Human Activity Recognition and Classification

Firas Ismail

02/05/2020

Overview

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, our goal will be to use data from accelerometers on the belt, forearm, arm, and dumbell of 6 participants who are performing barbell lifts to predict whether they're doing it correctly or not.
The data used is from *gruopware @LES*. more information and details about the data set can be found here

Getting the data

First we include the necessary packages , then borth a training set and a testing data set can be downloaded and imported to our environment through this R code:

```
library(ggplot2)
library(caret)
library(dplyr)
library(rattle)
library(corrplot)
library(stringi)
library(stringr)

download.file("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv",
              destfile = "train.csv")
download.file("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv",
              destfile = "test.csv")

dat<-read.csv("train.csv")
val<-read.csv("test.csv")
dim(dat)
```

```
## [1] 19622   160
```

```
dim(val)
```

```
## [1]  20 160
```

NOTE the "*test* data will be used as a validation data for the final quiz

Exploratory analysis and Pre-processing

We set the seed to **69** for later reproducibilty and get some idea about the outcome(**classe**) and the frequencies of its values.

```
set.seed(69)
table(dat$classe)/nrow(dat)
```

```
##
##         A         B         C         D         E
## 0.2843747 0.1935073 0.1743961 0.1638977 0.1838243
```

We plot some more feature tables:

```
table(dat$new_window)
```

```
##
##    no   yes
## 19216   406
```

```
table(is.na(dat$max_picth_belt))
```

```
##
## FALSE  TRUE
##   406 19216
```

we notice many variables have a lot of NA's and empty cases. Using the previous tables, it appears that those NA's are caused by the *new_window* variable.

```
table(dat$new_window)[1]/nrow(dat)
```

```
##        no
## 0.9793089
```

the above table indicates that 97.9308939% of the *new_window* values are "no" which cause the same percentage of NA's in other variables
Note since the NA's make up 98% of those columns , Imputing the missing values won't make much sense since we can't use 2% of the data to fill the other 98%. It would be wiser if we deleted the valriables.

Feature selection

We remove the NA's features from out data frame

```
x<-dat[dat$new_window=="no",]
nzv<-nearZeroVar(x,saveMetrics = TRUE)
dat<-dat[,!nzv$nzv]
val<-val[,!nzv$nzv]
```

now we remove the first 6 features (ID , name , timestamps ..) because of their irrelevance to our classe prediction.

```
dat<-dat[,-c(1:6)]
val<-val[,-c(1:6)]
```

We check if we still have NA's in our data frame:

```
table(is.na(dat))
```

```
##
##   FALSE
## 1039966
```

No more NA values.

Splitting the data

We split the data in *dat* to training and testing datasets.

```
intrain<-createDataPartition(dat$classe,p=0.7,list=FALSE)
ts<-dat[-intrain,] ##Testing dataset
tr<-dat[intrain,]  ##Training dataset

dim(tr)
```

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

```
dim(ts)
```

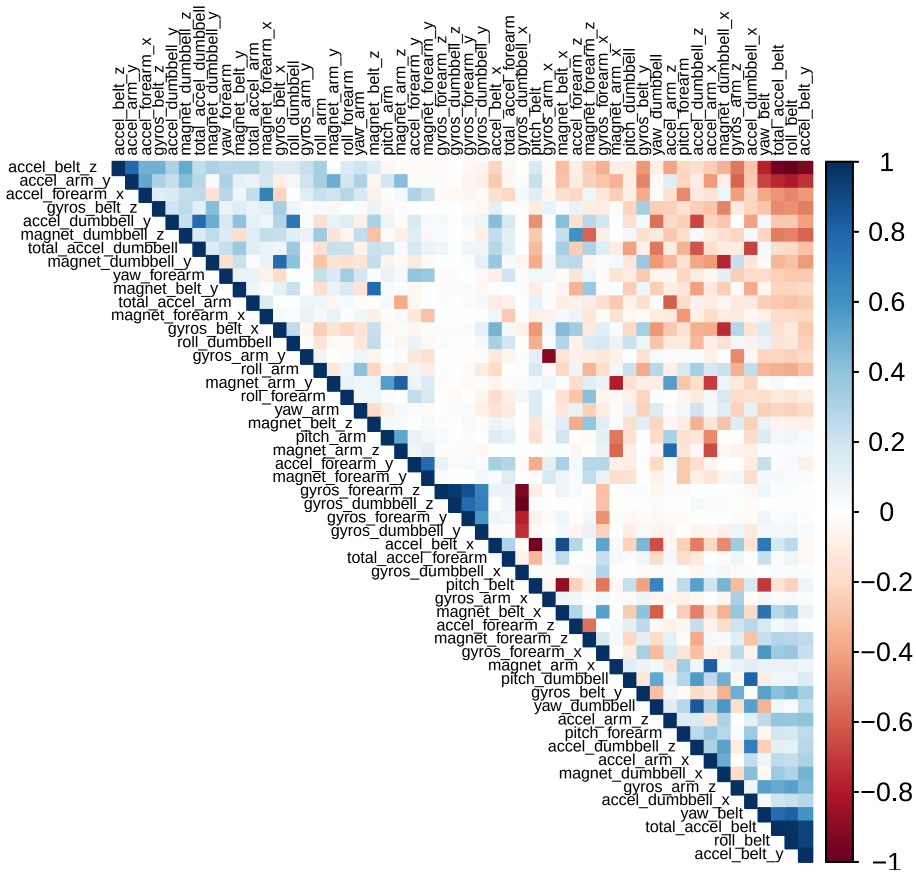
```
## [1] 5885   53
```

```
dim(val)
```

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

We plot the correlation table to see if we have a big cluster of correlated feature that can cause a problem:

```
corrplot(cor(tr[, -53]), order = "FPC", method = "color", type = "upper",
         tl.cex = 0.5, tl.col = rgb(0, 0, 0))
```



One last step before we try our models, we need to scale and center our variables.

```
prep<-preProcess(tr,method = c("center","scale"))
tr<-predict(prep,tr)
ts<-predict(prep,ts)
val<-predict(prep,val)
```

we make sure to apply the same pre-process with the same *mean* and *Std.deviation* to both the *testing* and *validation* datasets

Fitting a Decision Tree model

the decision tree model (CART) can have a good accuracy on classificaltion task when using cross validation and a good CP value.

we use our trainControl to set the cross validation to 5-folds repeated 5 times. then we Tune to different CP values in the tuneGrid function.

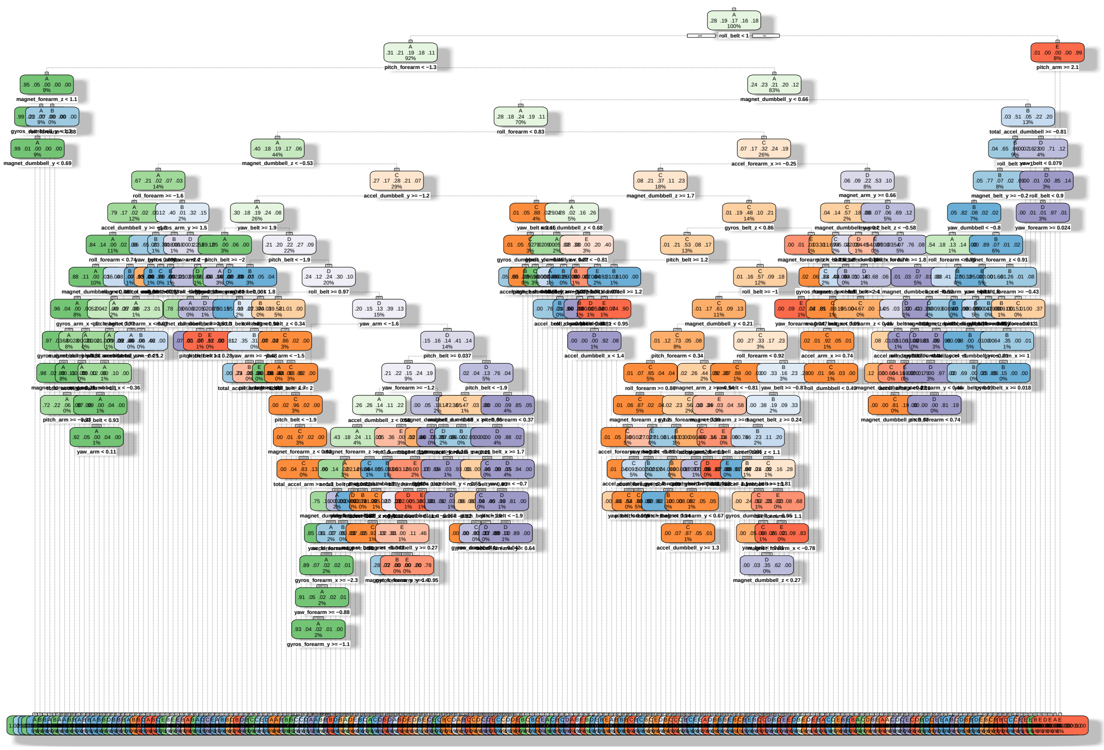
```
ctr<-trainControl(method="repeatedcv",number=5,repeats=5)
tune<-data.frame(cp=c(0.1,0.01,0.001,0.0001))
```

now we fit our model:

```
mf1<-train(classe~.,data=tr,method="rpart",trControl=ctr,tuneGrid=tune)
```

We can visualize our model using the *fancyRpartPlot* function from the *rattle* package

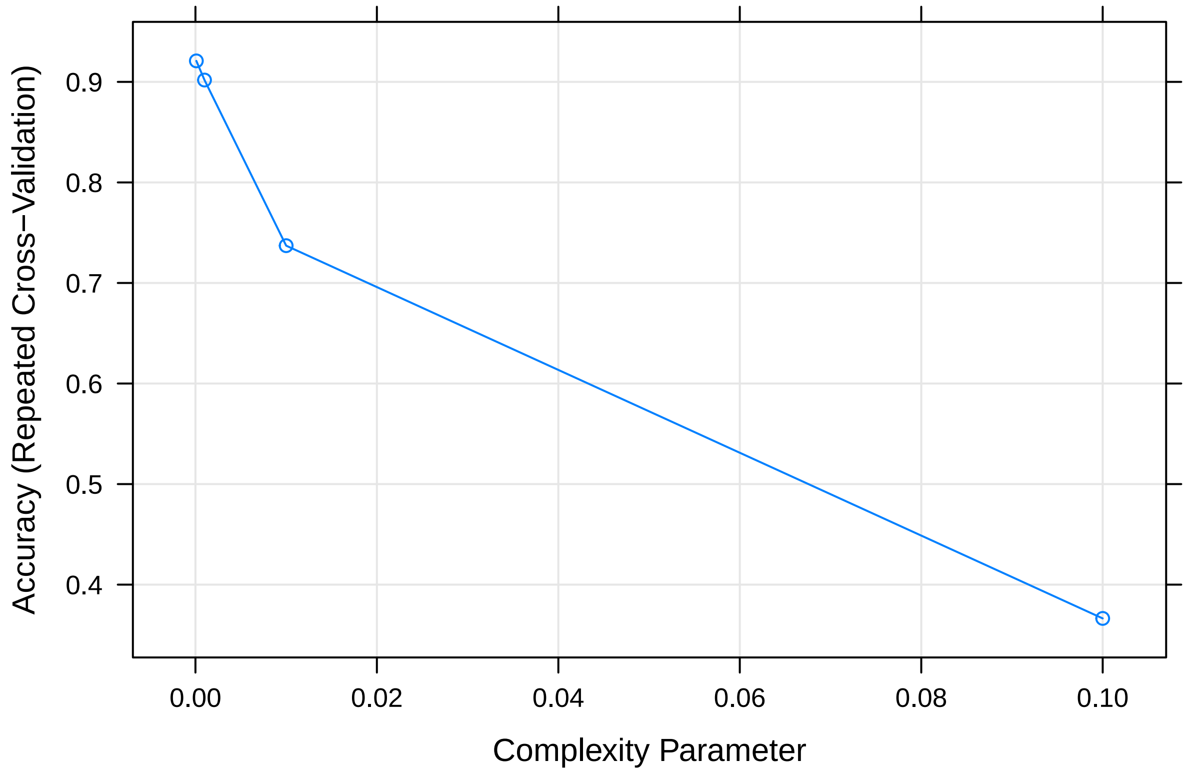
```
fancyRpartPlot(mf1$finalModel)
```



Rattle 2020−mai−02 11:06:34 ASUS

We can use the plot the model accuracy by its Complexity Parameter

```
plot(mf1)
```



We notice we get our best accuracy with a CP=0.0001.

Prediction we test the model using the testing data.

```
pre1<-predict(mf1,ts)
con<-confusionMatrix(pre1,ts$classe)
con$overall[1]
```

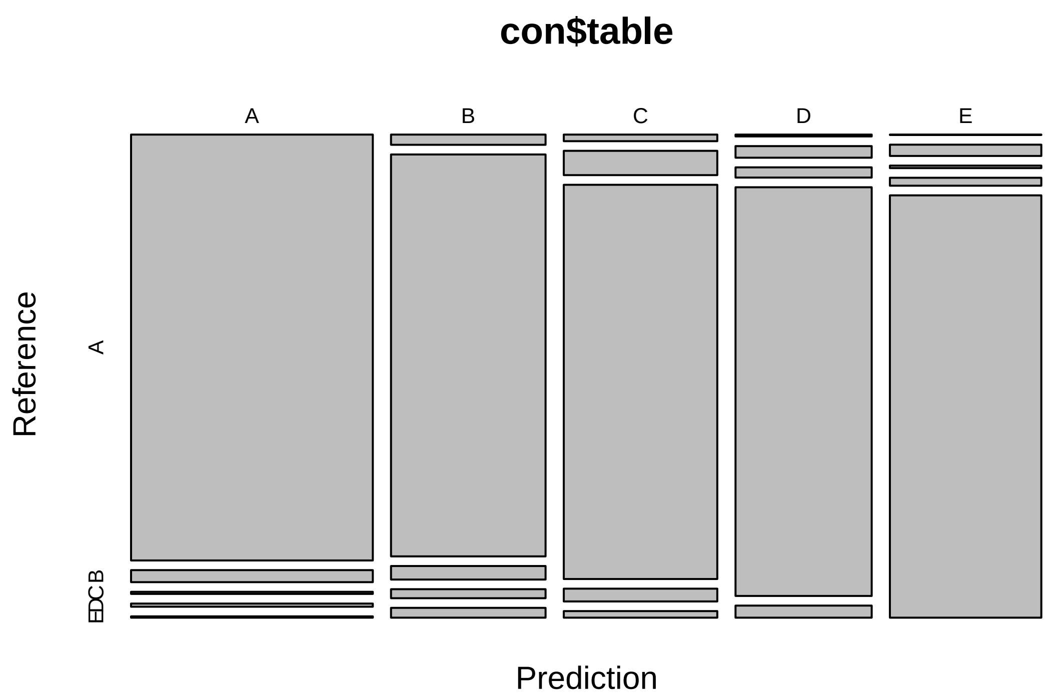
```
##  Accuracy
## 0.9274427
```

```
con$table
```

```
##           Reference
## Prediction    A    B    C    D    E
##          A 1628   46    8   12    5
##          B   25  982   33   22   24
##          C   16   59  957   31   16
##          D    4   25   22  880   26
##          E    1   27    6   19 1011
```

We get a good accuracy of 0.9274427 and our confusion table has good sensitivies and specificities.

```
plot(con$table)
```



We also predict the Validation set classe:

```
pre2<-predict(mf1,val)
pre2
```

```
##  [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
```

the above validation classification got me a 95% score on the course quizz, which makes it around 95% accurate for the validation test.

Error rate

the error rate 0.0725573 can be explained by random noise during the barbell exercice with each of the 6 participants doing his task slightly differenct than the others. it can also be caused by the features that got removed for excessive NA values. ### Fitting Other Models I've tried applied Random Forests and GBM but unfortunately the running time was taking too long because of the high data dimensions and i had to kill the process

Conclusion

Human Activity can be recognized and classified with a good accuracy even for a very specific task like barbell lifting, which can maybe in the futre help health professionals to study patients' movement patterns and predict a health problems