Practical Machine Learning HAR Clustering Model/Analysis using Random Forest
==========

To start with. the training data was partititioned into two - training (70%) and cross validation (30%). The partitioned training set was used to create the model and was validated using the cross validation set. Once the model parameters were satisfactory, the model was derived based on the full training set and this derived model was used to predit the test set.

Load the traning/test data

```
setwd("~/My Training/PML")

currentTrainPath = 'train.RData'
currentTestPath = 'test.RData'

training = T

if (training)
{
  trn = read.csv('pml_trn.csv',header=T)
} else {
  modFit<-readRDS(currentTrainPath)
  tst = read.csv('pml_tst.csv',header=T)
}
```

Data Cleanup and Feature Addition

```
if (training) {
  data<-trn
} else {
  data<-tst
}

data<-data[which(data$new_window=="no"),]
dim(data)

## [1] 13453   160
```

Cleaning and shaping up the data for initial analysis. Delete all columns containing NA

```
rmcol<-array()
j<-1
for (i in 1:ncol(data)) {
  if ((sum(is.na(data[,i])) > 0) | (sum(data[,i] == "") > 0)) {
```

```
    rmcol[j]<-i
    j<-j+1
  }
}

data<-data[,-rmcol]
rm(rmcol)

tm<-unlist(lapply(strsplit(as.character(data$cvtd_timestamp)," "),
function(x) x[2]))
time<-(lapply(strsplit(tm,":"),function(x) {return
(round((as.numeric(x[1])) + round(as.numeric(x[2])/60)))}))

data$cvtd_timestamp =
factor(NA,levels=c('midnight','morning','noon','afternoon','evening'))

data$cvtd_timestamp[time>22 | time<=6] <- 'midnight'
data$cvtd_timestamp[time>6 & time<=10] <- 'morning'
data$cvtd_timestamp[time>10 & time <= 13] <- 'noon'
data$cvtd_timestamp[time>13 & time <= 18] <- 'afternoon'
data$cvtd_timestamp[time>18 & time <= 22] <- 'evening'

rm(tm,time)
```

Train the Model. This model has been traing using Random Forest and it's defaults. After the initial analysis, it appeared that only 17 features are critical in determining the classifications. Hence, only 17 features were used to train the model.

```
library(caret)

## Loading required package: lattice
## Loading required package: ggplot2

nam<-names(data)
valCols<-
c(which(nam=="accel_dumbbell_x"),which(nam=="magnet_dumbbell_x"),which
(nam=="magnet_dumbbell_y"),which(nam=="magnet_dumbbell_z"),which(nam==
"roll_forearm"),which(nam=="pitch_forearm"),which(nam=="yaw_forearm"),
which(nam=="total_accel_forearm"),which(nam=="gyros_forearm_x"),which(
nam=="gyros_forearm_y"),which(nam=="gyros_forearm_z"),which(nam=="acce
l_forearm_x"),which(nam=="accel_forearm_y"),which(nam=="accel_forearm_
z"),which(nam=="magnet_forearm_x"),which(nam=="magnet_forearm_y"),whic
h(nam=="magnet_forearm_z"),which(nam=="classe"))
```

```
if (training) {
   data_b<-data
   values<-data_b$classe
   modFit<-train(classe~.,data<-data_b[,valCols],method="rf")
   prediction<-data$classe
   saveRDS(modFit,currentTrainPath)
   error<-sum(values!=prediction)/length(values)
   paste("Training Set Error:",(error)," ")

} else {
  values<-data$classe
  prediction<-predict(modFit,data[,valCols])
  error<-sum(values!=prediction)/length(values)
  paste("Test Set Error:",(error)," ")
}
```

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

```
## [1] "Training Set Error: 0  "
```