Practical machine learning course project

Rasmus Klitgaard

2025-01-23

```
library(caret)
```

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

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

```
library(randomForest)
```

```
## randomForest 4.7-1.2
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

First off, we will download and load the data.

```
download.file("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv", "training.csv")
download.file("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv", "testing.csv")

train <- read.csv("training.csv")
quiz_data <- read.csv("testing.csv")
```

First off, we will look at what is in the dataset. The dimension of the training is 19622, 160 and the dimension of the test is 20, 160. So there are 159 different features.

We will remove near zero variance parameters. First we split the training data in 80-20.

```
use_training <- createDataPartition(train$classe, p=0.80, list=FALSE)
training_set <- train[use_training,]
testing_set <- train[-use_training,]
dim(training_set)
```

```
## [1] 15699   160
```

and remoing the near zero variance parameters.

```
near_zero_variance <- nearZeroVar(training_set)
training_set <- training_set[, -near_zero_variance]
testing_set  <- testing_set [, -near_zero_variance]
dim(training_set)
```

```
## [1] 15699   102
```

```
dim(testing_set)
```

```
## [1] 3923  102
```

We should probably remove the name and timestamps and such. These are the first 5 parameters.

```
training_set <- training_set[ , -(1:5)]
testing_set  <- testing_set [ , -(1:5)]
dim(training_set)
```

```
## [1] 15699    97
```

```
dim(testing_set)
```

```
## [1] 3923   97
```

We should also remove variables with only NAs. We remove rows with NAs.

```
na_var <- sapply(training_set, function(x) mean(is.na(x)))
training_set <- training_set[, na_var == FALSE]
testing_set  <- testing_set [, na_var == FALSE]
```

We will train a random forest model with repeated cross validation. We will use 3 fold CV and repeat 2 times (in the interest of time). The Caret package gives us the traincontrol function, and we use this with <"repeatedcv">. Since my PC is slow, I randomly sample 2000 rows which hopefully include 1 of each classe.

```
train_control <- trainControl(method = "repeatedcv", number = 3, repeats = 2)

subsample <- training_set[sample(nrow(training_set), 2000), ]

fit <- train(classe ~ . , data = subsample, method = "rf", trControl = train_control)
```

And predicting to new data with confusion matrix

```
predictions <- predict(fit, new_data = testing_set)
confusion_matrix <- confusionMatrix(table(predictions, subsample$classe))
confusion_matrix
```

```
## Confusion Matrix and Statistics
##
##
## predictions   A   B   C   D   E
##           A 537   0   0   0   0
##           B   0 409   0   0   0
##           C   0   0 339   0   0
##           D   0   0   0 330   0
##           E   0   0   0   0 385
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.9982, 1)
##     No Information Rate : 0.2685
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
```

```
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            1.0000   1.0000   1.0000    1.000   1.0000
## Specificity            1.0000   1.0000   1.0000    1.000   1.0000
## Pos Pred Value         1.0000   1.0000   1.0000    1.000   1.0000
## Neg Pred Value         1.0000   1.0000   1.0000    1.000   1.0000
## Prevalence             0.2685   0.2045   0.1695    0.165   0.1925
## Detection Rate         0.2685   0.2045   0.1695    0.165   0.1925
## Detection Prevalence   0.2685   0.2045   0.1695    0.165   0.1925
## Balanced Accuracy      1.0000   1.0000   1.0000    1.000   1.0000
```

The confusion matrix indicates a very high degree of accuracy of 100% in all cases. I anticipate this will generalize outside the CV set as well.