First of all I fetched the training data from csv format to R.

training<-read.csv("C:/pml-training.csv")

It has a lot of features that are not useful for creating the model fit. The first candidate for eliminations are those factors that have low variance. So, we will have:

nsv<-nearZeroVar(training, saveMetrics=TRUE)

tmp<-training[,!nsv[,4]]

In our dataset we have a lot of covariates with NULL value. I wrote a removeNullColumns() function to detecting such features and remove them from the dataset as below:

removeNullColumns<-function(data){

loop<-dim(data)[2]

index<-logical(loop)

for(i in 1:loop){

if(sum(is.na(data[,i]))>(length(data[,i])/2)){

index[i]<-TRUE

}

}

data[,!index]

}

Finally, the first six columns are not very useful for classification tasks. The name of the participants, time span, etc distorts the results.

tmp<-tmp[,-c(1:6)]

Now the preprocessing is done and we can go to split the training data to test and training set:

#create data partition

sepIndex<-createDataPartition(training$classe,p=0.60,list=FALSE)

#create test and training set

trainingset<-training[sepIndex,]

testingset<-training[-sepIndex,]

We should make ourselves ready for cross-validation so,

#cross validation

ctrl<-trainControl(method="cv",number=10)

Now we are ready to make our Model. I applied the random forest for classification:

#create the model fit

modelFit<-train(classe~.,method="rf",data=trainingset,trControl=ctrl)

print(modelFit$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: 2

OOB estimate of error rate: 0.82%

Confusion matrix:

A B C D E class.error

A 3344 2 1 0 1 0.001194743

B 18 2250 11 0 0 0.012724879

C 0 17 2034 3 0 0.009737098

D 0 0 37 1893 0 0.019170984

E 0 0 2 5 2158 0.003233256

And at the end apply the created model on the testingset:

prediction<-predict(modelFit,newdata=testingset)

confusionMatrix(prediction, testingset$classe)

The result would be:

Confusion Matrix and Statistics

Reference

Prediction A B C D E

A 2231 12 0 0 0

B 1 1504 14 0 0

C 0 2 1347 36 0

D 0 0 7 1249 4

E 0 0 0 1 1438

Overall Statistics

Accuracy : 0.9902

95% CI : (0.9877, 0.9922)

No Information Rate : 0.2845

P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.9876

Mcnemar's Test P-Value : NA

Statistics by Class:

Class: A Class: B Class: C Class: D Class: E

Sensitivity 0.9996 0.9908 0.9846 0.9712 0.9972

Specificity 0.9979 0.9976 0.9941 0.9983 0.9998

Pos Pred Value 0.9947 0.9901 0.9726 0.9913 0.9993

Neg Pred Value 0.9998 0.9978 0.9967 0.9944 0.9994

Prevalence 0.2845 0.1935 0.1744 0.1639 0.1838

Detection Rate 0.2843 0.1917 0.1717 0.1592 0.1833

Detection Prevalence 0.2859 0.1936 0.1765 0.1606 0.1834

Balanced Accuracy 0.9987 0.9942 0.9894 0.9848 0.9985