Tutorial: 'cv.rf' function to tune the parameters of random forest

Alan Vazquez

2022-05-19

This document provides a tutorial about the function 'cv.rf'. The function calculates the cross-validation accuracy of random forest under different tuning parameter settings. In the next sections, we show the function's requirements, how to load the function and its correct usage.

Requirements

The function 'cv.rf' is available in the file "CrossValidation_RF.R". To use this function, you must have an R version 4.1.0 or above, and the library called 'randomForest'. To check the R version, run the command below.

```
R.version
```

```
##                _
## platform       x86_64-apple-darwin17.0
## arch           x86_64
## os             darwin17.0
## system         x86_64, darwin17.0
## status
## major          4
## minor          2.0
## year           2022
## month          04
## day            22
## svn rev        82229
## language       R
## version.string R version 4.2.0 (2022-04-22)
## nickname       Vigorous Calisthenics
```

The row 'version.string' shows the R version available on your computer. If you have an older version, you can upgrade by re-installing R. To this end, go to https://www.r-project.org/.

To install the library called 'randomForest', run the command below without the # symbol.

```
#install.packages("randomForest")
```

Set working directory

To start the project, we define a working directory which contains the required data sets and files. For example, to set the folder "/Users/STATS101B/Final_Project" as the working directory, run the command below without the # symbol.

```
#setwd("/Users/STATS101B/Final_Project")
```

Alternatively, you can choose the working directory in R Studio. To this end, go to the menu called "Session" and then "Set working directory". Next, select "Choose Directory..." and chose the folder "/Users/STATS101B/Final_Project".

Load the function

To load the function, we first put the file "CrossValidation_RF.R" on the working directory. Next, run the command below.

```
source("CrossValidation_RF.R")
```

```
## randomForest 4.7-1
```

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

The command puts the function 'cv.rf' on your current R environment.

The main function

The function 'cv.rf' has three arguments: `design`, `y` and `X`. The argument `design` is the experimental design that will be used to collect the cross-validation accuracy values. The arguments `y` and `X` are the responses and predictor matrix, respectively, of the auxiliary data set used to build a random forest. The following sections show the correct format of these arguments and how to use the main function.

Experimental design input

The argument `design` specifies the experimental design used to collect the data. It must be a data frame with seven columns, one for each tuning parameter of random forest. The design must contain at least two rows. **In the design, the factor levels must be in their actual units, not in their coded levels.** The columns in the data frame must also have an order. More specifically, the order of the columns must be "ntree", "mtry", "replace", "nodesize", "classwt", "cutoff" and "maxnodes". The details behind these tuning parameters are in the file "Final_Project_Spring_2022.pdf".

An example of a correct data frame for the argument `design` of the function 'cv.rf' is shown below.

```
print(design, row.names = FALSE)
```

```
##  ntree mtry replace nodesize classwt cutoff maxnodes
##    100    4       1      100     0.9    0.2        5
##    100    4       1      100     0.5    0.2     1000
##    100    4       0        5     0.9    0.8        5
##    500    2       1      100     0.5    0.2        5
##    100    4       0        5     0.5    0.2     1000
##    500    2       0      100     0.5    0.8        5
##    500    2       0        5     0.9    0.8        5
##    500    2       1        5     0.9    0.8     1000
##    100    4       0      100     0.5    0.8        5
##    100    2       1        5     0.5    0.8     1000
```

Auxiliary data sets

The arguments `y` and `X` are specific to the auxiliary data set to train a random forest. The input `y` is the response and `X` is the predictor matrix. These two inputs will be given to you in an object with extension ".RData". There are three possible objects, each of which containing a different data set. The objects are "diabetes.RData", "heart.RData", and "cardiovascular.RData". The assignment of each data set to a team is

available on the Google spreadsheet you use to register your team. A brief description of the auxiliary data sets is included in the document called "Data_Description_Spring_2022.pdf".

The corresponding data set must be located on the working directory. To load a data set, say "diabetes.RData", run the command below.

```
load("diabetes.RData")
```

The command above puts the objects 'y' and 'X' for the "diabetes" data set on the R environment.

Using the function

Once you define all three inputs, you run the function 'cv.rf' as follows:

```
results <- cv.rf(design, y, X)
```

```
## Collecting response on test combination  1
## Collecting response on test combination  2
## Collecting response on test combination  3
## Collecting response on test combination  4
## Collecting response on test combination  5
## Collecting response on test combination  6
## Collecting response on test combination  7
## Collecting response on test combination  8
## Collecting response on test combination  9
## Collecting response on test combination  10
```

```
print(results)
```

```
##    ntree mtry replace nodesize classwt cutoff maxnodes        CV
## 1    100    4       1      100     0.9    0.2        5 0.5000000
## 2    100    4       1      100     0.5    0.2     1000 0.7006933
## 3    100    4       0        5     0.9    0.8        5 0.5000000
## 4    500    2       1      100     0.5    0.2        5 0.6634271
## 5    100    4       0        5     0.5    0.2     1000 0.6306187
## 6    500    2       0      100     0.5    0.8        5 0.6449328
## 7    500    2       0        5     0.9    0.8        5 0.5000000
## 8    500    2       1        5     0.9    0.8     1000 0.7011687
## 9    100    4       0      100     0.5    0.8        5 0.6690356
## 10   100    2       1        5     0.5    0.8     1000 0.6719019
```

The output of the function (`results`) is a data frame containing the design and cross-validation accuracy values. Recall that the response of interest is the cross-validation accuracy of random forest. **Therefore, the data frame given by the function 'cv.rf' is the one you and your team will analyze.**