Predicting the 'Manner of Exercise' from wearable accelerometer data

Coursera: Machine Learning Course

SYNOPSIS

We partition the training data set into 'training' and 'cross-validation' sub-datasets. For noise reduction and efficency purposes, we eliminate near-zero variables from the list of possible predictors. Utilizing the 'training' sub-dataset, we then build a decision tree model with the 'classe' as the response variable. Then, we apply this model to predict the variable 'classe' in the cross-validation subset. By comparing the prediction with the true values, we compute the model's accuracy and misclassification rates.

Note: The 'real testing dataset is NOT utilized for model building, (nor could it be utilized, since the 'classe' variable is missing in it)

DATA PROCESSING

Load

```
##rm(list = ls())
library(tree)
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.1.1
```

```
## Loading required package: lattice
## Loading required package: ggplot2
```

```
##download.file("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-
training.csv", method = "curl", quiet = TRUE, destfile =
"training.csv")
##download.file("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-
testing.csv", method = "curl", quiet = TRUE, destfile =
"testing.csv")

traindata_raw       <- read.csv("training.csv")
testdata_raw        <- read.csv("testing.csv")    ## This is not
used in the model building process
##head(traindata_raw)
```

Preprocess

```
temp             <- traindata_raw[,2:160]          ## Don't use obs
number as a predcitor
nzv              <- nearZeroVar(temp)
traindata        <- temp[,-nzv]               ##Eliminate near-zero
variables, since they have little predictive value.

##Split training data into training and cross-validation
set.seed(3);
intrain       <- createDataPartition(y = temp$classe, p = 0.8,list
= FALSE)
trg           <- traindata[intrain,]
cross_val     <- traindata[-intrain,]
```

Model Building

```
## Build the model
tree.model1      <- tree( classe ~., data = trg)
print(summary(tree.model1))
```

```
##
## Classification tree:
## tree(formula = classe ~ ., data = trg)
## Variables actually used in tree construction:
##  [1] "cvtd_timestamp"       "magnet_dumbbell_y"
"var_accel_dumbbell"
##  [4] "magnet_dumbbell_z"    "raw_timestamp_part_1"
"gyros_dumbbell_y"
##  [7] "min_roll_dumbbell"    "gyros_arm_x"
"magnet_forearm_y"
## [10] "roll_belt"            "pitch_arm"
"stddev_roll_belt"
## [13] "avg_roll_dumbbell"    "magnet_belt_z"
"accel_forearm_x"
## Number of terminal nodes:  18
## Residual mean deviance:  0.354 = 107 / 302
## Misclassification error rate: 0.0688 = 22 / 320
```

```
## Model Improving
cv               <- cv.tree(tree.model1, FUN = prune.misclass)
print(names(cv))
```

```
## [1] "size"   "dev"    "k"      "method"
```

```
print(cv)
```

```
## $size
##  [1] 18 16 14 12 11  9  8  7  6  4  3  2  1
##
## $dev
##  [1]  68  68  69  77  71  75  84 103 119 119 175 174 238
##
## $k
##  [1] -Inf    0    1    3    5    6    9   11   13   15   29   32
67
##
## $method
## [1] "misclass"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

```
tree.model2      <- prune.misclass(tree.model1, best = 3)
```

Cross-validate and Calculate the Accuracy and Misclassification Rates

```
pred             <-  predict(tree.model1, newdata = cross_val, type
= "class")   ##Utilize the cross-val subset of the training data to
study goodness of the model
true             <-  cross_val$classe
result_matrix    <- table(pred,true)
cat("The confusion matrix is:\n")
```

```
## The confusion matrix is:
```

```
print(result_matrix)
```

```
##     true
## pred   A   B   C   D   E
##    A 844  53   1   0   0
##    B 272 701 492 107   0
##    C   0   0  15   1   1
##    D   0   0   0  11   1
##    E   0   5 176 524 719
```

```
accuracy         <- sum(diag(result_matrix))/sum(result_matrix)
misclass         <- 1- accuracy
```

RESULTS

```
cat("The accuracy and misclassification rates of the decsion tree
model are", signif(accuracy,2), " and ", signif(misclass,2), "
respectively")
```

```
## The accuracy and misclassification rates of the decsion tree
model are 0.58  and  0.42  respectively
```