Machine Learning

Pradeep Kumar

30 August 2018

Practical Machine Learning Project

Introduction

Using devices such as Jawbone Up, Nike FuelBand, and Fitbit it is now possible to collect a large amount of data about personal activity relatively inexpensively. These type of devices are part of the quantified self movement - a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks. One thing that people regularly do is quantify how much of a particular activity they do, but they rarely quantify how well they do it.

In this project, we will use data from accelerometers on the belt, forearm, arm, and dumbell of 6 participants to predict the manner in which they did the exercise.

Getting and Cleaning the Data

Loading the libraries

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.3.3
```

```
## Loading required package: lattice
```

```
## Warning: package 'lattice' was built under R version 3.3.3
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.3.3
```

```
library(rattle)
```

```
## Rattle: A free graphical interface for data science with R.
## Version 5.2.0 Copyright (c) 2006-2018 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.
```

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 3.3.3
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:rattle':
##
##     importance
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

Reading Data

Loading the data using URL's given.

```
trainingdata <- read.csv(url("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv"), header
= T, na.strings = c("NA", ""))
testingdata <- read.csv(url("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv"), header =
T, na.strings = c("NA", ""))
```

Processing Data

```
str(trainingdata)
```

```
## 'data.frame':    19622 obs. of  160 variables:
##  $ X                       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ user_name               : Factor w/ 6 levels "adelmo","carlitos",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ raw_timestamp_part_1    : int  1323084231 1323084231 1323084231 1323084232 1323084232 1323084232 13230
84232 1323084232 1323084232 1323084232 ...
##  $ raw_timestamp_part_2    : int  788290 808298 820366 120339 196328 304277 368296 440390 484323 484434
...
##  $ cvtd_timestamp          : Factor w/ 20 levels "02/12/2011 13:32",..: 9 9 9 9 9 9 9 9 9 9 ...
##  $ new_window              : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ num_window              : int  11 11 11 12 12 12 12 12 12 12 ...
##  $ roll_belt               : num  1.41 1.41 1.42 1.48 1.48 1.45 1.42 1.42 1.43 1.45 ...
##  $ pitch_belt              : num  8.07 8.07 8.07 8.05 8.07 8.06 8.09 8.13 8.16 8.17 ...
##  $ yaw_belt                : num  -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 ...
##  $ total_accel_belt        : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ kurtosis_roll_belt      : Factor w/ 396 levels "-0.016850","-0.021024",..: NA NA NA NA NA NA NA NA NA
NA ...
##  $ kurtosis_picth_belt     : Factor w/ 316 levels "-0.021887","-0.060755",..: NA NA NA NA NA NA NA NA NA
NA ...
##  $ kurtosis_yaw_belt       : Factor w/ 1 level "#DIV/0!": NA NA NA NA NA NA NA NA NA NA ...
##  $ skewness_roll_belt      : Factor w/ 394 levels "-0.003095","-0.010002",..: NA NA NA NA NA NA NA NA NA
NA ...
##  $ skewness_roll_belt.1    : Factor w/ 337 levels "-0.005928","-0.005960",..: NA NA NA NA NA NA NA NA NA
NA ...
##  $ skewness_yaw_belt       : Factor w/ 1 level "#DIV/0!": NA NA NA NA NA NA NA NA NA NA ...
##  $ max_roll_belt           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ max_picth_belt          : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ max_yaw_belt            : Factor w/ 67 levels "-0.1","-0.2",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ min_roll_belt           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ min_pitch_belt          : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ min_yaw_belt            : Factor w/ 67 levels "-0.1","-0.2",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ amplitude_roll_belt     : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ amplitude_pitch_belt    : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ amplitude_yaw_belt      : Factor w/ 3 levels "#DIV/0!","0.00",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ var_total_accel_belt    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ avg_roll_belt           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ stddev_roll_belt        : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ var_roll_belt           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ avg_pitch_belt          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ stddev_pitch_belt       : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ var_pitch_belt          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ avg_yaw_belt            : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ stddev_yaw_belt         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ var_yaw_belt            : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ gyros_belt_x            : num  0 0.02 0 0.02 0.02 0.02 0.02 0.02 0.02 0.03 ...
##  $ gyros_belt_y            : num  0 0 0 0 0.02 0 0 0 0 0 ...
##  $ gyros_belt_z            : num  -0.02 -0.02 -0.02 -0.03 -0.02 -0.02 -0.02 -0.02 -0.02 0 ...
##  $ accel_belt_x            : int  -21 -22 -20 -22 -21 -21 -22 -22 -20 -21 ...
##  $ accel_belt_y            : int  4 4 5 3 2 4 3 4 2 4 ...
##  $ accel_belt_z            : int  22 22 23 21 24 21 21 21 24 22 ...
##  $ magnet_belt_x           : int  -3 -7 -2 -6 -6 0 -4 -2 1 -3 ...
##  $ magnet_belt_y           : int  599 608 600 604 600 603 599 603 602 609 ...
##  $ magnet_belt_z           : int  -313 -311 -305 -310 -302 -312 -311 -313 -312 -308 ...
##  $ roll_arm                : num  -128 -128 -128 -128 -128 -128 -128 -128 -128 -128 ...
##  $ pitch_arm               : num  22.5 22.5 22.5 22.1 22.1 22 21.9 21.8 21.7 21.6 ...
##  $ yaw_arm                 : num  -161 -161 -161 -161 -161 -161 -161 -161 -161 -161 ...
##  $ total_accel_arm         : int  34 34 34 34 34 34 34 34 34 34 ...
##  $ var_accel_arm           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ avg_roll_arm            : num  NA NA NA NA NA NA NA NA NA NA
```

```
##  $ avg_roll_arm            : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ stddev_roll_arm         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ var_roll_arm            : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ avg_pitch_arm           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ stddev_pitch_arm        : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ var_pitch_arm           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ avg_yaw_arm             : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ stddev_yaw_arm          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ var_yaw_arm             : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ gyros_arm_x             : num  0 0.02 0.02 0.02 0 0.02 0 0.02 0.02 0.02 ...
##  $ gyros_arm_y             : num  0 -0.02 -0.02 -0.03 -0.03 -0.03 -0.03 -0.02 -0.03 -0.03 ...
##  $ gyros_arm_z             : num  -0.02 -0.02 -0.02 0.02 0 0 0 0 -0.02 -0.02 ...
##  $ accel_arm_x             : int  -288 -290 -289 -289 -289 -289 -289 -289 -288 -288 ...
##  $ accel_arm_y             : int  109 110 110 111 111 111 111 111 109 110 ...
##  $ accel_arm_z             : int  -123 -125 -126 -123 -123 -122 -125 -124 -122 -124 ...
##  $ magnet_arm_x            : int  -368 -369 -368 -372 -374 -369 -373 -372 -369 -376 ...
##  $ magnet_arm_y            : int  337 337 344 344 337 342 336 338 341 334 ...
##  $ magnet_arm_z            : int  516 513 513 512 506 513 509 510 518 516 ...
##  $ kurtosis_roll_arm       : Factor w/ 329 levels "-0.02438","-0.04190",..: NA NA NA NA NA NA NA NA NA NA
...
##  $ kurtosis_picth_arm      : Factor w/ 327 levels "-0.00484","-0.01311",..: NA NA NA NA NA NA NA NA NA NA
...
##  $ kurtosis_yaw_arm        : Factor w/ 394 levels "-0.01548","-0.01749",..: NA NA NA NA NA NA NA NA NA NA
...
##  $ skewness_roll_arm       : Factor w/ 330 levels "-0.00051","-0.00696",..: NA NA NA NA NA NA NA NA NA NA
...
##  $ skewness_pitch_arm      : Factor w/ 327 levels "-0.00184","-0.01185",..: NA NA NA NA NA NA NA NA NA NA
...
##  $ skewness_yaw_arm        : Factor w/ 394 levels "-0.00311","-0.00562",..: NA NA NA NA NA NA NA NA NA NA
...
##  $ max_roll_arm            : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ max_picth_arm           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ max_yaw_arm             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ min_roll_arm            : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ min_pitch_arm           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ min_yaw_arm             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ amplitude_roll_arm      : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ amplitude_pitch_arm     : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ amplitude_yaw_arm       : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ roll_dumbbell           : num  13.1 13.1 12.9 13.4 13.4 ...
##  $ pitch_dumbbell          : num  -70.5 -70.6 -70.3 -70.4 -70.4 ...
##  $ yaw_dumbbell            : num  -84.9 -84.7 -85.1 -84.9 -84.9 ...
##  $ kurtosis_roll_dumbbell  : Factor w/ 397 levels "-0.0035","-0.0073",..: NA NA NA NA NA NA NA NA NA NA .
..
##  $ kurtosis_picth_dumbbell : Factor w/ 400 levels "-0.0163","-0.0233",..: NA NA NA NA NA NA NA NA NA NA .
..
##  $ kurtosis_yaw_dumbbell   : Factor w/ 1 level "#DIV/0!": NA NA NA NA NA NA NA NA NA NA ...
##  $ skewness_roll_dumbbell  : Factor w/ 400 levels "-0.0082","-0.0096",..: NA NA NA NA NA NA NA NA NA NA .
..
##  $ skewness_pitch_dumbbell : Factor w/ 401 levels "-0.0053","-0.0084",..: NA NA NA NA NA NA NA NA NA NA .
..
##  $ skewness_yaw_dumbbell   : Factor w/ 1 level "#DIV/0!": NA NA NA NA NA NA NA NA NA NA ...
##  $ max_roll_dumbbell       : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ max_picth_dumbbell      : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ max_yaw_dumbbell        : Factor w/ 72 levels "-0.1","-0.2",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ min_roll_dumbbell       : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ min_pitch_dumbbell      : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ min_yaw_dumbbell        : Factor w/ 72 levels "-0.1","-0.2",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ amplitude_roll_dumbbell : num  NA NA NA NA NA NA NA NA NA NA ...
##   [list output truncated]
```

```
str(testingdata)
```

```
## 'data.frame':    20 obs. of  160 variables:
##  $ X                       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ user_name               : Factor w/ 6 levels "adelmo","carlitos",..: 6 5 5 1 4 5 5 5 2 3 ...
##  $ raw_timestamp_part_1    : int  1323095002 1322673067 1322673075 1322832789 1322489635 1322673149 13226
73128 1322673076 1323084240 1322837822 ...
##  $ raw_timestamp_part_2    : int  868349 778725 342967 560311 814776 510661 766645 54671 916313 384285 .
..
##  $ cvtd_timestamp          : Factor w/ 11 levels "02/12/2011 13:33",..: 5 10 10 1 6 11 11 10 3 2 ...
##  $ new_window              : Factor w/ 1 level "no": 1 1 1 1 1 1 1 1 1 1 ...
```

```
##  $ num_window              : int  74 431 439 194 235 504 485 440 323 664 ...
##  $ roll_belt               : num  123 1.02 0.87 125 1.35 -5.92 1.2 0.43 0.93 114 ...
##  $ pitch_belt              : num  27 4.87 1.82 -41.6 3.33 1.59 4.44 4.15 6.72 22.4 ...
##  $ yaw_belt                : num  -4.75 -88.9 -88.5 162 -88.6 -87.7 -87.3 -88.5 -93.7 -13.1 ...
##  $ total_accel_belt        : int  20 4 5 17 3 4 4 4 4 18 ...
##  $ kurtosis_roll_belt      : logi  NA NA NA NA NA NA ...
##  $ kurtosis_picth_belt     : logi  NA NA NA NA NA NA ...
##  $ kurtosis_yaw_belt       : logi  NA NA NA NA NA NA ...
##  $ skewness_roll_belt      : logi  NA NA NA NA NA NA ...
##  $ skewness_roll_belt.1    : logi  NA NA NA NA NA NA ...
##  $ skewness_yaw_belt       : logi  NA NA NA NA NA NA ...
##  $ max_roll_belt           : logi  NA NA NA NA NA NA ...
##  $ max_picth_belt          : logi  NA NA NA NA NA NA ...
##  $ max_yaw_belt            : logi  NA NA NA NA NA NA ...
##  $ min_roll_belt           : logi  NA NA NA NA NA NA ...
##  $ min_pitch_belt          : logi  NA NA NA NA NA NA ...
##  $ min_yaw_belt            : logi  NA NA NA NA NA NA ...
##  $ amplitude_roll_belt     : logi  NA NA NA NA NA NA ...
##  $ amplitude_pitch_belt    : logi  NA NA NA NA NA NA ...
##  $ amplitude_yaw_belt      : logi  NA NA NA NA NA NA ...
##  $ var_total_accel_belt    : logi  NA NA NA NA NA NA ...
##  $ avg_roll_belt           : logi  NA NA NA NA NA NA ...
##  $ stddev_roll_belt        : logi  NA NA NA NA NA NA ...
##  $ var_roll_belt           : logi  NA NA NA NA NA NA ...
##  $ avg_pitch_belt          : logi  NA NA NA NA NA NA ...
##  $ stddev_pitch_belt       : logi  NA NA NA NA NA NA ...
##  $ var_pitch_belt          : logi  NA NA NA NA NA NA ...
##  $ avg_yaw_belt            : logi  NA NA NA NA NA NA ...
##  $ stddev_yaw_belt         : logi  NA NA NA NA NA NA ...
##  $ var_yaw_belt            : logi  NA NA NA NA NA NA ...
##  $ gyros_belt_x            : num  -0.5 -0.06 0.05 0.11 0.03 0.1 -0.06 -0.18 0.1 0.14 ...
##  $ gyros_belt_y            : num  -0.02 -0.02 0.02 0.11 0.02 0.05 0 -0.02 0 0.11 ...
##  $ gyros_belt_z            : num  -0.46 -0.07 0.03 -0.16 0 -0.13 0 -0.03 -0.02 -0.16 ...
##  $ accel_belt_x            : int  -38 -13 1 46 -8 -11 -14 -10 -15 -25 ...
##  $ accel_belt_y            : int  69 11 -1 45 4 -16 2 -2 1 63 ...
##  $ accel_belt_z            : int  -179 39 49 -156 27 38 35 42 32 -158 ...
##  $ magnet_belt_x           : int  -13 43 29 169 33 31 50 39 -6 10 ...
##  $ magnet_belt_y           : int  581 636 631 608 566 638 622 635 600 601 ...
##  $ magnet_belt_z           : int  -382 -309 -312 -304 -418 -291 -315 -305 -302 -330 ...
##  $ roll_arm                : num  40.7 0 0 -109 76.1 0 0 0 -137 -82.4 ...
##  $ pitch_arm               : num  -27.8 0 0 55 2.76 0 0 0 11.2 -63.8 ...
##  $ yaw_arm                 : num  178 0 0 -142 102 0 0 0 -167 -75.3 ...
##  $ total_accel_arm         : int  10 38 44 25 29 14 15 22 34 32 ...
##  $ var_accel_arm           : logi  NA NA NA NA NA NA ...
##  $ avg_roll_arm            : logi  NA NA NA NA NA NA ...
##  $ stddev_roll_arm         : logi  NA NA NA NA NA NA ...
##  $ var_roll_arm            : logi  NA NA NA NA NA NA ...
##  $ avg_pitch_arm           : logi  NA NA NA NA NA NA ...
##  $ stddev_pitch_arm        : logi  NA NA NA NA NA NA ...
##  $ var_pitch_arm           : logi  NA NA NA NA NA NA ...
##  $ avg_yaw_arm             : logi  NA NA NA NA NA NA ...
##  $ stddev_yaw_arm          : logi  NA NA NA NA NA NA ...
##  $ var_yaw_arm             : logi  NA NA NA NA NA NA ...
##  $ gyros_arm_x             : num  -1.65 -1.17 2.1 0.22 -1.96 0.02 2.36 -3.71 0.03 0.26 ...
##  $ gyros_arm_y             : num  0.48 0.85 -1.36 -0.51 0.79 0.05 -1.01 1.85 -0.02 -0.5 ...
##  $ gyros_arm_z             : num  -0.18 -0.43 1.13 0.92 -0.54 -0.07 0.89 -0.69 -0.02 0.79 ...
##  $ accel_arm_x             : int  16 -290 -341 -238 -197 -26 99 -98 -287 -301 ...
##  $ accel_arm_y             : int  38 215 245 -57 200 130 79 175 111 -42 ...
##  $ accel_arm_z             : int  93 -90 -87 6 -30 -19 -67 -78 -122 -80 ...
##  $ magnet_arm_x            : int  -326 -325 -264 -173 -170 396 702 535 -367 -420 ...
##  $ magnet_arm_y            : int  385 447 474 257 275 176 15 215 335 294 ...
##  $ magnet_arm_z            : int  481 434 413 633 617 516 217 385 520 493 ...
##  $ kurtosis_roll_arm       : logi  NA NA NA NA NA NA ...
##  $ kurtosis_picth_arm      : logi  NA NA NA NA NA NA ...
##  $ kurtosis_yaw_arm        : logi  NA NA NA NA NA NA ...
##  $ skewness_roll_arm       : logi  NA NA NA NA NA NA ...
##  $ skewness_pitch_arm      : logi  NA NA NA NA NA NA ...
##  $ skewness_yaw_arm        : logi  NA NA NA NA NA NA ...
##  $ max_roll_arm            : logi  NA NA NA NA NA NA ...
##  $ max_picth_arm           : logi  NA NA NA NA NA NA ...
##  $ max_yaw_arm             : logi  NA NA NA NA NA NA ...
##  $ min_roll_arm            : logi  NA NA NA NA NA NA ...
##  $ min_pitch_arm           : logi  NA NA NA NA NA NA
```

```
##  $ min_pitch_arm            : logi  NA NA NA NA NA NA ...
##  $ min_yaw_arm              : logi  NA NA NA NA NA NA ...
##  $ amplitude_roll_arm       : logi  NA NA NA NA NA NA ...
##  $ amplitude_pitch_arm      : logi  NA NA NA NA NA NA ...
##  $ amplitude_yaw_arm        : logi  NA NA NA NA NA NA ...
##  $ roll_dumbbell            : num  -17.7 54.5 57.1 43.1 -101.4 ...
##  $ pitch_dumbbell           : num  25 -53.7 -51.4 -30 -53.4 ...
##  $ yaw_dumbbell             : num  126.2 -75.5 -75.2 -103.3 -14.2 ...
##  $ kurtosis_roll_dumbbell   : logi  NA NA NA NA NA NA ...
##  $ kurtosis_picth_dumbbell : logi  NA NA NA NA NA NA ...
##  $ kurtosis_yaw_dumbbell    : logi  NA NA NA NA NA NA ...
##  $ skewness_roll_dumbbell   : logi  NA NA NA NA NA NA ...
##  $ skewness_pitch_dumbbell : logi  NA NA NA NA NA NA ...
##  $ skewness_yaw_dumbbell    : logi  NA NA NA NA NA NA ...
##  $ max_roll_dumbbell        : logi  NA NA NA NA NA NA ...
##  $ max_picth_dumbbell       : logi  NA NA NA NA NA NA ...
##  $ max_yaw_dumbbell         : logi  NA NA NA NA NA NA ...
##  $ min_roll_dumbbell        : logi  NA NA NA NA NA NA ...
##  $ min_pitch_dumbbell       : logi  NA NA NA NA NA NA ...
##  $ min_yaw_dumbbell         : logi  NA NA NA NA NA NA ...
##  $ amplitude_roll_dumbbell : logi  NA NA NA NA NA NA ...
##    [list output truncated]
```

Removing all columns having missing values

```
trainingdata <- trainingdata[,colSums(is.na(trainingdata))==0]
testingdata <- testingdata[,colSums(is.na(testingdata))==0]
```

Removing the data that does not have much influence

```
nzv1 <- nearZeroVar(trainingdata, saveMetrics = TRUE)
nzv1
```

```
##                       freqRatio percentUnique zeroVar   nzv
## X                      1.000000  100.00000000   FALSE FALSE
## user_name              1.100679    0.03057792   FALSE FALSE
## raw_timestamp_part_1   1.000000    4.26562022   FALSE FALSE
## raw_timestamp_part_2   1.000000   85.53154622   FALSE FALSE
## cvtd_timestamp         1.000668    0.10192641   FALSE FALSE
## new_window            47.330049    0.01019264   FALSE  TRUE
## num_window             1.000000    4.37264295   FALSE FALSE
## roll_belt              1.101904    6.77810621   FALSE FALSE
## pitch_belt             1.036082    9.37722964   FALSE FALSE
## yaw_belt               1.058480    9.97349913   FALSE FALSE
## total_accel_belt       1.063160    0.14779329   FALSE FALSE
## gyros_belt_x           1.058651    0.71348486   FALSE FALSE
## gyros_belt_y           1.144000    0.35164611   FALSE FALSE
## gyros_belt_z           1.066214    0.86127816   FALSE FALSE
## accel_belt_x           1.055412    0.83579655   FALSE FALSE
## accel_belt_y           1.113725    0.72877383   FALSE FALSE
## accel_belt_z           1.078767    1.52379982   FALSE FALSE
## magnet_belt_x          1.090141    1.66649679   FALSE FALSE
## magnet_belt_y          1.099688    1.51870350   FALSE FALSE
## magnet_belt_z          1.006369    2.32901845   FALSE FALSE
## roll_arm              52.338462   13.52563449   FALSE FALSE
## pitch_arm             87.256410   15.73234125   FALSE FALSE
## yaw_arm               33.029126   14.65701763   FALSE FALSE
## total_accel_arm        1.024526    0.33635715   FALSE FALSE
## gyros_arm_x            1.015504    3.27693405   FALSE FALSE
## gyros_arm_y            1.454369    1.91621649   FALSE FALSE
## gyros_arm_z            1.110687    1.26388747   FALSE FALSE
## accel_arm_x            1.017341    3.95984099   FALSE FALSE
## accel_arm_y            1.140187    2.73672409   FALSE FALSE
## accel_arm_z            1.128000    4.03628580   FALSE FALSE
## magnet_arm_x           1.000000    6.82397309   FALSE FALSE
## magnet_arm_y           1.056818    4.44399144   FALSE FALSE
## magnet_arm_z           1.036364    6.44684538   FALSE FALSE
## roll_dumbbell          1.022388   84.20650290   FALSE FALSE
## pitch_dumbbell         2.277372   81.74498012   FALSE FALSE
## yaw_dumbbell           1.132231   83.48282540   FALSE FALSE
## total_accel_dumbbell   1.072634    0.21914178   FALSE FALSE
## gyros_dumbbell_x       1.003268    1.22821323   FALSE FALSE
## gyros_dumbbell_y       1.264957    1.41677709   FALSE FALSE
## gyros_dumbbell_z       1.060100    1.04984201   FALSE FALSE
## accel_dumbbell_x       1.018018    2.16593619   FALSE FALSE
## accel_dumbbell_y       1.053061    2.37488533   FALSE FALSE
## accel_dumbbell_z       1.133333    2.08949139   FALSE FALSE
## magnet_dumbbell_x      1.098266    5.74864948   FALSE FALSE
## magnet_dumbbell_y      1.197740    4.30129447   FALSE FALSE
## magnet_dumbbell_z      1.020833    3.44511263   FALSE FALSE
## roll_forearm          11.589286   11.08959331   FALSE FALSE
## pitch_forearm         65.983051   14.85577413   FALSE FALSE
## yaw_forearm           15.322835   10.14677403   FALSE FALSE
## total_accel_forearm    1.128928    0.35674243   FALSE FALSE
## gyros_forearm_x        1.059273    1.51870350   FALSE FALSE
## gyros_forearm_y        1.036554    3.77637346   FALSE FALSE
## gyros_forearm_z        1.122917    1.56457038   FALSE FALSE
## accel_forearm_x        1.126437    4.04647844   FALSE FALSE
## accel_forearm_y        1.059406    5.11160942   FALSE FALSE
## accel_forearm_z        1.006250    2.95586586   FALSE FALSE
## magnet_forearm_x       1.012346    7.76679238   FALSE FALSE
## magnet_forearm_y       1.246914    9.54031189   FALSE FALSE
## magnet_forearm_z       1.000000    8.57710733   FALSE FALSE
## classe                 1.469581    0.02548160   FALSE FALSE
```

```
nzv2 <- nearZeroVar(testingdata, saveMetrics = TRUE)
nzv2
```

```
##                      freqRatio percentUnique zeroVar   nzv
## X                     1.000000          100   FALSE FALSE
## user_name             2.000000           30   FALSE FALSE
## raw_timestamp_part_1  1.000000          100   FALSE FALSE
## raw_timestamp_part_2  1.000000          100   FALSE FALSE
## cvtd_timestamp        1.333333           55   FALSE FALSE
## new_window            0.000000            5    TRUE  TRUE
## num_window            1.000000          100   FALSE FALSE
## roll_belt             1.000000           90   FALSE FALSE
## pitch_belt            1.000000          100   FALSE FALSE
## yaw_belt              1.000000           90   FALSE FALSE
## total_accel_belt      1.200000           45   FALSE FALSE
## gyros_belt_x          1.500000           70   FALSE FALSE
## gyros_belt_y          2.250000           30   FALSE FALSE
## gyros_belt_z          2.500000           65   FALSE FALSE
## accel_belt_x          1.000000           85   FALSE FALSE
## accel_belt_y          1.500000           75   FALSE FALSE
## accel_belt_z          1.000000           85   FALSE FALSE
## magnet_belt_x         1.000000          100   FALSE FALSE
## magnet_belt_y         1.500000           80   FALSE FALSE
## magnet_belt_z         1.500000           80   FALSE FALSE
## roll_arm              8.000000           65   FALSE FALSE
## pitch_arm             8.000000           65   FALSE FALSE
## yaw_arm               8.000000           65   FALSE FALSE
## total_accel_arm       1.000000           85   FALSE FALSE
## gyros_arm_x           2.000000           95   FALSE FALSE
## gyros_arm_y           2.000000           95   FALSE FALSE
## gyros_arm_z           1.000000           90   FALSE FALSE
## accel_arm_x           2.000000           95   FALSE FALSE
## accel_arm_y           1.000000          100   FALSE FALSE
## accel_arm_z           1.000000          100   FALSE FALSE
## magnet_arm_x          1.000000          100   FALSE FALSE
## magnet_arm_y          2.000000           95   FALSE FALSE
## magnet_arm_z          1.000000          100   FALSE FALSE
## roll_dumbbell         1.000000          100   FALSE FALSE
## pitch_dumbbell        1.000000          100   FALSE FALSE
## yaw_dumbbell          1.000000          100   FALSE FALSE
## total_accel_dumbbell  2.000000           70   FALSE FALSE
## gyros_dumbbell_x      1.000000           90   FALSE FALSE
## gyros_dumbbell_y      1.000000           80   FALSE FALSE
## gyros_dumbbell_z      4.000000           85   FALSE FALSE
## accel_dumbbell_x      1.000000          100   FALSE FALSE
## accel_dumbbell_y      3.000000           90   FALSE FALSE
## accel_dumbbell_z      2.000000           95   FALSE FALSE
## magnet_dumbbell_x     1.000000          100   FALSE FALSE
## magnet_dumbbell_y     1.000000          100   FALSE FALSE
## magnet_dumbbell_z     2.000000           95   FALSE FALSE
## roll_forearm          2.000000           95   FALSE FALSE
## pitch_forearm         1.000000          100   FALSE FALSE
## yaw_forearm           1.000000          100   FALSE FALSE
## total_accel_forearm   1.500000           65   FALSE FALSE
## gyros_forearm_x       1.000000           90   FALSE FALSE
## gyros_forearm_y       1.000000          100   FALSE FALSE
## gyros_forearm_z       1.000000          100   FALSE FALSE
## accel_forearm_x       1.000000          100   FALSE FALSE
## accel_forearm_y       1.000000          100   FALSE FALSE
## accel_forearm_z       1.000000          100   FALSE FALSE
## magnet_forearm_x      1.000000          100   FALSE FALSE
## magnet_forearm_y      1.000000          100   FALSE FALSE
## magnet_forearm_z      1.000000          100   FALSE FALSE
## problem_id            1.000000          100   FALSE FALSE
```

First 5 columns are just for information and does not have much influene and by the above model removing 6th and 7th column New data

```
trainingdata <- trainingdata[,-c(1:7)]
testingdata <- testingdata[,-c(1:7)]
```

```
dim(trainingdata)
```

```
## [1] 19622    53
```

```
dim(testingdata)
```

```
## [1] 20 53
```

Fitting the training and test data

Splitting the training data into new training and testing data of 70%-30% ratio.

```
inTrain <- createDataPartition(y = trainingdata$classe, p = 0.7, list = F)
training <- trainingdata[inTrain,]
testing <- trainingdata[-inTrain,]
dim(training)
```

```
## [1] 13737    53
```

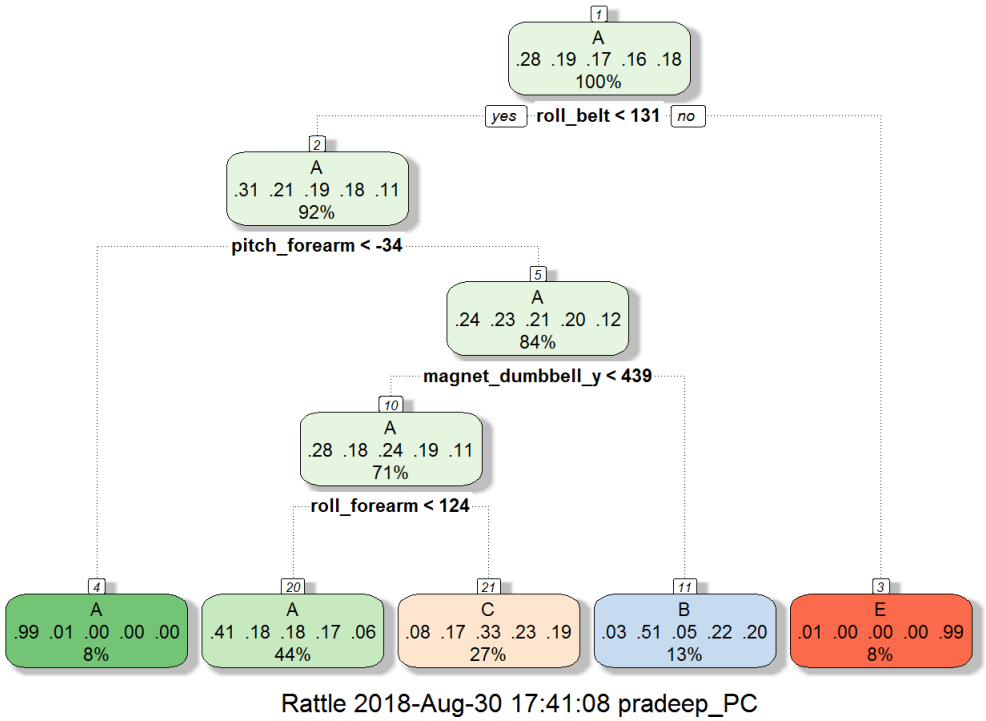
```
dim(testing)
```

```
## [1] 5885   53
```

Fitting ML alogrithms

Fitting a Decision Tree Model

```
fit_dt <- train(classe ~ ., data = training, method = "rpart")
fancyRpartPlot(fit_dt$finalModel)
```



Rattle 2018-Aug-30 17:41:08 pradeep_PC

```
predict_dt <- predict(fit_dt, newdata = testing)
confusionMatrix(predict_dt, testing$classe)$overall[1]
```

```
##  Accuracy
## 0.4997451
```

We can see that it has an accuracy of 0.49, which is quite low.

Fitting Random Forest Model

Fitting Random Forest Model

```
fit_rf <- randomForest(classe ~ ., data = training)
predict_rf <- predict(fit_rf, newdata = testing)
confusionMatrix(predict_rf, testing$classe)$overall[1]
```

```
##  Accuracy
## 0.9959218
```

This model has an accuracy of about 0.995 , which is quiet good.

Based on all above two models we can see that random forest model has the highest accuracy. So, using the random forest model to predict the testing data

Prediction

```
final_pred <- predict(fit_rf, newdata = testingdata)
final_pred
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
##  B  A  B  A  A  E  D  B  A  A  B  C  B  A  E  E  A  B  B  B
## Levels: A B C D E
```