Springboard Data Science Intensive
Capstone Project Final Report

Recognition of physical activities using smartphone sensors

Prepared by: Mehmet Solmaz (mesolmaz@gmail.com)
Mentor: Raghunandan Patthar

July, 2017

Introduction:
Motion sensors worn on body plays and important role recognizing human activity [1-3]. Sensing data can be obtained from smartphones in our pockets, wrist-worn motion sensors, or specially arranged sensors around the body. Recognition of activities is important to evaluate the lifestyle of an individual and perhaps to identify bad habits. Therefore, this data is connected to lifecare and healthcare applications.

Processing sensor data in real-time using smartphone's microprocessor is not a feasible option due to requirement for high data rate processing. A better approach would be identifying patterns of human activity ahead of time and use a learning model extracted from a classifier.

Simple activities (walking, running, biking) with repetitive nature can be easily identified with a single sensor data (Figure 1). However, complex activities such as eating, drinking, and smoking may require more sensory data because they are composed of specific hand gestures. Multiple sensors around the body can provide with more data to recognize the complex activities but the requirement to attach cables around the body and walking with extra electronics are not preferred for long-term activity monitoring [4]. Hence, smartphones or wrist-worn consumer products are the ultimate systems for long-term activity monitoring.

Figure 1. Activity recognition process [1].

This project investigates the Human Activities and Postural Transitions (HAPT) dataset and create models that predict 6 major and 6 transitional activities: 'WALKING', 'WALKING_UPSTAIRS', 'WALKING_DOWNSTAIRS', 'SITTING', 'STANDING', 'LAYING', 'STAND_TO_SIT', 'SIT_TO_STAND', 'SIT_TO_LIE', 'LIE_TO_SIT', 'STAND_TO_LIE', 'LIE_TO_STAND'. Creating such models can help with more accurate real-time activity tracking. Real time activity tracking helps the clients with following applications:

- Continuous monitoring of patients (Healthcare Management).
    - The doctors would be able to track the activities of critical patients for better healthcare management.
- Smart environments (Retail).
    - Tracking individuals movements could help create unique shopping plans for shoppers.
- Public surveillance (Security).
    - Activity tracking could be upgraded to trajectory tracking by including more classes and using physical characteristics of persons.

The dataset:

The data is collected from UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/datasets/Smartphone-Based+Recognition+of+Human+Activities+and+Postural+Transitions). It has two versions and the updated version includes 12 different activities instead of 6 in the initial dataset.

The dataset was obtained from 30 individuals wearing their smartphones that record the sensor data. The sensor data are triaxial linear acceleration and angular velocity signals using the phone accelerometer and gyroscope. The sensor RAW data is originally in time-series but goes through a post processing step to extract the following time and frequency domain signals (Figure 2). The final data is multivariate.

Name	Time	Freq.
Body Acc	1	1
Gravity Acc	1	0
Body Acc Jerk	1	1
Body Angular Speed	1	1
Body Angular Acc	1	0
Body Acc Magnitude	1	1
Gravity Acc Mag	1	0
Body Acc Jerk Mag	1	1
Body Angular Speed Mag	1	1
Body Angular Acc Mag	1	1

(a)

Function	Description
mean	Mean value
std	Standard deviation
mad	Median absolute value
max	Largest values in array
min	Smallest value in array
sma	Signal magnitude area
energy	Average sum of the squares
iqr	Interquartile range
entropy	Signal Entropy
arCoeff	Autorregresion coefficients
correlation	Correlation coefficient
maxFreqInd	Largest frequency component
meanFreq	Frequency signal weighted average
skewness	Frequency signal Skewness
kurtosis	Frequency signal Kurtosis
energyBand	Energy of a frequency interval
angle	Angle between two vectors

(b)

Figure 2. a) The signals obtained from accelerometer and gyroscope sensors; b) Each signal on the left has the measures on the right for computing the feature vectors.

The Features are statistical data from smartphone accelerometer and gyroscope sensors. A total of 17 signals, both time and frequency components were extracted from these sensors and statistical measures (mean, std, skewness etc.) were applied to obtained a total of 561 features.

The original dataset is in TXT format and is already split into 70-30% Train-Test split. Feature names, Target Activities and Subject ID (The volunteer who performs the activities) are provided in separate TXT files. There are a total of 8 *TXT* files: (X_test.txt, X_train.txt, y_test.txt, y_train.txt, features.txt, activity_labels.txt, subject_id_test.txt, subject_id_train.txt).

Methodology:

This is a supervised learning problem where the labels are given. It is also a classification problem where there are 12 classes. Several libraries (Pandas, numpy, scikit-learn, matplotlib, seaborn) were used for data wrangling, descriptive statistics, matrix operations, model creation and evaluation, dimensionality reduction, data visualization.

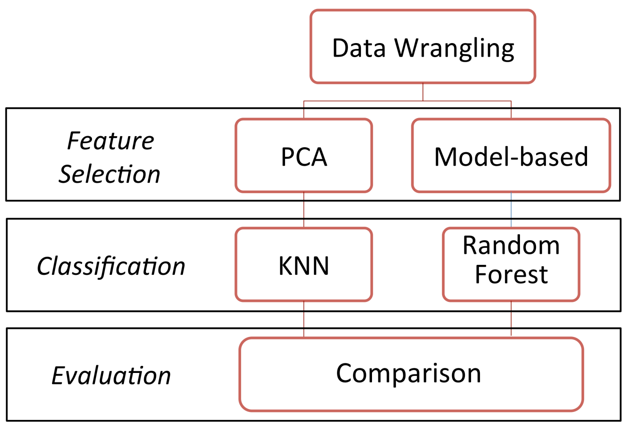


Figure 3. Methodology flow chart.

The flow chart of the methodology is given above. After combining the dataset and extracting distributions of target activities, I chose to implement 2 different methods where Feature Selection is followed by Classification (KNN and RandomForest). At the end, we compare the performance of each method during Model Evaluation using a single-metric: *f1-score*.
Both classifiers were tested with the whole (561 features) and reduced (n<561) datasets. Cross Validation and hyperparameter tuning were used on both Classifiers. CV is an approach to prevent over-fitting. The best choice of parameters for both KNN and RF were tested with k-fold CV.

Data Wrangling:

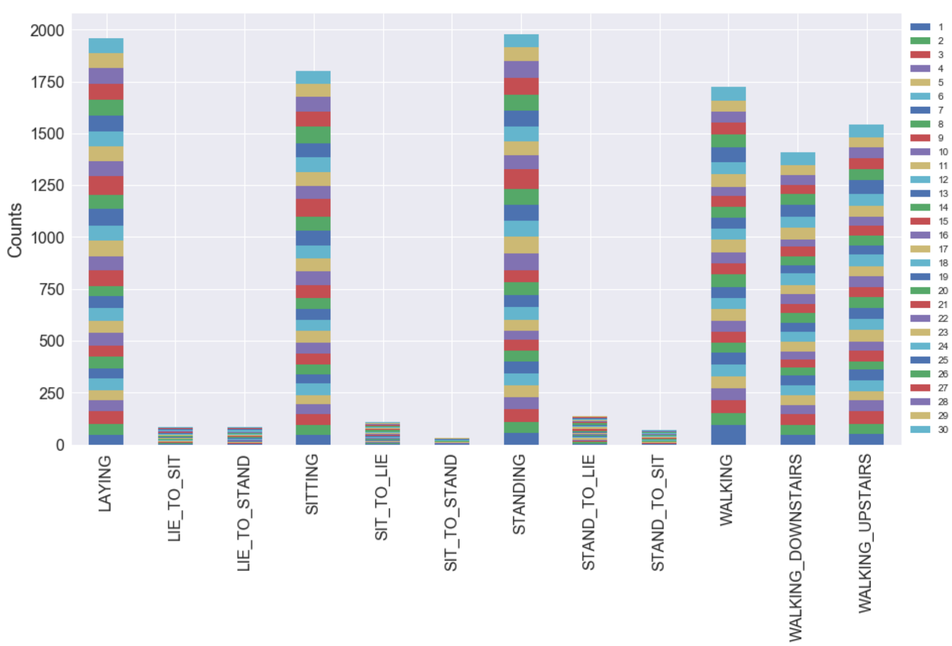
Data wrangling part is done as follows:

- Loading TXT files.
- Renaming column names with feature_names in feature dataset
- Mapping activity numbers to names in target train-test datasets.

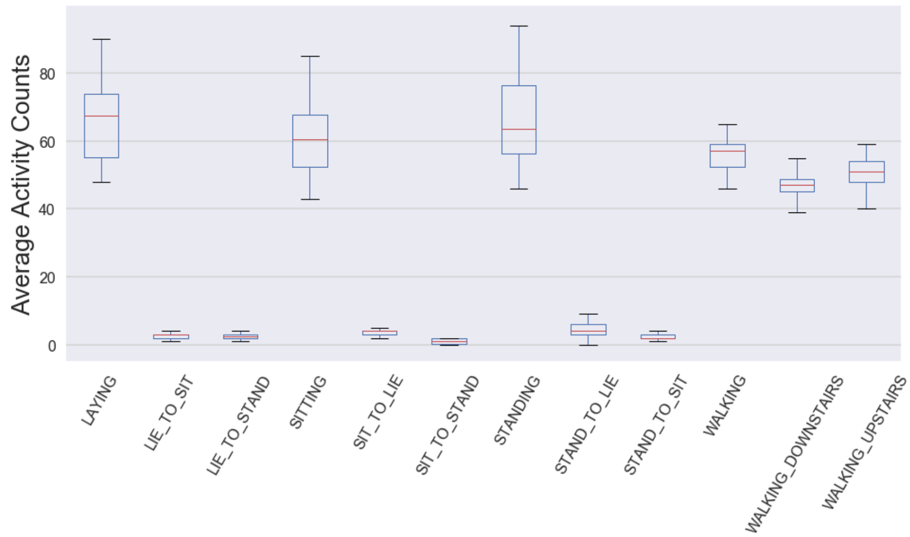
- Concatenating train-test feature dataset to form a dataframe with size 10929x561. It was already split for us to use, but this is not good if we want to do K-fold Cross Validation.
- Concatenating train-test target dataset to form a dataframe with size 10929x1.
- Extracting and comparing activity counts per subject (Volunteer who performs the activities)
- Crosstabbing subject_id and target_activity to see if the data is evenly distributed among 12 activities.

The data has no missing data points, and the only cleaning with the dataset was done to reduce dimensionality.

The original dataset has 6 target activities whereas the updated version has 12. The extra 6 activities are Postural Transitions. The main Human activities were recorded for 5 seconds each where as postural transitions took less time. Hence each subject's recordings should have much less data for transitions. When we crosstab the target_activity with subject_id, we first can look at Activity counts per subject (Figure 4a). We can see that major Human activity counts are much higher than postural transition activities, which are represented with much less data. In Figure 4b, we also see that average activity counts collected from subjects are above 50 while postural transition activities are less than 10. <u>There is an imbalance among targets, which could represent a challenge for correct classification</u>.



(a)



(b)

Figure 4. a) Activity counts per subject; b) The average activity counts of 30 persons. The data imbalance between main and transitional activities is well observed.

PCA for Feature selection:

My main goal in this study is to correctly recognize all main and transitional human activities. The feature size is currently 561 and feature selection methods were used to eliminate unimportant features.

One approach to reduce the feature size is Principal Component Analysis (PCA). PCA is an unsupervised learning method where the input data is transformed to principal components, to a lower dimension. A good rule of thumb with PCA is that we should be able to explain 98% of the variance with the reduced dimensions. If we plot # of components vs. Variance Retained, we see that the Explained Variance increases steadily and saturates (Figure 5). This is called a Scree plot. We calculate that 123 principal components are enough to explain 98% of the variance. This is reduction from 561 to 123, 78% decrease.

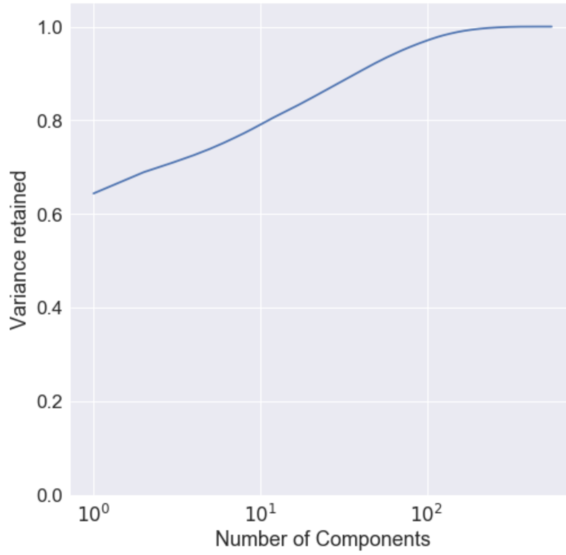


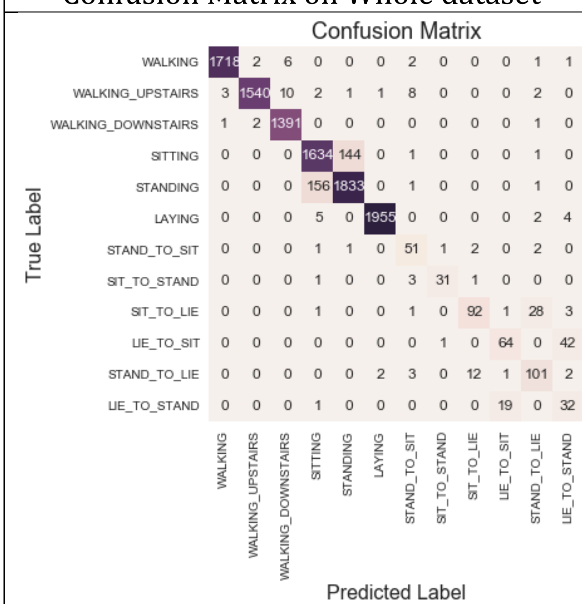
Figure 5. The results of PCA analysis showing how explained variance is changing with the number of components.

The next steps are playing with Model-Based Feature selection and visualization of most important features. Then I will start using classifiers on the reduced dataset.

Classification with KNN:

K-nearest neighbors is a simple algorithm that does classification based on distance function. It does not make any assumptions on the data. The number k defines how many neighbors affect the classification. If k=1, the algorithm turns into nearest neighbor algorithm. KNN is especially useful with large feature set, as in the case of computer vision and face recognition.

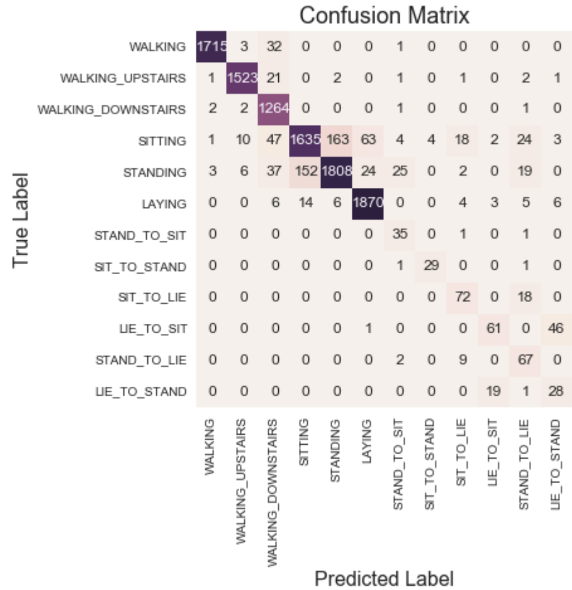
Confusion Matrix on Whole dataset



Classification Report on Whole dataset

	precision	recall	f1-score	support
LAYING	1.00	0.99	1.00	1966
LIE_TO_SIT	0.75	0.60	0.67	107
LIE_TO_STAND	0.38	0.62	0.47	52
SITTING	0.91	0.92	0.91	1780
SIT_TO_LIE	0.86	0.73	0.79	126
SIT_TO_STAND	0.94	0.86	0.90	36
STANDING	0.93	0.92	0.92	1991
STAND_TO_LIE	0.73	0.83	0.78	121
STAND_TO_SIT	0.73	0.88	0.80	58
WALKING	1.00	0.99	1.00	1730
WALKING_DOWNSTAIRS	0.99	1.00	0.99	1395
WALKING_UPSTAIRS	1.00	0.98	0.99	1567
avg / total	0.96	0.96	0.96	10929

Confusion Matrix on Reduced dataset	Classification Report on Reduced dataset



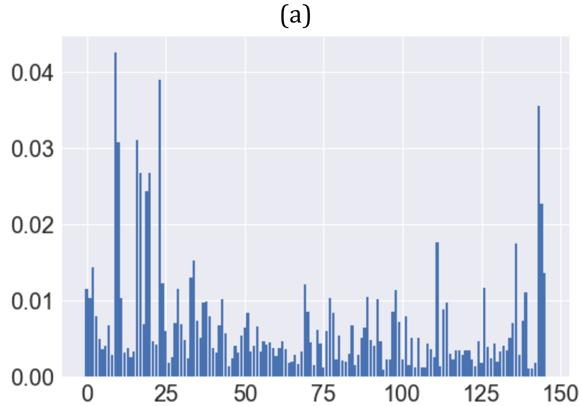
	precision	recall	f1-score	support
LAYING	0.96	0.98	0.97	1914
LIE_TO_SIT	0.72	0.56	0.63	108
LIE_TO_STAND	0.33	0.58	0.42	48
SITTING	0.91	0.83	0.87	1974
SIT_TO_LIE	0.67	0.80	0.73	90
SIT_TO_STAND	0.88	0.94	0.91	31
STANDING	0.91	0.87	0.89	2076
STAND_TO_LIE	0.48	0.86	0.62	78
STAND_TO_SIT	0.50	0.95	0.65	37
WALKING	1.00	0.98	0.99	1751
WALKING_DOWNSTAIRS	0.90	1.00	0.94	1270
WALKING_UPSTAIRS	0.99	0.98	0.98	1552
avg / total	0.93	0.92	0.93	10929

Figure 6. The Confusion Matrix and Classification Report when KNN was trained and tested with whole (561 features) and reduced (123 features) datasets.

Model-based Feature selection:

PCA was able to reduce the dimensions from 561 to 123. However, the features are transformed and only principal components with large variance are kept.
We can also try Model-Based feature selection and have a look at the original features we selected. Tree based classifiers provide the importance of features. Model-based feature selection is able to reduce the feature size to 146 by keeping the threshold three quarters of the mean. The barcode like figure (Figure 7a) shows selected (white) and discarded (black) features.

(a)



(b)

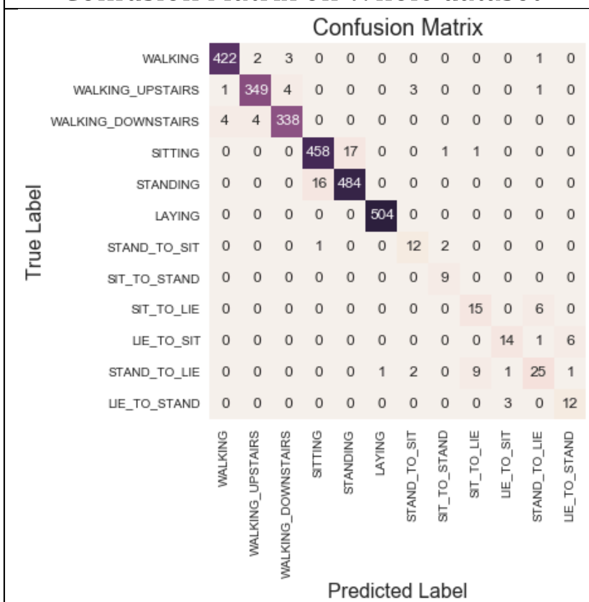
Figure 7. a) The selected and omitted features; b) Feature importance score plotted against Feature number.

We can also look at the importance level of 146 features (Figure 7b). According to the RF model, the strongest feature leading to model evaluation is tBodyAcc-max()-X, Maximum body acceleration in X direction.

Classification with Random Forest (RF):

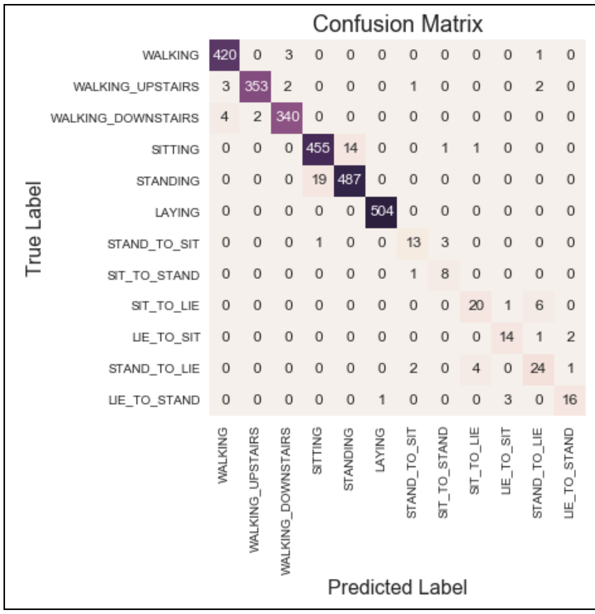
RF is a tree-based algorithm with multiple trees grown. Each tree votes for the class and the forest choose the classification with most votes. There is no pruning in RF, instead tree depth parameter can be adjusted. RF is an algorithm requiring intensive calculations, especially with increasing number of cross-validations. Here the CV was kept as 5.

Confusion Matrix on Whole dataset	Classification Report on Whole dataset



	precision	recall	f1-score	support
LAYING	1.00	1.00	1.00	504
LIE_TO_SIT	0.78	0.67	0.72	21
LIE_TO_STAND	0.63	0.80	0.71	15
SITTING	0.96	0.96	0.96	477
SIT_TO_LIE	0.60	0.71	0.65	21
SIT_TO_STAND	0.75	1.00	0.86	9
STANDING	0.97	0.97	0.97	500
STAND_TO_LIE	0.74	0.64	0.68	39
STAND_TO_SIT	0.71	0.80	0.75	15
WALKING	0.99	0.99	0.99	428
WALKING_DOWNSTAIRS	0.98	0.98	0.98	346
WALKING_UPSTAIRS	0.98	0.97	0.98	358
avg / total	0.97	0.97	0.97	2733

Confusion Matrix on Reduced dataset	Classification Report on Reduced dataset



	precision	recall	f1-score	support
LAYING	1.00	1.00	1.00	504
LIE_TO_SIT	0.78	0.82	0.80	17
LIE_TO_STAND	0.84	0.80	0.82	20
SITTING	0.96	0.97	0.96	471
SIT_TO_LIE	0.80	0.74	0.77	27
SIT_TO_STAND	0.67	0.89	0.76	9
STANDING	0.97	0.96	0.97	506
STAND_TO_LIE	0.71	0.77	0.74	31
STAND_TO_SIT	0.76	0.76	0.76	17
WALKING	0.98	0.99	0.99	424
WALKING_DOWNSTAIRS	0.99	0.98	0.98	346
WALKING_UPSTAIRS	0.99	0.98	0.99	361
avg / total	0.97	0.97	0.97	2733

Figure 8. The Confusion Matrix and Classification Report when RF was trained and tested with whole (561 features) and reduced (146 features) datasets.

Results and Model Evaluation:

f1-scores for both classifiers on both datasets were used to summarize their performance (Figure 9). RF and KNN results are color coded with blue and red, respectively.

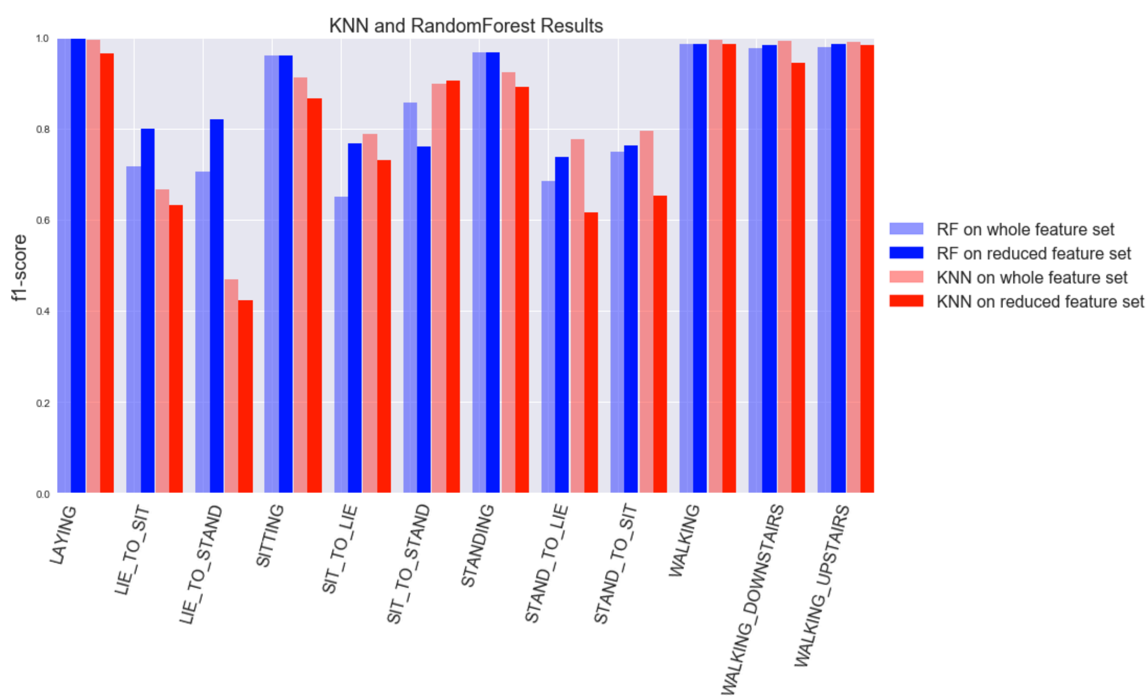


Figure 9. f1-scores summarize the performance of KNN and RF classifiers for all 12 activities.

We can also deduct the average f1-scores for Major and Transitional activities (Table 1 and 2).

Table 1. f1-score for Postural Transitional Activity Classes

	KNN	RF
Original Dataset	0.74	0.71
Reduced Dataset	0.65	0.77

Table 2. f1-score of Complete Classes

	KNN	RF
Original Dataset	0.96	0.97
Reduced Dataset	0.93	0.97

- There is no perfect recall and precision. There are confusions about all of the classes in both classifiers. The perfect 1.00 score is due to rounding.
- The average precision and recall for Major Activities (Laying, Sitting, Standing, Walking, Walking_Downstairs, and Walking_Upstairs) are over 0.95 for both classifiers. There are not many false positives for these classes except for Standing and for Sitting classes, which are confused with each other.
- The Transitional Activities show worse f1-scores than the Major Activities (Tables 1,2).
- The Transitional Activities for KNN classifier show worse average f1-score for Reduced Dataset compared to original dataset.
- As for RF classifier, the classification scores of Major and Transitional Activities marginally increased compared to KNN and the confusion rate between Standing and for Sitting classes decreased.
- RF Classifier is able to predict better in a data imbalance situation, i.e. between Major and Transitional Activities. A tuned RF performs better than KNN for HAPT classification problem.
- KNN performed worse with Reduced dataset when the Whitening option of PCA was used.

Client Recommendations:

- Classification of Main Human Activities can be done with almost perfect accuracy. This results in correctly tracking major activities using a smartphone. For healthcare management, predicting major activities could be enough for following patient's mobility and calorie tracking. Along with other information such as age, height and weight, it would be easy to track an individual's life style.
- As for smart environments and public surveillance, people tend to switch between activities or not repetitively perform major activities. Since the HAPT dataset was obtained in ideal conditions where volunteers were asked to follow a specific pattern of movements, the data for target classes were very well documented and close to perfection. Although the postural transition data was much smaller compared to major activities, the classifiers were able to detect them with >70% accuracy. Companies in smart

environments and public surveillance should focus on collecting more data on postural transitions to increase the detection ability.

- Companies in public surveillance should invest in tracking an individual's trajectory using Human Activity Dataset. It is possible to build a trajectory for an individual if more classes are added as targets. For instance, the data to be collected when turning right and left, and changing lanes could help with drawing a trajectory.
- The company should look at the most important features that help classify an activity. Using less number of features will help energy effective real-time tracking due to less intensive CPU use.
-

References:

[1] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine," in *International workshop on ambient assisted living*, 2012, pp. 216-223.

[2] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A Public Domain Dataset for Human Activity Recognition using Smartphones," in *ESANN*, 2013.

[3] A. Bayat, M. Pomplun, and D. A. Tran, "A study on human activity recognition using accelerometer data from smartphones," *Procedia Computer Science,* vol. 34, pp. 450-457, 2014.

[4] A. M. Khan, Y.-K. Lee, S. Y. Lee, and T.-S. Kim, "A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer," *IEEE transactions on information technology in biomedicine,* vol. 14, pp. 1166-1172, 2010.