Data Wrangling

Group 20

2023-12-08

Data Wrangling

```r
# reading in the data
liver_data <- read.csv(file = "RNAseq_LIHC.csv")
mut_data <- read.table(file = "data_mutations.txt", header = TRUE, sep = "\t")
patient_data <- read.table(file = "data_clinical_patient.txt", header = TRUE, sep = "\t")

# the patient IDs are 12 characters long
unique_patient_data <- length(unique(patient_data$PATIENT_ID))

# number of unique patient IDs in patient data
unique_patient_data
```

```
## [1] 372
```

```r
# using gsub to extract the first 12 characters of the ID columns
unique_mutation_data <- length(unique(gsub("^(.{12}).*$", "\\1", mut_data$Tumor_Sample_Barcode)))

# number of unique patient IDS in mutation data
unique_mutation_data
```

```
## [1] 357
```

```r
unique_RNAseq_data <- length(unique(gsub("^(.{12}).*$", "\\1", colnames(liver_data)[-1])))

# number of unique patient IDs in RNAseq data
unique_RNAseq_data
```

```
## [1] 371
```

```r
colnames(liver_data) <- gsub("\\.", "-", colnames(liver_data))

patient_ids <- patient_data$PATIENT_ID

mut_ids <- unique(gsub("^(.{12}).*$", "\\1", mut_data$Tumor_Sample_Barcode))

RNAseq_ids <- unique(gsub("^(.{12}).*$", "\\1", colnames(liver_data)[-1]))
```

```r
all_patients <- length(intersect(RNAseq_ids, intersect(patient_ids, mut_ids)))

# number of patients shared between all three datasets
all_patients
```

```
## [1] 352
```

```r
# the IDs of patients shared between all three datasets
all_patients_ids <- intersect(RNAseq_ids, intersect(patient_ids, mut_ids))
```

Shared Patient Dataframes

```r
# making new dataframes for each set but only with shared patients

# patient data
index_vec <- c()

for (i in 1:length(patient_data$PATIENT_ID)) {
    for (j in 1:length(all_patients_ids)) {
        if (patient_data$PATIENT_ID[i] == all_patients_ids[j]) {
            index_vec <- c(index_vec, i)
        }
    }
}

patient_data_new <- patient_data[index_vec, ]
write.csv(patient_data_new, file = "patient_data_shared.csv")

# mutation data
index_vec2 <- c()
mut_patients <- gsub("^(.{12}).*$", "\\1", mut_data$Tumor_Sample_Barcode)

for (i in 1:length(mut_patients)) {
    for (j in 1:length(all_patients_ids)) {
        if (mut_patients[i] == all_patients_ids[j]) {
            index_vec2 <- c(index_vec2, i)
        }
    }
}

mut_data_new <- mut_data[index_vec2, ]
write.csv(mut_data_new, file = "mutation_data_shared.csv")

# rnaseq data
index_vec3 <- c()
rnaseq_patients <- gsub("^(.{12}).*$", "\\1", colnames(liver_data)[-1])

for (i in 1:length(rnaseq_patients)) {
    for (j in 1:length(all_patients_ids)) {
        if (rnaseq_patients[i] == all_patients_ids[j]) {
            index_vec3 <- c(index_vec3, i)
        }
    }
}
# corrected for gene name column
index_vec3 <- index_vec3 + 1
index_vec3 <- c(1, index_vec3)

liver_data_new <- as.data.frame((liver_data[, index_vec3]))  #get rid of transpose?

liver_gene_names <- liver_data_new[, 1]

liver_data_new <- as.data.frame(sapply(liver_data_new, as.numeric))
```

```
## Warning in lapply(X = X, FUN = FUN, ...): NAs introduced by coercion
```

```
rownames(liver_data_new) <- liver_gene_names

liver_data_new <- liver_data_new[, -1]

liver_data_cst_removed <- as.data.frame(t(liver_data_new))

# some patients were duplicated in the RNAseq matrix, therefore, these
# instances were removed until each patient left in the matrix was unique
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.1
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
set.seed(1)

liver_data_cst_removed <- liver_data_cst_removed[!duplicated(gsub("^(.{12}).*$",
    "\\1", rownames(liver_data_cst_removed))), ]  ##

# change the patient names to be just the patient IDs
rownames(liver_data_cst_removed) <- gsub("^(.{12}).*$", "\\1", rownames(liver_data_cst_removed))

liver_data_cst_removed <- liver_data_cst_removed[, colSums(liver_data_cst_removed) >
    1]  #remove columns that add up to 0 or 1

write.csv(liver_data_cst_removed, file = "rnaseq_data_shared.csv")
```