	1	0	1	FIM	BAIXA
11	FALSE	FALSE	0	0	0	1	0	1	1	0	0	INICIO	BAIXA
12	FALSE	FALSE	0	0	1	0	0	1	1	0	0	INICIO	BAIXA
13	FALSE	FALSE	0	1	0	0	1	0	1	0	0	MEIO	BAIXA
14	FALSE	FALSE	0	2	0	0	0	2	2	0	0	MEIO	BAIXA
15	FALSE	FALSE	2	0	0	1	0	3	3	0	0	FIM	MEDIA
16	FALSE	FALSE	0	0	0	0	0	0	0	0	0	FIM	NULL
17	FALSE	FALSE	0	0	0	0	0	0	0	0	0	FIM	NULL
18	FALSE	FALSE	1	0	2	1	1	3	4	0	0	INICIO	MEDIA
19	FALSE	FALSE	0	0	0	1	0	1	1	0	0	INICIO	MEDIA
20	FALSE	FALSE	1	0	1	1	2	1	2	1	0	MEIO	MEDIA
21	FALSE	FALSE	0	0	1	0	0	1	1	0	0	MEIO	MEDIA
22	FALSE	FALSE	0	1	1	0	2	0	2	0	0	FIM	MEDIA
23	FALSE	FALSE	0	1	0	0	1	0	0	1	0	FIM	BAIXA
24	FALSE	FALSE	0	1	0	0	0	1	0	1	0	FIM	BAIXA

Com o conjunto de dados neste formato, foi convertido para formato *.arff* (Figura 9) para ser trabalhado no *MOA*. Para um arquivo ser convertido para o formato *.arff,* é necessário criar um cabeçalho apresentando toda a descrição (formato) dos registros (Figura 9). Algumas ferramentas criam este arquivo automaticamente, por não ser uma tarefa que exigia muito esforço, criamos este arquivo manualmente.

```
@RELATION Ocorrencias

@ATTRIBUTE DIA_MES NUMERIC
@ATTRIBUTE VESP_FERIADO {TRUE,FALSE}
@ATTRIBUTE FERIADO {TRUE,FALSE}
@ATTRIBUTE MADRUGADA NUMERIC
@ATTRIBUTE MANHA NUMERIC
@ATTRIBUTE TARDE NUMERIC
@ATTRIBUTE NOITE NUMERIC
@ATTRIBUTE SEXO_F NUMERIC
@ATTRIBUTE SEXO_M NUMERIC
@ATTRIBUTE PESSOA NUMERIC
@ATTRIBUTE ESTAB NUMERIC
@ATTRIBUTE RESID NUMERIC
@ATTRIBUTE PERSEMANA {INICIO,MEIO,FIM}
@ATTRIBUTE class {BAIXA,MEDIA,ALTA,NULL}

@DATA
3,FALSE,FALSE,0,1,0,0,1,0,0,1,0,FIM,BAIXA
4,FALSE,FALSE,0,0,1,1,1,1,1,1,0,INICIO,BAIXA
5,FALSE,FALSE,0,0,0,0,0,0,0,0,0,INICIO,NULL
6,FALSE,FALSE,1,1,0,0,1,1,2,0,0,MEIO,BAIXA
7,FALSE,FALSE,0,0,1,2,3,0,1,1,1,MEIO,MEDIA
8,FALSE,FALSE,0,0,0,0,0,0,0,0,0,FIM,NULL
9,FALSE,FALSE,1,0,0,0,0,1,1,0,0,FIM,BAIXA
10,FALSE,FALSE,2,0,0,0,2,0,1,0,1,FIM,BAIXA
```

Figura 9. Exemplo de arquivo no formato .arff

Até então, todo o processo realizado foi bastante manual tendo em vista a limitação do sistema hoje existente para o controle das ocorrências policiais geradas. Foi necessário ter muita atenção em todos estes processos para evitar que a integridade dos dados fosse afetada. Optamos por trabalhar com estes dados referentes a seis meses do ano, pois o sistema apresentou problemas no mês de setembro e outubro do mesmo ano, causando uma lacuna de tempo nos registros o que poderia prejudicar o resultado final da pesquisa. Desde então, este sistema apresentou outras falhas e prejudicou a retirada de dados que fossem relativos a um ano completo.

3.2 Métricas de Avaliação

A taxa de acerto foi a principal métrica de avaliação desta pesquisa. Sabemos que é através de métricas que conseguimos identificar o melhor desempenho e, pela natureza do conjunto de dados, não seria interessante outra métrica de avaliação senão esta. O conjunto de dados preparado para a execução deste trabalho, sofreu o mesmo

tipo de limpeza, classificação e organização que é feito quando existe a necessidade da criação de algum tipo de relatório que apresente informações até então desconhecidas sobre a criminalidade de uma determinada região da cidade.

Reconstruímos em laboratório todos os processos de análise de ocorrências que são realizados a cada 3, 7 e 10 dias pelos policiais militares responsáveis pelo acompanhamento das ocorrências no 8º BPM. Estes períodos pré-definidos pelos responsáveis por estruturar as Operações Policiais da região devem ao fato de que:

- 3 dias por estar relacionado com finais de semana (sexta-feira, sábado e domingo) e que não existe um controle em tempo real pelos responsáveis por se tratar de horários extras ao expediente normal de trabalho;
- 7 dias e 10 dias pelo fato de estar relacionado a uma semana inteira de ocorrências ou a um determinado período de operação. Com estes valores é possível planejar o policiamento e medir os resultados ao final de cada período sem que ocorra uma alta nos índices de criminalidade e que seja constatada de forma tardia.

Procuramos trazer a realidade mais próxima da pesquisa, para que, com base nos resultados apresentados, novas pesquisas possam ser criadas e que seus resultados possam ser utilizados de forma positiva para a elaboração dos planos estratégicos operacionais da corporação, comparando-os com as análises feitas hoje em dia pelos especialistas em segurança para a tomada de decisão. Outro fator importante e que pode ser respondido com esta pesquisa, é o de que se existe aumento da criminalidade em determinados períodos do ano como por exemplo carnaval, feriados prolongados, férias e períodos chuvosos.

Acredita-se que, para os tipos de crimes analisados, estes fatores não apresentem influências, pois não trata-se de um tipo de crime realizado através da ocasião, mas sim, uma espécie de preparação é feita como por exemplo um levantamento do local, identificação dos alvos e outros itens que ajudam os criminosos a realizar o ato delituoso.

3.2.1 Procedimentos

Para a execução das análises, alguns procedimentos foram seguidos de acordo com a metodologia proposta. Estes procedimentos são os usuais para as avaliações de classificadores como por exemplo, o método de avaliação.

- **A configuração *batch*:** é utilizada para uma quantidade limitada de dados, ou seja, um "lote" de dados e que podem utilizar diferentes estratégias de separação dos conjuntos de treinamento e teste como por exemplo (*Holdout, Cross-Validation e Leave-one-out*). Um estudo mais aprofundado sobre esta configuração pode ser encontrado em (GOMES, 2012) e (KOHAVI, 1995).
- **A configuração *online*:** escolhida para a execução do trabalho pois podemos considerar a entrada de dados de ocorrências policiais, um conjunto de dados *online* que referem-se a entradas de ligações via 190 para uma Central onde então são geradas as ocorrências policiais e posteriormente encaminhadas as viaturas. Este modelo de configuração trabalha corretamente quando existe uma entrada volumosa de dados. No caso da Polícia Militar de Joinville, estas entradas chegam a atingir uma média de 350 ocorrências por dia, aumentando ou diminuindo conforme o período do dia, da semana e do mês e que precisam ser filtradas e analisadas conforme a necessidade do trabalho pretendido.

O primeiro método de avaliação para a configuração *online* é o método ***Periodic Holdout***, utilizado para rastrear a evolução ou regressão da taxa de acerto do modelo ao longo do tempo. Segundo (GOMES, 2012), testar o algoritmo muito frequentemente pode influenciar negativamente o tempo de processamento para a avaliação do modelo. Exemplos do conjunto ***Holdout*** (retenção) devem ser exemplos ainda sem nenhuma interferência do método, ou seja, devem ter sido recém fornecidos pela *stream*.

O segundo método de avaliação é chamada ***Interlevead Test-Then-Train*** que atua intercalando o treinamento e o teste, ou seja, uma quantidade do conjunto é utilizada para realizar o teste e, logo em seguida, para treinar o classificador. Como este modelo consegue prover a taxa de acerto média, chega muito próximo do que atualmente é realizado quando os dados de ocorrências policiais são analisados, ou seja, uma análise da semana é realizada e depois disso, operações são realizadas por uma semana para que seja possível medir o resultado apresentado.

O terceiro método e o escolhido para a realização dos experimentos é chamado ***Prequential*** que une os dois métodos apresentados anteriormente, ou seja, é bastante similar com o método ***Test-Then-Train*** intercalando teste e treinamento, porém, possui o fator de desvanecimento que, segundo GOMES (2012) faz com que diminua o impacto que as predições incorretas iniciais tem sobre as predições futuras. O modelo ***Prequential*** também possui algumas características do método ***Periodic Holdout*** que fornece a taxa de acerto do conjunto.

Podemos representar o método escolhido na Figura 10 onde, quando customizada uma janela de 3 dias, ocorrerá o teste com 3 instâncias e treino com as outras 3 instâncias seguintes até a última instância do conjunto, a Figura 11 representa a customização para uma janela de 7 dias e desta vez testando com 7 instâncias e treinando com as próximas 7 instâncias e a Figura 12 representa janela de 10 dias e os testes e treinamentos ocorrem a cada conjunto de 10 instâncias. Estas parametrizações de dias foram escolhidas por representarem o processo executado manualmente pelos especialistas em Segurança Pública atualmente.

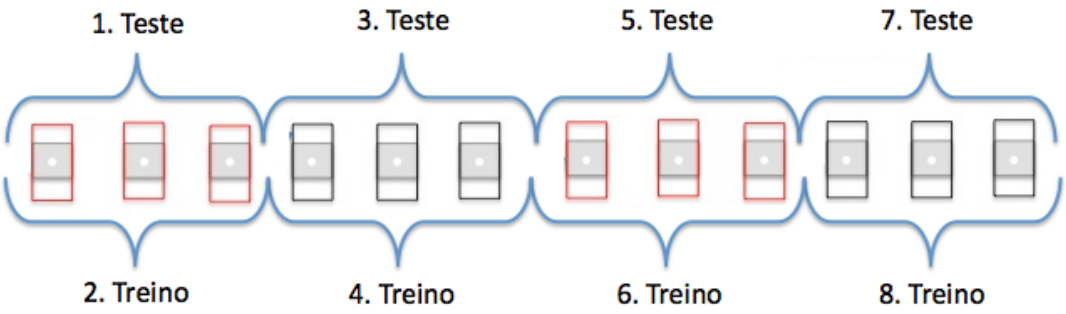

Figura 10. Modelo método Prequential em janela de 3 dias

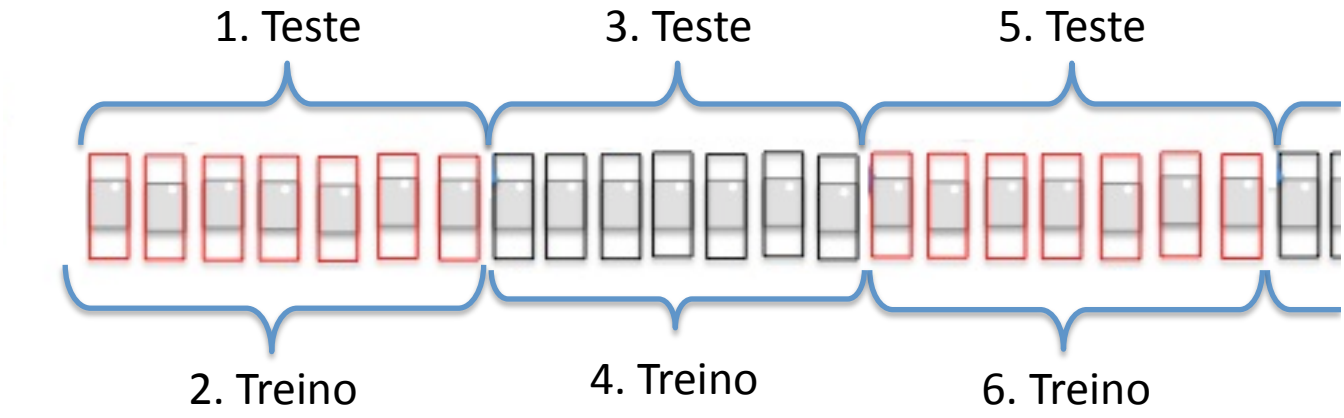

Figura 11. Modelo método Prequential em janela de 7 dias.

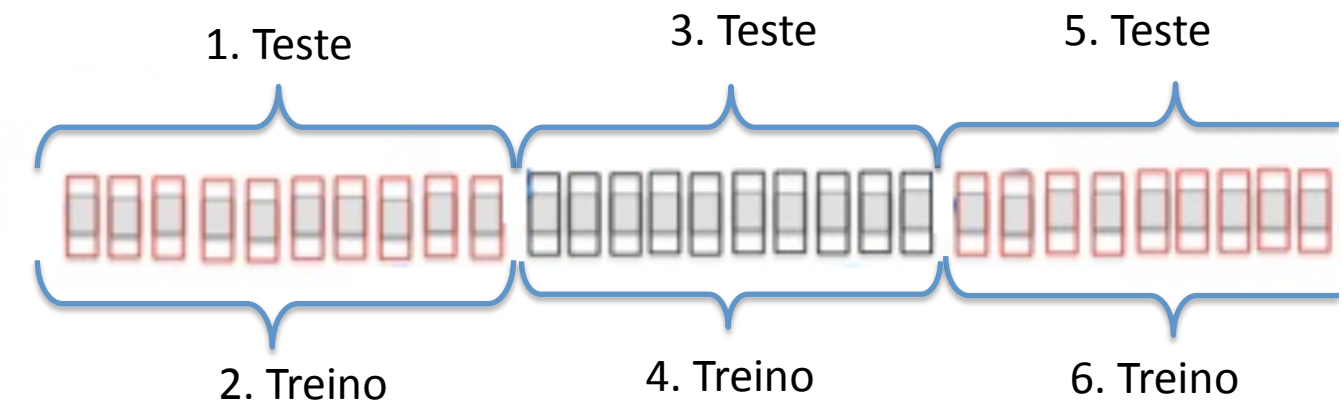

Figura 12. Modelo método Prequential em janela de 10 dias.

O fator utilizado para avaliar o desempenho dos algoritmos é denominado medidas de desempenho. Existem três medidas propostas:

- Taxa de acerto das instâncias classificadas corretamente;
- Quantidade de Classificadores do Conjunto que é uma métrica avaliada a cada passo da fase de treinamento e,
- Idade média dos classificadores no conjunto, métrica definida pela somatória da idade dos classificadores, dividido pelo total de classificadores do conjunto.

Para a realização destes experimentos, consideramos somente a taxa de acerto (porcentagem de instâncias classificadas corretamente pelos algoritmos) pois queremos analisar o comportamento dos algoritmos quando trabalhando com *stream* de dados reais. A quantidade de classificadores do conjunto e a sua idade média não foram levados em consideração uma vez que o conjunto de dados é pequeno e estas métricas não são o foco da pesquisa.

O Quadro 2 apresenta um modelo criado para o controle do processo de execução dos classificadores sobre o conjunto de dados de ocorrências policiais. Com este modelo, todas as análises foram realizadas, utilizando sempre as métricas padrão de cada classificador e posteriormente algumas customizações foram realizadas buscando melhores resultados de classificação, porém, as customizações efetuadas não apresentaram resultados melhores dos que obtidos com as métricas padrão dos classificadores.

O conjunto de dados foi submetido a testes e treinamentos com janelas de (03, 07 e 10 dias) para que as simulações chegassem mais próximas da realidade das análises realizadas pelos especialistas em Segurança Pública. Foram utilizados 04 algoritmos

(*Leveraging Bag* e *ADWIN* disponíveis no framework *MOA* e o *DWM* implementado por GOMES (2012) e *ADD Expert* implementado neste trabalho). Para cada um dos 04 algoritmos, foram selecionados 05 classificadores base (baseados em árvores e no método *online Bagging* (*Naive Bayes, Hoeffding Tree, Leveraging Bag, Ozabag ADWIN, e ASHoeffding Tree*)). Para os classificadores base, não levamos em consideração a alteração de seus algoritmos para as execuções no conjunto de dados proposto, as alterações dos parâmetros ficou limitada aos (parâmetros de β (*Beta*), γ (*Gamma*), tamanho das janelas e quantidade de experts).

Quadro 2. Modelo de Controle para realização das análises.

Algoritmo		Tamanho Janela (dias) Teste e Treino	Classificador Base utilizado				
ADD	WEAKEST	3	NB	HT	LB	OZA	ASHT
		7	NB	HT	LB	OZA	ASHT
		10	NB	HT	LB	OZA	ASHT
	OLDEST	3	NB	HT	LB	OZA	ASHT
		7	NB	HT	LB	OZA	ASHT
		10	NB	HT	LB	OZA	ASHT
DWM		3	NB	HT	LB	OZA	ASHT
		7	NB	HT	LB	OZA	ASHT
		10	NB	HT	LB	OZA	ASHT
LEVBAG		3	NB	HT	LB	OZA	ASHT
		7	NB	HT	LB	OZA	ASHT
		10	NB	HT	LB	OZA	ASHT
ADWIN		3	NB	HT	LB	OZA	ASHT
		7	NB	HT	LB	OZA	ASHT
		10	NB	HT	LB	OZA	ASHT

Legenda

NB – Naive Bayes
HT – Hoeffding Tree
LB – Leveraging Bag
OzA – Ozabag ADWIN
ASHT – ASHoeffding Tree

3.2.2. Implementação do algoritmo *ADDExp.D*

Uma das propostas deste trabalho, é a implementação do algoritmo *ADDExp.D* que foi proposto por KOLTER e MALOOF (2005) no qual, segundo os próprios autores, é muito semelhante ao algoritmo DWM, apresentado no Capítulo 2. O algoritmo foi implementado com base no pseudocódigo proposto pelos autores (ver

Figura 13) e incorporado ao framework MOA. Este algoritmo é utilizado para analisar classes discretas e diferencia-se dos demais algoritmos de predição pelo fato de que novos especialistas podem ser adicionados durante o processo de aprendizagem online e para cada classificação possível, uma soma dos pesos é realizada para que se faça a classificação.

Algorithm *ADDExp.D* $(\{x,y\}^T,\beta,\gamma)$
Parameters:

$\{x,y\}^T$: training data with class $y \in Y$
$\beta \in [0,1]$: factor for decreasing weights
$\gamma \in [0,1]$: factor for new expert weight

Initialization:

1. Set the initial number of experts $N_1 = 1$.
2. Set the initial expert weight $\omega_{1,1} = 1$.

For $t = 1,2,...,T$:

1. Get expert predictions $\varepsilon_{t,1,...,}\varepsilon_{t,Nt} \in Y$

2. Output prediction:

$$\hat{y}_t = \arg\max_{c \in Y} \sum_{i=1}^{N_t} \omega_{t,i}\left[c = \varepsilon_{t,i}\right]$$

3. Update expert weights:

$$\omega_{t+1,i} = \omega_{t,i}\beta^{\left[y_t \neq \varepsilon_{t,i}\right]}$$

4. If $\hat{y}_t \neq y_t$ then add a new expert:

$$N_{t+1} = N_t + 1$$

$$\omega_{t+1,N_t+1} = \gamma \sum_{i=1}^{N_t} \omega_{t,i}$$

5. Train each expert on example x_t, y_t

Figura 13. Pseudocódigo *ADDExp* para classes discretas. (KOLTER e MALOOF, 2005)

Um fragmento da implementação do *ADDExp.D* onde os *experts* são selecionados pelo treino e adicionados a lista de experts, o cálculo da predição e a atualização dos pesos dos especialistas é feita, está apresentado na Figura 14.

```
@Override
public double[] getVotesForInstance(Instance instnc) {
    //output
    double votes[] = new double[instnc.numClasses()];

    if(experts.isEmpty()){
        ExpertAddExp exp = new ExpertAddExp((Classifier) getPreparedClassOption(baseLearnerOption));
        exp.trainOnInstance(instnc);
        experts.add(exp);
    }
    //get predictions from experts
    ArrayList<Integer> previsoes = new ArrayList<Integer>();
    //calculates the output prediction
    for(ExpertAddExp e : experts){
        previsoes.add(Utils.maxIndex(e.getVotesForInstance(instnc)));
        votes[Utils.maxIndex(e.getVotesForInstance(instnc))] += e.getWeight();
    }
    //updates expert weights
    for(int i = 0; i < experts.size(); i++){
        if(Utils.maxIndex(votes) != previsoes.get(i)){
            //atualiza o peso antigo
            experts.get(i).setWeight(betaOption.getValue() * experts.get(i).getWeight() );
        }
```

Figura 14. Fragmento da implementação do *ADDExp.D*.

As técnicas propostas por KOLTER e MALOOF (2005), *Oldest First e Weakest First* (Figura 15) foram implementadas em arquivos diferentes, a estrutura do algoritmo é basicamente a mesma, a diferença entre eles está no momento da poda, onde uma versão fará a poda dos especialistas mais antigos e a outra podará os especialistas com menor peso.

```
//adiciona novo expert caso a previsão seja errada
double valorEsperado = instnc.classValue();
int indiceObtido = Utils.maxIndex(votes);
if(valorEsperado != indiceObtido){
    ExpertAddExp exp = new ExpertAddExp((Classifier) getPreparedClassOption(baseLearnerOption));
    double sumWeights = 0.0;
    for(ExpertAddExp e : experts){
        sumWeights += e.getWeight();
    }
    exp.setWeight(gammaOption.getValue() * sumWeights);
    experts.add(exp);
    System.out.println("ERREI");
}

//removendo expert mais antigo
if(experts.size() > maxExpertsOption.getValue()){
    //choose the expert to be removed
//removendo expert com menor peso
ExpertAddExp toRemove = experts.get(0);
    for(ExpertAddExp ite : experts){
        if(ite.getWeight() < toRemove.getWeight()){
            toRemove = ite;
        }
    }

    //remove it from the ensemble
    experts.remove(toRemove);
}
```

Figura 15. Remoção de Especialistas (*Weakest First e Oldest First*).

O *framework MOA* encontra-se disponível para download no site http://moa.cms.waikato.ac.nz/ e trata-se de um arquivo formato (.jar) que contém todos os algoritmos disponíveis para experimentos. Abrindo o arquivo jar do *MOA* no NetBeans, encontramos todos os pacotes onde estão organizados os algoritmos que são executados pela ferramenta. No pacote moa.classifiers foram adicionadas as classes *DWM* implementada por GOMES (2012) e *ADDExp.D,* implementada neste trabalho.

Considerações Finais

Neste capítulo, descrevemos todos os procedimentos metodológicos da preparação do conjunto de dados, das ferramentas e do algoritmo desenvolvido para as análises. Podemos ver que a etapa de preparação do conjunto de dados e a escolha da ferramenta de análise é fundamental para que se obtenha bons resultados.

Capítulo 4

Resultados e Análise

Neste capítulo são apresentados os resultados obtidos nos experimentos realizados. Nesse momento, nosso objetivo é analisar o comportamento geral dos algoritmos, comparando as taxas de acerto entre cada um nas diferentes janelas e customizações configuradas com o *framework MOA.* Além disso, será analisada a viabilidade da utilização de algoritmos baseados em conjuntos de classificadores, trabalhados em conjuntos de dados reais e relacionados a ocorrências policiais.

4.1 Média da Taxa de Acerto

Damos início à apresentação dos resultados obtidos com uma breve descrição dos processos realizados para adquiri-los. Apresentamos no Capítulo 3 deste trabalho todos os procedimentos relacionados a preparação do conjunto de dados, escolha do ambiente e implementação dos algoritmos. Vários ciclos foram realizados com os algoritmos baseados em conjuntos de classificadores, aqui apresentados, até que fosse encontrada a melhor parametrização da ferramenta. O Quadro 3 apresenta a média da taxa de acerto ao final da execução dos algoritmos, estes valores ficaram definidos como os melhores em todas as execuções.

Como estamos avaliando o comportamento de um algoritmo analisando uma *stream* de dados reais, escolhemos o método *Prequential* que trabalha com configuração *online* e apresenta boa performance quando submetido a grande volume de dados. Neste momento, trabalhamos com um conjunto de dados consideravelmente reduzido, no entanto, conforme visto anteriormente, uma média de 300 ocorrências distintas são geradas diariamente e este método trabalhando em tempo real nas entradas de dados de ocorrências seria o mais adequado. As etapas seguintes foram a escolha do algoritmo e do classificador base. A escolha pelos classificadores base foi feita com base em outros estudos que relatam uma boa adaptação destes em *streams* de dados reais, resultando em boa performance e bons resultados.

Para cada algoritmo foi parametrizado 5 classificadores base e executados em janelas de 3, 7 e 10 dias conforme apresentado no Quadro 2. As parametrizações dos classificadores base, permaneceram as *default*. Em cada uma das execuções, os parâmetros de β (*Beta*) que é a constante multiplicativa para decrementar o peso dos *experts* (Figura 13), γ (*Gamma*) que á a constante multiplicativa para o pesos de novos *experts*, que possuem como padrão 0,5 e a variável K que é a constante para definição de remoção de *experts*, foram configuradas com diferentes parâmetros até que fosse possível encontrar a melhor execução para o conjunto de dados propostos. O Quadro 3 apresenta a melhor média da taxa de acerto encontrada durante as simulações.

Quadro 3. Média da Taxa de Acerto dos algoritmos.

			Classificador Base (% média de acerto)				
Algoritmo		**Tamanho Janela (dias)**	**NB**	**HT**	**LB**	**OZA**	**ASHT**
ADD	**WEAKEST**	**3**	68,18	68,18	59,71	**68,88**	68,18
		7	67,77	67,77	59,07	**68,39**	67,77
		10	68,36	68,36	59,68	**68,99**	68,36
	OLDEST	**3**	84,77	84,77	76,2	**87,19**	84,77
		7	84,48	84,48	75,76	**86,90**	84,48
		10	84,24	84,24	75,45	**86,73**	84,24
DWM		**3**	**69,36**	57,08	60,6	56,47	57,08
		7	**68,9**	56,7	60,13	56,13	56,7
		10	**69,48**	57,01	60,68	56,4	57,01
LEVBAG		**3**	**60,97**	**60,97**	58,92	57,38	**60,97**
		7	**60,31**	**60,31**	58,3	56,63	**60,31**
		10	**60,92**	**60,92**	58,92	56,89	**60,92**
ADWIN		**3**	**67,92**	**67,92**	58,36	58,85	**67,92**
		7	**67,39**	**67,39**	57,66	58,32	**67,39**
		10	**67,92**	**67,92**	57,86	58,48	**67,92**

No Gráfico 2, apresentamos somente a execução com o algoritmo *ADDExp.D* método *Oldest First* com o classificador base *OzaBagAdwin*. O Quadro 3, coluna **OZA**, valores apresentados em vermelho, servem como referência para a construção do gráfico abaixo.

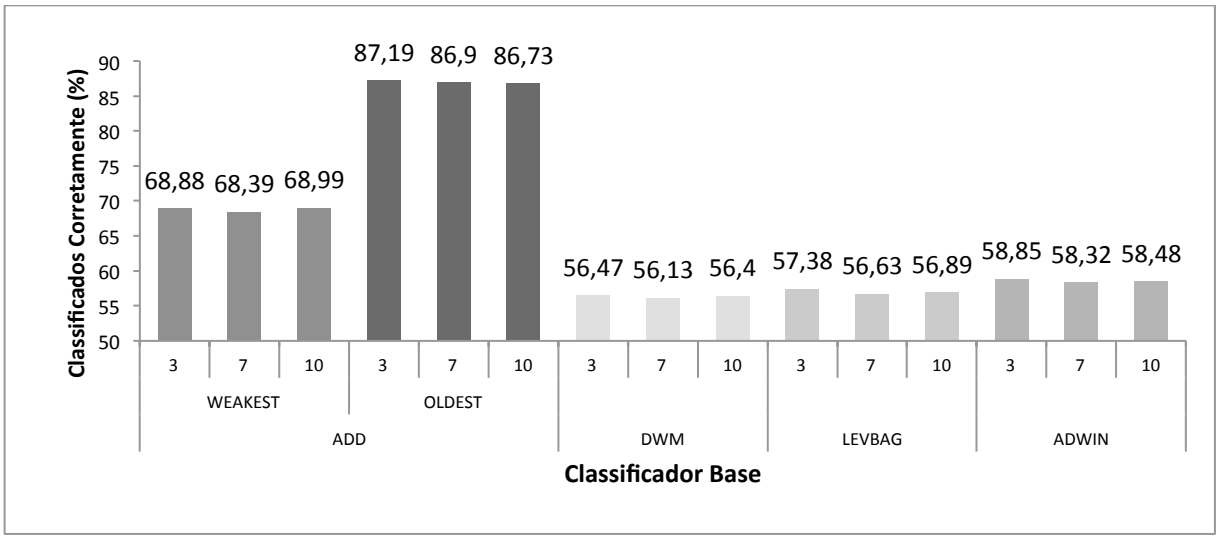

Gráfico 2. Média da taxa de acerto do algoritmo ADDExp.D com classificador OzaBagAdwin.

O algoritmo *ADDExp.D* com o classificador base *OzaBagAdwin* apresentou resultados mais altos quando comparado com os demais classificadores base.

Outra análise realizada com o *ADDExp.D* utilizando como algoritmo classificador o *OzaBagAdwin* foi o aumento da janela de dias. Nos testes foram parametrizadas além das janelas de 3, 7, e 10 dias previstas para o desenvolvimento do trabalho, e que tem relação direta com os processos atualmente realizados pela PMSC, e janelas de 15, 20 e 30 dias, para que pudéssemos analisar o comportamento do algoritmo com janelas de tempo maior, porém, estas últimas 3 janelas escolhidas somente para acompanhar os resultados apresentados em períodos de tempo maior, sem intuito de ser analisado de uma forma mais aprofundada, tendo em vista que estes períodos fogem dos padrões realizados pela PMSC. O Quadro 6 e o Gráfico 3 apresentam os valores obtidos e podemos ver que o algoritmo vai se tornando menos eficiente quando as janelas aumentam. Podemos explicar este comportamento pelo fato de que a criminalidade não tem um padrão específico para acontecer e que períodos com diferença de tempo muito grande podem ter sido influenciados por diversos fatores externos que podem contribuir significativamente para a mudança deste padrão.

Quadro 4. Média de acerto do algoritmo com janelas de diferentes tamanhos.

ADDExp	**OLDEST FIRST**	**3 dias**	**87,19**
		7 dias	**86,9**
		10 dias	**86,73**
		15 dias	**86,41**
		20 dias	**85,36**
		30 dias	**85,28**

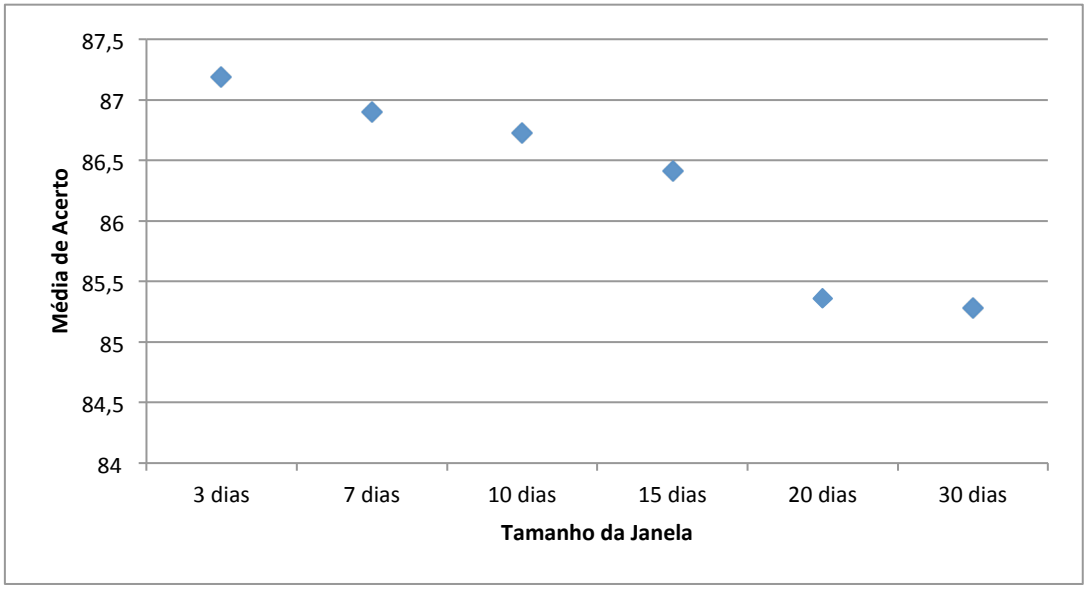

Gráfico 3. Média de acerto do algoritmo com janelas de diferentes tamanhos

Após a análise dos valores obtidos, identificamos que o conhecimento empírico sobre a criminalidade é comprovado com os números apresentados, ou seja, para os especialistas em Segurança Pública, existe uma maior probabilidade de se obter sucesso em operações, quando utilizam ocorrências relativas a um período curto de tempo, como por exemplo, 3, 7 e 10 dias, do que utilizar dados de crimes ocorridos em um espaço de tempo mais antigo, como por exemplo, um período acima de 20 dias. Na prática, as informações mais utilizadas são de períodos de até 10 dias e o que for acima desse período é utilizado somente como dado estatístico para identificar quantidade ocorrida em determinadas áreas.

OzaBagADWIN é um método *bagging online* de OZA e RUSSEL, que contribuiu para os resultados apresentados. Atuou como um reforço do algoritmo *ADDExp.D* em seus dois métodos de poda (*Oldest First* e *Weakest First*). Sua contribuição se deve ao fato de que este método trabalha fazendo um sorteio de valores aleatórios para cada instância determinando se a instância será ou não utilizada para treinamento de cada classificador do conjunto. É importante lembrar que a reposição da reamostragem do conjunto é feita utilizando a distribuição de Poisson[6], sendo usada para modelar o número de eventos que ocorrem dentro de um determinado intervalo de tempo, ou seja, ela expressa a probabilidade de uma série de eventos ocorrer num certo período de tempo se eles ocorreram independentemente de quando ocorreu o último evento.

[6] Distribuição de Probabilidade de variável aleatória discreta que expressa a probabilidade de uma série de eventos ocorrer num certo período de tempo.

Com o conjunto de dados analisado, o algoritmo *ADDExp.D* se adapta rapidamente a mudanças de conceitos. No Gráfico 4, podemos ver que, para todos os classificadores base analisados, a janela de 3 dias foi a que apresentou melhores resultados com relação as demais janelas analisadas. Desta forma, podemos dizer que, determinados tipos de crimes possuem um ciclo temporal de acontecimento. Observando por um lado mais próximo da realidade da PMSC (8º BPM), os criminosos praticam este tipo de crime com um cuidado maior que os demais tipos de crime, tendo em vista que não é um crime realizado baseado na oportunidade, mas sim que exige um certo planejamento para que tudo saia como o esperado na visão dos criminosos.

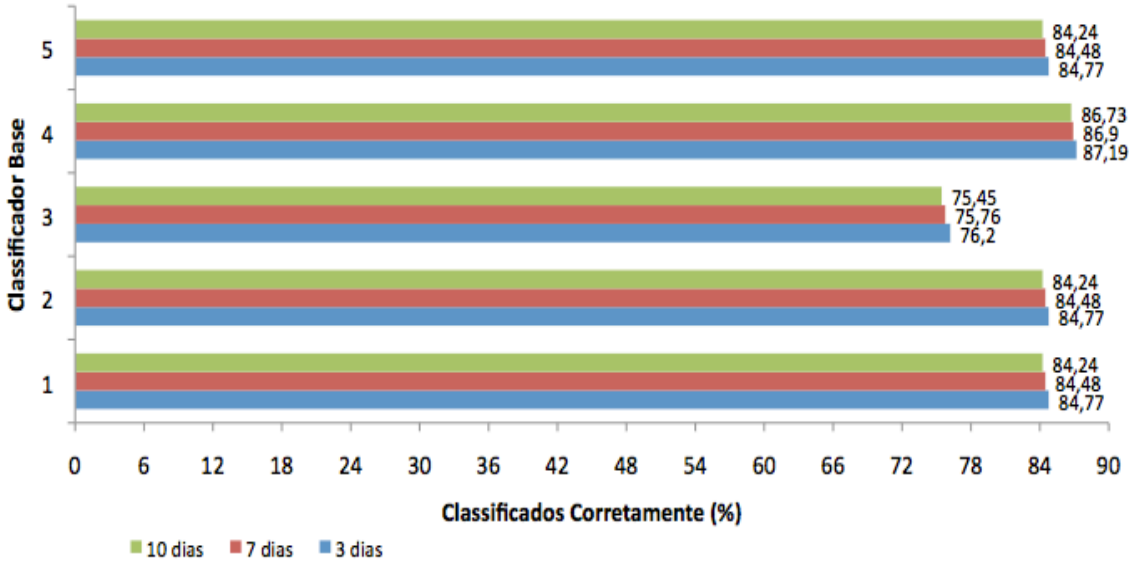

Gráfico 4. ***ADDExp.D Oldest First***

Legenda Classificadores
1 – *Naive Bayes*
2 – *Hoeffding Tree*
3 – *Leveraging Bag*
4 – *Ozabag ADWIN*
5 – *ASHoeffding Tree*

Somente o classificador *Leveraging Bag* não apresentou resultados acima de 80% de classificação correta das instâncias. Em GOMES e ENEMBRECK (2013) e BARDDAL, GOMES e ENEMBRECK (2014), existem relatos de estudos onde o algoritmo *Leveraging Bag* obteve uma taxa de acerto muito abaixo dos outros algoritmos quando os conjuntos de dados não possuíam *drifts*. Tanto o resultado apresentado neste trabalho quanto os resultados apresentados pelos referidos autores levam a acreditar que o algoritmo *Leveraging Bag* não é indicado para conjuntos de dados sem *Drift.*

É possível dizer que os resultados obtidos foram vistos como relevantes pelos especialistas da Segurança Pública, uma vez que gerenciar a criminalidade em curtos períodos de tempo é um processo mais eficaz que deixar para gerenciá-los em períodos acima de 20 dias. Podemos afirmar que, a criminalidade possui uma alteração muito grande quando comparamos um mês e outro, uma vez que assim como o dinheiro muda de mãos (consumidores para lojistas), o foco dos criminosos também muda entre pessoas, comércios e residências.

4.2 Análise do Algoritmo

Para que pudéssemos analisar todos os resultados com o máximo de embasamento das situações ocorridas, obtivemos no site do INMEP – Instituto Nacional de Meteorologia[7], registros das chuvas ocorridas durante o primeiro semestre do ano de 2010, período de nossa análise, em número de dias (Quadro 7 e Gráfico 5). Convém relatar que estes números são inerentes a quantidade de dias que choveu no mês, não sendo possível apontar com precisão o dia do mês que o fato ocorreu. De acordo com os especialistas da Segurança Pública, fatores como clima tem forte incidência na variação das tipificações criminais, ou seja, para eles, existe uma diminuição no registro destes tipos de crimes, principalmente nos crimes de roubo contra pessoas pelo fato de que o movimento de pessoas pela via pública nestes dias apresenta uma forte queda.

Quadro 5. Dias com chuva em Joinville.

Mês	N. Dias com Chuva
Janeiro	15
Fevereiro	18
Março	22
Abril	17
Maio	13
Junho	8
Julho	11

Fonte: INMEP (2013).

[7] www.inmep.gov.br

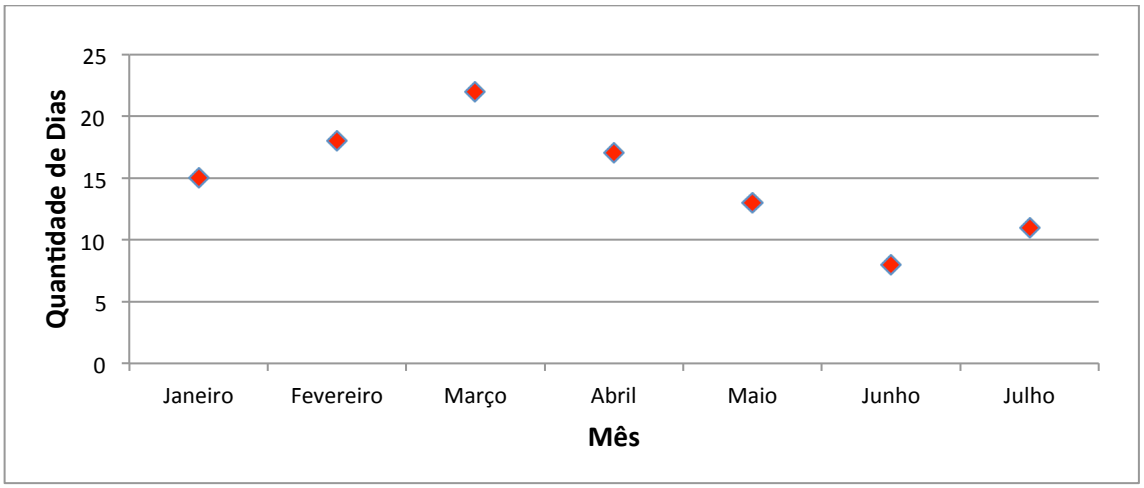

Gráfico 5. Número de dias com chuva em Joinville.

Com base nos dados apresentados, podemos ver que o mês de Março de 2010 foi o mais chuvoso de todo o período analisado. Vamos procurar relacionar os resultados apresentados pela ferramenta com estes dados para tentar confirmar se os resultados condizem com as informações empíricas apresentadas pelos especialistas em Segurança Pública.

4.2.1 Resultados Algoritmo *ADDExp.D*

Como apresentado no Quadro 3, duas implementações foram feitas para o algoritmo *ADDExp.D* e iniciaremos abordando o método *Weakest First* que teve definido como parâmetro 0,143 para o β (*Beta*) e 0,136 para o γ (*Gamma*) e tamanho da janela variando em 3, 7 e 10 dias.

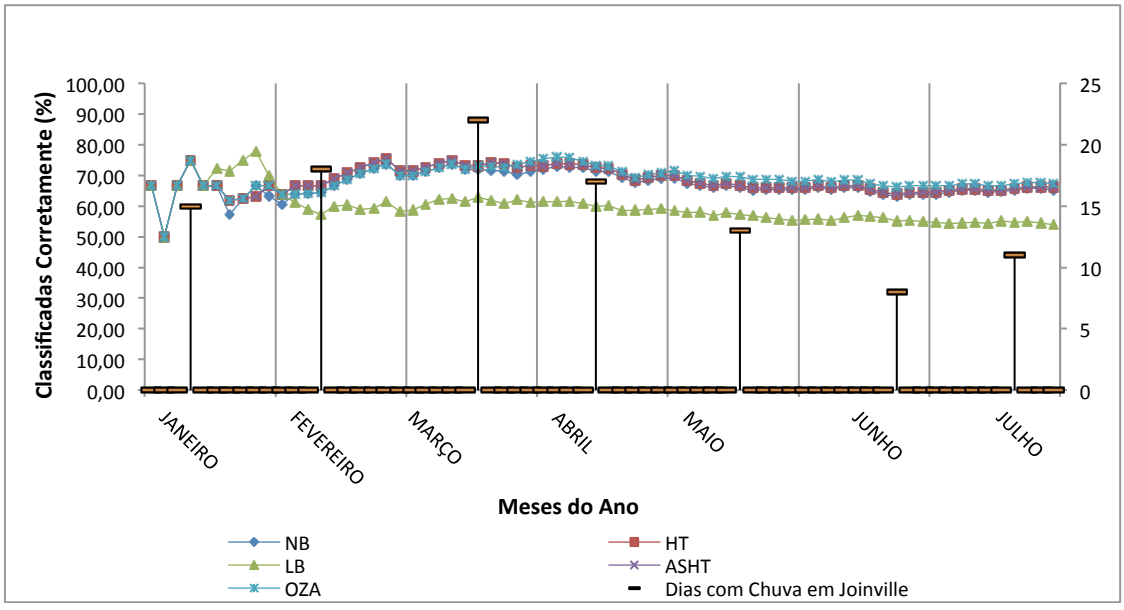

Gráfico 6. ADDExp.D - Weakest First – 3 dias

Apresentamos nos Gráficos 6, 7 e 8 os resultados do algoritmo *ADDExp.D* executado com os 5 classificadores base. Consideramos importante acrescentar neste gráfico, a quantidade de dias que choveu e o mês a que cada período está caracterizando. Para o período analisado, não comprovamos o relacionamento apresentado pelos especialistas em Segurança Pública de que, nos períodos de chuva na região, a criminalidade diminui. Analisando os resultados, vemos que a presença da chuva não influenciou para reduzir ou aumentar os índices estudados.

O classificador base *Leveraging Bag* apenas no mês de janeiro onde atingiu 78% de instâncias classificadas corretamente, ou seja, enquanto o conjunto de dados ainda apresentava *drifts* ele obteve uma taxa de acertos eficiente, porém, quando o conjunto estabilizou, seus resultados apresentaram um *Drift gradual*. Percebemos um *Drift* abrupto entre o fim do mês de janeiro e início do mês de fevereiro, esta mudança de conceito pode estar relacionada com algum evento que tenha ocorrido na região. Vemos no Gráfico 1, na classificação das instâncias entre 31 e 40 que todos os índices apresentaram baixa. Verificando no calendário anual, constatamos que o período em questão foi comemorado o carnaval e como todo período de feriado prolongado, as pessoas tendem a viajar ficando alguns dias fora de casa e isso acaba refletindo nas ocorrências analisadas.

O classificador base *ASHT* (*Adaptive Size Hoeffding Tree*) apresenta *drifts* graduais partindo de 66% indo a mais de 75% de taxas de acerto no mês de fevereiro e depois outros *drifts graduais* fizeram com que ele estabilizasse em 60% no final do período. Como as árvores são frequentemente reiniciadas, este comportamento não influencia negativamente a capacidade de classificação do conjunto devido ao peso atribuído a cada árvore. Analisando a estabilização das taxas de acerto em um curto período de tempo, podemos ter a ideia da criação de uma árvore pequena pela sua fácil adaptação aos novos conceitos, tornando as classificações seguintes bem estáveis e com pouca existência de *drifts*, ou seja, os fatos foram recorrentes.

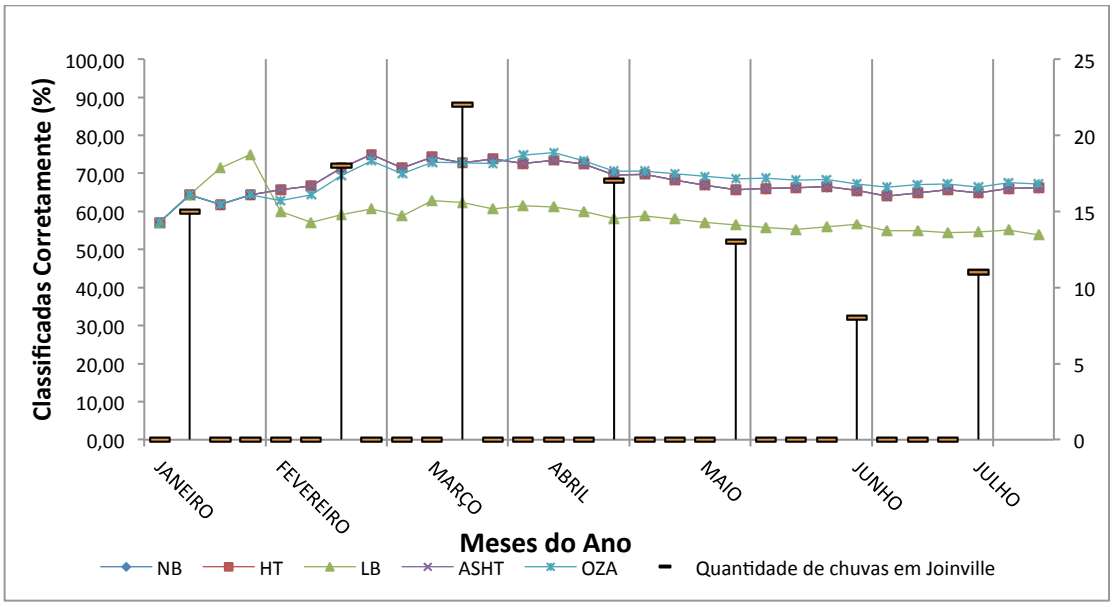

Gráfico 7. ADDExp.D - Weakest First - 7 dias

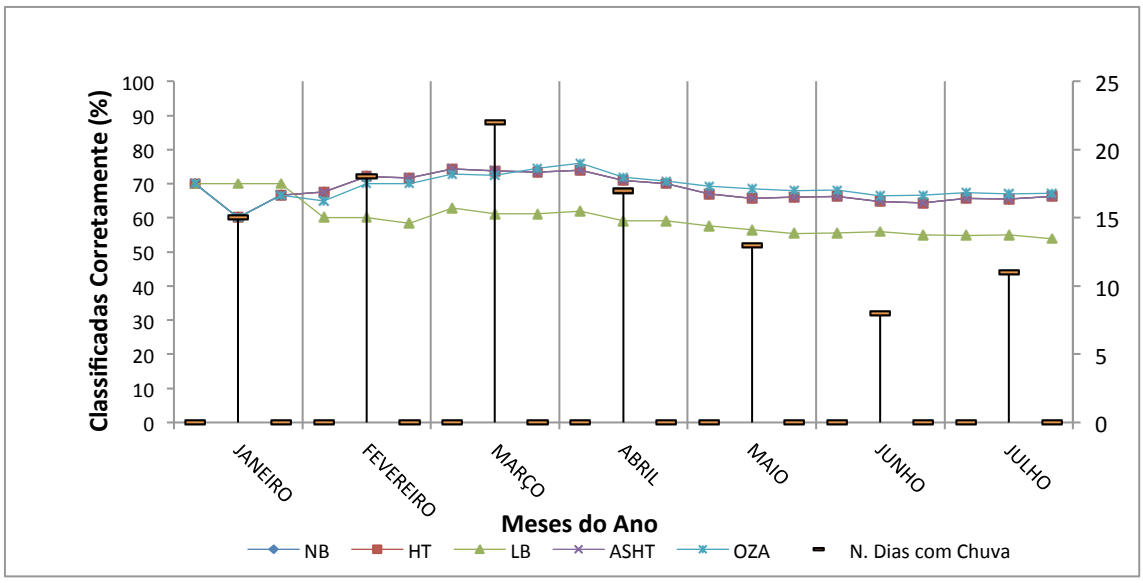

Gráfico 8. ADDExp.D - Weakest First - 10 dias

O algoritmo *ADDExp.D* com o método *Weakest First* executando em diferentes tamanhos de janelas (3, 7 e 10 dias) apresentou resultados bem próximos em todos os classificadores base trabalhados, com exceção do *Leveraging Bag* que apresentou em todas as análises resultados abaixo dos demais, porém acima de 60% de classificações corretas, convém citar novamente GOMES e ENEMBRECK (2013) e BARDDAL, GOMES e ENEMBRECK (2014) para entendermos este comportamento.

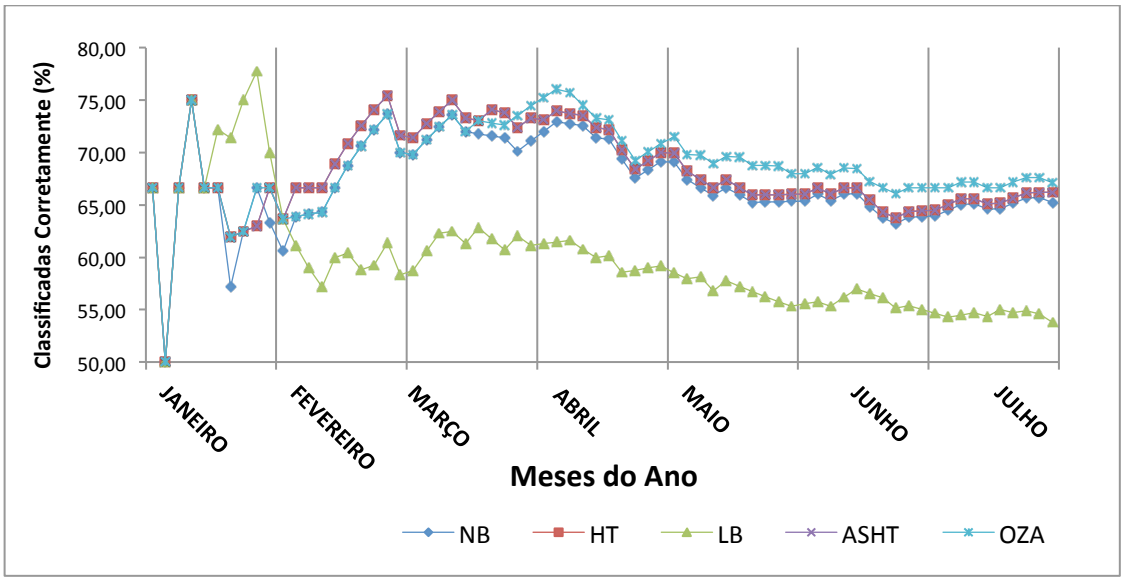

Gráfico 9. Média de acertos - ADDExp.D - Weakest First - 3 dias

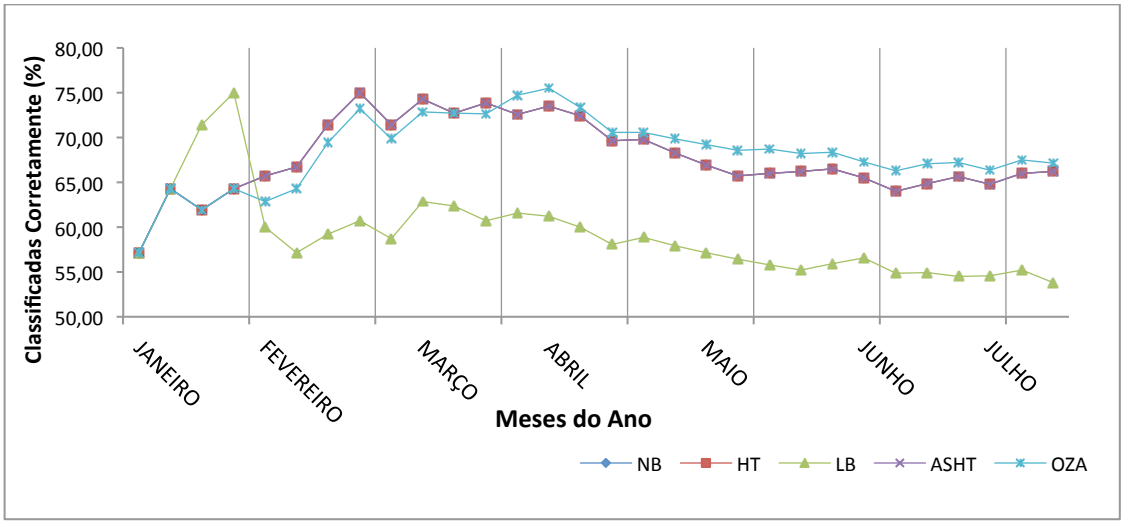

Gráfico 10. Média de acertos - ADDExp.D - 7 dias

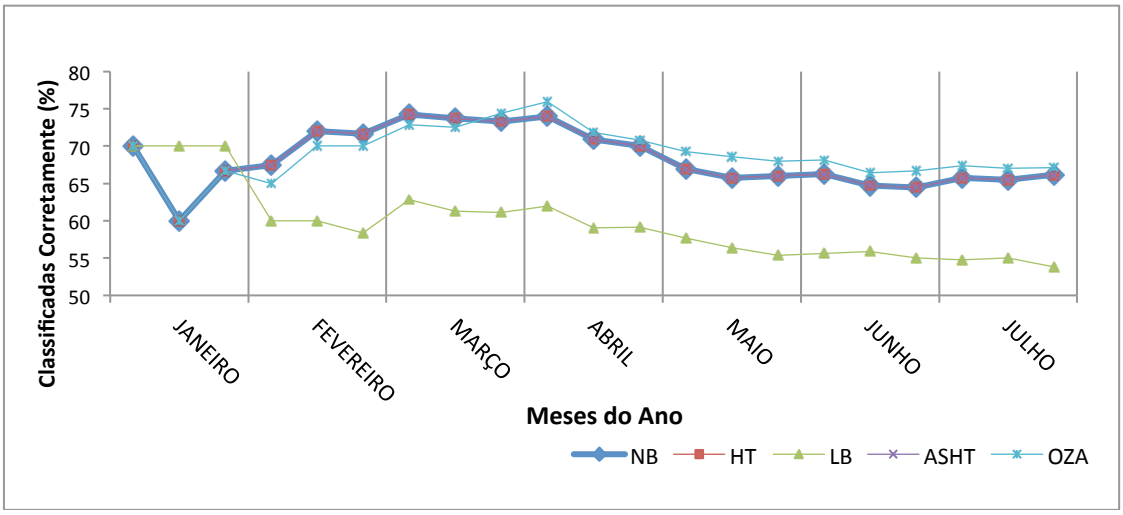

Gráfico 11. Média de acertos - ADDExp.D - 10 dias

Analisando os Gráficos 9, 10 e 11, podemos ver que a remoção de especialistas com menor peso tende a deixar o conjunto mais instável surgindo *drifts abruptos* quando trabalha com janelas de 3 dias, porém, constatamos que esta instabilidade reduz bastante conforme ocorre o aumento da janela surgindo então *drifts graduais*. Como os algoritmos base possuem funcionamento bem semelhante uns com os outros, o resultado apresentado tem bastante proximidade tornando este um fator positivo do estudo.

Depois que os resultados foram analisados pelos especialistas em Segurança Pública conseguimos obter informações suficientes para comprovar a viabilidade do uso destes algoritmos em conjuntos de dados que envolvam dados da Polícia Militar em específico os dados relacionados a roubos e assaltos contra pessoas, estabelecimentos e residências. É uma outra forma de ver os resultados que até então não haviam sido analisados por ferramentas específicas e nem haviam sido submetidos a outros tipos de análises e tratamentos.

Outro ponto interessante a ser considerado é quanto a quantidade de dias com chuva em um determinado mês, este fator não foi decisivo para uma melhor classificação nas taxas de acerto, ou seja, fatores como chuva não influenciam este tipo de ação delituosa. O método *Weakest First* mantém a classificação estável e o método *Oldest First* em alguns pontos como por exemplo nos meses de abril e maio até diminuem justamente quando o índice de chuvas também se torna baixo. De uma outra visão, temos dois feriados bem próximos sendo o primeiro do dia 21 de abril, momento em que o conjunto apresenta um *Drift* e o segundo *Drift* no dia 01 de maio.

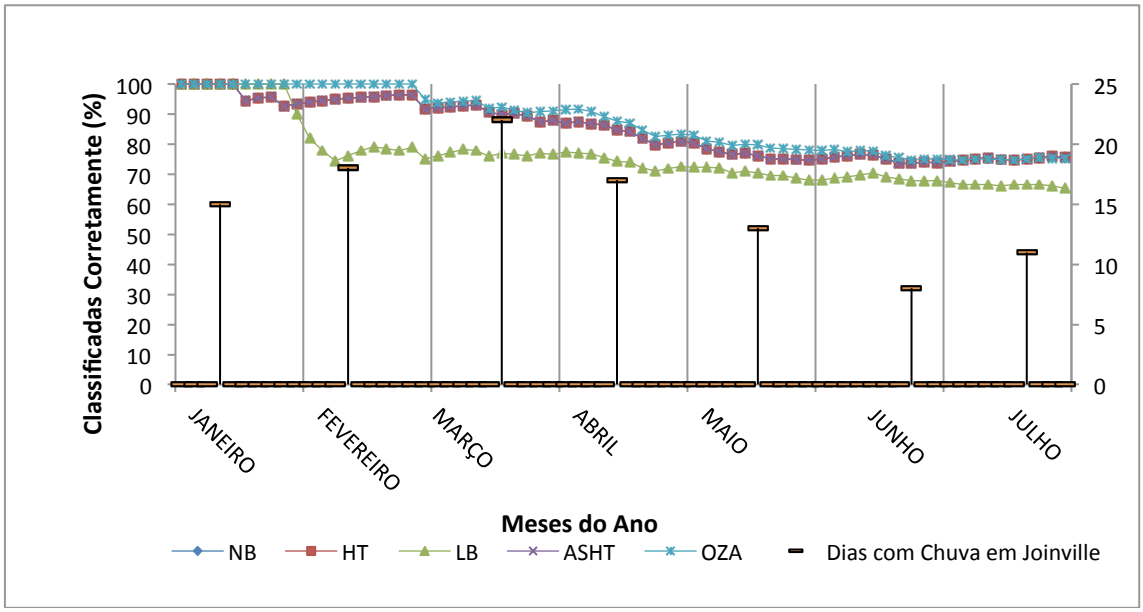

Gráfico 12. *ADDExp.D - Oldest First* - 3 dias

De uma forma geral, os resultados apresentados através do método *Oldest First* (remoção dos classificadores mais velhos) são bem superiores quando comparados com os resultados obtidos através do método *Weakest First*. Os classificadores *Naive Bayes, Hoeffding Tree* e *Adaptive Size Hoeffding Tree* apresentaram a mesma média de 84,77%, *Leveraging Bag* não só nessa execução mas em todas as outras realizadas, apresentou resultados abaixo de 80%, ou seja, quando o conjunto de dados é considerado "estacionário" (sem *Drift*) o algoritmo obtém resultados muito baixos.

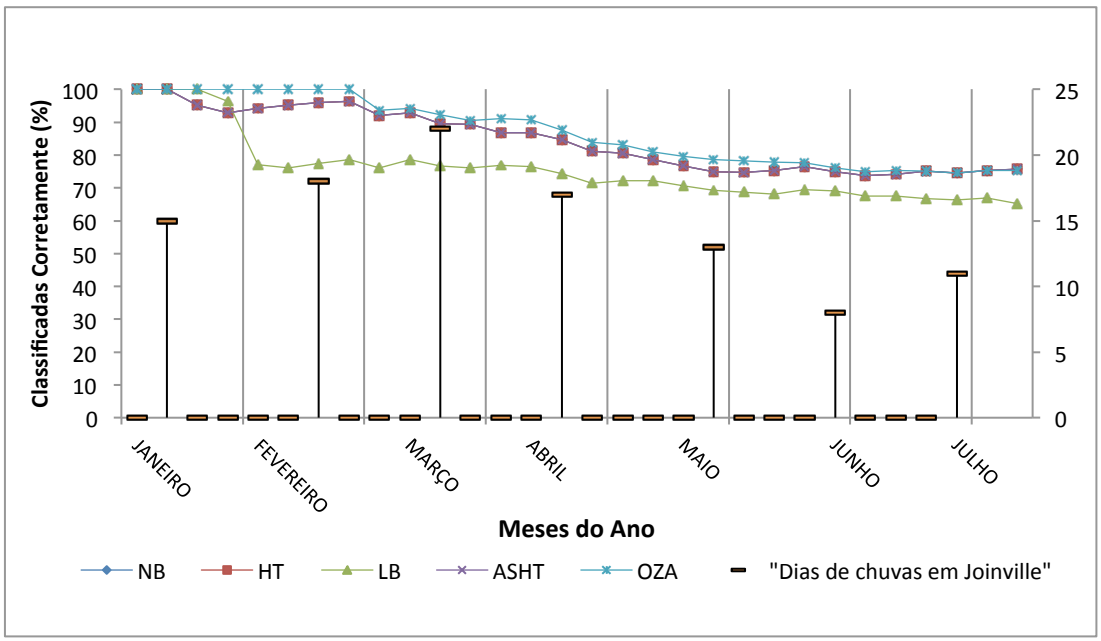

Gráfico 13. *ADDExp.D - Oldest First* - 7 dias

Drifts graduais são apresentados a partir de aproximadamente 70 instâncias (terceiro mês) do conjunto de dados, a ocorrência deles pode estar relacionada com o método *Oldest First* uma vez que conceitos antigos não estão ficando armazenados, mas sim, novos conceitos estão sendo inseridos no conjunto. Podemos considerar que isto é importante com relação aos dados analisados, pois independente do conceito anteriormente passado, novos tipo de ocorrências estão sendo geradas pela CRE190 e com isso, novos conceitos vem surgindo e que até então não se tinha o conhecimento.

Segundo os especialistas em Segurança Pública, na atividade prática, o *modus operandi* do indivíduo vem mudando e tentando se adaptar a novos padrões tanto de segurança em que os proprietários dos estabelecimentos ou residências instalam para tentar coibir a criminalidade, como na consciência de que ela está presente a qualquer momento pelas pessoas. Comportamentos simples das pessoas, como por exemplo, evitar em ir a determinados tipos de caixas eletrônicos que não estão estrategicamente instalados ou seja, caixas eletrônicos que a partir de determinados horários do dia ou perí-

odos da semana ficam praticamente abandonados sem nenhuma movimentação de pessoas. Isso torna o local um ponto interessante para se fazer novas vítimas e então possa ocorrer um roubo ou assalto contra pessoa com subtração de veículos, dinheiro ou outros bens.

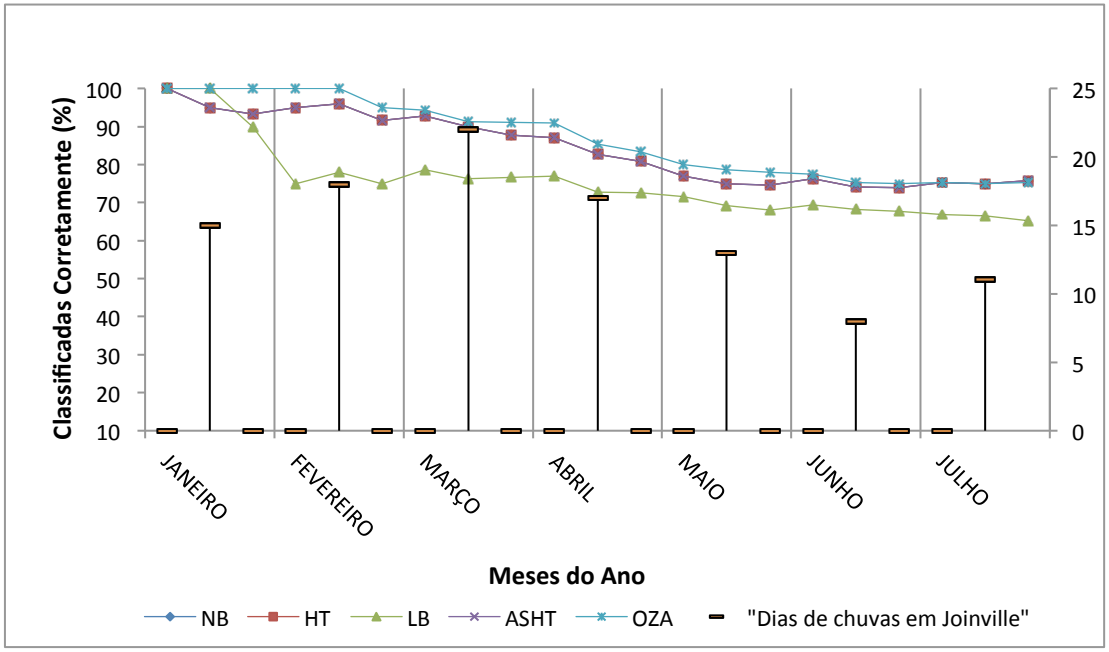

Gráfico 14. ***ADDExp.D - Oldest First*** **- 10 dias**

Fica claro para nós que, as implementações aqui apresentadas do *ADDExp.D – Oldest e Weakest First* obtiveram um bom comportamento quando submetidas a análise de conjuntos de dados reais em especial com dados relacionados à Segurança Pública.

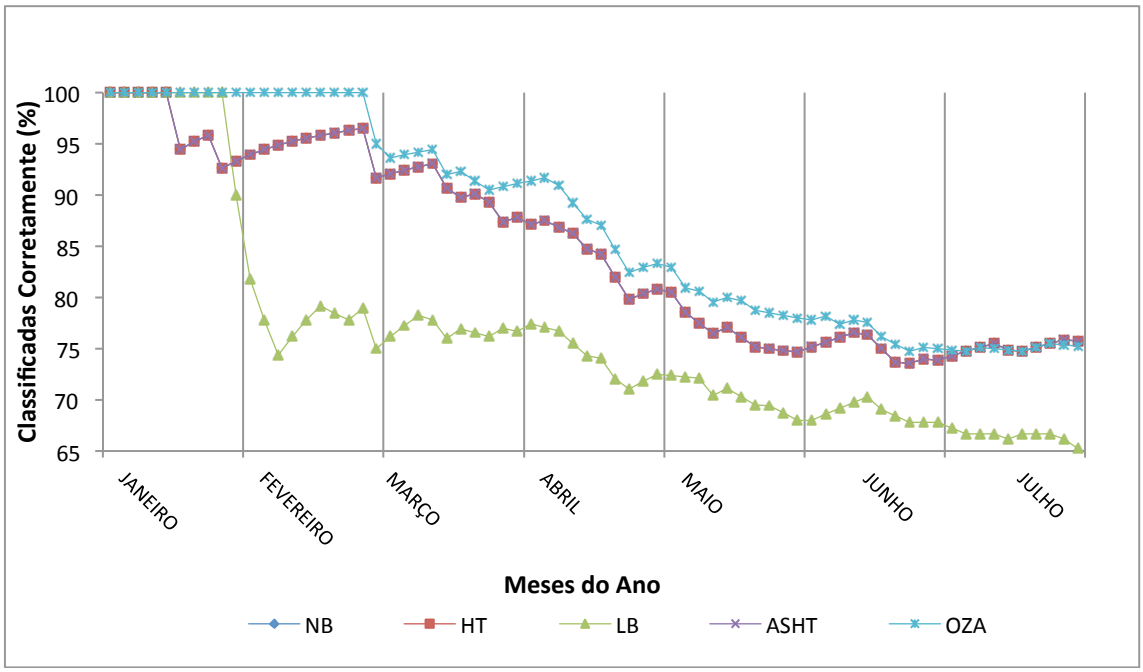

Gráfico 15. ***ADDExp.D - Oldest First*** **- 3 dias**

Em uma visão mais aproximada das taxas de acerto do algoritmo com os classificadores base vemos que *Hoeffding Tree, Adaptive Size Hoeffding Tree* e o *Naive Bayes* (Gráfico 15) se adaptam lentamente às mudanças de conceito no conjunto de dados, apesar de serem classificadores *online.* Este fato pode estar relacionado ao tamanho do conjunto de dados analisado. Vemos que, em determinados momentos, existem *Drifts* abruptos e que logo na sequencia alguns *Drifts* moderados, porém, o *Leveraging Bag* apresenta o *Drift* abrupto ao final do primeiro mês, seguindo em uma média de taxa de acerto de aproximadamente 70%.

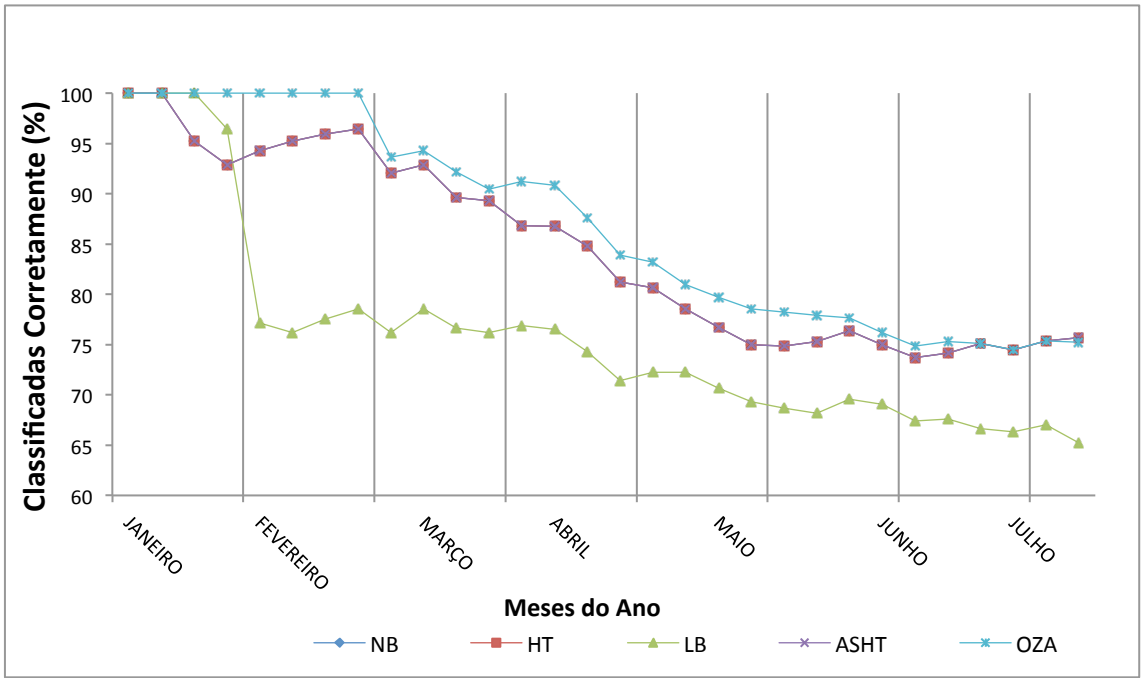

Gráfico 16. *ADDExp.D - Oldest First* - 7 dias

Comparando a classificação dos valores das classes apresentadas no Gráfico 1 onde 84% das primeiras 56 instâncias estavam nas faixas baixa e média com os resultados obtidos com os algoritmos (Gráficos 16 e 17) nos primeiros 8 rounds de 7 dias, onde a classificação foi 100% correta, podemos constatar que a ausência de *Drift* pode estar relacionado a estabilidade do conjunto. Conforme a quantidade de instâncias aumenta entre as faixas média e alta, a porcentagem de instâncias classificadas corretamente apresenta um *Drift* gradual vai sendo apresentado até estabiliza com aproximadamente 75%. Somente para o classificador *Leveraging Bag* que ocorre um *Drift* abrupto no segundo round, mantendo um *Drift* moderado durante boa parte da execução e outro *Drift* gradual ao final da execução. Este classificador apresentou os níveis mais baixos de classificação das instâncias.

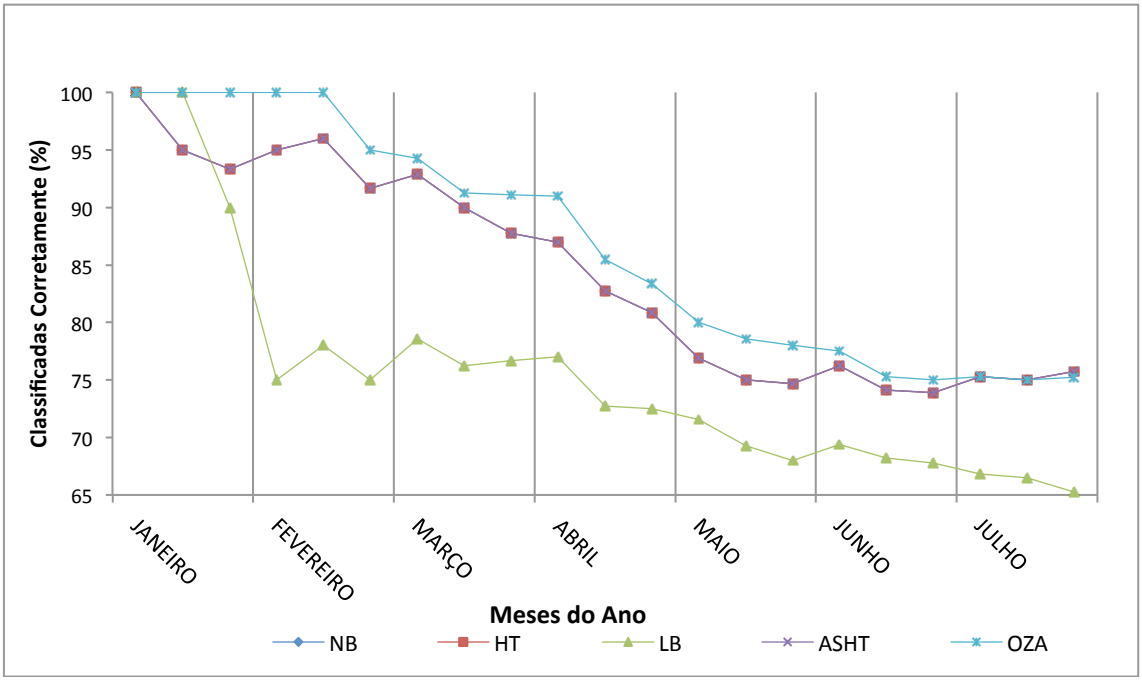

Gráfico 17. ADDExpert.D - Oldest First - 10 dias

Os resultados apresentados pela aplicação, evidenciam a importância da existência de uma ferramenta de apoio à decisão que possa analisar ocorrências policiais de uma determinada região e que estejam relacionadas a um determinado tipo de crime.

Outros resultados que foram obtidos com as análises dos dados:

a) O algoritmo *DWM* apresentou seus melhores resultados quando executado com o Naive Bayes, atingindo uma média de 69%, isso significa que o Naive Bayes se recuperou melhor à mudança que os demais classificadores base;

b) Os classificadores *OzaBagADWIN, Hoeffding Tree, Leveraging Bag* e *Adaptive Size Hoeffding Tree* não apresentaram bons resultados quando executados em conjunto com o algoritmo *DWM* (Quadro 3. Média da Taxa de Acerto dos algoritmos.Quadro 5)*;*

c) *Leveraging Bag* e *ADWIN* apresentaram resultados muito abaixo dos demais analisados e portanto, podem não ser a melhor opção para analisar conjuntos de dados reais que envolvam a criminalidade de uma região.

Com os resultados obtidos, podemos afirmar que a melhor opção para trabalhar em conjuntos de dados reais que envolvam a criminalidade de uma região mais especificamente crimes de roubo ou assalto contra pessoas, estabelecimentos comerciais e residências é o algoritmo *ADDExpert.D* tendo como classificador base o algoritmo

OzaBagADWIN com o algoritmo padrão *Hoeffding Tree.* Muitos estudos podem evoluir a partir deste, tendo em vista a quantidade de tipos de ocorrências que a Polícia Militar possui em sua base de dados.

Outras técnicas ainda podem ser exploradas para que novos conhecimentos sejam obtidos e possivelmente, agir de forma positiva tanto para a sociedade pesquisadora quanto para a própria Segurança Pública. Os resultados aqui apresentados, foram confrontados com as práticas atualmente realizadas pela Polícia Militar de Joinville e muita similaridade foi encontrada com os resultados e o existente, portanto, podemos afirmar que este procedimento se torna útil para ser aplicado como ferramenta de apoio à Segurança Pública, mais especificamente ao 8º Batalhão de Polícia Militar de Santa Catarina.

Apesar de ser um conhecimento que até então sugeria-se que fosse acontecer, o aumento da criminalidade em períodos pós chuva, nossos resultados mostraram que os algoritmos não levaram em consideração esta informação, ou seja, como a criminalidade está sempre mudando seus indivíduos onde velhos criminosos deixam a prisão e surgem também novos criminosos a todo momento, podemos dizer que eles praticarão os crimes indiferente de tempo, hora, dia ou outro item que até então considerássemos relevante.

Pelo conhecimento adquirido pelos policiais militares, sabe-se que no início do mês, crimes de roubos contra pessoa aumentam significantemente e roubos contra empresas também aumentam, onde o foco principal dos ladrões são malotes de pagamentos. Passados os dias de pagamento das empresas, tende a aumentar os índices de roubos contra estabelecimentos comerciais uma vez que, teoricamente, o dinheiro está mudando de mãos. Já para os roubos a residências, não existe um período de ocorrências mais frequentes, podendo ocorrer tanto no início do mês quanto no final dele, pois os objetos dos roubos à residências, em sua grande maioria, são jóias, eletrônicos e veículos. Com este conhecimento disponibilizado aos gestores, será possível trabalhar com um melhor planejamento das operações, distribuindo de uma forma mais eficiente os policiais militares participantes, assim como também o melhor uso dos meios disponíveis para o policiamento e proposição de políticas publicas de segurança adequadas.

Capítulo 5

Considerações Finais

Considerando a carência dos órgãos da Segurança Pública em relação à existência de ferramentas adequadas para a prevenção da criminalidade, buscamos nessa pesquisa, implementar técnicas até então não implementadas em conjuntos de classificadores relacionados à Crimes e Contravenções. O objetivo desta pesquisa, é incentivar que novos trabalhos sejam desenvolvidos tendo esta pesquisa como um modelo de referência para que novos modelos sejam implementados utilizando algoritmos baseados em conjuntos de classificadores. Desta forma, buscaremos estar um passo à frente de situações criminais que até então são resolvidas de uma forma muito genérica e empírica.

Diversas técnicas foram apresentadas neste trabalho e um estudo foi desenvolvido com base em dados reais de ocorrências, buscamos mostrar a possibilidade de implementação destes modelos. A detecção de mudanças torna-se essencial para trabalharmos com dados de ocorrências *online* uma vez que, o melhor uso do efetivo e dos recursos pode ser feito com base nos resultados fornecidos pela ferramenta.

Com as análises realizadas, vemos que os algoritmos baseados em conjuntos de classificadores trabalharam de maneira adequada quando submetidos a análises de bases de dados reais. Todo o processo de análise do conjunto de dados foi realizado com a maior proximidade possível do que hoje é realizado para a obtenção de resultados coerentes que possam resultar em operações realizadas pela Polícia Militar. Foi observado que as técnicas de *Drift Detection* apresentaram resultados satisfatórios quando confrontados com os resultados hoje existentes e trabalhados pela corporação. Por outro lado, podemos ver que, mudanças devem ser feitas no sistema hoje existente na CRE190 para que as técnicas aqui propostas possam ser implementadas tendo em vista a dificuldade de se conseguir extrair e manipular dados no sistema atual e que estejam de acordo com as entradas necessárias para que novas análises sejam realizadas. Estudos vem sendo elaborados para que alterações no sistema sejam realizadas, porém, como trata-se de uma instituição pública, diversos fatores influenciam mudanças desse porte, como por

exemplo, questões de centralização de todos os registros realizados nos aproximadamente 240 municípios do estado.

As maiores dificuldades encontradas no desenvolvimento deste trabalho, estão relacionadas a preparação dos dados para a realização dos experimentos, uma vez que, todo o processo de seleção de dados teve que ser feito manualmente. Todas as ocorrências tiveram que ser lidas para que outros detalhes importantes para os experimentos fossem extraídos e, questões de cunho pessoal e profissional no que diz respeito ao entendimento de cada uma das pessoas que atuam na CRE190 no fechamento destas ocorrências, tendo em vista que o que pode ser considerado um roubo contra pessoa para um profissional, pode ser considerado furto contra pessoa para outra e isso apresenta uma grande interferência nos resultados finais dos experimentos.

Vemos que, cada vez mais, as ferramentas tecnológicas podem atuar apoiando também órgãos da Segurança Pública, porém, muito esforço no que diz respeito a atualização dos parques tecnológicos destes órgãos deve ser realizado, uma vez que, ferramentas para este fim exigem equipamentos com boa capacidade de processamento para que problemas não surjam durante a execução dos experimentos. Também exige a atuação de profissionais com conhecimento adequado para atuar com as técnicas aqui propostas, ou seja, para que estas técnicas possam ser utilizadas de forma positiva pela corporação em questão, muitos paradigmas devem ser quebrados de forma que se atinja resultados positivos tanto para a Segurança Pública quanto para a Sociedade em que ela está inserida.

Todos os objetivos propostos no trabalho foram alcançados, como por exemplo a implementação do algoritmo *ADDExpert.D* e a análise comparativa entre ele e demais algoritmos baseados em conjunto de classificadores. Por outro lado, novas análises podem ser realizadas com uma quantidade maior de instâncias e também com outros grupos de ocorrências como por exemplo brigas, acidentes de trânsito, furtos de veículos entre outros tipos de ocorrências geradas pela CRE190.

Outras técnicas também podem ser estudadas e propostas para trabalhos futuros assim como também outros algoritmos buscando resultados ainda mais satisfatórios e que possam ser utilizados de forma positiva pela instituição. Muito se tem estudado sobre a criminalidade em si, porém, pouco se tem trabalhado no que diz respeito a extração do conhecimento nos registros pela instituição mantidos. Uma resposta para a pequena quantidade de estudos na área proposta deve-se ao fato de que, uma baixa porcentagem de pessoas dentro da corporação possui conhecimento suficiente em ferramentas

tecnológicas para conseguir concluir projetos voltados a áreas de tecnologia. Outro fator importante, é a existência de pessoas que apoiem a iniciativa e queiram levar o projeto adiante, colaborando positivamente para a sua conclusão.

Muito ainda pode ser estudado em trabalhos futuros como por exemplo a utilização das técnicas aqui propostas para conseguir mapear perfis criminosos com base nas informações armazenadas na base de dados de ocorrências policiais geradas pela CRE190. Trabalhos voltados para outros tipos de emergências também podem ser desenvolvidos tendo em vista que, ocorrências que envolvam mau súbito em pessoas também são geradas e mantidas nesse ambiente, ou seja, trabalhos que envolvam problemas de saúde da população também podem ser elaborados e trabalhados utilizando as mesmas técnicas aqui propostas.

Este trabalho, e os resultados apresentados, podem contribuir positivamente para a área da Segurança Pública, que até então não possuía pesquisas relacionadas com técnicas de *Drift Detection* e a previsão de crimes em determinadas regiões de uma cidade. Vale ressaltar, que este modelo pode ser adequado a qualquer órgão de Segurança Pública que esteja buscando ferramentas de apoio à Gestão, tendo em vista que estamos caminhando para um novo conceito de Segurança Pública diretamente interligada com todos os tipos de ferramentas de Gestão disponíveis no mercado. Abre-se aqui, um caminho ainda que estreito, porém de grande importância para a comunidade acadêmica e também para a Segurança Pública para trabalhar cada vez mais e com maior apoio em pesquisas que envolvam novas técnicas e ferramentas tecnológicas e ocorrências policiais.

Referências Bibliográficas

AHA, D. W., KIBLER, D., ALBERT, M. K. *Instance-based Learning Algorithms.* Machine Learning, n. 6, v.1, 1991, p. 37 – 66.

BAENA-GARCIA, M.; DEL CAMPO-ÁVILA, J.; FIDALGO, R.; BIFET, A.; GAVALDÀ, R.; MORALES-BUENO, R. *Early drift detection method.* In: Proc. ECML/PKDD 2006, Work. Knowledge Discovery from Data Streams, 2006, p. 77-86.

BARDDAL, J. P.; GOMES, H. M.; ENEMBRECK, F. *SFNClassifier:A Scale-free Social Network Method to Handlew Concept Drift. In: ACM Symposium on Applied Computing 2014. To appear: Proceedings of the 29th ACM Symposium on Applied Computing.*

BIFET, A.; HOLMES, G.; KIRKBY, R.; PFAHRINGER, B. DATA STREAM MINING – A Practical Approach. COSI: Centre for Open Software Innovation. p.179, 2011 .

BIFET, A.; HOLMES, G.; PFAHRINGER, B.; KRANEN, P.; KREMER, H.; JANSEN, T.; SEIDL, T. MOA: *Massive Online Analysis, a Framework for Stream Classification and Clustering. JMLR – Proceedings Track*, 2010.

BIFET, A.; HOLMES, G.; PFAHRINGER, B.; GAVALDÀ, R. *Improving Adaptive Bagging Methods for Envolving Data Streams.* In: Proc. ACML 2009, Springer-Verlag, Berlin, Heildelberg, p. 23-37.

BISHOP, C. M. *Pattern Recognition and Machine Learning.* 2006, Springer, Heildelberg.

BREIMAN, L. *Bagging Predictors.* In: Machine Learning, Kluwer Academic Publishers, Boston, Manufactured in Netherlands, 1996, p. 123 – 140.

CASTELA, E. M. Investigação Criminal na Era do Governo Eletrônico, Inteligência Artificial X Boletim de Ocorrência (BO), soluções em KMAI. Dissertação de Mestrado, Universidade Federal de Santa Catarina – UFSC, 2003.

COPPIN, B. *Inteligência Artificial*. Tradução e Revisão do livro Artificial Intelligence illuminated, First Edition, LTC, Rio de Janeiro, 2010. p. 638.

DELANY, S. J.; CUNNINGHAM, P. *An Analysis of Case-Base Editing in a Spam Filtering System*. 7th European Conference on Case-Base Reasoning (ECCBR`04). V.3155 of LNAL., Springer, 2004, p. 128 - 141.

DE PAULA, A. C. M. P.; AVILA, B. C.; SCALABRIN, E.; ENEMBRECK, F. *Using Distributed Data Mining and Distributed Artificial Intelligence for Knowledge Integration.* In: 11th Cooperative Information Agents – CIA`07, 2007, Delft. Cooperative Information Agents XI, 11th International Workshop, CIA 2007. Berlin: Springer-Velag, 2007, v. 4676, p. 89 - 103.

ENEMBRECK, F.; AVILA, B. C.; SCALABRIN, E. E.; BARTHÉS, A. *Drifting Negotiations, Applied Artificial Intelligence*. Taylor & Francis, pp. 861 – 881, v.21, n. 9, 2007. ISSN: 1087-6545.

ENEMBRECK, F.; TACLA, C. A.; BARTHÉS, J. P. *Learning Negotiation Policies Using Ensemble-Based Drift Detection Techniques*. International Journal on Artificial Intelligence Tools – World Scientific Publishing Company, v.18, p. 173 – 196, 2009.

FREUND, Y.; SCHAPIRE, R. E. *Experiments with a new boosting algorithm. Proceedings of the 13th International Conference on Machine Learning.* 1996, pp. 148-156.

GAMA, J.; MEDAS, P.; CASTILLO, G.; RODRIGUES, P. *Learning with drift detection*. In: Proc. 17th Brazilian Symp. Artificial Intelligence, 2004, p. 285-295.

GOMES, H. M. Teoria de Redes Sociais aplicada ao problema de classificação *online* com mudança de conceito. Dissertação de Mestrado, 2012 – Curitiba – Pr.

GOMES, H. M.; ENENMBRECK, F. SAE: *Social Adaptive Ensemble Classifier for Data Strams, 2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM).*

GONÇALVES, A. E. Geocodificação e análise do mapeamento da criminalidade na cidade de Ipatinga. Monografia de Especialização, 2002 – UFMG – Universidade Federal de Minas Gerais.

HALL, M.; FRANK, E.; HOLMES, G.; PFAHRINGER, B.; REUTEMANN, P.; WITTEN, I. H. *The WEKA Data Mining Software: An Update*. In Proceedings of the 15th ACM SIGKDD – International Conference on Knowledge Discovery and Data Mining, 2009.

HELMBOLD, D. P., LONG, P. M. *Tracking drifting concepts by minimizing disagreements*. Journal of Machine Learning, 1994, p. 27 - 45.

KLINKENBERG, R. *Learning drifting concepts: Example selection vs. example weighting.* Intelligent Data Analysis, Special Issue on Incremental Learning Systems Capable of Dealing with Concept Drift, Vol.8, No. 3, 2004, p. 281-300.

KOLTER, J.Z.; MALOOF, M.A. *Dynamic Weighted Majority: A New Ensemble Method for Tracking Concept Drift*. IEEE Conference on Data Mining, 2003, p. 123.

KOLTER, J.Z.; MALOOF, M.A. *Using Additive Expert Ensembles to Cope with Concept Drift*. In Proceedings of the 22nd International Conference on Machine Learning, 2005, p. 449 – 456.

MACHADO, D.M.S. O uso da informação na gestao inteligente da Segurança Pública. Sao Paulo, 2008.

MARTINS, J. J. *Classificação de páginas de internet*. São Carlos, 2003, p. 76.

MELLIT, A.; KALOGIROU, S. A. *Artificial Intelligence techniques for photovoltaic applications: A Review*. *Progress in Energy and Combustion Science*, v.34, 2008, p. 574 – 632.

MIRANDA, A. P. M.; GUEDES, S. L.; BORGES, D.; BEATO, C.; SOUZA, E.; TEIXEIRA, P. A. S. A Análise Criminal e o Planejamento Operacional. Volume 1 – Série Análise Criminal – 1ª edição, Rio de Janeiro, 2008, ISBN 978-85-60502-5

Mitchell, Tom M. *Machine Learning*, McGraw-Hill, 1997.

NALEPA, G. M. Detecção de *Drifts* em um Processo de Negociação Bilateral Utilizando Rede Bayesiana e IB3, Curitiba, 2010, p. 69.

NALEPA, G.; ÁVILA, B. C.; ENEMBRECK, F.; SCALABRIN, E. *Learning Negotiation Policies Using IB3 and Bayesian Networks.* In: IDEAL 2010 – *Intelligent Data Engineering and Automated Learning*, *11th International Conference*, *Paysley, UK*.

NATH, S. V. Crime Pattern Detection Using Data Mining. In: WI-IAT 2006 Workshops. 2006 IEEE/WIC/ACM International Conference, p. 41-44.

NISHIDA, K.; YAMAUCHI, K.; OMORI, T. *ACE: Adaptive classifiers-ensemble system for concept-drifting environments.* In: Proc. 6th Int. Work. Multiple Classifier Systems, 2005, p. 176-185.

NISHIDA, K.; YAMAUCHI, K. *Detecting Concept Drift Using Statistical Testing*. Discovery Science: 10th international conference, DS 2007, Sendai, Japan, October 1-4, 2007, proceedings (3-540-75487-3, 978-3-540-75487-9), DS 2007, p. 264 – 269.

NISHIDA, K. *Learning and Detecting Concept Drift*. Hokkaido University, 2008, p. 123.

OHNO, A. *Detecção de Mudanças em problemas de classificação a partir de classificadores sociais*. Dissertação de Mestrado, 2011, Curitiba – PR.

PARADELA, R. B. *Utilizando Pesos Estáticos e Dinâmicos em Sistemas Multi-Classificadores com Diferentes Níveis de Diversidade.* Dissertação de Mestrado, 2007, Universidade Federal do Rio Grande do Norte, Natal.

PEREIRA, F. R.; BANASZEWSKI, R. F.; SIMAO, J. M.; TACLA, C. A. *Método baseado em detecção de mudanças para determinar preço de oferta de pedidos de clientes no ambiente TAC – SCM.* II MOPP 2010 - II Mostra de Pesquisa e Pós-Graduação da UTFPR, Agosto, 2010.

POELMANS, J.; ELZINGA, P.; VIAENE, S.; DEDENE, G. Curbing domestic violence: instantiating C-K theory with formal concept analysis and emergent self-organizing maps. In: International Journal Intelligent Systems in Accounting, Finance and Management, 17: 167-191. Doi: 10.1002/isaf.319, 2010.

POLIKAR, R.; ELWELL, R. *Incremental Learning of Variable Rate Concept Drift.* In: MCS 2009. LCNS, v. 5519, p. 142 – 151, Springer, Heidelberg (2009).

SENGER, L. J. *Escalonamento de processos: uma abordagem dinâmica e incremental para a exploração de características de aplicações paralelas.* USP - São Carlos, Dezembro, 2004. p. 236.

SIKLÓSSY, L.; AYEL, M. *Datum Discovery.* In: IDA-97 – Advances in Intelligent Data Analysis, Springer-Velag Berlin Heidelberg, 1997, LCNS 1280, p. 459 – 463.

SOUZA, A. J.; Análise dos Dados Relativos as Ocorrencias Registradas na Central de Emergência 190 da Polícia Militar na Região de Joinville, Buscando a Definição de Modelos de Previsão, Utilizando Ferramentas da Mineração de Dados. 2005.

STANLEY, K. O. *Learning concept drift with a committee of decision trees,* Tech. Report UT-AI-TR-03-302, Department of Computer Sciences, University of Texas at Austin, USA, 2003.

TSYMBAL, A. *The Problem of Concept Drift: Definitions and Related Work.* Technical Report TCD-CS-2004-15, Computer Science Department, Trinity College, Dublin, Ireland, 2004.

WANG, T.; RUDIN, C.; WAGNER, M.; SEVIERI, R. *Learning to Detect Patterns of Crime.* In Forthcoming in European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, 2013.

WANG, R.; SHI, L.; MICHEÁL O. F.; ROBSON, E. *A META-LEARNING METHOD FOR CONCEPT DRIFT.* In Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, p. 257-262, Waterford, Ireland, 2010, p. 227-243.

WIDMER, G.; KUBAT, M. *Effective learning in dynamic environments by explicit context tracking*, In: Proc. ECML 1993, Springer-Verlag, LNCS 667, 1993, p. 227-243.

WIDMER, G.; KUBAT, M. *Learning in the presence of concept drift and hidden contexts*, Machine Learning, 1996, p. 69 - 101.

Anexos

Declaração

O Anexo 1 trata-se de uma declaração do Comandante do CRE190 Joinville autorizando o uso dos dados de ocorrências policiais.

ESTADO DE SANTA CATARINA
POLÍCIA MILITAR DE SANTA CATARINA
5°REGIÃO DE POLÍCIA MILITAR
CENTRAL REGIONAL DE EMERGÊNCIA 190

DECLARAÇÃO

Joinville, 05 setembro de 2013.

Declaro, para os devidos fins, que:

ANDERSON JOSÉ DE SOUZA, CI/PM 926150-8, matriculado no Programa de Mestrado em Informática Aplicada na PUC-PR, está autorizado a utilizar os dados das ocorrências policiais de Roubo ou Assalto Contra Pessoa, Estabelecimentos Comerciais e Residências, da Região de Joinvile, registradas na Central de Emergência 190, compreendidas no período de 01/01/2010 a 30/07/2010.

E por ser verdade firmo a presente.

Dirceu Neundorf - Ten Cel PM
Oficial Comandante

Anexo 1. Declaração para uso dos dados da PMSC Joinville.