Lab 3 - Stat 215A, Fall 2020

Due: October 26, 11:59 PM

As usual, push a folder called "lab3" to your `stat-215-a` GitHub repository by the deadline. At a minimum, your "lab3" folder should contain the files below:

- **lab3.Rmd**: the raw report + code with your name
- **lab3.pdf**: the output of lab3.Rmd. This output should not contain any code.
- **R**/: a folder containing R scripts that will be sourced in lab3.Rmd.

You may create additional folders if you wish, but please keep your project directory organized, and do not push your **data**/ folder or other files that are not necessary to reproduce your report. Note also that you do not have to create blinded files.

1 Parallelizing k-means

We will be investigating the stability of k-means using a popular procedure outlined in Ben-Hur et al. [2001], which uses stability as a guide for picking k. The procedure is outlined in Algorithm 1. You should consult the paper for more details, particularly regarding the similarity measures you can use.

Algorithm 1 Calculation of clustering similarities in k-means

```
for $k = 2$ to $k_{max}$ do
    for $i = 1$ to $N$ do
        $\text{sub}_1 = \text{subsample}(X, m)$, a subsample of fraction $m$ of dataset $X$
        $\text{sub}_2 = \text{subsample}(X, m)$, a subsample of fraction $m$ of dataset $X$
        $L_1 = \text{cluster}(\text{sub}_1)$
        $L_2 = \text{cluster}(\text{sub}_2)$
        $\text{intersect} = \text{sub}_1 \cap \text{sub}_2$
        $S(i, k) = \text{similarity}(L_1(\text{intersect}), L_2(\text{intersect}))$
    end for
end for
```

In this lab, you will be implementing this method on the binary-coded linguistic data from Lab 2. Please use the .Rdata file included in this lab to ensure consistency. Set the maximum number of clusters to consider: $k_{max} = 10$; the number of repeated iterations: $N = 100$; and the sampling proportion, m as large as you can get it while also running in a reasonable time (m should be no less than 0.2, no more than 0.8). This will require a decent amount of computation, so you should try to run this in parallel on the SCF cluster.

1. To compute the similarity of clusterings, you will (in theory) be dealing with a $q \times q$ matrix, where q is the the number of data points that are common to each subsample (if $m = 0.8$, q will be approximately 29,000, meaning that the similarity matrix, C, will be a 6GB matrix). This could be prohibitively large. For the following questions, you can use any similarity measure mentioned in the paper - correlation, Jaccard, matching.

    (a) Write an R function called `similarity()` to calculate the similarity between two membership vectors in R that will actually complete in reasonable time for inputs up to size 5000. Write this function in a file called `similarity.R` in your `R/` folder.

    (b) Write a memory-efficient version of the similarity function to calculate similarity in Rcpp. Name this function `similarityRcpp()` in a file called `similarity.cpp` in your `R/` folder. Can you avoid storing the $q \times q$ matrix? If so, how do you avoid it?

    (c) Discuss and compare the timing of your R and Rcpp implementations of the similarity function when applied to the lingBinary.Rdata for various choices of m.

2. Implement Algorithm 1, using your Rcpp implementation of the similarity function to compute the similarity. Also, to speed up computation, parallelize the outer for loop of this method using `foreach`, and run your job with the lingBinary.Rdata on the SCF cluster using `sbatch`. Please save the resulting S matrix (or data.frame) as a *.csv* file in a folder named `results/`.
   See the following reference for detailed instruction on using the cluster: `http://statistics.berkeley.edu/computing/servers/cluster`. If you do not have an SCF account, you will need to request one, by going to `https://scf.berkeley.edu/account`.

3. Make a plot similar to Figure 3 in Ben-Hur et al. [2001]. What would you pick as k for this dataset and why? Do you trust this method?

In this lab, you will graded more carefully than usual on your code. As always, please follow the Google R style guide. Make sure your variable names are meaningful, write comments liberally, and document your functions. Please also provide all scripts necessary to reproduce your results. This includes any .R or .sh scripts used to run your code on the SCF.

1.1 Reproducibility

I will use the scf cluster to test the reproducibility of your code, by cloning into your repository and attempting to compile your reports. For your convenience, I provide you with the script I will be using to test your code. If you wish, you can use the script to test your code in the following way:

1. Go into your lab2 directory, for example using my directory:

   ```
   cd /Users/omerronen/Documents/Phd/215a2021/stat-215-a-gsi/lab3
   ```

2. Run

   ```
   bash test.sh .
   ```

To perform testing on scf, you may need to transfer the data file there which can be done using *SCP* (see detailed instructions on this https://statistics.berkeley.edu/computing/copying-fileslink).

Lastly if you wish to use any R packages that require installation, please ask for approval here - https://docs.google.com/spread
I will try to install it on scf, and approve if possible.

References

Asa Ben-Hur, André Elisseeff, and Isabelle Guyon. A stability based method for discovering structure in clustered data. In *Pacific symposium on biocomputing*, volume 7, pages 6–17, 2001.