Supervised Machine Learning: Classification and Regression in R

Hannah Rose Kirk

07/07/2020

Pre-Tutorial Instructions

These are the pre-workshop instructions for "Supervised Machine Learning: Classification and Regression in R", a virtual tutorial co-hosted by Ladies Who Tech and R-Ladies Beijing.

Your choice:

- Sit back, listen and enjoy learning about networks
- Download all the materials and follow live in your own RStudio console/script

If you choose the second option, you may wish to complete these steps before the tutorial starts.

What to Expect

Overview:

- *What is Supervised Machine Learning?*
- *When can we use it?*
- *What approaches can we take?*

Part 1:

- A foundational introduction to importing, cleaning and visualizing data, and how to implement one machine learning model to make predictions
- **datatset**: predicting malign/benign breast cancer tumors

Part 2:

- A more in-depth overview into how we choose and fine-tune different machine learning models.
- **dataset**: predicting the occurance of diabetes and the diabetes risk from pregnancies

Step 1: Downloading RStudio

R and RStudio are free, open-sourced software which are avaliable for most operating systems. Regardless of your operating system, you need to download R before installing RStudio.

R can be dowloaded from "http://lib.stat.cmu.edu/R/CRAN/".

Then download RStudio Desktop from "https://rstudio.com/products/rstudio/download/".

Follow the installation instructions with the default installation options.

Now you can open RStudio from your computer and start coding!

Step 2: Downloading the Materials

If you want to follow the tutorial in live time then you will need to download 2 datasets from my website. Go to https://www.hannahrosekirk.com/research/ and under 'Coding Tutorials' download the zip file with the materials.

- "dataset1_cancer.csv"
- "dataset2_diabetes.csv"

Place these datasets in a folder on your desktop, in your documents, wherever you like, but remember where the folder is and copy its location path.

Step 3: Setting the working directory

We need to tell R where to retrieve files and information from whilst running our script on networks. We want our workspace to be the folder we have placed the downloaded materials in. R will also save our output files to this location.

We can use the *setwd()* R function, and copy and paste the location path of our folder:

```
setwd("paste file path here")
```

You can also do this manually in R Studio with clicking: 'Session > Set Working Directory > Choose Directory' then just click on the folder with the materials.

Step 4: Installing packages

An R package is a pre-made collection of functions (and sometimes datasets) which has been developed by the R community. Packages increase the functionality of base R and make our job easier when we want to do more advanced analysis and visualiation.

We need to install these packages into R before we can work with their contents. We are using CRAN as our repository. A repository is a place where packages are located so we can access them and install them. CRAN is the official repository maintained by the R community around the world but you could also download packages from GitHub for example.

```
install.packages("dplyr", repos = "http://cran.us.r-project.org")
install.packages("magrittr", repos = "http://cran.us.r-project.org")
install.packages("tidyverse", repos = "http://cran.us.r-project.org")
install.packages("corrplot", repos = "http://cran.us.r-project.org")
install.packages("caret", repos = "http://cran.us.r-project.org")
install.packages("class", repos = "http://cran.us.r-project.org")
install.packages("randomForest", repos = "http://cran.us.r-project.org")
install.packages("rpart", repos = "http://cran.us.r-project.org")
install.packages("rpart.plot", repos = "http://cran.us.r-project.org")
```

You can always get help with a package with the command:

```
packageDescription("package name")
help(package = "package name")
```

You are ready to go!