

# PML peer graded assignment

Osiepere

8 February 2019

```
library(lattice)
library(ggplot2)
library(caret)
library(rpart)
library(rpart.plot)
library(corrplot)

## corrplot 0.84 loaded

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)

## 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

library(RColorBrewer)

set.seed(1813)

url_train <- "http://d396qusza40orc.cloudfront.net/predmachlearn/pml-
training.csv"
url_quiz  <- "http://d396qusza40orc.cloudfront.net/predmachlearn/pml-
testing.csv"

data_train <- read.csv(url(url_train), strip.white = TRUE, na.strings =
c("NA", ""))
```

```

data_quiz <- read.csv(url(url_quiz), strip.white = TRUE, na.strings =
c("NA", ""))

dim(data_train)

## [1] 19622 160

dim(data_quiz)

## [1] 20 160

in_train <- createDataPartition(data_train$classe, p=0.75, list=FALSE)
train_set <- data_train[ in_train, ]
test_set <- data_train[ -in_train, ]
dim(train_set)

## [1] 14718 160

dim(test_set)

## [1] 4904 160

nzv_var <- nearZeroVar(train_set)
train_set <- train_set[ , -nzv_var]
test_set <- test_set [ , -nzv_var]
dim(train_set)

## [1] 14718 121

dim(test_set)

## [1] 4904 121

na_var <- sapply(train_set, function(x) mean(is.na(x))) > 0.95
train_set <- train_set[ , na_var == FALSE]
test_set <- test_set [ , na_var == FALSE]
dim(train_set)

## [1] 14718 59

dim(test_set)

## [1] 4904 59

train_set <- train_set[ , -(1:5)]
test_set <- test_set [ , -(1:5)]
dim(train_set)

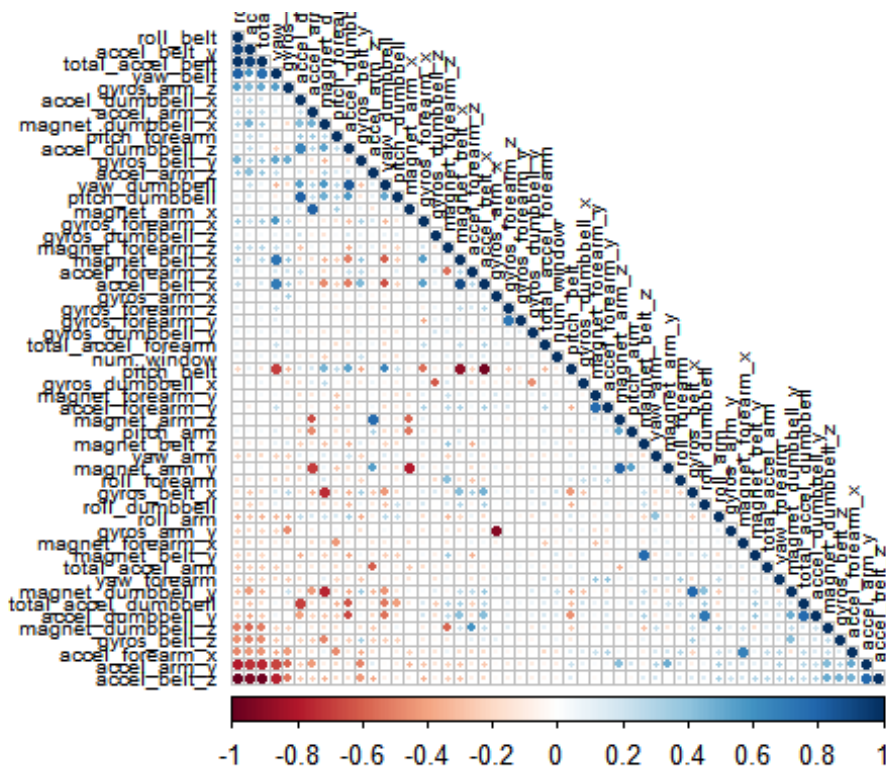
## [1] 14718 54

dim(test_set)

## [1] 4904 54

```

```
corr_matrix <- cor(train_set[, -54])
corrplot(corr_matrix, order = "FPC", method = "circle", type = "lower",
         tl.cex = 0.6, tl.col = rgb(0, 0, 0))
```



```
set.seed(1813)
fit_decision_tree <- rpart(classe ~ ., data = train_set, method="class")
fancyRpartPlot(fit_decision_tree)

## Warning: labs do not fit even at cex 0.15, there may be some overplotting
```



```
##
##               Class: A Class: B Class: C Class: D Class: E
## Sensitivity      0.8946  0.6207  0.7801  0.6144  0.7492
## Specificity      0.9139  0.9469  0.9474  0.9234  0.9520
## Pos Pred Value   0.8052  0.7372  0.7580  0.6114  0.7785
## Neg Pred Value   0.9562  0.9123  0.9533  0.9243  0.9440
## Prevalence       0.2845  0.1935  0.1743  0.1639  0.1837
## Detection Rate   0.2545  0.1201  0.1360  0.1007  0.1376
## Detection Prevalence 0.3161 0.1629 0.1794 0.1648 0.1768
## Balanced Accuracy 0.9043  0.7838  0.8638  0.7689  0.8506
```

The predictive accuracy of the decision tree model is relatively low at 74.9 %.

```
plot(conf_matrix_decision_tree$table, col =
conf_matrix_decision_tree$byClass,
     main = paste("Decision Tree Model: Predictive Accuracy =",
                  round(conf_matrix_decision_tree$overall['Accuracy'], 4)))
```

**Decision Tree Model: Predictive Accuracy = 0.749**

