

# Análise de Perfil de Motoristas: Detecção de Eventos por meio de Smartphones e Aprendizado de Máquina

Jair Ferreira Júnior<sup>1,2</sup>, Gustavo Pessin<sup>1,2</sup>

<sup>1</sup>Instituto de Ciências Exatas e Naturais  
Universidade Federal do Pará (UFPA) – Belém, PA, Brasil

<sup>2</sup>Laboratório de Computação Aplicada  
Instituto Tecnológico Vale – Belém, PA, Brasil

jair.ferreira@pq.itv.org, gustavo.pessin@itv.org

**Resumo.** *O comportamento do motorista influencia fortemente a segurança no trânsito, o consumo de combustível e a emissão de gases poluentes. A Análise de Perfil de Motoristas (APM) é uma ferramenta que visa entender e potencialmente influenciar positivamente o comportamento de motoristas. Usualmente, tarefas de APM envolvem coletas automatizadas de dados de condução e aplicação de modelos computacionais a fim de gerar classificações que caracterizem o perfil de agressividade de condutores. Diferentes sensores e métodos de classificação têm sido empregados nesta tarefa, entretanto, soluções de baixo custo e alto desempenho ainda são alvos de pesquisa. Neste artigo é realizada uma investigação com diferentes sensores, presentes em um smartphone Android, e diferentes algoritmos de classificação, a fim de avaliar qual conjunto de sensor/método permite a classificação com maior acuracidade. Os resultados mostram que combinações específicas de sensores e métodos inteligentes permitem melhorar o desempenho das classificações.*

## 1. Introdução

O comportamento do motorista influencia fortemente a segurança no trânsito [Xiaoqiu et al. 2011] e é causa da grande maioria dos acidentes [Evans 2004]. O custo econômico total destes acidentes — apenas nos Estados Unidos em 2010 — foi estimado em mais de \$240 bilhões, decorrentes de cerca de 32 mil mortes, 3,9 milhões de feridos e 24 milhões de veículos danificados [Blincoe et al. 2015]. Adaptações neste comportamento podem aumentar a segurança e diminuir o consumo de combustível/energia e emissão de gases dos veículos [N. Haworth 2001, Van Mierlo et al. 2005]. A Análise de Perfil de Motoristas (APM) é uma das ferramentas que visa entender e potencialmente influenciar positivamente o comportamento de condutores, podendo promover uma forma de dirigir mais eficiente do ponto de vista energético e também mais segura.

A APM consiste na utilização de sensores para coleta automatizada de dados de condução de veículos (e.g., velocidade, aceleração, frenagem, esterçamento, localização) e aplicação de um modelo computacional de análise a fim de gerar uma pontuação ou classificação que caracterize a forma de dirigir do motorista. Esta pontuação representa o seu grau de risco ou segurança em um trajeto. A coleta de dados de direção em APM pode ser realizada por diversos tipos de sensores, desde os presentes nos atuais smartphones,

até equipamentos específicos para esse fim, como câmeras de monitoramento, caixas telemáticas<sup>1</sup> e adaptadores OBD (On-Board Diagnostic)<sup>2</sup>.

Atualmente, os smartphones são equipados com diversos sensores utilizados em soluções APM. Estes sensores incluem, entre outros, GPS (Global Positioning System), acelerômetro, giroscópio e magnetômetro de 3 eixos (lateral, longitudinal e vertical). Experimentos [Paefgen et al. 2012, Skog et al. 2013, Handel et al. 2014] demonstraram que, no contexto de APM, os dados fornecidos por estes sensores estão fortemente correlacionados com medições de referência de caixas telemáticas instaladas fixamente nos veículos e, apesar de haver espaço para melhorias, os smartphones se provaram uma forma acessível e interessante de instrumentar um veículo para coleta de dados. A relevância da APM têm crescido nos últimos anos em diversos domínios de aplicação. Na gestão de frotas, há interesse em informações detalhadas e precisas sobre o uso dos veículos. No domínio de telemática de seguros, planos do tipo Usage-Based Insurance (UBI) têm como objetivo calcular o bônus de seguro em função do perfil individual de condução do motorista, ao invés de recorrer a estatísticas baseadas em grupos, como idade, sexo e estado civil. Adicionalmente, a pontuação do motorista está relacionada com quão ecológica é sua forma de dirigir [Van Mierlo et al. 2004]. Neste caso, o principal objetivo é incentivar os motoristas a buscar uma boa pontuação para otimizar o consumo de combustível e energia de seus veículos, gerando economia. Por fim, é possível também prevenir a ocorrência de acidentes ao fornecer *feedback* ao motorista. Tal *feedback* é mais útil e eficaz quando ocorre em tempo real, notificando-o, por exemplo, sempre que acelerar demais ou quando sua pontuação refletir uma direção agressiva.

Diversos trabalhos de APM podem ser encontrados na literatura. Dentre estes, podemos citar [Johnson and Trivedi 2011, Castignani et al. 2013, Saiprasert et al. 2014, Castignani et al. 2015] que utilizam uma fusão de dados dos sensores do smartphone para identificar eventos de direção agressivos (e.g., curvas, acelerações e freadas agressivas) e, conseqüentemente, classificar a forma de dirigir do motorista. Outro trabalho [Araujo et al. 2012] utiliza dados de sensores do próprio veículo para fornecer dicas de direção e avaliar a eficiência de consumo de combustível em função da forma de dirigir do motorista. Os algoritmos de aprendizado de máquina empregados nestes trabalhos se resumem à lógica fuzzy ou variações de Dynamic Time Warping<sup>3</sup> (DTW). Adicionalmente, a maioria dos trabalhos usa detecção de eventos agressivos como base para a classificação de perfis de direção e peca ao não fornecer uma avaliação quantitativa da metodologia empregada.

Portanto, este artigo tem como objetivo a avaliação quantitativa das performances de combinações de algoritmos de aprendizado de máquina e sensores de smartphone Android, na tarefa de detecção de eventos de direção. Assim, ao final desta avaliação, será possível identificar qual combinação de sensor, seu(s) eixo(s) e algoritmo melhor detecta um determinado tipo de evento agressivo. O restante do artigo é organizado da seguinte forma: a Seção 2 resume alguns trabalhos relacionados com APM. A Seção 3 descreve

---

<sup>1</sup>A “caixa telemática” é um equipamento de medição fixamente instalado no veículo que pode ter sensores próprios ou se conectar aos sensores internos do veículo através do barramento CAN.

<sup>2</sup>O OBD é um sistema que confere aos veículos atuais a capacidade de se auto-diagnosticar e fornecer dados (e.g., velocidade) em tempo real via uma porta de comunicação padronizada.

<sup>3</sup>*Dynamic Time Warping* é uma técnica utilizada para encontrar padrões em séries temporais e foi originalmente empregada no problema de reconhecimento de fala [Sakoe and Chiba 1978].

a metodologia da avaliação realizada, seguida de seus resultados na Seção 4. Por fim, na Seção 5 são apresentadas as conclusões e trabalhos futuros.

## 2. Trabalhos Relacionados

Esta seção descreve resumidamente alguns trabalhos recentes da literatura relacionados com APM. É importante ressaltar que várias soluções comerciais de APM estão disponíveis atualmente. A maioria delas aplicada aos domínios de telemática de seguros e gestão de frotas. Exemplos incluem Aviva Drive, Greenroad, Snapshot e SeeingMachines. Entretanto, uma análise técnica destas soluções é inviável já que seus métodos de coleta e modelos de análise de dados não são publicamente disponíveis.

O trabalho [Johnson and Trivedi 2011] apresenta o aplicativo *MIROAD* para iPhone que faz uma fusão do magnetômetro, acelerômetro, giroscópio e GPS do smartphone para identificar eventos de direção agressivos e, conseqüentemente, classificar a forma de dirigir como agressiva ou não-agressiva. Estes eventos são detectados por um único classificador baseado no algoritmo DTW. Todo o processamento é feito em tempo real no próprio smartphone, sem necessidade de equipamentos externos. A análise experimental executada mostrou que 97% dos eventos agressivos foram corretamente identificados. O trabalho [Araujo et al. 2012] propõe um aplicativo de smartphone para avaliar a eficiência de consumo de combustível em função da forma de dirigir do motorista. O aplicativo também fornece dicas de direção em tempo real do tipo “troque de marchas mais cedo” e “aceleração muito alta”. O processamento é feito localmente no smartphone e em tempo real. O aplicativo não utiliza nenhum sensor do smartphone. Ao invés disso, coleta dados de velocidade, aceleração, altitude, sinal do acelerador, consumo instantâneo de combustível e RPM (Rotações Por Minuto) dos sensores do próprio veículo por meio de um adaptador OBD Bluetooth e do aplicativo Torque Pro para smartphones. Após a coleta, é realizada uma extração de características e três classificadores ( $h_1$ ,  $h_2$  e  $h_3$ ) são aplicados.  $h_1$  é um discriminante linear que identifica a “Condição de Direção” como *urbana*, *auto-estrada* ou *mista*.  $h_2$  e  $h_3$  são sistemas fuzzy que fornecem, respectivamente, a “Avaliação do Consumo de Combustível” e a “Dica de Direção”. Uma avaliação preliminar da aplicação foi realizada através de uma viagem de pouco mais de uma hora. Nesta avaliação,  $h_1$  reconheceu corretamente as diferentes condições de direção presentes, o classificador  $h_2$  informou a melhor avaliação na maior parte do tempo e o classificador  $h_3$  forneceu dicas para trocar de marcha mais cedo e acelerar mais delicadamente sempre que o motorista acelerava de forma mais intensa. Não há menção à eficácia dos classificadores  $h_2$  e  $h_3$  na referida avaliação preliminar.

O artigo [Castignani et al. 2013] propõe um mecanismo de avaliação de motoristas baseado em lógica fuzzy que utiliza o acelerômetro, magnetômetro e sensor de gravidade de smartphones Android como entrada, classifica o motorista nos estilos de direção *Normal*, *Moderado* e *Agressivo*, e calcula seu score de direção de 0 (melhor) a 100 (pior). A classificação e o score não são calculados em tempo real, visto que os dados de sensores são coletados pelo aplicativo *UBI-Meter* e armazenados no smartphone para posterior envio a um servidor remoto na Internet que agrega dados de diversos motoristas e os analisa. Foi realizado experimento com 5 motoristas, totalizando 2360 km, 33 horas e 87 viagens. Neste experimento, os scores gerais dos motoristas 1 ao 5 foram 68,57, 83,46, 82,72, 77,15 e 60,12. A eficácia do mecanismo no experimento não é informada.

O objetivo do trabalho [Saiprasert et al. 2014] é utilizar o GPS, acelerômetro e magnetômetro dos smartphones para classificar a forma de dirigir de motoristas nos perfis de segurança *muito seguro*, *seguro*, *agressivo*, e *muito agressivo*. Esta classificação é realizada por meio da detecção, em tempo real, de eventos de condução relevantes que podem ocorrer durante uma viagem. A detecção é feita por um algoritmo de reconhecimento de padrões [Saiprasert et al. 2013] baseado na técnica de DTW. Os eventos detectados por este algoritmo são *frenagem agressiva*, *aceleração brusca*, *curva acentuada*, *mudança de faixa agressiva* e *velocidade excessiva*. Na extensa análise experimental realizada (30 viagens em rota de 315 km e 30 viagens em outra rota de 780 km), a grande maioria dos motoristas foi classificada nos perfis intermediários *seguro* (28 viagens) e *agressivo* (17 viagens). Poucos foram classificados nos perfis *muito seguro* (3 viagens) e *muito agressivo* (8 viagens). O artigo não menciona a eficácia de seu método no experimento realizado.

O artigo [Castignani et al. 2015] apresenta o *SenseFleet*, uma plataforma de APM baseada em smartphone Android. O aplicativo coleta dados do acelerômetro, magnetômetro, sensor de gravidade e GPS do smartphone com o objetivo de detectar eventos de risco (velocidade, aceleração, frenagem e esterçamento excessivos) ocorridos durante uma viagem. Uma nova pontuação do motorista é gerada a cada viagem percorrida em função da detecção destes eventos, podendo variar entre 0 (pior) a 100 (melhor) pontos. Tal detecção é realizada por um sistema fuzzy cujos limites são obtidos dinamicamente através de um processo de calibragem executado na primeira vez que o aplicativo é utilizado em uma nova combinação de motorista, smartphone e veículo. Todo o processamento é feito em tempo real no próprio aplicativo, independentemente do modelo de smartphone e veículo. Adicionalmente, os eventos detectados, os dados coletados e as pontuações podem ser enviados a um servidor central que agrega informações de diversos motoristas e viagens para futura análise e geração de relatórios. Diversos experimentos foram realizados pelos autores. Em um deles foi observada uma taxa pouco maior que 90% de eventos corretamente identificados. Em outro experimento que utilizava dois smartphones bastante diferentes instalados no mesmo veículo, foi constatado que a detecção de eventos segue praticamente o mesmo padrão, independentemente do dispositivo utilizado.

Ao analisar os trabalhos apresentados nos parágrafos anteriores, podemos identificar as seguintes características comuns: (i) os trabalhos utilizam uma fusão de dados de vários sensores como entrada para os modelos de detecção de eventos; (ii) a maioria dos trabalhos [Araujo et al. 2012, Castignani et al. 2013, Castignani et al. 2015] emprega lógica fuzzy e o restante [Johnson and Trivedi 2011, Saiprasert et al. 2014] utiliza DTW; (iii) a maioria dos trabalhos usa os sensores disponíveis em smartphones como meio de coleta de dados, ao invés dos sensores do próprio veículo (via interface OBD) ou caixas telemáticas; e (iv) a maioria dos trabalhos não menciona a eficácia quantitativa das metodologias propostas, tornando difícil avaliá-las. Isso ocorre em [Castignani et al. 2013, Saiprasert et al. 2014] e parcialmente em [Araujo et al. 2012].

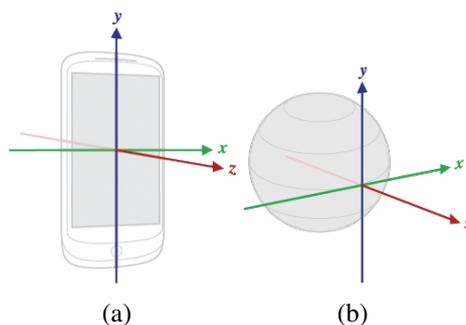
### 3. Metodologia

A avaliação proposta neste trabalho tem como objetivo identificar as triplas {*sensor*, *eixo(s)*, *algoritmo de aprendizado*} capazes de melhor detectar um determinado tipo de evento de direção. Para tanto, modelamos a avaliação como um problema de aprendizado supervisionado no qual as classes são os tipos de eventos, e os dados dos sensores dão

origem a vetores de atributos que formam os *datasets*. Ao aplicar diversos algoritmos de classificação, obtemos os scores ROC [Fawcett 2006] para cada classe e dataset. Este score varia de 0,5 (pior resultado) a 1,0 (melhor resultado) e será utilizado para determinar a eficácia do algoritmo de classificação dados um tipo de evento de direção (classe), um sensor e seu(s) eixo(s).

Utilizamos os seguintes sensores de smartphone Android nesta avaliação: aceleração linear (AclLin), magnetômetro (Mag), gravidade (Grav), vetor de rotação (VRot), acelerômetro (Acel) e giroscópio (Gir). O acelerômetro mede a força da aceleração causada pelo movimento do telefone. O sensor de aceleração linear funciona como o acelerômetro, mas excluindo a aceleração da gravidade. O giroscópio mede o grau de rotação em torno dos eixos do dispositivo. O magnetômetro mede a força do campo magnético, funcionando como uma bússola. O sensor de vetor de rotação representa a orientação do smartphone como uma combinação da rotação de um ângulo  $\theta$  sobre um eixo ( $x$ ,  $y$  ou  $z$ ). Por último, o sensor de gravidade fornece um vetor tridimensional indicando a direção e magnitude da gravidade.

Os dados fornecidos pelos sensores do smartphone são uma série temporal com precisão de nanosegundos e taxas de amostragem de  $\sim 100$  Hz ou  $\sim 240$  Hz, dependendo do sensor. Todos os sensores utilizados, exceto o de vetor de rotação, fornecem amostras com três valores ( $x$ ,  $y$  e  $z$ ) referentes ao sistema de coordenadas do dispositivo exibido na Figura 1(a). Já o sensor de vetor de rotação fornece amostras com quatro valores (*componentes escalar*,  $x$ ,  $y$  e  $z$ ) referentes ao sistema de coordenadas da Terra exibido na Figura 1(b). Cada sensor dá origem a um dataset contendo dados de todas as suas coordenadas, mais um dataset para cada coordenada isoladamente. Por exemplo, o acelerômetro dá origem a quatro datasets: *acelerômetro* (com dados de  $x$ ,  $y$  e  $z$  juntos), *acelerômetro\_x*, *acelerômetro\_y* e *acelerômetro\_z*. Portanto, esta avaliação foi realizada em um total de 25 datasets. Detalhes sobre os vetores de atributos criados serão apresentados na Seção 3.1.



**Figura 1. Sistema de coordenadas relativo ao smartphone (a) utilizado por todos os sensores desta avaliação, exceto pelo de vetor de rotação que usa o sistema de coordenadas relativo à Terra (b).**

Os algoritmos de classificação usados nesta avaliação são o Multi Layer Perceptron (MLP), Bagging de “Reduced Error Pruning Trees” (REPTrees) e Support Vector Machines (SVM). Utilizamos as implementações destes algoritmos no popular framework de mineração de dados e aprendizado de máquina WEKA [Hall et al. 2009] e LIBSVM [Chang and Lin 2011]. Configuramos os algoritmos com os parâmetros padrão do WEKA, conforme Tabela 1.

**Tabela 1. Algoritmos e suas configurações no WEKA.**

Algoritmo	Configuração
Multi Layer Perceptron (MLP)	-L 0.3 -M 0.2 -N 500 -V 0 -E 20 -H a
Bagging de REPTree	-P 100 -I 10 -W weka.classifiers.trees.REPTree -- -M 2 -V 0.001 -N 3 -L -1
Support Vector Machines (SVM)	-S 0 -K 2 -D 3 -G 0.0 -R 0.0 -N 0.5 -M 40.0 -C 1.0 -E 0.001 -P 0.1

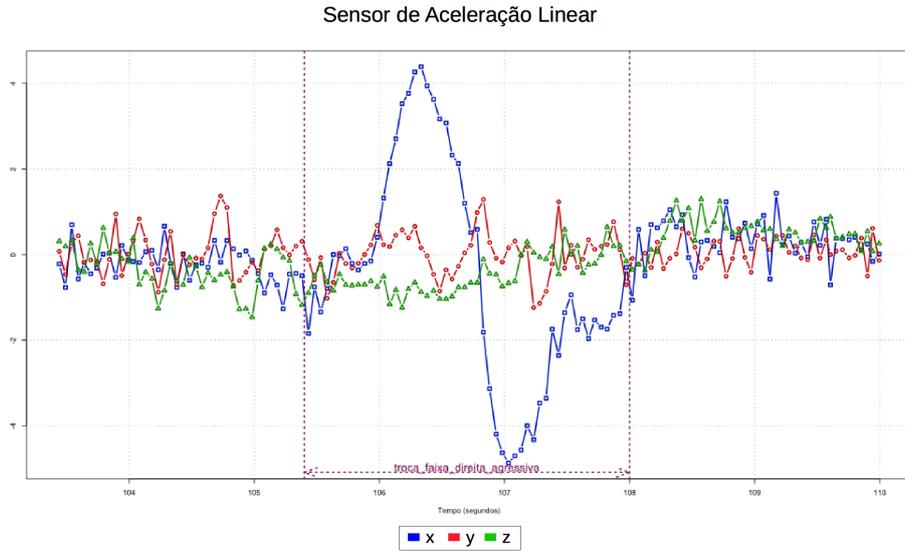
Cada algoritmo de classificação é treinado e avaliado com os datasets por meio de validação cruzada de 10 subconjuntos. Como possuímos 25 datasets e 3 algoritmos, avaliamos  $(25 * 3) = 75$  combinações de  $\{sensor, eixo(s), algoritmo\}$  de aprendizado, obtendo o ROC Score para cada tipo de evento de direção. As classes de aprendizado desta avaliação foram baseadas nos tipos de eventos de direção detectados em [Saiprasert et al. 2014]. Com o objetivo de obter leituras de sensores do smartphone para estes eventos, realizamos um experimento no qual estas manobras de direção foram executadas em um veículo real durante duas viagens com duração de aproximadamente 5 minutos cada, enquanto um aplicativo Android capturava dados dos sensores do smartphone. Os tempos iniciais e finais dos eventos foram registrados para servir como *ground-truth*. Os tipos de eventos de direção e seus números de amostras no experimento são: Aceleração Agressiva (2 eventos), freada Agressiva (3 eventos), Curva Agressiva à Direita (4 eventos), Curva Agressiva à Esquerda (2 eventos), Troca de Faixa Agressiva à Direita (2 eventos), Troca de Faixa Agressiva à Esquerda (2 eventos) e Evento não agressivo (9 eventos). O último tipo de evento agrupa manobras executadas de forma não agressiva (e.g. curva suave, aceleração suave).

Realizamos o experimento nas seguintes condições: (i) o veículo foi um Honda Civic 2011; (ii) o smartphone foi um Samsung Galaxy S6 afixado na posição vertical no para-brisas do veículo por meio de suporte apropriado; (iii) o mesmo motorista com 20 anos de experiência conduziu o veículo em ambas as viagens; (iv) a via era asfaltada, seca, com poucos buracos e de trânsito leve; (v) o clima era ensolarado; e (vi) a taxa de amostragem dos sensores de aceleração linear, gravidade, vetor de rotação e magnetômetro era de  $\sim 100$  Hz, enquanto que a do acelerômetro e giroscópio era de  $\sim 240$  Hz. O gráfico da Figura 2 exibe os dados dos eixos do sensor de aceleração linear durante um evento de troca de faixa agressiva à direita coletado no experimento. Nota-se o formato característico do eixo  $x$ .

### 3.1. Vetor de Atributos

Cada sensor origina um dataset bruto contendo basicamente o instante da coleta (em nanossegundos) e os valores dos eixos dos sensores naquele momento. Estes dados brutos são tratados antes de serem submetidos aos classificadores. Este tratamento consiste em gerar uma janela móvel de 7 segundos deslizando em incrementos de 1 segundo sobre a série temporal. O tamanho da janela foi experimentalmente definido em 7 segundos devido ao fato de que, ao analisar a *ground-truth* do experimento, percebemos que os eventos de direção duram entre 2 (para os mais agressivos) e 6 (para os não agressivos) segundos. Portanto, com essa duração, a janela captura completamente qualquer evento de direção coletado no experimento.

A janela é expressa por um vetor de atributos. Para calcular este vetor, consideramos  $t_0$  o conjunto de amostras do segundo atual da série temporal,  $t_{-1}$  o con-



**Figura 2. Eixos x, y e z do sensor de aceleração linear em um evento de troca de faixa agressiva à direita. Nota-se o formato característico do eixo x.**

junto de amostras do segundo anterior, e assim por diante, até  $t_{-6}$ . Calculamos então as médias ( $M_0 = M_{[t_0]}$ ,  $M_{-1} = M_{[t_{-1}, t_0]}$ , ...,  $M_{-6} = M_{[t_{-6}, t_0]}$ ), medianas ( $MN_0 = MN_{[t_0]}$ ,  $MN_{-1} = MN_{[t_{-1}, t_0]}$ , ...,  $MN_{-6} = MN_{[t_{-6}, t_0]}$ ) e desvios padrões ( $DP_0 = DP_{[t_0]}$ ,  $DP_{-1} = DP_{[t_{-1}, t_0]}$ , ...,  $DP_{-6} = DP_{[t_{-6}, t_0]}$ ) acumulados. Adicionalmente, as tendências de crescimento ou diminuição das médias das amostras de cada segundo em relação às do segundo atual ( $t_0$ ) são calculadas pelas suas proporções da seguinte forma:  $T_{-1} = \frac{M_{t_{-1}}}{M_{t_0}}$ ,  $T_{-2} = \frac{M_{t_{-2}}}{M_{t_0}}$ , ...,  $T_{-6} = \frac{M_{t_{-6}}}{M_{t_0}}$ . A classe é então adicionada ao vetor quando os tempos inicial e final do evento (obtido por meio da ground-truth) estiverem ambos entre  $t_{-6}$  e  $t_0$  da janela. Caso contrário, o vetor de atributos não é gerado para esta janela em particular. A Figura 3 exibe a estrutura do vetor de atributos para um eixo da série temporal. Quando o dataset possuir diversos eixos, os vetores de cada eixo (sem as classes) são concatenados horizontalmente e a classe é adicionada como último campo do vetor resultante. Por fim, como os vetores de atributos são atemporais, agrupamos os vetores de várias viagens em um único dataset para o mesmo sensor e eixo(s).



**Figura 3. Vetor de atributos representando uma janela deslizante de 7 segundos (de  $t_{-6}$  a  $t_0$ ) da série temporal para um eixo de um sensor.**

#### 4. Resultados

Executamos os 3 algoritmos de classificação da Tabela 1 sobre os 25 datasets contendo os vetores de atributos descritos na Seção 3.1 e extraímos as triplas  $\{sensor, eixo(s), algoritmo\}$  de maiores ROC Scores médios para cada evento de direção. Cada tripla foi executada com 10 sementes aleatórias diferentes. Os resultados da avaliação podem ser vistos na Figura 4.

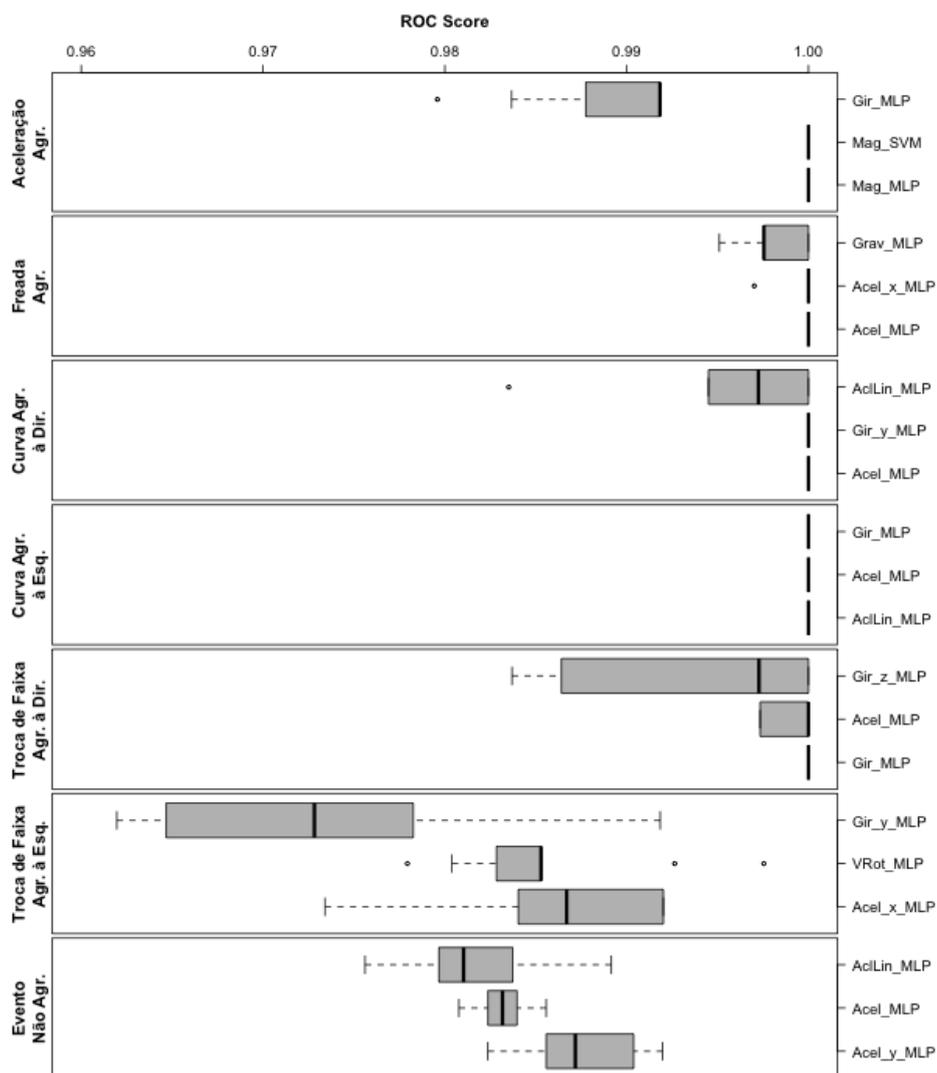


Figura 4. ROC Scores de 10 execuções das 3 melhores triplas (*sensor, eixo(s), algoritmo*) para cada tipo de evento de direção. O melhor algoritmo de detecção entre os avaliados é o MLP e o melhor sensor é o acelerômetro, exceto para o evento de aceleração agressiva.

Ao agrupar os resultados por eventos, notamos que a aceleração agressiva é melhor detectada com o magnetômetro em conjunto com os algoritmos MLP ou SVM que obtiveram a mesma performance máxima. Na freada agressiva, a melhor tripla foi {*acelerômetro, todos os eixos, MLP*}. Para a curva agressiva à direita, o algoritmo MLP foi o melhor classificador, obtendo a mesma performance máxima com o giroscópio (eixo y apenas) e acelerômetro. O mesmo ocorreu com a curva agressiva à esquerda que também foi melhor detectada pelo MLP, obtendo performance máxima com o giroscópio, acelerômetro e sensor de aceleração linear. A troca de faixa agressiva à direita foi melhor detectada pela tripla {*giroscópio, todos os eixos, MLP*} que obteve o ROC Score máximo. A troca de faixa agressiva à esquerda foi melhor detectada pela tripla {*acelerômetro, y, MLP*}, mas seguida de perto por {*sensor de vetor de rotação, todos os eixos, MLP*}. O evento não agressivo foi melhor classificado pela tripla {*acelerômetro, y, MLP*}. Por fim, ao analisar os resultados, notamos que o algoritmo MLP em conjunto com o acelerômetro

figuram entre as 3 melhores performances na grande maioria dos eventos, com exceção do evento de aceleração agressiva cujos melhores resultados não incluem o acelerômetro.

## 5. Conclusões e Trabalhos Futuros

Apresentamos neste trabalho uma avaliação quantitativa das performances dos algoritmos MLP, SVM e Bagging de REPTree na detecção de eventos de direção utilizando dados de sensores de smartphones Android. Obtivemos amostras de 7 tipos de eventos de direção por meio de um experimento no qual os dados de 6 sensores foram registrados durante a execução dos referidos eventos em um veículo real. Os tempos iniciais e finais de cada evento executado foram registrados para compor a ground-truth do experimento. Os resultados de 10 execuções dos algoritmos de aprendizado para cada combinação de sensor e eixo(s) mostram (i) quais combinações de sensor, eixo(s) e algoritmo de aprendizado melhor detectam cada tipo de evento de direção; (ii) que o algoritmo MLP obteve as melhores performances; e (iii) que o acelerômetro se mostrou o sensor com melhores performances, exceto para o evento de aceleração agressiva.

Como trabalhos futuros, esperamos realizar experimentos com viagens mais longas, com motoristas e modelos de smartphones diferentes e em condições de via, clima e temperatura distintos para obter um conjunto maior e mais diversificado de amostras dos eventos de direção. Também esperamos comparar algoritmos de aprendizado de máquina adicionais, incluindo os baseados em lógica fuzzy e DTW. Por fim, pretendemos utilizar os melhores algoritmos e sensores identificados diretamente em um aplicativo para smartphone Android a fim de detectar os eventos de direção em tempo real e, consequentemente, classificar o perfil de agressividade de direção do motorista.

## Agradecimentos

Agradecemos ao Prof. Dr. Cleidson de Souza e aos colegas Sérgio Viademonte, Carolina Quintão e Adalberto Silva Júnior pelas conversas, perguntas, sugestões, inspirações e ajuda na implementação deste trabalho. Agradecemos ao apoio financeiro recebido através da Chamada 59/2013 MCTI/CT-Info/CNPq, processo 440880/2013-0.

## Referências

- Araujo, R., Igreja, A., de Castro, R., and Araujo, R. (2012). Driving coach: A smartphone application to evaluate driving efficient patterns. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 1005–1010.
- Blincoe, L., Miller, T. R., Zaloshnja, E., and Lawrence, B. A. (2015). The Economic and Societal Impact of Motor Vehicle Crashes, 2010 (Revised). Technical report.
- Castignani, G., Derrmann, T., Frank, R., and Engel, T. (2015). Driver behavior profiling using smartphones: A low-cost platform for driver monitoring. *Intelligent Transportation Systems Magazine, IEEE*, 7(1):91–102.
- Castignani, G., Frank, R., and Engel, T. (2013). Driver behavior profiling using smartphones. In *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, pages 552–557.
- Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

- Evans, L. (2004). *Traffic safety*. Science Serving Society.
- Fawcett, T. (2006). An introduction to roc analysis. *Pattern Recogn. Lett.*, 27(8):861–874.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18.
- Handel, P., Ohlsson, J., Ohlsson, M., Skog, I., and Nygren, E. (2014). Smartphone-based measurement systems for road vehicle traffic monitoring and usage-based insurance. *Systems Journal, IEEE*, 8(4):1238–1248.
- Johnson, D. and Trivedi, M. (2011). Driving style recognition using a smartphone as a sensor platform. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, pages 1609–1615.
- N. Haworth, M. S. (2001). Driving to Reduce Fuel Consumption and Improve Road Safety. *Road Safety Research, Policing and Education Conference, 2001, Melbourne, Victoria, Australia*, (5):7.
- Paefgen, J., Kehr, F., Zhai, Y., and Michahelles, F. (2012). Driving behavior analysis with smartphones: Insights from a controlled field study. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia, MUM '12*, pages 36:1–36:8, New York, NY, USA. ACM.
- Saiprasert, C., Pholprasit, T., and Pattara-Atikom, W. (2013). Detecting driving events using smartphone. In *Proceedings of the 20th ITS World Congress*.
- Saiprasert, C., Thajchayapong, S., Pholprasit, T., and Tanprasert, C. (2014). Driver behaviour profiling using smartphone sensory data in a v2i environment. In *Connected Vehicles and Expo (ICCVE), 2014 International Conference on*, pages 552–557.
- Sakoe, H. and Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 26(1):43–49.
- Skog, I., Handel, P., Ohlsson, M., and Ohlsson, J. (2013). Challenges in smartphone-driven usage based insurance. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 1135–1135.
- Van Mierlo, J., Maggetto, G., van de Burgwal, E., and Gense, R. (2004). Driving style and traffic measures - influence on vehicle emissions and fuel consumption. 218(D1):43–50+.
- Van Mierlo, J., Maggetto, G., Van de Burgwal, E., and Gense, R. (2005). Driving style and traffic measures-influence on vehicle emissions and fuel consumption. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 218(1):43–50.
- Xiaoqi, F., Jinzhang, J., and Guoqiang, Z. (2011). Impact of driving behavior on the traffic safety of highway intersection. In *Measuring Technology and Mechatronics Automation (ICMTMA), 2011 Third International Conference on*, volume 2, pages 370–373.