Final Report on Merge Automation

Joshua R. Harkness

BA Information Science

Minor Computer Science

A.A.S. Criminal Justice

**General Overview**

The purpose of the R script written was to automate the merging process of clinical data to the already combined gene expressions, mutations, copy number variations, and methylation.

The datasets were merged based on the patient’s bcr (breakpoint cluster region protein) barcode. Editions had to be made to the clinical dataset for the barcodes to match exactly, due to a difference between hyphens and periods, the inclusion of the word “data”, and capitalization. Another edition made to clinical was to add .01 at the end of the barcode, because each barcode at least contained .01 (symbol for cancerous sample) at the end in the other dataset’s barcode. The other endings, .10 and .13, stood for normal samples and metastasized samples respectfully. This ensured that all the proper barcodes were given the additional variables from clinical, without any repeated values.

An outer join was used to ensure no variables were lost while still merging identical barcodes together. The outer join was used through R’s merge function and setting the “all” parameter to true. By merging the clinical data with the others mentioned above, an additional 3719 variables were introduced.

**Understanding the Code**

Comments were used throughout the code the help clarify each block of code’s purpose. The following table can also be used as a guide to understanding the R script used.

|  |  |  |
| --- | --- | --- |
| **Line Number(s)** | **Code** | **Description** |
| 5 - 7 | memory.limit(60000)  library(tidyverse)  library(data.table) | Sets the memory limit passed the required and loads the required packages to run the code. |
| 10, 12 | merged\_data <- read.csv("data/BRCAMerged-NoClin.csv") | Sets the already merged data without clinical to the variable named merged\_data. Line 12 provides the size of this dataset. |
| 15, 17 | clinical\_data <- read.csv("data/BRCA.clin.merged.csv") | Sets the clinical data to the variable named clinical\_data. Line 17 provides the size of this dataset. |
| 20, 21 | clinical\_data <- as.data.frame(clinical\_data)  merged\_data <- as.data.frame(merged\_data) | Sets the data as a data frame for safer manipulation. This will be addressed in the Possible Future Changes section. |
| 24, 25 | merged\_data <- t(merged\_data)  clinical\_data <- t(clinical\_data) | Transposes the dataframes for easier merging (adding columns instead of rows). |
| 28 - 36 | clinical\_data <- cbind(Col.Names = row.names(clinical\_data), clinical\_data)  colnames(clinical\_data) <- clinical\_data[1, ]  clinical\_data <- clinical\_data[-1,]  rownames(clinical\_data) <- NULL  merged\_data <- cbind(Col.Names = row.names(merged\_data), merged\_data)  colnames(merged\_data) <- merged\_data[1, ]  merged\_data <- merged\_data[-1,]  rownames(merged\_data) <- NULL | Sets the column names and changes the row names into an actual column, instead of being solely a heading. This is done to both dataframes. |
| 41 - 47 | clinical\_data[, "patient.bcr\_patient\_barcode"] <- gsub("-", ".", clinical\_data[, "patient.bcr\_patient\_barcode"], fixed = TRUE) %>%  toupper()  clinical\_data[, "patient.bcr\_patient\_barcode"] <- paste0("Data.", clinical\_data[, "patient.bcr\_patient\_barcode"])  clinical\_data[, "patient.bcr\_patient\_barcode"] <- paste0(clinical\_data[, "patient.bcr\_patient\_barcode"], ".01")  colnames(clinical\_data)[colnames(clinical\_data) == "patient.bcr\_patient\_barcode"] <- "Des.GeneSymbol" | Manipulates the barcode in clinical\_data to match the format of the barcode in merged\_data.  clinical\_data’s barcode row name is then changed to match the name where the barcodes are in merged\_data. |
| 50, 51 | all\_merged <- merge(merged\_data, clinical\_data, all = TRUE)  all\_merged <- t(all\_merged) | Merges the dataframes together, based on patient barcode then transposes the data back to the original format. The new dataframe is set to the variable all\_merged. |
| 54 - 57 | all\_merged <- cbind(all\_merged, Row.Names = rownames(all\_merged))  colnames(all\_merged) <- all\_merged[1, ]  all\_merged <- all\_merged[-1,]  rownames(all\_merged) <- NULL | Moves row names from just a header to an actual column. |
| 60 - 63 | all\_merged <- subset(all\_merged, select = c(Des.Description, Data.TCGA.3C.AAAU.01:Des.GeneSymbol))  all\_merged <- subset(all\_merged, select = c(Des.Platform, Des.Description:Des.GeneSymbol))  all\_merged <- subset(all\_merged, select=c(Des.GeneSymbol, Des.Platform:Des.GeneSymbol))  all\_merged <- all\_merged[, !duplicated(colnames(all\_merged))] | Move the row names to their suggested positions; Des.GeneSymbol first, Des.Platform second, and Des. Description third. After these three, the patient barcodes follow. |
| 66 | rownames(all\_merged) <- c(1:nrow(all\_merged)) | Creates row numbers as the row names |
| 69 - 71 | all\_merged[, "Des.Platform"][is.na(all\_merged[, 'Des.Platform'])] <- "Clinical"  remove\_gene\_sym <- which(colnames(clinical\_data) == "Des.GeneSymbol")  all\_merged[, "Des.Description"][is.na(all\_merged[, "Des.Description"])] <- colnames(clinical\_data)[-remove\_gene\_sym] | Replaces NA values with proper descriptions. |
| 74, 75 | all\_merged <- as.data.frame(all\_merged)  all\_merged <- setDT(all\_merged)[all\_merged, on = c("Des.GeneSymbol==Des.Description"), Des.GeneSymbol := "-"][] | Removes duplicated values from Des.Description in Des.GeneSymbol and replaces them with “-“. This is done through the use of datatables. |
| 78 | all\_merged <- all\_merged[order(all\_merged[, "Des.Platform"]), ] | Sorts Des.Platform |
| 80 | write.csv(all\_merged, file = "data/BRCAMergeAutomated.csv") | Creates a csv file for the all\_merged datatable. |

**User Manual**

To use the R script properly, all that is required from the user is to place the combined data of gene expressions, mutations, copy number variations, and methylation into line 10. The user enters the file path between the quotes in read.csv. Be sure the file is a csv before using this function. If the file is not currently a csv, save it as one.

The next step required by the user is to use the same instructions as mentioned above except this time, the user will place the file path for the clinical data in line 15, between the quotes in read.csv.

The final step involves the last line of code, line 80. The user must write the file path they wish to save the newly merged data to between the quotes in write.csv.

Also, if not done previously, the user must install the packages tidyverse and data.table.

After these steps are finished, the user simply needs to press run for each line of code, until the code is complete. A warning message will appear when the user attempts the merge on line 50, but it is not an issue. This is a timely process.

**Possible Future Changes**

Currently, running the entire R script takes a substantial amount of time. Throughout learning the programming language, I discovered changing the dataframe to a datatable allowed for significantly faster processing. Unfortunately, I did not have enough time to go back and make the changes necessary for the conversion of a dataframe to a datatable. These changes could significantly reduce the amount of time required to run the entire script.

Another minor change that could be added is placing the entire script within one function, this way the user would only have to enter the parameters on one line, and press run once to run the entire script.

To address a goal that I was given for the grant of “Add to the script a function extracting the data for a set of genes provided as input parameter”, it seems this already exists within R. If I understand what is being asked correctly, this can be done through R without creating an additional function. When seeking all data for one set of genes, the user should be able to enter the genes wanted as the first parameter of the all\_merged variable, and the place a comma with nothing after to receive all the data for that set of genes. One example would be all\_merge[A1BG, ]. This should provide all the data regarding this gene symbol, but it has not been tested.