Predictivie analysis of quality of fitness movements based on wearable equipmenent data

Data aquiring and cleaning

In this section training and testing datasets are load into R. In addition, due to issues of algorithms to work with NA values, the columns with NA in them are removed.

```
set.seed(355)
library(caret); library(ggplot2); library(RANN); library(klaR); library(dplyr); library(randomForest)
setwd("D:/Coursera/MLFirst")
testing3<- read.csv("pml-testing.csv", na.strings=c("NA","#DIV/0!", ""))
training3<- read.csv("pml-training.csv", na.strings=c("NA","#DIV/0!", ""))
training3<-training3[ , apply( training3, 2, function(x) !any(is.na(x)))]
```

Afterwards, knn pre processing should be performed on the training data set to impute NA values if they exist. We named the training data as validation and devided the test data into a training and a test datasets.

```
prePt<- preProcess(training3, method=c("knnImpute"))
training1<- predict(prePt,training3)
testing1<- predict(prePt,testing3)
training1$classe <- factor(training1$classe)
validation<- testing1[,-c(1:7)]
training2<- training1[,-c(1:7)]
INtraining<- createDataPartition(training2$classe, p=0.7, list=FALSE)
training<- training2[INtraining,]
testing<- training2[-INtraining,]
```

To introuce cross validation into the solution we chose a 5 number K-fold strategy.

```
train_control <- trainControl(method="cv", number=5)
```

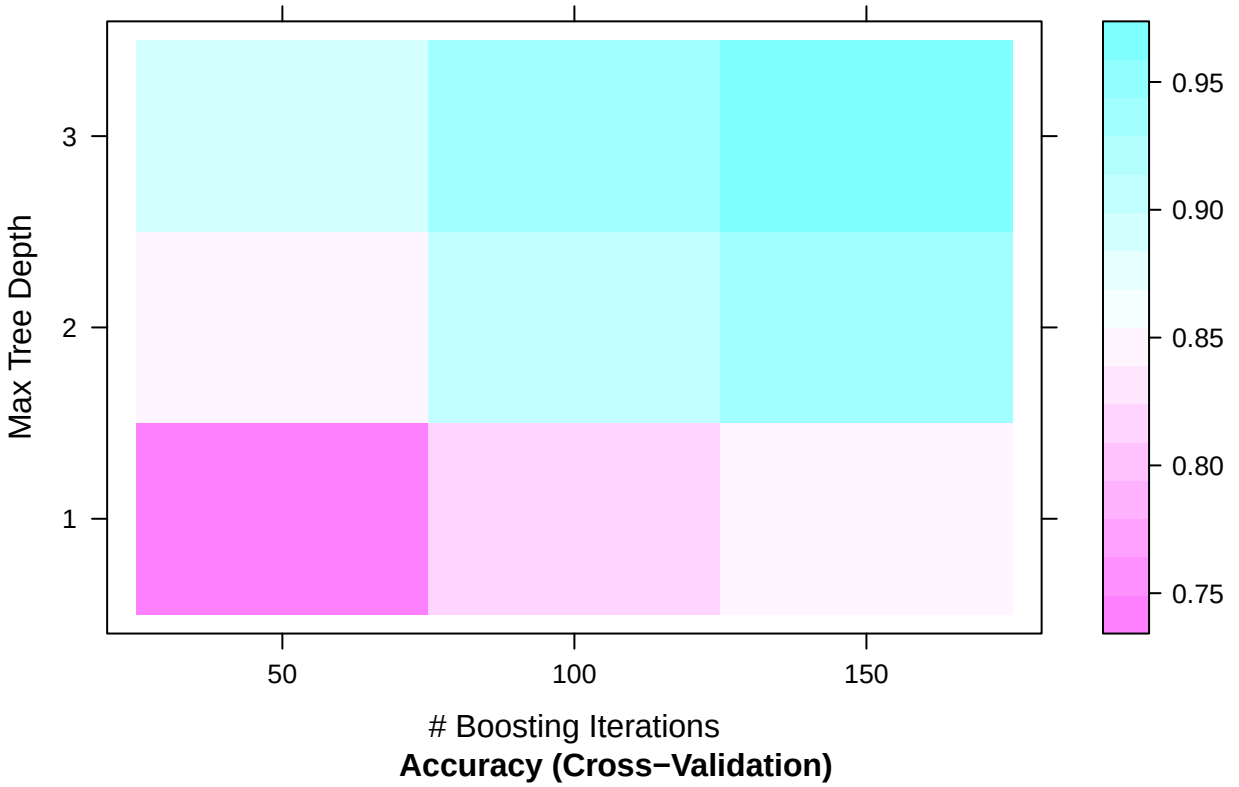
The first model that we trained on our training dataset was a naive bayesian algorithm. However accuracy of this model is not desireable so we decided to add three models and use those sacked models to predict the outcome.

```
fitnb<- train(classe~., data=training, trControl=train_control, method="nb", na.action=na.pass)
fittednb<- predict(fitnb, testing)
CMnb<- confusionMatrix(fittednb,testing$classe)
```

The secong model that was trained on the training data was boosting algorithm.

```
fitgbm<- train(classe~., data=training, trControl=train_control, method="gbm", na.action=na.pass, verbos
fittedgbm<- predict(fitgbm, testing)
CMgbm<- confusionMatrix(fittedgbm,testing$classe)
```

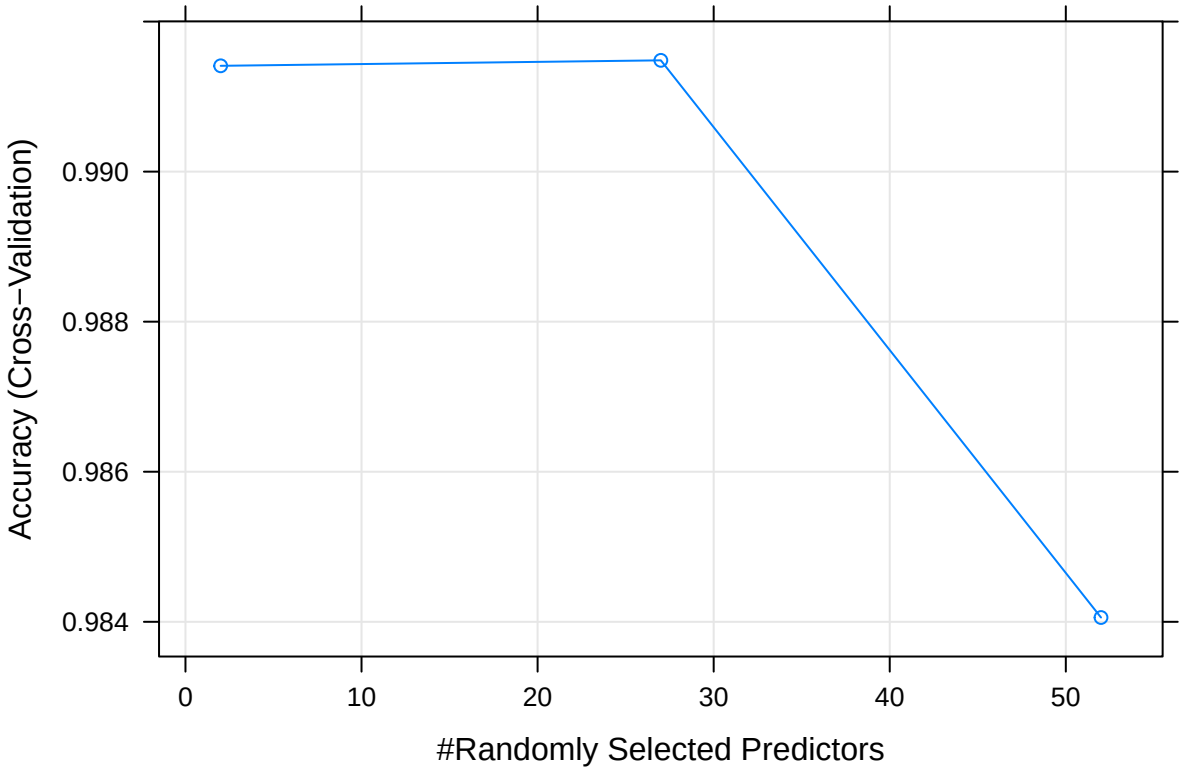
```
plot(fitgbm,plotType = "level")
```



The third model that was trained on the training data was rand forest algorithm.

```
fitrf<- train(classe~., data=training, trControl=train_control, method="rf", na.action=na.pass)
fittedrf<- predict(fitrf, testing)
CMrf<- confusionMatrix(fittedrf,testing$classe)
```

```
plot(fitrf)
```



Finally, all these three models are stacked and another random forest algorithm was trained on the test data.

```
prDF<-data.frame(fittednb,fittedrf, fittedgbm, classe=testing$classe)
fitcombined<- train(classe~., data=prDF, trControl=train_control, method="rf", na.action=na.pass)
fittedcombined<- predict(fitcombined, prDF)
CMcombined<- confusionMatrix(fittedcombined,testing$classe)
```

```
Combined.accuracy <- CMcombined$overall['Accuracy']
rf.accuracy <- CMrf$overall['Accuracy']
gbm.accuracy <- CMgbm$overall['Accuracy']
nb.accuracy <- CMnb$overall['Accuracy']
AccuracyDF<- data.frame(nb=nb.accuracy, gbm=gbm.accuracy,rf=rf.accuracy,combined=Combined.accuracy)
AccuracyDF
```

```
##                  nb       gbm        rf  combined
## Accuracy 0.7425658 0.9658454 0.9920136 0.9920136
```

Finallly all four models are used to predict the classe outcome based on the data in the validation data set. A comparison of all four models can be seen below.

```
fittedrfV<- predict(fitrf, validation)
fittedgbmV<- predict(fitgbm, validation)
fittednbV<- predict(fitnb, validation)
prDFV<-data.frame(fittednb=fittednbV,fittedrf=fittedrfV,fittedgbm=fittedgbmV)
fittedcombinedV<- predict(fitcombined, newdata=prDFV)
```

```
ResultDF<- data.frame(nb=fittednbV, gbm=fittedgbmV,rf=fittedrfV,combined=fittedcombinedV)
ResultDF
```

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