Atividade 4: Exploração de dados de atividades físicas

Rodney Rick
RA 181.814
Disciplina MO444
Prof. Dr. Anderson Rocha

***Resumo*—O objetivo deste trabalho é estudar a representação de dados disponíveis em um janela de tempo e como agrupá-los, transformá-los (quando necessário), e através de técnicas de validação o quão preciso está o modelo gerado e avaliar os resultados finais obtidos. Também abordado discussão dos dados de saída e problemas enfrentados para as execuções.**

I. Introdução

As seções seguintes apresentam o conjunto de dados, como foi extraídos valores, técnicas de árvores de decisão e para melhores explorações dos dados, usado uma técnica de redução de dimensionalidade. O foco para este projeto é explorar técnicas ainda não trabalhada nas atividades anteriores.

A base de dados é composta por amostras de atividades de usuário (estes designados por códigos), com um "*snapshot*"temporal e valores de aceleração X, Y e Z retirados através do uso de algum aparelho de medição. Para cada nova amostra, existe um novo "*snapshot*", e caso seja o mesmo usuário e a mesma atividade, a próxima amostra descreve a continuação do movimento daquela atividade 50 milissegundos a frente. Mais informações sobre o conjunto de dados está descrita na seção II.

Para composição desses "*snapshots*"ao longo do tempo, necessita-se o agrupamento desses dados, demonstrados na seção III. A utilização de técnicas para a transformação dos dados tornou-se de grande utilidade para maiores ganhos nos valores de acurária balanceada.

Trabalhar com validações cruzadas usando técnicas como a Matriz de Confusão, o qual trabalha com cálculos de Sensitividade, Especificidade, Taxa de Falso Positivo, Taxa de Falso Negativo entre outros cálculos. Uma tabela da matrix de confusão está presente no Apêndice A, figura 13.

Ao final do relatório (seção V) faz-se o levantamento dos resultados e discussão das técnicas aplicadas conforme a resolução. Na seção de Apêndice A tem algumas informações extras de validação dos dados.

Para esse trabalho foram considerados algumas leitura essenciais para este trabalho:

- Capítulo 15 (*Random Forest*) do livro de Hastie et al.;
- Capítulo 12 (*PCA*) do livro de Bishop;
- e, notas de aula e artigos na internet.

II. O conjunto de dados

O conjunto de dados é composto por 2.980.763 registros de atividades, os quais s ao compostos por seis atributos: usuários, atividade física, tempo em *timestamp* (registro temporal) e as acelerações nos eixos X, Y e Z.

Ainda, a base está ordenada por usuário, atividade física realizada e tempo (nesta ordem de prioridade), os quais, dado um novo registro da base original, o mesmo usuário e atividade, demonstra o próximo tempo da atividade (50 milissegundos a frente) com as novos valores para X, Y e Z, exibindo continuidade da atividade. A tabela I exemplifica. A tabela II representa a distribuição dos dados conforme a atividade (a coluna *offset* é esclarecido na seção III).

Tabela I: Tabela exemplo do conjuntos de dados

Usuário	Atividade	Timestamp	X	Y	Z
1679	Walking	1370520469556	0.29413	-0.63560	-0.22693
1679	Walking	1370520469606	-0.49968	-0.60445	-0.22602
1679	Walking	1370520469656	-2.17834	0.71349	0.37201

Tabela II: Tabela exemplo do conjuntos de dados

Atividade	Quantidade	%	*offset*
Walking	1,255,923	42.1	300
Jogging	438,871	14.7	90
Stairs	57,425	1.9	20
Sitting	663,706	22.3	160
Standing	288,873	9.7	100
LyingDown	275,967	9.3	69

Deve-se atentar que para os 225 usuários disponíveis nas amostras, nem todos praticam as 6 atividades físicas.

III. Exploração dos dados

Nesta seção, trabalha-se a composição dos dados a fim de formar uma série temporal e assim caracterizar um determinada atividade.

Para que a identificação seja possível, deve-se criar uma "janela de tempo"com alguns segundos de cada uma das atividades. Escolhendo dois intervalos, de 5 e 10 segundos, tem-se a composição de 100 e 200 amostras, respectivamente. No entanto, obriga-se ainda alguns tratamentos. Na tabela II, existe a coluna *offset*, a qual demonstra a tratativa de passo dado em cada janela de tempo. Suponha que dado uma lista composta de 22 números (0 à 21) e uma janela (ou intervalo) de 10 posições, trabalha-se com o passo (*offset*) de 5 posições. Logo será aproveitado todas as listas completas de 10 valores e descartados o resto da lista que não atender a esse critério. Segue exemplo na tabela III.

Essa abordagem de janela "deslizante"de p posições torna-se interessante para a criação de mais exemplos de amostras, sem a necessidade da sintetização de novos registros através da combinação de outros dois ou mais, conforme a técnica

Tabela III: Tabela de janelas e descartes

Parte da Lista	Quantidade
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]	10
[5, 6, 7, 8, 9, 10, 11, 12, 13, 14]	10
[15, 16, 17, 18, 19, 20, 21]	7

Tabela IV: Separação dos dados em Treino e Validação

		Janela de Tempo (em segundos)	
		5	**10**
Quantidade de amostras	**Treino**	25000	20000
	Validação	25000	20000

SMOTE que visa sintetizar mais dados para bases desbalanceadas, como enunciada no artigo de Chawla et al..

Porém, mesmo para as classes com maior amostragem (conforme tabela II) como *LyingDown* do que a classe *Stairs*, necessita-se o tratamento de um *offset* pequeno, devido a menor de amostragem por usuário, lembrando que deve-se completar as janelas de 100 e 200 posições (obs: são 100 posições de dados referentes ao vetor X, 100 do vetor Y e 100 do vetor Z, o mesmo vale para 200 posições). A escolha desses valores de passos foi para equilibrar o número de amostras, mantendo ainda uma certa quantidade de balanceamento desproporcional original da base.

Ainda nesta seção, trabalhou-se com duas abordagens para a composição dos vetores dos valores de x, y, z para cada janela de tempo. Segue abaixo (n representa o tamanho máximo da janela de tempo):

(a) $x_1, y_1, z_1, x_2, y_2, z_2, ..., x_n, y_n, z_n$, onde $n = [1, 100]$;
(b) $x_1, x_2, ..., x_n, y_1, y_2, ..., y_n, z_1, z_2, ..., z_n$

Apesar da modelagem do item (b) parecer mais interessante, pois facilita investigar com algumas técnicas nos vetores separadamente para depois concatená-los com os resultados obtidos de cada uma das técnicas, a abordagem com o (a) mostrou-se com melhores valores nas técnicas aplicadas e abordadas na seção IV. Um dos maiores motivos para a escolha do item (a) foi devido ao processamento da técnica de de *Fast Fourier transform* (*FFT*) passando a trabalhar com dados no domínio da frequência.

Para trabalhar com validações de modelos, como demonstra a técnica da Matriz de Confusão, para este experimento, no momento de separação das bases de treino e validação, deve-se garantir que um (ou mais) usuário(s) praticando uma mesma atividade não esteja presente nas duas bases. Garantindo assim o propósito da validação sem o possível *overfitting* na etapa de treinamento.

A proporção de dados explorados em cada janela de tempo está descrita na tabela IV.

IV. As técnicas

Para os experimentos desta seção, foram escolhidos algumas etapas de processamento:

1) Combinação dos vetores $x_1, y_1, z_1, x_2, y_2, z_2, ..., x_n, y_n, z_n$;
2) Aplicação da técnica *PCA* para redução da dimensionalidade dos dados;
3) Se aplicado *FFT*, usa-se novamente *PCA*, conforme passo 1.
4) Exploração de alguns algoritmos:
   i Árvore de decisão (usando o pacote da linguagem de programação R, o *rpart*);
   ii Random Forest (da linguagem de programação R);

Por experiências nos trabalhos anteriores, foi aplicado alguns escolhas sobre os atributos. Trabalhou-se com a variância dos atributos e selecionou-se os 50% de atributos mais variados e aplicando a técnica do item 4.a. O resultado deste processo foi intrigante. Apesar de operar com metades dos atributos, a árvore de decisão do *rpart*, foi possível construir, uma árvore igual mesmo que utilizado 100% dos atributos em uma janela de tempo de 5 segundos. Onde o processamento consumiu quase que $2/3$ a menos no processamento, tanto no tempo quanto no consumo de recursos computacionais. As figuras 1 e 2 demonstram as mesmas árvores.

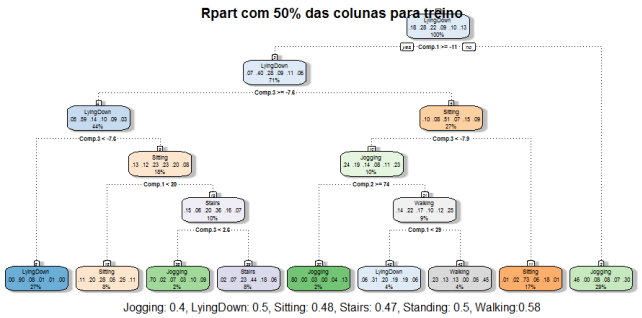

Figura 1: Aplicação de Árvore de Decisão com 50% dos atributos e com janela de 5 segundos.

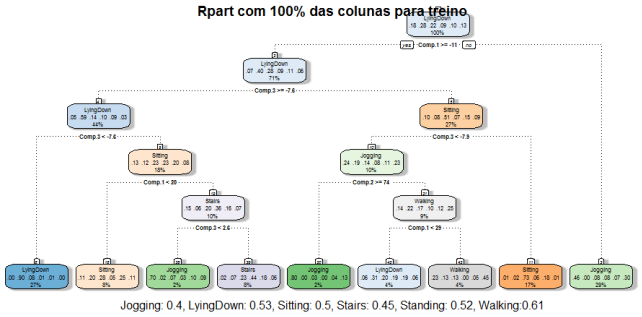

Figura 2: Aplicação de Árvore de Decisão com 100% dos atributos e com janela de 5 segundos.

Para o caso de *Random Forest*, utilizou-se 25% dos atributos (nas janelas de tempo de 5 e 10 segundos) devido ao tempo de processamento.

Na figura 3, constata-se a geração de até 500 árvores com os cortes aleatórios de acordo com o erro calculado em

cada árvore. O que torna essa figura interessante é o menor erro obtido somente utilizando a composição da Assimetria, Curtose, Média, Mínino e Máximo de cada um dos registros. Para esse experimento utilizou-se a janela de 10 segundos (200 atributos concatenados) e que o erro obtido foi melhor que a figura 6 que trabalha com mais atributos e mesma janela de tempo.

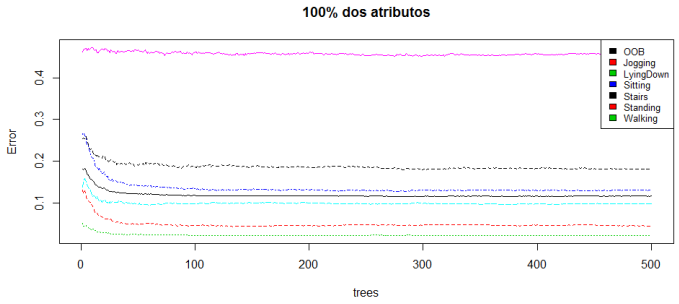

Figura 3: Composição de valores extraídos de cada amostra com Assimetria (*Skewness*), Curtose (*Kurtosis*), Média, Mínimo e Máximo.

Outros experimentos estão presentes no apêndice A nas figuras 8, 7. Para a exploração utilizando a transformada de *Fourier*, foi usando *Random Forest*, por ter resultado em melhores valores finais e demonstrados nas figuras 9 e 12.

As matrizes de confusão, exemplificadas nas figuras 4 e 5, demonstram o dados com janela de 10 segundos, aplicação de: *PCA*, *FFT* para trabalhar com dados no domínio de frequência, reaplicação de *PCA* e posterior utilização de *Random Forest*. Esse procedimento foi feito utilizando base de treino e testando com base de validação (figura 4) e invertendo as bases na sequência (figura 5).

	Class: Sitting	Class: Jogging	Class: Standing	Class: Walking	Class: Stairs	Class: LyingDown
Sensitivity	0.62628	0.6771	0.13514	0.2181	0.14174	0.000
Specificity	0.67989	0.7068	0.91098	0.9117	0.93260	1.000
Pos Pred Value	0.22109	0.2520	0.23981	0.3318	0.51059	NaN
Neg Pred Value	0.92614	0.9375	0.83523	0.8530	0.68655	0.925
Prevalence	0.12670	0.1273	0.17205	0.1673	0.33160	0.075
Detection Rate	0.07935	0.0862	0.02325	0.0365	0.04700	0.000
Detection Prevalence	0.35890	0.3421	0.09695	0.1100	0.09205	0.000
Balanced Accuracy	0.65309	0.6920	0.52306	0.5649	0.53717	0.500

Figura 4: Resultado da Matriz de Confusão com treino e validação.

	Class: Stairs	Class: Walking	Class: Jogging	Class: Standing	Class: Sitting	Class: LyingDown
Sensitivity	0.95486	0.16927	0.08484	0.008226	0.04034	0.0000
Specificity	0.09303	0.98792	0.98758	0.998837	0.96315	1.0000
Pos Pred Value	0.10671	0.57935	0.52339	0.432432	0.26127	NaN
Neg Pred Value	0.94782	0.92365	0.87034	0.903371	0.75652	0.6713
Prevalence	0.10190	0.08950	0.13850	0.097250	0.24415	0.3287
Detection Rate	0.09730	0.01515	0.01175	0.000800	0.00985	0.0000
Detection Prevalence	0.91185	0.02615	0.02245	0.001850	0.03770	0.0000
Balanced Accuracy	0.52394	0.57860	0.53621	0.503532	0.50175	0.5000

Figura 5: Resultado da Matriz de Confusão com validação e treino (ordem inversa para treino e validação).

V. Conclusões

Para este trabalho, houve um grande aprendizado em como explorar a combinação dos atributos para novas componentes como:

(a) utilizando curtose, assimetria, média, mínimo e máximo (e que resultou em melhores resultados)
(b) trabalhar com com as transformações dos dados para o domínio das frequências, utilizando *Fast Fourier Transform* (*FFT*);
(c) aplicação de tratamentos para redução de dimensionalidade utilizando *PCA*, e para cada nova etapa a ser aplicada, resguardando e não descaracterizando os dados para classificação adequada;
(d) Utilizar técnicas como o Árvore de Decisão mais comum, presente no pacote do *R* (*rpart*) e *Random Forest*.
(e) trabalhar a variação das quantidades de atributos utilizados, escolhendo os atributos através da variância;
(f) explorar janelas de tempo de diferentes tamanhos (5 e 10 segundos) e investigar os tamanhos dos passos, enunciados como *offset*.

Etapas mais interessantes perante os resultados:

- Devido ao processamento rápido, tanto os itens (a) e (b) (este um pouco mais custoso em termos de processamento), mas ainda com ótimos resultados;
- trabalhar com a variância aplicada em cada um dos atributos e capturar somente os mais variados, tornou-se rápido em termos de processamento e tempo, e será mais explorado no futuro com novas implementações;
- com janelas de tempo diferenciadas, houve somente um ganho pequeno nos resultados de acurácia balanceada na Matriz de Confusão dos dados explorados.

Outros resultados podem ser encontrado no apêndice A.

Referências

[1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*, ser. Springer Series in Statistics. Springer, 2009. [Online]. Available: https://books.google.com.br/books?id=tVIjmNS3Ob8C

[2] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[3] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

Apêndice

As figuras presentes neste apêndice reflete os experimentos realizados e suas validações em janelas de tempo de 5 segundos (100 atributos) e 10 segundos (200 atributos).

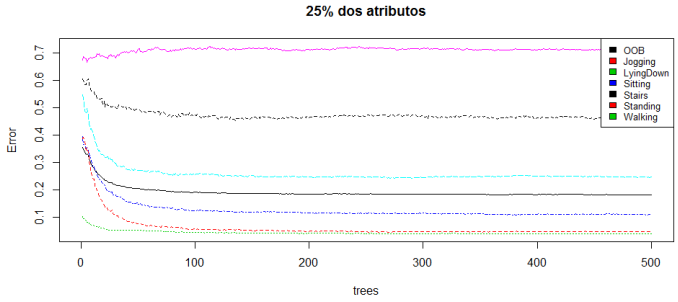

Figura 6: Resultado de erro calculado em *Random Forest* com 25% dos atributos com janela de 10 segundos.

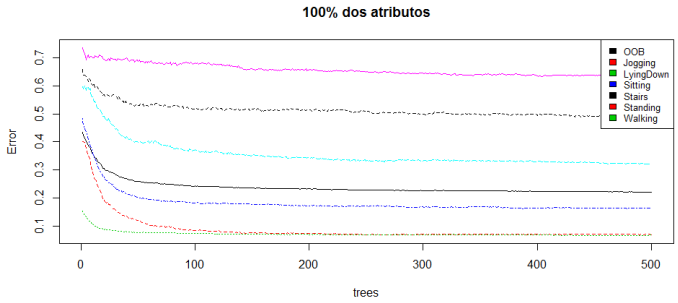

Figura 7: Resultado de erro calculado em *Random Forest* com 100% dos atributos com janela de 5 segundos.

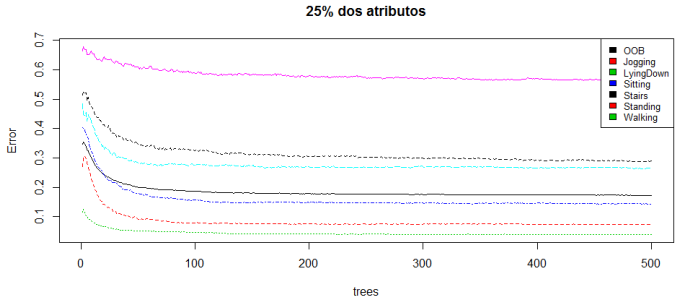

Figura 8: Resultado de erro calculado em *Random Forest* com 25% dos atributos com janela de 5 segundos.

	Class: Walking	Class: Stairs	Class: Sitting	Class: Standing	Class: Jogging	Class: LyingDown
Sensitivity	0.6557	0.18096	0.4493	0.2695	0.19385	0.004253
Specificity	0.7639	0.90438	0.7419	0.8352	0.92551	0.993487
Pos Pred Value	0.4070	0.41372	0.2002	0.2796	0.25945	0.063492
Neg Pred Value	0.8998	0.74756	0.9035	0.8281	0.89505	0.905756
Prevalence	0.1981	0.27160	0.1258	0.1918	0.11865	0.094050
Detection Rate	0.1299	0.04915	0.0565	0.0517	0.02300	0.000400
Detection Prevalence	0.3192	0.11880	0.2822	0.1849	0.08865	0.006300
Balanced Accuracy	0.7098	0.54267	0.5956	0.5523	0.55968	0.498870

Figura 9: Matrix de Confusão com 25% dos atributos com janela de 5 segundos, trabalhando com treino e validação.

Statistics by Class:

	Class: Walking	Class: Stairs	Class: Standing	Class: Jogging	Class: Sitting	Class: LyingDown
Sensitivity	0.23893	0.8024	0.45954	0.02411	0.03301	0.0000
Specificity	0.99095	0.6581	0.43406	0.99597	0.98837	1.0000
Pos Pred Value	0.73203	0.2218	0.08752	0.35088	0.50000	NaN
Neg Pred Value	0.92640	0.9648	0.87177	0.91859	0.74364	0.6512
Prevalence	0.09375	0.1083	0.10565	0.08295	0.26055	0.3488
Detection Rate	0.02240	0.0869	0.04855	0.00200	0.00860	0.0000
Detection Prevalence	0.03060	0.3918	0.55470	0.00570	0.01720	0.0000
Balanced Accuracy	0.61494	0.7302	0.44680	0.51004	0.51069	0.5000

Figura 10: Matrix de Confusão com 25% dos atributos com janela de 5 segundos, treinando com validação e validando com treino.

OOB estimate of error rate: 19.58%
Confusion matrix:

	Jogging	LyingDown	Sitting	Stairs	Standing	Walking	class.error
Jogging	1502	0	8	4	15	130	0.09463532
LyingDown	0	6640	195	74	52	15	0.04816514
Sitting	70	195	4517	209	137	83	0.13317981
Stairs	6	30	346	1603	0	181	0.25992613
Standing	55	61	1204	131	566	96	0.73213441
Walking	413	13	49	76	68	1256	0.33013333

Figura 11: Trabalhando com 100% dos atributos, usando *FFT* e uma janela de tempo de 5 segundos.

	Class: Walking	Class: Stairs	Class: Sitting	Class: Standing	Class: Jogging	Class: LyingDown
Sensitivity	0.6059	0.1524	0.6839	0.1194	0.27686	0.001595
Specificity	0.8089	0.8943	0.6490	0.9271	0.88586	0.999945
Pos Pred Value	0.4393	0.3497	0.2189	0.2800	0.24616	0.750000
Neg Pred Value	0.8925	0.7389	0.9345	0.8161	0.90099	0.906081
Prevalence	0.1981	0.2716	0.1258	0.1918	0.11865	0.094050
Detection Rate	0.1201	0.0414	0.0860	0.0229	0.03285	0.000150
Detection Prevalence	0.2732	0.1184	0.3929	0.0818	0.13345	0.000200
Balanced Accuracy	0.7074	0.5234	0.6664	0.5233	0.58136	0.500770

Figura 12: Matrix de confusão com 100% dos atributos, usando *FFT* e uma janela de tempo de 5 segundos.

		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	**True positive**	**False positive** (Type I error)
	Predicted condition negative	**False negative** (Type II error)	**True negative**
	Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	True positive rate (TPR), Sensitivity, Recall = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$
		False negative rate (FNR), Miss rate = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	True negative rate (TNR), Specificity (SPC) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$

Figura 13: Matrix de Confusão (https://en.wikipedia.org/wiki/Confusion_matrix)