

Machine learning project

Sarah Elsayed

July 7, 2019

Executive summary

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. The goal of this project is to predict the manner in which they did the exercise. This is the “classe” variable in the training set.

Cleaning The data

```
url1 <- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv"
url2 <- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv"
#Check if zip has already been downloaded in projectData directory?
if(!file.exists("training.csv")){
  download.file(url1, "training.csv")
  download.file(url2, "testing.csv")
}
TrainigData <- read.csv("training.csv")
TestingData <- read.csv("testing.csv")
head(TrainigData)
```

```
##      X user_name raw_timestamp_part_1 raw_timestamp_part_2  cvtd_timestamp
## 1 1  carlitos      1323084231              788290 05/12/2011 11:23
## 2 2  carlitos      1323084231              808298 05/12/2011 11:23
## 3 3  carlitos      1323084231              820366 05/12/2011 11:23
## 4 4  carlitos      1323084232              120339 05/12/2011 11:23
## 5 5  carlitos      1323084232              196328 05/12/2011 11:23
## 6 6  carlitos      1323084232              304277 05/12/2011 11:23
##      new_window num_window roll_belt pitch_belt yaw_belt total_accel_belt
## 1          no         11      1.41      8.07   -94.4              3
## 2          no         11      1.41      8.07   -94.4              3
## 3          no         11      1.42      8.07   -94.4              3
## 4          no         12      1.48      8.05   -94.4              3
## 5          no         12      1.48      8.07   -94.4              3
## 6          no         12      1.45      8.06   -94.4              3
##      kurtosis_roll_belt kurtosis_picth_belt kurtosis_yaw_belt
## 1
## 2
## 3
## 4
## 5
## 6
##      skewness_roll_belt skewness_roll_belt.1 skewness_yaw_belt max_roll_belt
## 1                                     NA
## 2                                     NA
```

##	3					NA
##	4					NA
##	5					NA
##	6					NA
##		max_pitch_belt	max_yaw_belt	min_roll_belt	min_pitch_belt	min_yaw_belt
##	1	NA		NA	NA	
##	2	NA		NA	NA	
##	3	NA		NA	NA	
##	4	NA		NA	NA	
##	5	NA		NA	NA	
##	6	NA		NA	NA	
##		amplitude_roll_belt	amplitude_pitch_belt	amplitude_yaw_belt		
##	1	NA		NA		
##	2	NA		NA		
##	3	NA		NA		
##	4	NA		NA		
##	5	NA		NA		
##	6	NA		NA		
##		var_total_accel_belt	avg_roll_belt	stddev_roll_belt	var_roll_belt	
##	1	NA	NA		NA	NA
##	2	NA	NA		NA	NA
##	3	NA	NA		NA	NA
##	4	NA	NA		NA	NA
##	5	NA	NA		NA	NA
##	6	NA	NA		NA	NA
##		avg_pitch_belt	stddev_pitch_belt	var_pitch_belt	avg_yaw_belt	
##	1	NA	NA	NA	NA	
##	2	NA	NA	NA	NA	
##	3	NA	NA	NA	NA	
##	4	NA	NA	NA	NA	
##	5	NA	NA	NA	NA	
##	6	NA	NA	NA	NA	
##		stddev_yaw_belt	var_yaw_belt	gyros_belt_x	gyros_belt_y	gyros_belt_z
##	1	NA	NA	0.00	0.00	-0.02
##	2	NA	NA	0.02	0.00	-0.02
##	3	NA	NA	0.00	0.00	-0.02
##	4	NA	NA	0.02	0.00	-0.03
##	5	NA	NA	0.02	0.02	-0.02
##	6	NA	NA	0.02	0.00	-0.02
##		accel_belt_x	accel_belt_y	accel_belt_z	magnet_belt_x	magnet_belt_y
##	1	-21	4	22	-3	599
##	2	-22	4	22	-7	608
##	3	-20	5	23	-2	600
##	4	-22	3	21	-6	604
##	5	-21	2	24	-6	600
##	6	-21	4	21	0	603
##		magnet_belt_z	roll_arm	pitch_arm	yaw_arm	total_accel_arm
##	1	-313	-128	22.5	-161	34
##	2	-311	-128	22.5	-161	34
##	3	-305	-128	22.5	-161	34
##	4	-310	-128	22.1	-161	34
##	5	-302	-128	22.1	-161	34
##	6	-312	-128	22.0	-161	34
##		avg_roll_arm	stddev_roll_arm	var_roll_arm	avg_pitch_arm	stddev_pitch_arm

##	1	NA	NA	NA	NA	NA
##	2	NA	NA	NA	NA	NA
##	3	NA	NA	NA	NA	NA
##	4	NA	NA	NA	NA	NA
##	5	NA	NA	NA	NA	NA
##	6	NA	NA	NA	NA	NA
##		var_pitch_arm	avg_yaw_arm	stddev_yaw_arm	var_yaw_arm	gyros_arm_x
##	1	NA	NA	NA	NA	0.00
##	2	NA	NA	NA	NA	0.02
##	3	NA	NA	NA	NA	0.02
##	4	NA	NA	NA	NA	0.02
##	5	NA	NA	NA	NA	0.00
##	6	NA	NA	NA	NA	0.02
##		gyros_arm_y	gyros_arm_z	accel_arm_x	accel_arm_y	accel_arm_z
##	1	0.00	-0.02	-288	109	-123
##	2	-0.02	-0.02	-290	110	-125
##	3	-0.02	-0.02	-289	110	-126
##	4	-0.03	0.02	-289	111	-123
##	5	-0.03	0.00	-289	111	-123
##	6	-0.03	0.00	-289	111	-122
##		magnet_arm_y	magnet_arm_z	kurtosis_roll_arm	kurtosis_pitch_arm	
##	1	337	516			
##	2	337	513			
##	3	344	513			
##	4	344	512			
##	5	337	506			
##	6	342	513			
##		kurtosis_yaw_arm	skewness_roll_arm	skewness_pitch_arm	skewness_yaw_arm	
##	1					
##	2					
##	3					
##	4					
##	5					
##	6					
##		max_roll_arm	max_pitch_arm	max_yaw_arm	min_roll_arm	min_pitch_arm
##	1	NA	NA	NA	NA	NA
##	2	NA	NA	NA	NA	NA
##	3	NA	NA	NA	NA	NA
##	4	NA	NA	NA	NA	NA
##	5	NA	NA	NA	NA	NA
##	6	NA	NA	NA	NA	NA
##		min_yaw_arm	amplitude_roll_arm	amplitude_pitch_arm	amplitude_yaw_arm	
##	1	NA	NA	NA	NA	
##	2	NA	NA	NA	NA	
##	3	NA	NA	NA	NA	
##	4	NA	NA	NA	NA	
##	5	NA	NA	NA	NA	
##	6	NA	NA	NA	NA	
##		roll_dumbbell	pitch_dumbbell	yaw_dumbbell	kurtosis_roll_dumbbell	
##	1	13.05217	-70.49400	-84.87394		
##	2	13.13074	-70.63751	-84.71065		
##	3	12.85075	-70.27812	-85.14078		
##	4	13.43120	-70.39379	-84.87363		
##	5	13.37872	-70.42856	-84.85306		

```

## 6      13.38246      -70.81759      -84.46500
## kurtosis_picth_dumbbell kurtosis_yaw_dumbbell skewness_roll_dumbbell
## 1
## 2
## 3
## 4
## 5
## 6
## skewness_pitch_dumbbell skewness_yaw_dumbbell max_roll_dumbbell
## 1 NA
## 2 NA
## 3 NA
## 4 NA
## 5 NA
## 6 NA
## max_picth_dumbbell max_yaw_dumbbell min_roll_dumbbell min_pitch_dumbbell
## 1 NA NA NA
## 2 NA NA NA
## 3 NA NA NA
## 4 NA NA NA
## 5 NA NA NA
## 6 NA NA NA
## min_yaw_dumbbell amplitude_roll_dumbbell amplitude_pitch_dumbbell
## 1 NA NA
## 2 NA NA
## 3 NA NA
## 4 NA NA
## 5 NA NA
## 6 NA NA
## amplitude_yaw_dumbbell total_accel_dumbbell var_accel_dumbbell
## 1 37 NA
## 2 37 NA
## 3 37 NA
## 4 37 NA
## 5 37 NA
## 6 37 NA
## avg_roll_dumbbell stddev_roll_dumbbell var_roll_dumbbell
## 1 NA NA NA
## 2 NA NA NA
## 3 NA NA NA
## 4 NA NA NA
## 5 NA NA NA
## 6 NA NA NA
## avg_pitch_dumbbell stddev_pitch_dumbbell var_pitch_dumbbell
## 1 NA NA NA
## 2 NA NA NA
## 3 NA NA NA
## 4 NA NA NA
## 5 NA NA NA
## 6 NA NA NA
## avg_yaw_dumbbell stddev_yaw_dumbbell var_yaw_dumbbell gyros_dumbbell_x
## 1 NA NA NA 0
## 2 NA NA NA 0
## 3 NA NA NA 0

```

```

## 4          NA          NA          NA          0
## 5          NA          NA          NA          0
## 6          NA          NA          NA          0
##   gyros_dumbbell_y gyros_dumbbell_z accel_dumbbell_x accel_dumbbell_y
## 1          -0.02          0.00          -234          47
## 2          -0.02          0.00          -233          47
## 3          -0.02          0.00          -232          46
## 4          -0.02          -0.02          -232          48
## 5          -0.02          0.00          -233          48
## 6          -0.02          0.00          -234          48
##   accel_dumbbell_z magnet_dumbbell_x magnet_dumbbell_y magnet_dumbbell_z
## 1          -271          -559          293          -65
## 2          -269          -555          296          -64
## 3          -270          -561          298          -63
## 4          -269          -552          303          -60
## 5          -270          -554          292          -68
## 6          -269          -558          294          -66
##   roll_forearm pitch_forearm yaw_forearm kurtosis_roll_forearm
## 1          28.4          -63.9          -153
## 2          28.3          -63.9          -153
## 3          28.3          -63.9          -152
## 4          28.1          -63.9          -152
## 5          28.0          -63.9          -152
## 6          27.9          -63.9          -152
##   kurtosis_pitch_forearm kurtosis_yaw_forearm skewness_roll_forearm
## 1
## 2
## 3
## 4
## 5
## 6
##   skewness_pitch_forearm skewness_yaw_forearm max_roll_forearm
## 1                      NA
## 2                      NA
## 3                      NA
## 4                      NA
## 5                      NA
## 6                      NA
##   max_pitch_forearm max_yaw_forearm min_roll_forearm min_pitch_forearm
## 1          NA          NA          NA          NA
## 2          NA          NA          NA          NA
## 3          NA          NA          NA          NA
## 4          NA          NA          NA          NA
## 5          NA          NA          NA          NA
## 6          NA          NA          NA          NA
##   min_yaw_forearm amplitude_roll_forearm amplitude_pitch_forearm
## 1                      NA          NA
## 2                      NA          NA
## 3                      NA          NA
## 4                      NA          NA
## 5                      NA          NA
## 6                      NA          NA
##   amplitude_yaw_forearm total_accel_forearm var_accel_forearm
## 1                      36          NA

```

```

## 2          36          NA
## 3          36          NA
## 4          36          NA
## 5          36          NA
## 6          36          NA
##   avg_roll_forearm stddev_roll_forearm var_roll_forearm avg_pitch_forearm
## 1          NA          NA          NA          NA
## 2          NA          NA          NA          NA
## 3          NA          NA          NA          NA
## 4          NA          NA          NA          NA
## 5          NA          NA          NA          NA
## 6          NA          NA          NA          NA
##   stddev_pitch_forearm var_pitch_forearm avg_yaw_forearm
## 1          NA          NA          NA
## 2          NA          NA          NA
## 3          NA          NA          NA
## 4          NA          NA          NA
## 5          NA          NA          NA
## 6          NA          NA          NA
##   stddev_yaw_forearm var_yaw_forearm gyros_forearm_x gyros_forearm_y
## 1          NA          NA          0.03          0.00
## 2          NA          NA          0.02          0.00
## 3          NA          NA          0.03         -0.02
## 4          NA          NA          0.02         -0.02
## 5          NA          NA          0.02          0.00
## 6          NA          NA          0.02         -0.02
##   gyros_forearm_z accel_forearm_x accel_forearm_y accel_forearm_z
## 1         -0.02          192          203          -215
## 2         -0.02          192          203          -216
## 3          0.00          196          204          -213
## 4          0.00          189          206          -214
## 5         -0.02          189          206          -214
## 6         -0.03          193          203          -215
##   magnet_forearm_x magnet_forearm_y magnet_forearm_z classe
## 1          -17          654          476          A
## 2          -18          661          473          A
## 3          -18          658          469          A
## 4          -16          658          469          A
## 5          -17          655          473          A
## 6          -9          660          478          A

```

```
#Check NA sum in Columns knowing that the total training size is 19622
```

```
column_NAstatus<-colSums(is.na(TrainigData))
```

```
column_NAstatus<-data.frame(column_NAstatus)
```

```
#by viewing the NA counts there are many column with 19216 NA values which makes these columns not impo
```

```
column_NAstatus[20:40,]
```

```
## [1] 0 19216 19216 0 19216 19216 0 19216 19216 19216 19216
```

```
## [12] 19216 19216 19216 19216 19216 19216 0 0 0 0
```

```
# get rid of columns that have NA values more than 0.5 of total values
```

```
NACol<-colSums(is.na(TrainigData))<nrow(TrainigData)/2
```

```
TrainigData1<-TrainigData[,NACol]
```

```
Comp<-c(ncol(TrainigData),ncol(TrainigData1))
names(Comp)<-c("before","after")
# 67 unnecessary columns have been removed
Comp
```

```
## before after
##      160     93
```

```
# apply for the testing data
TestingData<-TestingData[,NACol]

#Get rid of empty columns
Emptycol<-colSums(TrainigData1!="")>nrow(TrainigData1)/2
TrainigData2<-TrainigData1[,Emptycol]
Comp<-c(ncol(TrainigData1),ncol(TrainigData2))
names(Comp)<-c("before","after")
# 33 unnecessary columns with 19216 empty values from 19622 values have been removed
Comp
```

```
## before after
##       93     60
```

```
TrainigData<-TrainigData2
# apply for the testing data
TestingData<-TestingData[,Emptycol]

# filter out highly correlated predictors
CorrFilter<-preProcess(TrainigData, method = "corr")
TrainigData <- predict(CorrFilter, TrainigData)
TestingData<- predict(CorrFilter, TestingData)

#Remove the non-predictors columns
TrainigData <-TrainigData[,~c(1:7)]
TestingData <-TestingData[,~c(1:7)]

dim(TestingData)
```

```
## [1] 20 46
```

```
dim(TrainigData)
```

```
## [1] 19622    46
```

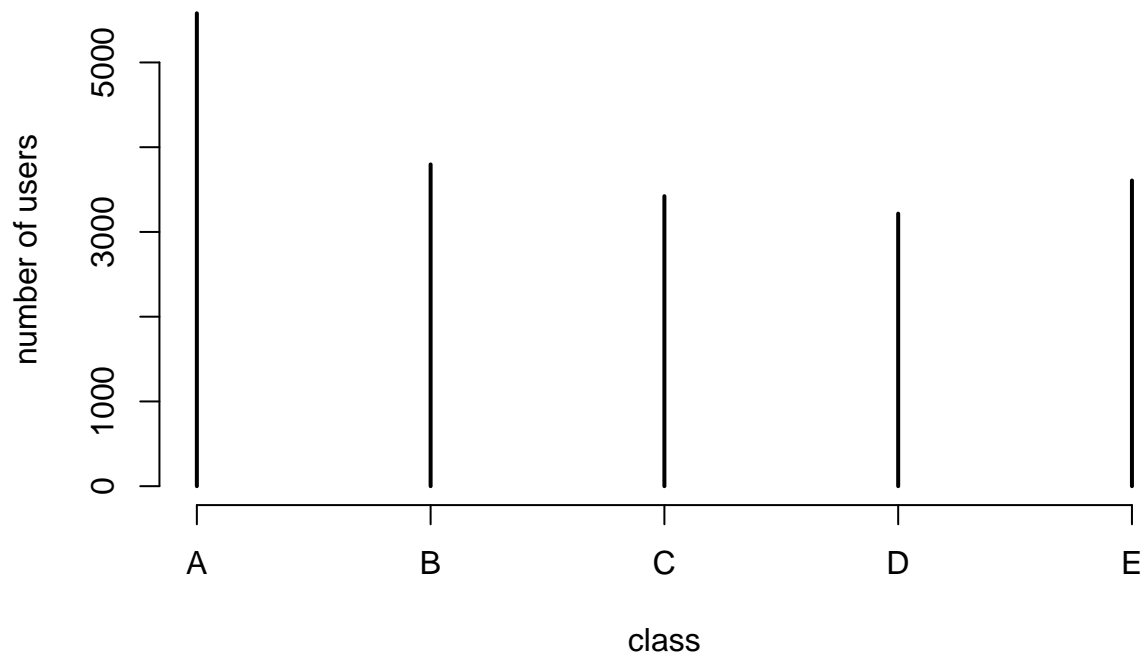
Pre__Processing

```
#Check for near zero values
preProc<-preProcess(TrainigData, method = "nzv")
#non near zero values found
preProc
```

```
## Created from 19622 samples and 1 variables
##
## Pre-processing:
##   - ignored (1)
```

Exploratory data analysis

```
#plot the class variabe, It is noted that the arrange of the classes A,B,E,C,D
plot(table(TrainigData$classe),xlab = "class", ylab ="number of users")
```



```
# Have a look on the users
table(TrainigData$user_name)
```

```
## < table of extent 0 >
```

The cleaned data contains 19622 observations and 53 variables.

Training and prediction


```

#Split the data into training and validation
set.seed(333)
inTrain      <- createDataPartition(TrainigData$classe, p = 0.6, list = FALSE)
Training     <- TrainigData[inTrain,]
validation   <- TrainigData[-inTrain,]

# Using randomforest algorithm and 3 folds
controlRF <- trainControl(method="cv", number=3, verboseIter=FALSE)
trainingModel <- train(classe ~ ., data=Training, method="rf",trControl=controlRF)
trainingModel$finalModel

```

```

##
## Call:
##  randomForest(x = x, y = y, mtry = param$mtry)
##              Type of random forest: classification
##              Number of trees: 500
## No. of variables tried at each split: 23
##
##              OOB estimate of  error rate: 0.82%
## Confusion matrix:
##      A      B      C      D      E class.error
## A 3343      2      1      0      2 0.001493429
## B   24 2250      5      0      0 0.012724879
## C      0   20 2025      9      0 0.014118793
## D      0      0   23 1905      2 0.012953368
## E      0      0      2      6 2157 0.003695150

```

```

#Prediction on the validation set
predict<- predict(trainingModel, validation)
cm<-confusionMatrix(validation$classe, predict)

```

```

#The validation accuracy
cm$overall['Accuracy']

```

```

## Accuracy
## 0.9917155

```

```

#Error calculating
Error <- 1 - cm$overall['Accuracy']
Error

```

```

## Accuracy
## 0.008284476

```

```

#Prediction on the test set
predict<- predict(trainingModel, TestingData)
predict

```

```

## [1] 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

```