A predictive analysis of Accelerometer data to discern type of activity

Introduction:

An accelerometer is a device that is used to measure the actual acceleration of the body on which it is placed [1], where actual acceleration is defined as the weight experienced by a small test mass present on the device. In modern times, these devices have become an integral part of hand held cellular devices. An accelerometer records the acceleration in the x, y and z directions, which can further be divided into a number of sub-fields such as the body acceleration, acceleration frequency, acceleration jerk etc. [2]. The present dataset that was analyzed consisted of numerous such acceleration measurements and the corresponding activity that was being carried out with each of the measurements namely, **sitting, laying, standing, walking, walking up or walking down**. The aim of the present analysis was to develop a set of classifiers that could be used to predict the type of activity being undertaken by an individual, being given the acceleration data from the Smartphone of the individual. The applications of this technique could be in the fitness sector, in healthcare, in advertising and in other areas too.

In the present analysis a large dataset was given with accelerometer measurements for 21 individuals performing all the six tasks stated above. The total number of rows of data was 7352 and there were 561 types of accelerations data recorded per observation. To conduct an unbiased analysis, the data set was divided into two parts, the test set and training set. Initially, exploratory analysis was carried out on the training set to recognize the fields which could distinguish the data well. Then, based on the exploratory analysis, statistical analysis was carried out on the data using decision tree bagging and boosting methods and also a combined voting method using the two techniques. The result of the analysis showed that in this case, the Random Forest approach [3] showed the best results among all three approaches with a misclassification error of 4.85% on the test data.

Methods:

Data Collection

The data was downloaded from the following URL: https://spark-public.s3.amazonaws.com/dataanalysis/samsungData.rda on March 6, 2013 using the R programming language [4]. The data was taken from Smartlab's experimental data at the DITEN University, Geneva [2].

Exploratory Analysis

The first step of the exploratory analysis was pre-processing the raw data to make it easier to analyze. Firstly, the data was checked for missing and infinite values. No such data was found. Then the columns were renamed with generic names to make the accessing of individual columns simpler. The activity data was converted to factor form to make it simpler to analyze this data using R modules. Then the data was divided into the training and the test sets. The test set is the one on which the accuracy of the predictive algorithm is finally tested and in this case it contained data from subjects 27, 28, 29 and 30. The training set is the one on which the analyses are carried out, i.e. where the predictive algorithm is trained. This consisted of data of all the other subjects excluding the four subjects in the test set. A few exploratory graphs were plotted with some of the columns in the data set versus the data index. These were scatter plots showing the column acceleration data on the y-axis and the index number of the data or another acceleration parameter on the x-axis, colored according to the activity being carried out by the individual. These graphs showed, as also shown in Prof. Leek's videos, that some fields, such as *tBodyAcc-mean()* in x and y directions did not help separate the data, while *tBodyAcc-max()* in the x and y directions separated the sedentary activities were from those involving motion. Fig. 1 (a) attached separately shows one such graph, plotting tGravity Acceleration Max in X and Y directions, colored by the type of data. This graph shows that these parameters can easily separate between the sedentary activities like Laying and Standing and the other activities. I tried *glm()* with the Gaussian linear modeling on some variables against the output data but did not get promising results. This is because the data did not have linear trends but rather grouped non-linear tendencies. Hence I decided to proceed to use decision trees in my analysis to better separate out the data.

Statistical Modeling

Initially I attempted a k-fold bagged tree approach as described in the course lectures. I used a ten-fold approach and got a misclassification error of 16.52%. As per the lectures, the Random Forest method is essentially a bagged tree approach but includes some additional steps which optimize the process and also make it more robust, though it might bias the prediction analysis to the training data. This approach picks a fraction of the data and a fraction of the input variables and creates a decision tree. This process is repeated multiple times and the large number of tress which are made (500 in the present analysis) all vote in the final prediction to give accurate predictions. Another approach which I used to increase the number of classifiers was the gradient boosting tree approach [5]. In this approach there are some inbuilt loss functions of residuals which are calculated and at each step a variable is incrementally added to the expression until the loss function minimizes. This process is repeated in a stage wise manner till the loss function converges. The third approach I applied was simply a combination of the above two techniques. This may or may not improve the overall analysis, so I decided to build all three models and see which one of the three models gives the best results.

Results:

The following are the results of the various models that were applied. Initially linear models were applied, but when the result from there outputs were not too accurate, it could be concluded that the data was not of the form that could be properly fitted with a line or a curve. Following this three approaches were applied: Random Forest, Gradient Boosting and a combination of the two using voting methods [6].

1. Generalized Linear Model:

Initially, a generalized linear model considering only the variables which had a P-value of less than 0.001 (significant association) using a Gaussian model was applied. The misclassification error was set as the criterion for deciding the accuracy of a prediction model. Since the linear model gave a numeric output, its prediction of the activity data of the training set was rounded off to a whole number. This was then compared to the actual activity values of the training set and a misclassification error of 38.3% was observed.

2. Random Forest:

The random forest method was applied on the training set, considering all 561 columns of accelerometer data. Predictions were calculated for the training set and they were stored both as the predicted outcome activity and as a matrix containing the probabilities of each of the rows of data being a type of activity. The importance function applied to the random forest model gave the relative importance of each of the columns of data in terms of the magnitude of the mean decrease of the Gini coefficient associated with the model. On the basis of this, the important parameters were angle(X, gravityMean), angle(Y, gravityMean), fBodyAccMag-energy(), fBodyGyro-maxInds-Z, fBodyAccJerk-bandsEnergy()-1,16, tGravityAcc-arCoeff()-Y,2, tGravityAcc-energy()-Y, tGravityAcc-energy()-X", tGravityAcc-min()-Y, tGravityAcc-min()-X, tGravityAcc-max()-Y, tGravityAcc-max()-X, tGravityAcc-mean()-Y, tGravityAcc-mean()-X and tBodyAcc-correlation()-X,Y. One of the trees of the random forest has been plotted in Fig. 1 (b). The tree shows that at the initial splits, Laying, Sitting and Standing get separated out while it is more difficult to separate Walk, Walk up and Walk down.

This model was applied to the training set, and misclassification errors were 97/5867, or 1.65% misclassification error.

3. Gradient Boosted Trees

This method was used to provide an independent classifier to the predictive model. A drawback of this approach was that it required more pre-processing of the data, as this (as understood by reading from resource [5]) method can only report output data as true and false i.e. as binary outcomes. Hence six columns were added to the training data, isSitting, isLaying, is Standing, isWalk, isWalkup, isWalkdown, corresponding to the kind of activity being performed by the person in that row. Thus, if the ith row had data of a person sitting, isSitting[i] would be TRUE and all others would be FALSE. Gradient boosting was applied for each of these columns versus the data. Then a prediction of the training set was calculated using voting method, using all six of the boosted trees. The calculated misclassification error was observed to be 151/5867 = 2.57%.

4. Applying both the models above using averaging of results

Both the above approaches have been combined into a single model and the prediction probability values for each type of activity was calculated by taking the average from the two above models. It was noticed in this case that having two classifiers made no improvement in classification in this data set with the percentage of misclassification error being 105 errors out of 5867 data points corresponding to misclassification errors of 1.79%. A possible cause for this could be that both methods are tree based, and perhaps an independent classifier might have had more success in improving the results.

5. Application of the three models on the test set

All three models were applied on the test set finally, to see which algorithm was superior, when applied to data which was independent of the training set. These results are tabulated below.

Table 1: Misclassification error when predictive algorithms were applied to the test set

Mode Type	Misclassifications	Total Data Points	Misclassification Error
1. Random Forest	72	1485	4.85%
2. Gradient Boosted Tree	109	1485	7.34%
3. Combined 1. And 2.	93	1485	6.26%

Based on these results the most accurate results were given by the **Random Forest method**. Fig. 1 (c) shows the test data plotted with the same axes as Fig. 1 (a), colored by the activity. The size of points is enlarged when the point is incorrectly classified. It can be observed that the misclassifications are mainly in separating Walk, Walk up and Walk down, as was expected from data observation during expository analysis.

Conclusion:

The present report carried out a predictive analysis of using accelerometer data from a Smart Phone to predict the type of activity being undertaken by the individual whose data is being recorded. The appropriate method for the prediction study was the Random Forest approach. This approach, when applied to the training data correctly classified the data 98.35% of the time and when it was applied once to the test set, showed correct classification 95.15% of the time. Other approaches of Gradient Boosting and combining boosting and bagging were also attempted, but did not give results as accurate as the Random Forest method.

Other methods of statistical analysis might have made the analysis more robust by finding the predictive variables which had a greater impact on the output. These methods include SVD decomposition and Support Vector Machine (SVM) methods. SVD could reduce the dimensions of the problem while SVM could have been another classifier in the combined approach. Another possible step that might have assisted the analysis was to look into the meanings of each of the variables and the possible impacts they might have on the output.

Reference

[1] Wikipedia “Accelerometer” Webpage. URL: http://en.wikipedia.org/wiki/Accelerometer. Accessed on: 03/09/13

[2] “Human Activity Recognition Using Smartphones Dataset” Webpage. URL: https://www.dropbox.com/s/rrsm7nv5j7y3rsi/analysis2info.pdf. Accessed on 03/09/13

[3] “Package ‘Random Forest’” URL: http://cran.r-project.org/web/packages/randomForest/randomForest.pdf. Accessed on 03/09/13

[4] R Core Team (2012). “R: A language and environment for statistical computing.” URL: http://www.R-project.org

[5] “Pragmatic Programming Techniques” Webpage. URL: http://horicky.blogspot.com.au/2012/06/predictive-analytics-decision-tree-and.html. Accessed on 03/09/13

[6] “Combining Predictors”, Week 7 lecture, Jeff Leek, http://coursera.org/.