Human Activity Recognition Using Smartphone Data

Stanford University

Fjoralba Shemaj, Nicholas Canova

Problem

As more sensors are being built into mobile phones to measure our movements, positioning and orientation, the opportunity to understand this data and make improvements in our daily lives increases. Using sensor data obtained from study participants performing six different activities (walking, walking upstairs, walking downstairs, sitting, standing, laying), our objective is to build a model that accurately classifies which of these activities is being performed.

Our project falls into the scope of Activity Recognition, a field that offers many benefits and enables many new applications, for example step counters on your smartphone, as well as applications for elderly assistance and personal health monitoring.

Dataset

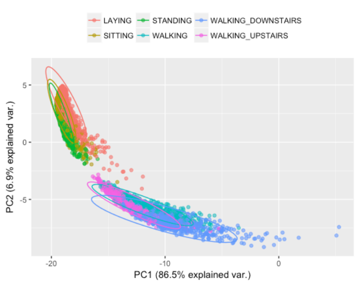
Our data comes from experiments with 30 participants, each of whom wore a Samsung Galaxy S2 smartphone containing an accelerometer and a gyroscope while performing the six activities mentioned previously. The smartphone collected 3-axial linear acceleration and angular velocity measurements, each at a constant rate of 50 hertz. Each individual observation is a construction of sensor signals received over a 2.56 second interval, with consecutive observations overlapping by 50% in time.

Features for the dataset were then constructed by calculating variables from the accelerometer signals in the time and frequency domain, including the mean, standard deviation, signal magnitude area, and others. In total, each observation corresponds to 561 constructed features from the data collected.

Data Visualization

We have implemented both PCA and t-SNE to visualize our data on a two-dimensional plane, and to compare which algorithm better separates the different activities. Since the principal components are linear combinations of our features, PCA can only capture the linear structures of the features. Alternatively, t-SNE uses a different objective function that maintains the local distances between the data and captures non-linear paths.

Both algorithms effectively distinguish between activities of motion (walking, walking upstairs, walking downstairs) and static activities (sitting, laying, standing). Within static activities, t-SNE does better in separating standing from sitting and laying than does PCA. Further, both graphics clearly show more variability in activities of motion than in static activities, which is consistent with our intuition that people tend to have subtle differences in the way they walk, but generally sit, stand and lay similarly.



Conclusions

The similar test errors from the various models suggests that increasing the complexity of the model doesn't necessarily improve the performance. Models with linear decision boundaries (LDA, multinomial, and SVM with linear kernel) did perform slightly better than GBM and SVMs with radial basis and polynomial kernels. Visualizations show that certain activities overlap significantly, and that there are differences in the way they are performed by individuals. The leave-one-out cross validation supports this, and we propose that a larger training set with more than 30 participants could significantly improve the performance of our model, in particular a training set with a more diverse set of participants.

Models Analyzed

Our main objective for this problem is to construct a highly accurate classifier that generalizes well on data from new users. To do this, we have tested the performance of different classifiers on our data, and assessed why some performed well while others performed poorly. Algorithms implemented, as well as our motivation for each algorithm, include:

- Multinomial model – one of the less complex models implemented. Also serves as a good baseline for the performance of other models.
- Support vector machines – since some activities are very similar to each other, as indicated by PCA, maximizing margins when separating these similar activities is necessary for high performance.
- Generalized boosted classification models – our data is high-dimensional and there is a high level of interaction among the features, both of which GBMs tend to handle well.
- Linear discriminant analyses – a reasonable choice due to evidence of normally distributed features.

Results / Error Analysis

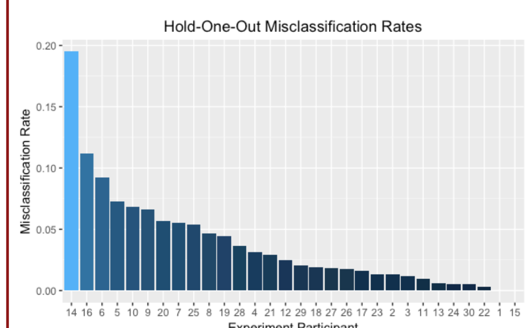
Based on our objective to construct a classifier that generalizes well on data from new users, our results are calculated from the activity predictions of users not included in the training data. Had our models been trained and tested on the same users, we likely would have received lower misclassification rates.

Our dataset contains roughly an equal number of observations for each of the six activities. Additionally, while certain applications of Activity Recognition may require that some activities be more accurately classified than others, given our general analysis we chose to weight each activity equally. As a result, we use the overall misclassification rate on the test data as our performance measure. The train and test errors for each of our analyses are displayed below, along with a confusion matrix for the SVM with linear kernel:

Machine Learning Algorithm	Train Error	Test Error
SVM with Linear Kernel	0.60%	3.66%
SVM with Radial Basis Kernel	0.33%	3.70%
SVM with Polynomial Kernel	0.64%	3.97%
Gradient Boosted Trees	0.00%	5.29%
Linear Discriminant Analysis	1.43%	3.77%
Multinomial Model	0.42%	3.33%

Activity	Laying	Sitting	Standing	Walking	Down	Up
Laying	**537**	0	0	0	0	0
Sitting	0	**434**	17	0	0	0
Standing	0	55	**515**	0	0	0
Walking	0	0	0	**492**	4	18
Downstairs	0	0	0	3	**410**	2
Upstairs	0	2	0	1	6	**451**
Accuracy %	100	88.4	96.8	99.2	97.6	95.8

The most difficult activity to classify is sitting, which has an overall misclassification rate of 11.6% and is incorrectly classified as standing. Alternatively, standing is occasionally misclassified as sitting, and walking upstairs is occasionally misclassified as walking.



Intuitively, the accuracy of the model when applied to a new user may depend on how similar that new user is (the specifics of their motions, movements, etc.) relative to the users that the model was trained on. To evaluate this, we performed leave-one-out cross validation with linear kernel SVM, and the results confirmed our expectation. The error rate by user ranged from 0.0% to 19.5%, with results varying in between.