RAC1 survival analysis

Katia Otterstedt

2023-10-04

Contents

Introduction

This script is designed for the execution of survival analysis utilizing TCGA clinical and RNA-Seq data. In this instance, we employ RNA-Seq and survival COAD (Colon Adenocarcinoma) data to conduct survival analysis.

Libraries

This analysis requires 'survminer' package for cutpoint determination based on survival prognosis and 'survival' package to prepare de fit.

```
library(tidyverse) # pipes (%>%) and dplyr data munging
library(RTCGA.clinical) # survival times
library(RTCGA.rnaseq) # genes expression
library(survminer)
library(survival)
```

Retrieving survival data

The initial phase of our analysis involves retrieving survival data pertaining to participants within our study cohort. This dataset encompasses the days until the last follow-up, denoting the absence (0) or presence (1) of our specific event of interest, which, in the context of this analysis, is mortality.

```
COAD.surv <- survivalTCGA(COAD.clinical)
```

Extracting expression regarding gene of interest

In light of our intention to stratify patients into two groups based on RAC1 expression, our initial step involves extracting this expression data from the TCGA repository. We begin by reformatting the extracted dataset for enhanced clarity and integrate this information with the previously obtained survival data.

```
#Look for gene ID and extract expression
RAC1_expression <- expressionsTCGA(COAD.rnaseq,extract.cols = "RAC1|5879")
#Rename variables and cut sample names
RAC1_expression <- RAC1_expression %>% rename(cohort = dataset,
                                              RAC1 = `RAC1|5879`)%>%
  mutate(bcr_patient_barcode = substr(bcr_patient_barcode, 1, 12))

#Merge with survival data
COAD.surv.Rac1 <- COAD.surv %>%
  left_join(RAC1_expression,
            by = "bcr_patient_barcode")

COAD.surv.Rac1 <- COAD.surv.Rac1 %>%
  filter(!is.na(RAC1))
```

Group stratification

Cutpoint determination

Employing survival information and expression data, we invoke the 'surv_cutpoint' function to discern a precise cutpoint. The purpose of this cutpoint is to categorize patients according to their RAC1 expression levels. We generate the variable "gene of interest" so that the script can be modified and adapted to different analysis.

```
COAD.cut <- surv_cutpoint(
  COAD.surv.Rac1,
  time = "times",
  event = "patient.vital_status",
  variables = "RAC1")

# Getting the value of cutpoint in CPM and statistic
summary(COAD.cut)
```

```
##      cutpoint statistic
## RAC1  10123.4  2.162516
```

```r
# Checking some statistics related to out gene of interest
mean(COAD.surv.Rac1$RAC1, na.rm = TRUE)
```

```
## [1] 7517.785
```

```r
sd(COAD.surv.Rac1$RAC1, na.rm = TRUE)
```

```
## [1] 1985.764
```

```r
median(COAD.surv.Rac1$RAC1, na.rm = TRUE)
```

```
## [1] 7157.457
```

Patient stratification

Subsequently, we classify our patients based on the previously established cutpoint.

```r
COAD.cat <- surv_categorize(COAD.cut)
COAD.cat <- cbind(COAD.surv.Rac1$bcr_patient_barcode, COAD.cat)
colnames(COAD.cat)[1] <- "bcr_patient_barcode"

#Refactoring so that our control "low" goes first in the plot
COAD.cat <- COAD.cat %>%
  mutate(across(all_of("RAC1"), ~ factor(., levels = c("low", "high"))))

#Adjusting with survfit
fit <- surv_fit(Surv(times, patient.vital_status) ~ RAC1, data = COAD.cat)

#Obtaining pvalue for each fit
surv_pvalue(fit)
```

```
##   variable       pval   method pval.txt
## 1     RAC1 0.06993956 Log-rank p = 0.07
```

Plot generation

Finally, we generate a Kaplan Meier plot to illustrate survival patterns across the two patient cohorts.

```r
#Creating the plots
ggsurvplot(fit=fit,
                    data=COAD.cat,
                    risk.table = TRUE,
                    pval = TRUE,
                    pval.size = 3,
                    conf.int = TRUE,
                    title = "RAC1 Stratification",
                    xlab = "Time (Years)",
                    xlim = c(0,1825),
```

```
                ylab = "Survival probability",
                xscale = 365,
                break.time.by = 365,
                ggtheme = theme_bw(),
                palette = c("#006FAB", "#DA3926"),
                risk.table.col = "strata",
                risk.table.y.text = FALSE,
                risk.table.fontsize = 3,
                tables.theme = theme_bw(),
                risk.table.y.text.col = T,
                font.legend=5)
```