|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | Directory | Description | Source |
| lung\_gene | lung\_gene<- read.table("~/DLSPH/CHL5208/data/lung\_gene.txt",  header = TRUE, sep = "\t"  ) %>% data.frame() | Gene expression dataset | Original from TCGA |
| lung\_phe | lung\_phe <- read.table("data/phenotype.txt",  header = TRUE, sep = "\t"  ) %>%  data.frame() | Phenotype data | Original from TCGA |
| clinical\_dat | clinical\_dat <- read\_delim("data/lusc/clinical.tsv",  delim = "\t", escape\_double = FALSE,  trim\_ws = TRUE  ) | Demographic info | Original |
| Work\_dat1 | work\_dat1 <- inner\_join(clinical\_dat1, phe\_gen\_dat,  by = c("case\_submitter\_id" = "X\_PATIENT")  ) %>%  filter(!is.na(OS.time)) %>%  filter(!(OS == 1 & OS.time == 0)) %>%  filter(!(OS == 0 & OS.time == 0)) | All data  996 samples,  Used **for Univariate data analysis** |  |
| **Uni\_dat** | uni\_data <- work\_dat1 %>%  select(OS,OS.time,OS.time\_yr,  OS.time\_month,  project\_id,  age\_at\_diagnosis,  gender,race,  ajcc\_pathologic\_stage,cancer\_stage) | 一个work\_dat1的subset，感觉更轻便一点 |  |
| Work\_dat2 | work\_dat2 <- work\_dat1 %>%  filter(!is.na(age\_at\_diagnosis)&!is.na(cancer\_stage)) | 这里我筛选了age and cancer stage not na的，供958例，用于后面的real data analysis, 因为我们有adjusted for demo info |  |
| Gene\_dat | gene\_dat <- work\_dat2 %>%  select(-colnames(.)[!(colnames(.)%in% gene\_names)]) %>%  apply(.,2,as.numeric)  colnames(gene\_dat) <- rlang::set\_names(paste0('X\_',  make.names(colnames(gene\_dat))))# Assuming your independent variables start from the third column | 这里我把gene 数据选出来了 变为numeric的 |  |
| Outcome\_dat | outcome\_dat <- work\_dat2 %>% select(OS.time\_month,OS) %>%  rlang::set\_names("time","status") | Outcome 数据为一个数据集 |  |
| Cov\_dat | cov\_dat <- work\_dat2 %>%  select(age\_at\_diagnosis,cancer\_stage\_cat2) %>%  rlang::set\_names('age\_at\_diagnosis','cancer\_stage') | Covariate作为一个数据集 |  |
| **Cox\_dat** | cox\_dat <- cbind(outcome\_dat,cov\_dat, gene\_dat) | 所有需要的列放在一起，后面的分析就是基于这个数据集的 |  |

Then, 基于cox\_dat，adjusting for age and cancer stage, univariate data analysis was done one each gene, and the output was stored in **uni\_cox\_res2.rdata,**

**save(uni\_cox\_res2,file = 'output/uni\_cox\_res2.rdata')**

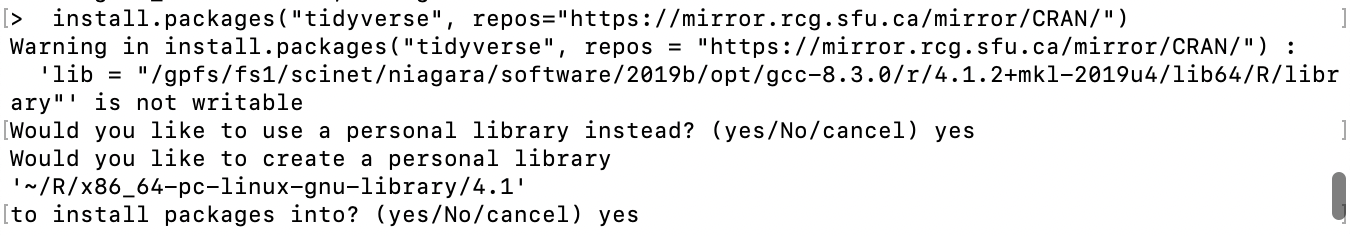
further, the top 200 significant snps were selected for the following analysis

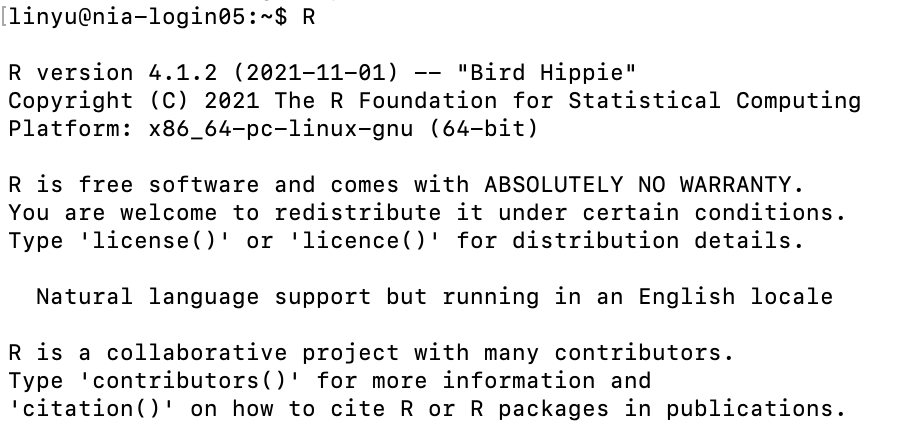
|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| Uni\_cox\_res2 | save(uni\_cox\_res2,file = 'output/uni\_cox\_res2.rdata') |  |  |
| Uni\_cox\_top200 | uni\_cox\_top200 <- uni\_cox\_res2%>%  arrange(BH) %>%  filter(BH <= BH[200])  save(uni\_cox\_top200,file="**output**/uni\_cox\_top200.rdata") | Top 200 univariate data analysis output |  |
| Reg\_dat | reg\_dat <-cox\_dat %>%  select("time", "status", "age\_at\_diagnosis", "cancer\_stage",  (uni\_cox\_top200$var\_name)) | Subset of cox\_dat 只有**200个genes** |  |
| trainingSet  and  testSet | set.seed(20240123)  trainIndex <- createDataPartition(reg\_dat$status,  p = 0.8, list = FALSE)  trainingSet <- reg\_dat[trainIndex,]  testSet <- reg\_dat[-trainIndex,] |  |  |
| lung\_boot\_lst | lung\_boot\_lst <- boot\_f1(  df = trainingSet,  nboot = 100,  boot\_ft = 20,  gene\_index = 5,  cov\_vars = c('age\_at\_diagnosis','cancer\_stage'),  seed = 20231106  ) | 767 obs, 200 genes |  |

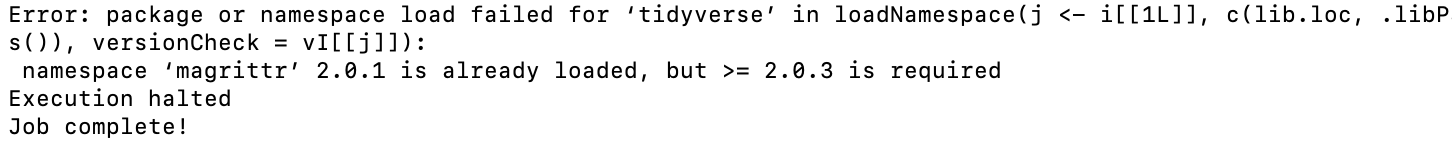
Simulation set up

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| Sim\_set1 | boot\_ft1 <- c(5:10)  boot\_ft2 <- c(5:15)  boot\_ft3 <- c(5:20)  sim\_set1 <- lapply(boot\_ft1,function(boot\_ft){  boot\_f (df = sim\_set[[1]],  nboot = 100,  boot\_ft = boot\_ft,  seed = 20231106)  }) | Feature=25时，取不同的boot\_ft (bootstrap set里面的feature数量) 超参数 |  |
| Sim\_set2 | sim\_set2 <- lapply(boot\_ft2,function(boot\_ft){  boot\_f (df = sim\_set[[2]],  nboot = 100,  boot\_ft = boot\_ft,  seed = 20231106)  }) | Feature = 50时，取不同的boot\_ft超参数 |  |
| Sim\_set3 | sim\_set3 <- lapply(boot\_ft3,function(boot\_ft){  boot\_f (df = sim\_set[[3]],  nboot = 100,  boot\_ft = boot\_ft,  seed = 20231106)  }) | Feature=100时，不同的bootstrap feature 超参数 |  |
| Reg\_dat50 | /04\_top200\_selection | Here I generated, which is a subset of reg\_dat with the top 50 significant genes , as a starting point |  |

A document of errors installing packages on CCDB:



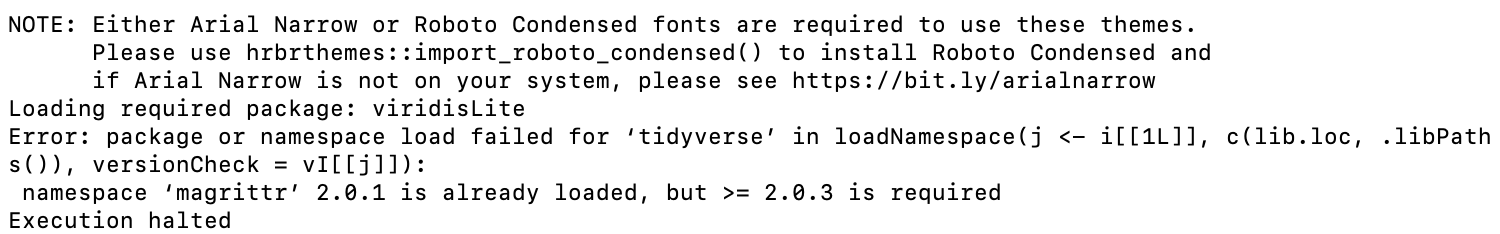




The downloaded source packages are in

'/tmp/RtmpLSGCjC/downloaded\_packages'

magrittr



# Unload the 'caret' package

library(tidyverse, unload = TRUE)

# Install or update the 'generics' package

install.packages("magrittr") # Replace with the correct version if needed

# Load the 'generics' package

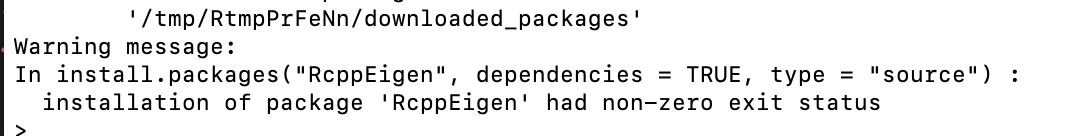
library(magrittr)

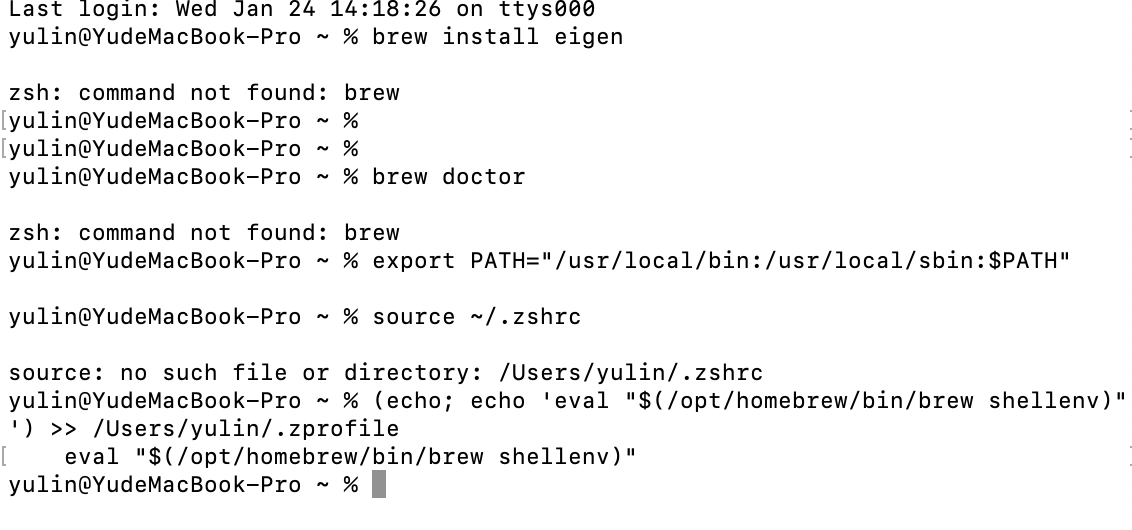
# Load the 'caret' package

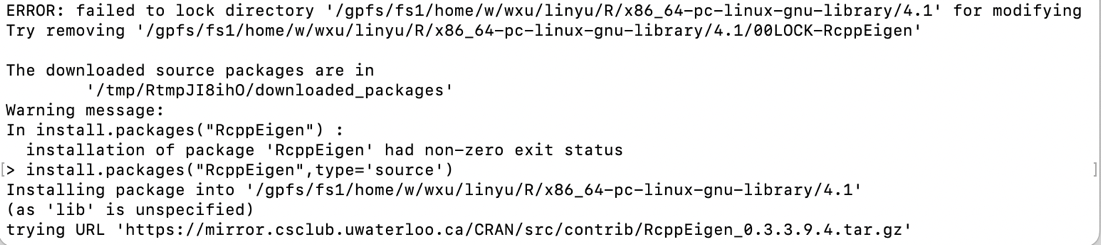
library(tidyverse)

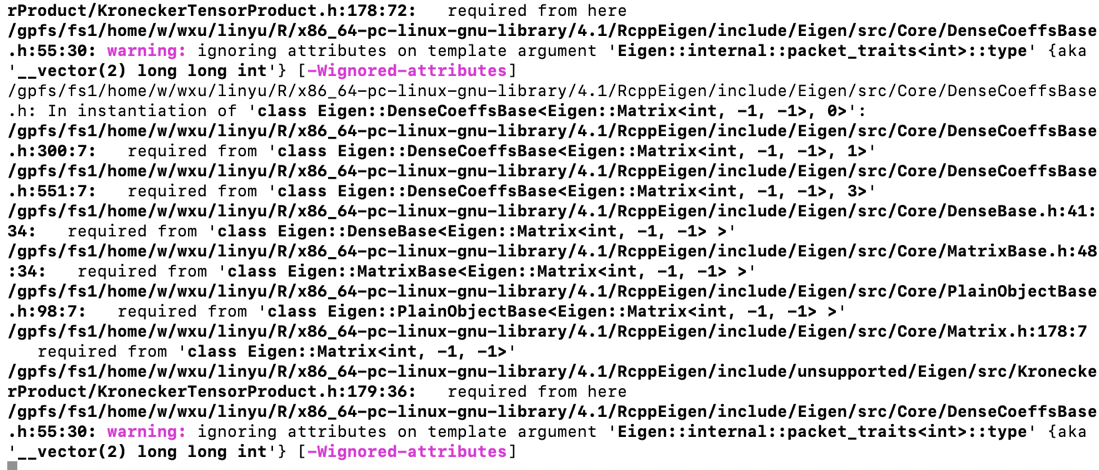
# Check the version of the 'generics' package

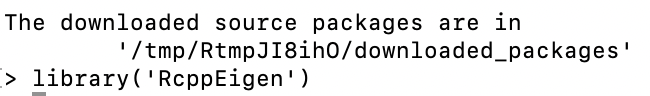
packageVersion("generics")

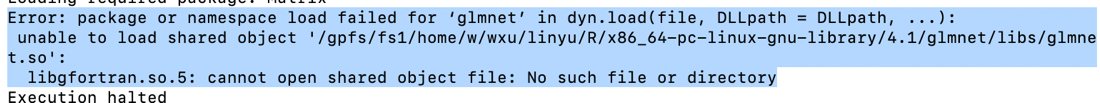












Loading required package: Matrix

Error: package or namespace load failed for ‘glmnet’ in dyn.load(file, DLLpath = DLLpath, ...):

unable to load shared object '/gpfs/fs1/home/w/wxu/linyu/R/x86\_64-pc-linux-gnu-library/4.1/glmnet/libs/glmnet.so':

libgfortran.so.5: cannot open shared object file: No such file or directory

Lmod is automatically replacing "gcc/8.3.0" with "intel/2019u4".

