ADVANCED PROGRAMMING
MASTER IN STATISTICS FOR DATA SCIENCE. 2018-19
RCPP ASSIGNMENT: PROGRAMMING NEAREST NEIGHBOUR IN C++

1.5 POINTS

Introduction

The aim of this assignment is to program in C++ a simple but useful machine learning method: KNN or k-nearest neighbor. I have already programmed KNN with K=1 in R, which you will find in Aula Global.

The aim of the assignment is:

1. **(0.50 points)** To translate my R code into C++. I will value if you are able to use at least one vectorized sugar function
2. **(0.50 points)** To modify the R code so that it works with **K=2** neighbors
3. **(0.50 points)** To translate the R code of step 2 into C++
4. **(extra +0.25 points)** If you modify the C++ code to work with "inverse euclidean distance"

In all cases:

- Compile the C++ code with *sourceCpp*
- Use the library microbenchmark in order to determine whether the C++ version is faster than:
    - the R version of the code,
    - and the FNN knn.reg that belongs to the CRAN library *FNN* (*library(FNN)*)
- Note: you will probably need to use:
    - && ("and" in R)
    - & ("and" in C++)
    - != ("inequality" in R and C++)

What to hand in:

1. Write a very short report (1 page) about the code you wrote and the results of the microbenchmark and whether you get the same result with all methods (R, C++, and *FNN::knn.reg*).
2. Hand in your code and report in Aula Global. It should be a zip file containing:
    a. The report
    b. A file with code for step 1
    c. A file with code for step 2
    d. A file with code for step 3