
Coisas estranhas que podemos fazer com dados coletados em dispositivos móveis: bye-bye privacidade!

Human Activity Recognition

Fabrício J. Barth

fabricio.barth@gmail.com

Faculdade BandTec e Watson Group IBM

Novembro de 2014

Exemplos de dispositivos e dados que podem ser coletados

Celulares



Coleta informações sobre:

- Localização (latitude e longitude);
- Movimentação (acelerômetro, giroscópio);
- Ambiente (audio, proximidade, luminosidade);
- Social (histórico de ligações, contatos).

Relógios



Coleta informações sobre:

- Frequência cardíaca;
- Pressão arterial.

Outras aplicações (talvez menos úteis)



Monitora atividades físicas



Monitora:

- Que atividade está sendo realizada;
- Qual a duração;
- Qual a frequência.

Disney



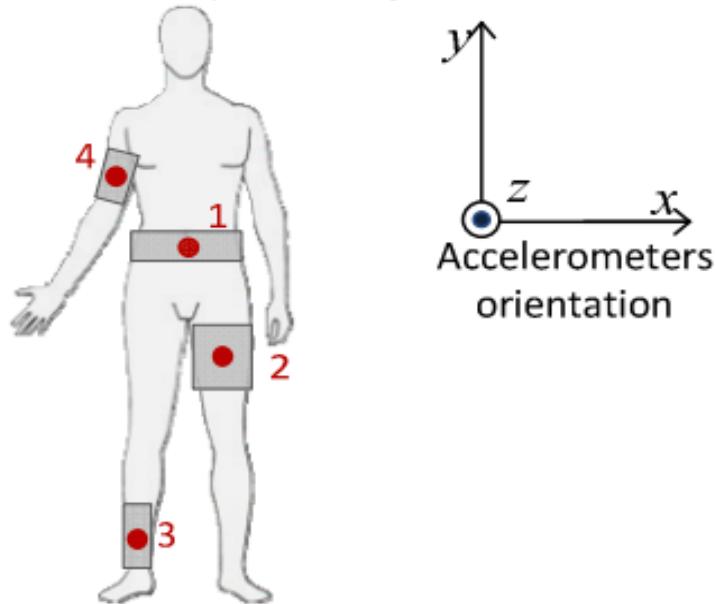
Monitora:

- Quando entrou e saiu do parque;
- Por onde andou;
- O que e quando comprou;
- Em quais parques foi e quando foi.

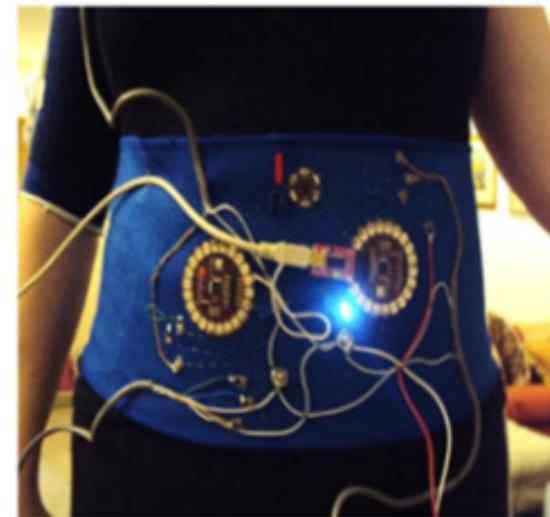
O que podemos fazer
com os dados coletados?

Primeiro exemplo de aplicação [2]

Scheme of positioning and orientation



User wearing the device



O objetivo deste exemplo é construir um classificador capaz de dizer que atividade (**sitting**, **sitting down**, **standing**, **standing up**, **walking**) uma pessoa está realizando a partir de dados coletados de acelerômetros presentes no corpo desta pessoa.

Pipeline do processo para reconhecimento de atividades [1]



Dados coletados e filtrados

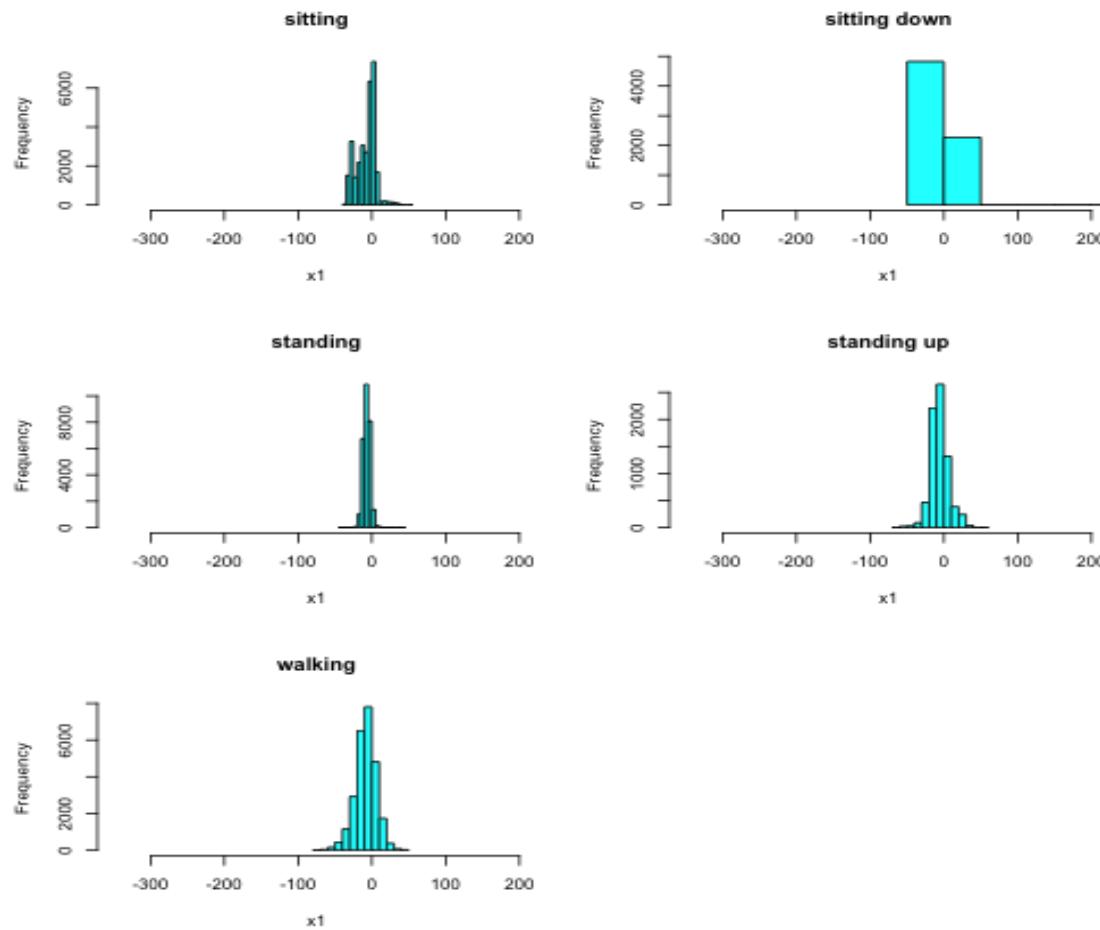
- Dados coletados a partir de 4 tri-axial acelerômetros.
- Foram consideradas janelas de tempo de 1 segundo, com overlapping de 150ms.
- Medidas de roll, pitch e módulo de aceleração foram adquiridas.
- A amostra dentro da janela de tempo foi agrupada e atributos foram gerados (i.e., variância, média).
- Foram filtrados 12 atributos finais - três para cada acelerômetro.

Construção do classificador

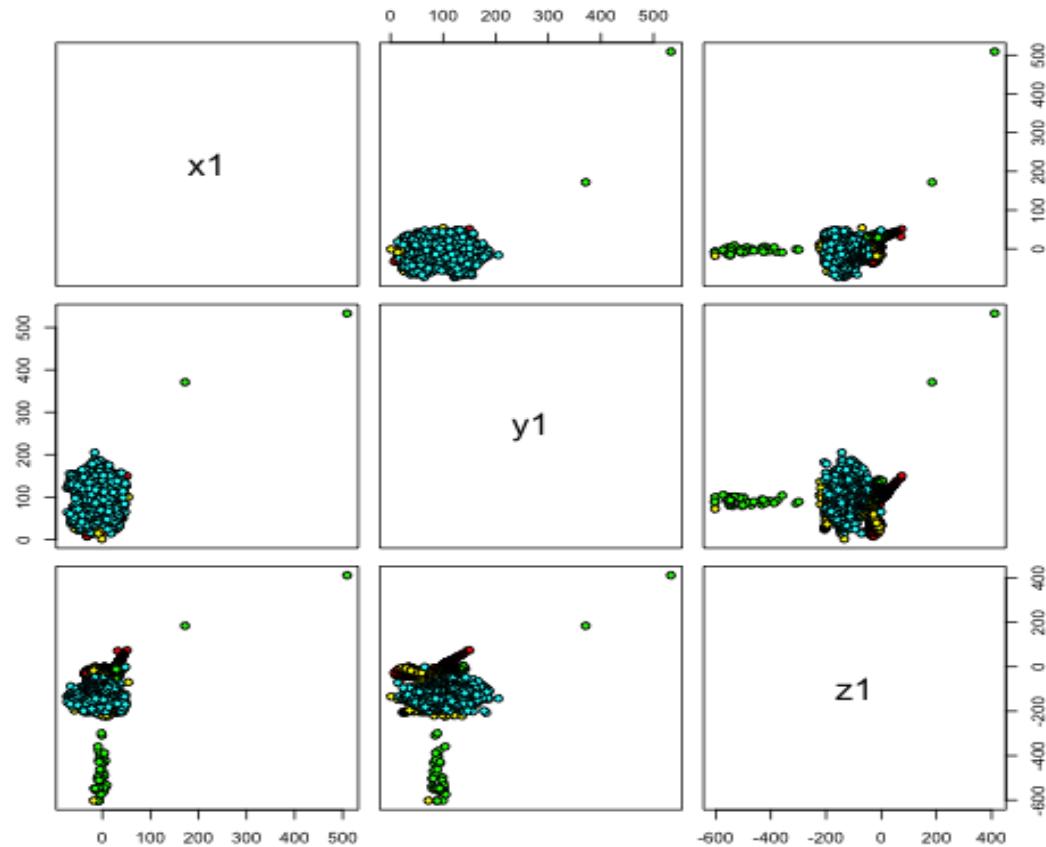
- O dataset possui 165.633 exemplos e 19 atributos:
 - ★ user, gender, age, how_tall_in_meters, weight, body_mass_index, $x_1, y_1, z_1, \dots, x_4, y_4, z_4$, **class**
- O dataset foi dividido em conjunto de treinamento e teste, respeitando a proporção dos valores do atributo **class**.

Referência: <http://rpubs.com/fbarth/har01>

Alguns resultados da análise descritiva



Distribuição das atividades levando-se em consideração dados do sensor



Algoritmo utilizado para criação do modelo: **Random Forest**

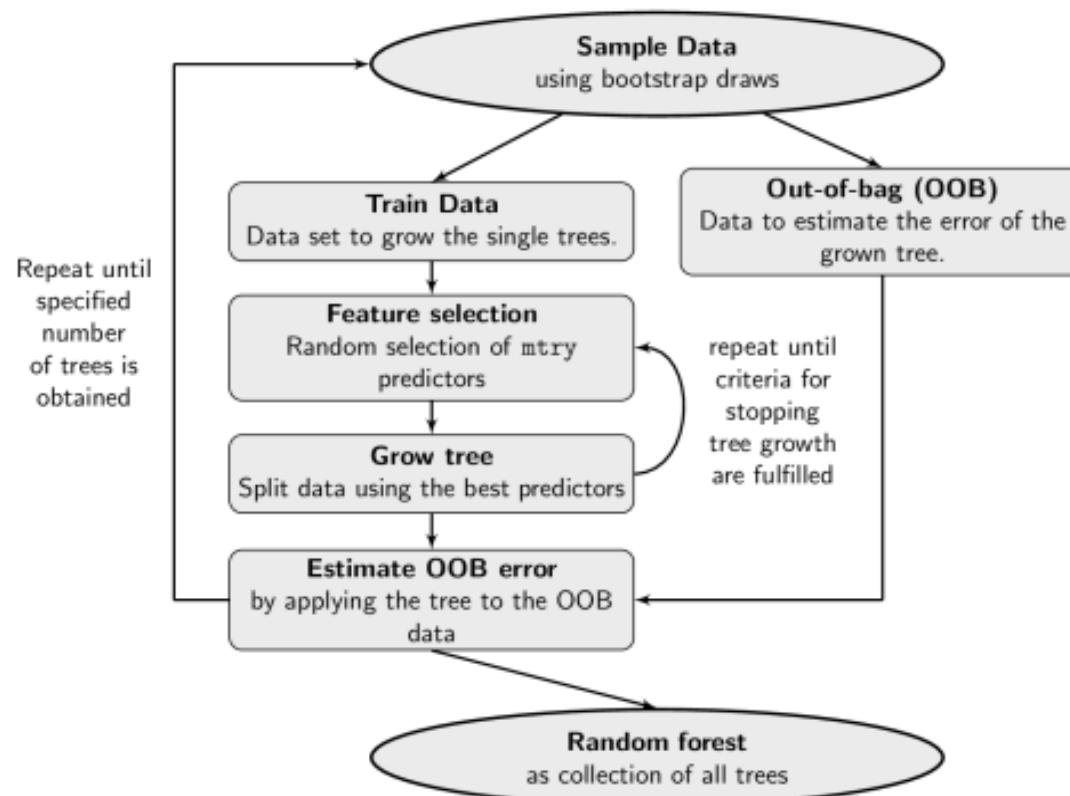


Figure 1: Random Forest Algorithm

Criando o modelo...

```
library(randomForest)
```

```
## randomForest 4.6-7  
## Type rfNews() to see new features/changes/bug fixes.
```

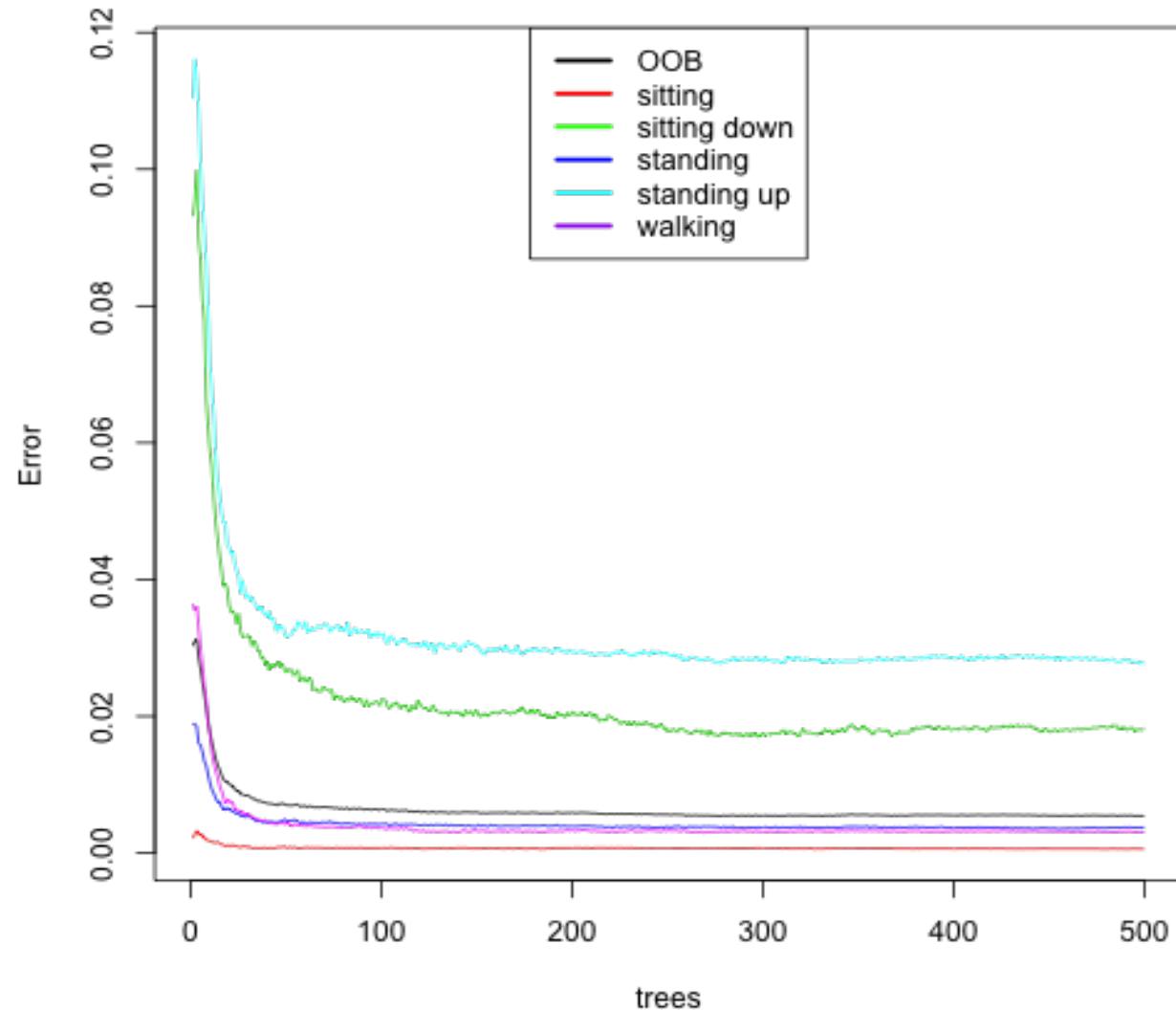
```
formula <- class ~ x1 + y1 + z1 + x2 + y2 + z2 + x3 + y3 + z3 + x4 + y4 + z4  
model <- randomForest(formula, data = treinamento, do.trace = 100, importance = TRUE)
```

```
## ntree      OOB      1      2      3      4      5  
##   100: 0.64% 0.07% 2.25% 0.43% 3.20% 0.37%  
##   200: 0.59% 0.07% 2.07% 0.39% 2.94% 0.34%  
##   300: 0.54% 0.07% 1.76% 0.37% 2.81% 0.30%  
##   400: 0.55% 0.06% 1.82% 0.37% 2.85% 0.32%  
##   500: 0.54% 0.06% 1.82% 0.37% 2.77% 0.30%
```

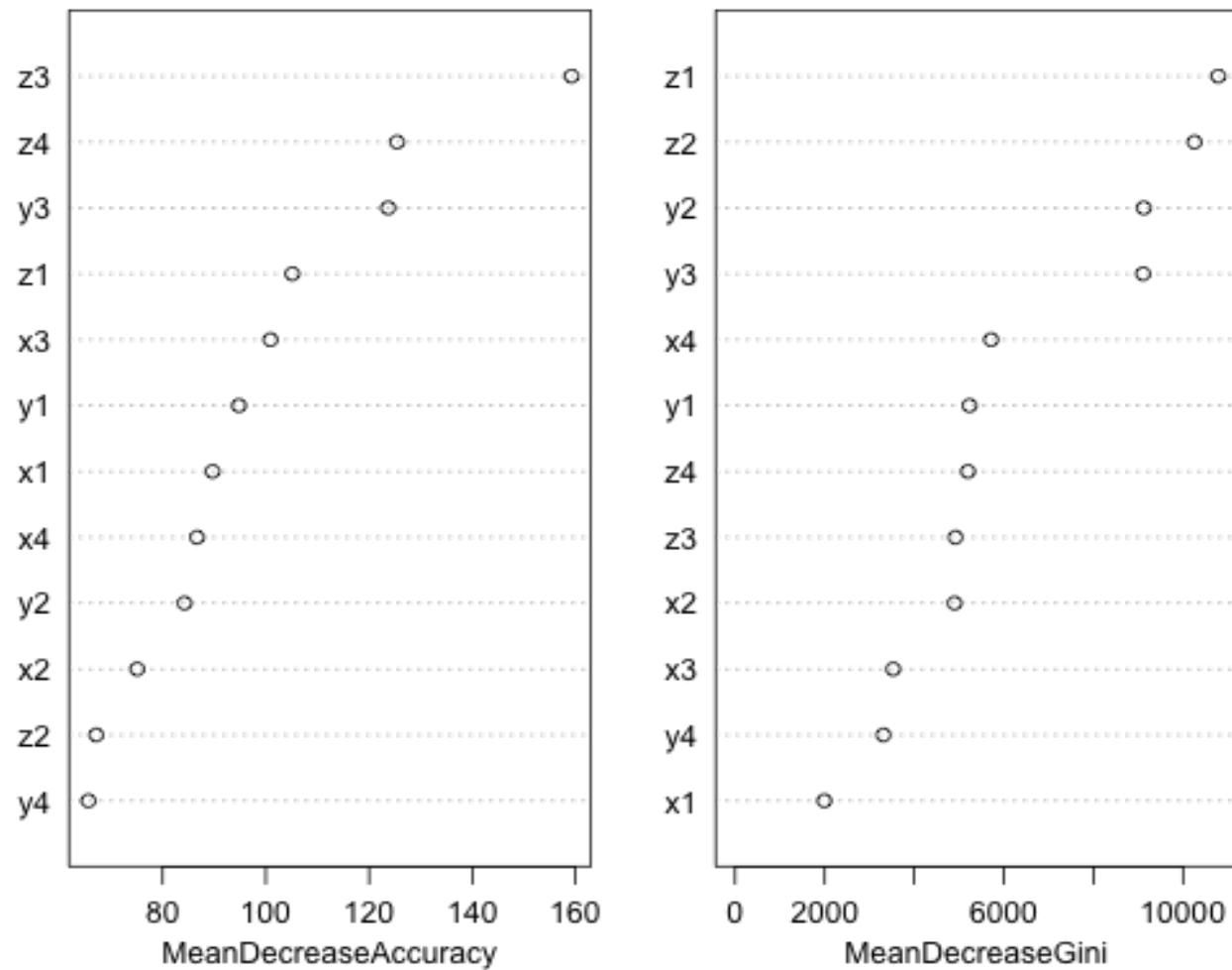
```
model
```

```
##  
## Call:  
##   randomForest(formula = formula, data = treinamento, do.trace = 100,      importance =  
TRUE)  
##           Type of random forest: classification  
##                         Number of trees: 500  
## No. of variables tried at each split: 3  
##  
##           OOB estimate of  error rate: 0.54%  
## Confusion matrix:  
##             sitting  sittingdown standing standingup walking class.error  
## sitting       30361        4         0        14         0  0.0005925  
## sittingdown     3        6968       23        60        43  0.0181767  
## standing       0         0      28316        8        98  0.0037295  
## standingup      8        77       53       7243       68  0.0276547  
## walking        0        14       47        18      25955  0.0030345
```

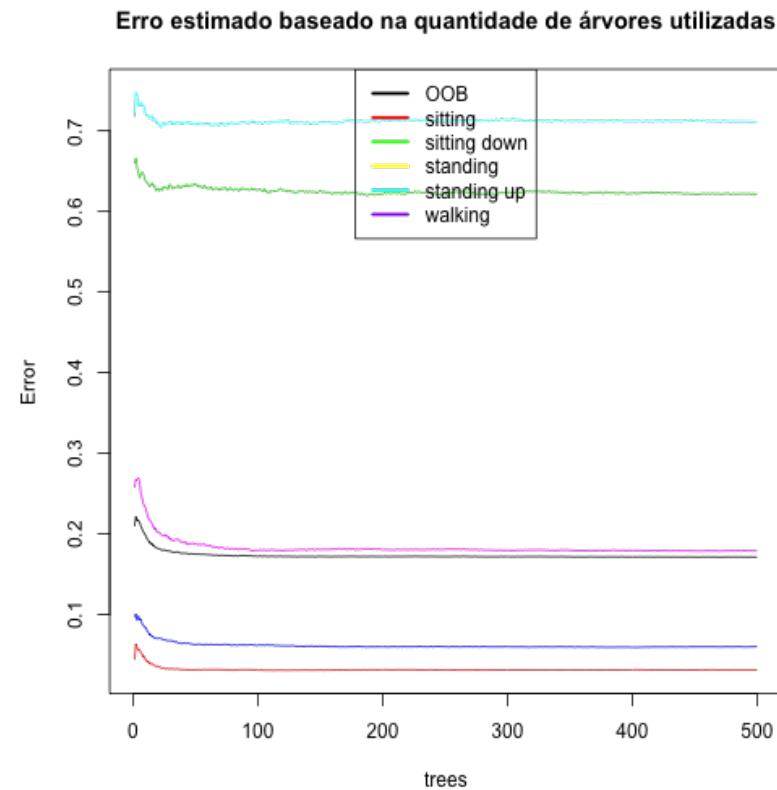
Erro estimado baseado na quantidade de árvores utilizadas



Importância dos atributos ao classificar as observações



Outro modelo



Modelo que utiliza apenas dados do acelerômetro localizado na cintura (Erro estimado: 17.12%)

Validando o modelo completo com o conjunto de testes

```
testPred <- predict(model, newdata = teste)
t <- table(testPred, teste$class)
confusionMatrix(t)
```

```
## Confusion Matrix and Statistics
##
##          sitting sittingdown standing standingup walking
##  sitting       20236           2        0         3        0
##  sittingdown      0        4651        0        54       20
##  standing        0        14     18879        39       43
##  standingup       16        39         6       4819        9
##  walking          0        24        63        51     17284
##
## Overall Statistics
##
##               Accuracy : 0.994
##                 95% CI : (0.994, 0.995)
##   No Information Rate : 0.306
##   P-Value [Acc > NIR] : <2e-16
##
```

Resultados do trabalho original [2]

We used AdaBoost with 10 iterations and configured the C4.5 tree for a confidence factor of 0.25. The overall recognition performance was of 99.4% (weighted average) using a 10-fold cross validation testing mode, with the following accuracies per class: “sitting” 100%, “sitting down” 96.9%, “standing” 99.8%, “standing up” 96.9%, and “walking” 99.8%. The confusion matrix is presented in Table 3.

Table 3. Confusion Matrix

Predicted class					
Sitting	Sitting down	Standing	Standing Up	Walking	Actual class
50,601	9	0	20	1	Sitting
10	11,484	29	297	7	Sitting down
0	4	47,342	11	13	Standing
14	351	24	11,940	85	Standing up
0	8	27	60	43,295	Walking

The results obtained in this research are very close to the top results of the literature (99.4% in [14], and 99.6% in [15]), even though, it is hard to compare them. Each research used a different dataset, a different set of classes, and different test modes.

Será que é possível determinar quem está realizando a atividade?

-
-

References

- [1] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and JorgeL. Reyes-Ortiz. Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In José Bravo, Ramón Hervás, and Marcela Rodríguez, editors, *Ambient Assisted Living and Home Care*, volume 7657 of *Lecture Notes in Computer Science*, pages 216–223. Springer Berlin Heidelberg, 2012.
- [2] Wallace Ugulino, Débora Cardador, Katia Vega, Eduardo Velloso, Ruy Milidiú, and Hugo Fuks. Wearable computing: Accelerometers' data classification of body postures and movements. In LelianeN. Barros, Marcelo Finger, AuroraT. Pozo, GustavoA. Giménez-Lugo, and Marcos Castilho,

editors, *Advances in Artificial Intelligence - SBIA 2012*, Lecture Notes in Computer Science, pages 52–61. Springer Berlin Heidelberg, 2012.