

ECCA_project

Shuai

2022-07-14

```
library(tidyr, quietly = TRUE)
library(ggplot2, quietly = TRUE)
# Packages for normalization and DE analysis
# Packages for plotting
library(ggpubr)
library(ggfortify)
library(scales)
library(pheatmap)
library(patchwork)
library(RColorBrewer)
library(psych) # correlation plot
library(tidyverse)
library(hrbrthemes)
library(viridis)
library(rstatix)
# R part for clustering
library(rpart)
library('partykit')
library(survival)
library(rpart.plot)
# Survival analysis
library('survminer')
```

Prepare the input vector of joint/individual value #####

```
# Load the results from ECCA fit
load("eccca_result_without_main_effect.RData")
outECCA = result3

# Load and form survival response #####
pfi_data = read.csv("PanCan_ClinicalData_V4_20171009.csv")
load("prostate_timer_pit3.RData")
sampleIDs = substr(rownames(its), 1, 12)
```

```
# This is a list of the model without intercept, with
# U1 - joint from DeMixT, V1 - corresponding loadings
# U2 - joint from TIMER, V2 - corresponding loadings
# Z1, A1 - individual scores and loadings (scale-dependent) for DeMixT
# Z2, A2 - individual scores and loadings (up to orthogonal rotation, see below) for TIMER
# To match the loadings with cell types, see
U1 = (outECCA$U1)
U2 = (outECCA$U2)
Z1 = outECCA$Z1 # individual score for DeMixT
```

```

Z2 = outECCA$Z2 # individual score for TIMER
names(U1) = names(U2) = names(Z1) = rownames(Z2) = sampleIDs

# Prepare the Gleason score information
load("./tcga_prostate_detailed_gelason.RData")
row.names(meta.info.GS) = gsub(x = row.names(meta.info.GS), pattern = "\\.", replacement = "-")
meta.info.GS = meta.info.GS[,-1]
TCGA.PRAD.GS = meta.info.GS[sampleIDs,]

GScore = as.character(TCGA.PRAD.GS$Reviewed.Gleason)
names(GScore) = row.names(TCGA.PRAD.GS)

GScore.filter = GScore[-which(GScore == "3+3")]
GScore.filter[which(GScore.filter == "3+5" |
                    GScore.filter == "4+4" |
                    GScore.filter == "4+5" |
                    GScore.filter == "5+3" |
                    GScore.filter == "5+4" |
                    GScore.filter == "5+5" |
                    GScore.filter == "5+6" |
                    GScore.filter == "5+7" )] = "8+"

# Keep samples with gleason score >= 7
U1.filter = U1[names(GScore.filter)]
U2.filter = U2[names(GScore.filter)]
Z1.filter = Z1[names(GScore.filter)]
Z2.filter = Z2[names(GScore.filter),]

# Take one joint score by combining U1 and U2, they have super high colinearity
# Rotate individual scores and loading for TIMER to make both orthogonal

crossprod(U1.filter, U2.filter) # 0.7827502

##           [,1]
## [1,] 0.7827502

Ujoint = svd(cbind(U1.filter, U2.filter))$u[, 1]
timerIndSvd = svd(tcrossprod(Z2.filter, outECCA$A2))
Individual_score_TIMER_1 = timerIndSvd$u[, 1]
Individual_score_TIMER_2 = timerIndSvd$u[, 2]

# Match with patients in outECCA

colnames(pfi_data)[1] = "bcr_patient_barcode" # Adjust the first column name.
pfi_time =
  pfi_data %>% select(bcr_patient_barcode, PFI, PFI.time, age_at_initial_pathologic_diagnosis) %>%
  filter(bcr_patient_barcode %in% sampleIDs) %>%
  mutate(bcr_patient_barcode = as.character(bcr_patient_barcode),
         PFI = as.numeric(as.character(PFI)),
         PFI.time = as.numeric(as.character(PFI.time)))
rownames(pfi_time) = pfi_time$bcr_patient_barcode
pfi_time = pfi_time[sampleIDs, ]

names(Ujoint) = names(GScore.filter)

```

```
names(Individual_score_TIMER_1) = names(GScore.filter)
names(Individual_score_TIMER_2) = names(GScore.filter)
```

Prepare the input data frame for figs #####

```
data <- data.frame(
  Age = pfi_time[names(GScore.filter),4],
  Joint_score = Ujoint[names(GScore.filter)],
  Individual_score_DeMixT = Z1.filter,
  Individual_score_TIMER_1 = Individual_score_TIMER_1[names(GScore.filter)],
  Individual_score_TIMER_2 = Individual_score_TIMER_2[names(GScore.filter)],
  Gleason_score = factor(GScore.filter, levels = c("3+4", "4+3", "8+")),
  PFI = pfi_time[names(GScore.filter),2],
  PFI.time = pfi_time[names(GScore.filter),3])
dim(data)
```

```
## [1] 239 8
```

```
save(data, file = "./ecca_result_analysis_7+.RData")
```

Boxplot show ECCA scores and gleason groups #####

```
load(file = "./ecca_result_analysis_7+.RData")
```

```
length(which(data$Gleason_score == "3+4"))
```

```
## [1] 89
```

```
length(which(data$Gleason_score == "4+3"))
```

```
## [1] 72
```

```
length(which(data$Gleason_score == "8+"))
```

```
## [1] 78
```

```
# Fig4A - 1 Ujoint and Gleason score
```

```
cls = c( "#6883bc", "#56B4E9", "#E69F00")
```

```
Med.ECCA= data.frame(Cancer.type=unique(as.character(data$Gleason_score)),
```

```
                      med = c(by(data = data$Joint_score, INDICES = as.character(data$Gleason_score), FUN =
```

```
                      mad = c(by(data = data$Joint_score, INDICES = as.character(data$Gleason_score), FUN =
```

```
Med.ECCA$Gleason_score=rownames(Med.ECCA)
```

```
Med.ECCA[order(Med.ECCA$med, decreasing = T), "order"] = 1:3
```

```
Med.ECCA.reorder=Med.ECCA[order(Med.ECCA$order),]
```

```
data.plot=merge(data, Med.ECCA.reorder, by="Gleason_score")
```

```
violin_plot =
```

```
ggplot(data.plot, aes(x=reorder(Gleason_score, Joint_score, FUN=median, decreasing = T), y= Joint_score,
```

```
  geom_violin(aes(fill=Gleason_score, color=Gleason_score), alpha = 0.7, position=position_dodge(0.8)) +
```

```
  scale_y_continuous(limits=c(-0.1, -0.00), breaks=c(-0.1, -0.08, -0.06,-0.04, -0.02)) +
```

```
  labs(title="", x="", y = "Joint Score") + guides(fill=FALSE) +
```

```
  geom_vline(xintercept = 1:(3-1)+0.5, col="grey", lty=3) +
```

```
  geom_jitter(width = 0.15, size=0.15) +
```

```
  scale_color_manual(values=cls) +
```

```
  scale_fill_manual(values=cls) +
```

```
  geom_segment(aes(x=order-0.4, xend=order+0.4, y=med, yend=med), color="black", size=1) +
```

```
  theme(title=element_text(size=6), axis.text.x=element_text(size=10, angle=60, hjust=1),
```

```
axis.title.x=element_text(size=6), axis.text.y=element_text(size=12),
axis.title.y=element_text(size=12, face="bold"), legend.position = "none",
legend.text=element_text(size=6), legend.title=element_text(size=6))
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

```
stat.test <- data %>%
  t_test(Joint_score ~ Gleason_score) %>%
  adjust_pvalue(method = "BH") %>%
  add_significance()
```

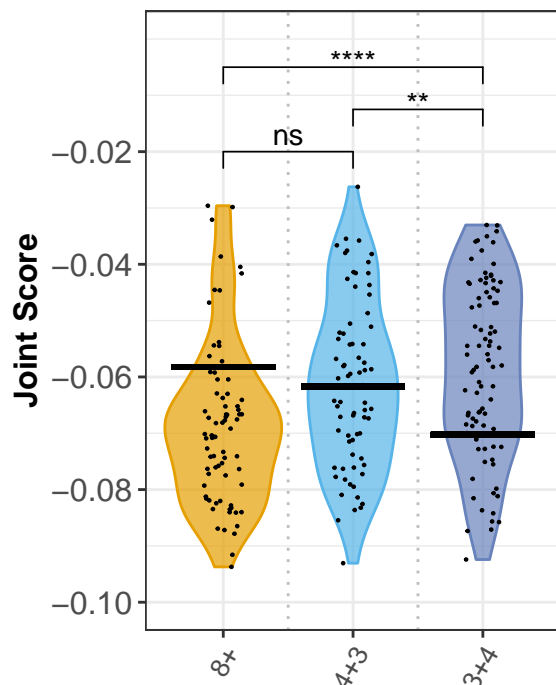
```
stat.test
```

```
## # A tibble: 3 x 10
##   .y.      group1 group2    n1    n2 statistic    df      p    p.adj p.adj.signif
##   <chr>   <chr>   <chr>   <int> <int>   <dbl> <dbl>   <dbl>   <dbl>   <chr>
## 1 Joint_~ 3+4    4+3      89    72     1.01  154. 3.14e-1 3.14e-1 ns
## 2 Joint_~ 3+4    8+      89    78     4.31  163. 2.8 e-5 8.4 e-5 ****
## 3 Joint_~ 4+3    8+      72    78     3.15  147. 2 e-3 3 e-3 **
```

```
stat.test <- stat.test %>% add_xy_position(x = "Gleason_score")
p = violin_plot + stat_pvalue_manual(stat.test, label = "p.adj.signif", y.position = c(-0.02, -0.005, -0.015))
print(p)
```

```
## Warning: Removed 1 rows containing non-finite values (stat_ydensity).
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
ggsave(plot = p, filename = "./Image/1_Ujoint_vs_Gleason.pdf", width = 3, height = 4, dpi = 300)
```

```
## Warning: Removed 1 rows containing non-finite values (stat_ydensity).
```

```
## Removed 1 rows containing missing values (geom_point).
```

```

# Fig4A - 2 Individual DeMixT and Gleason score
cls = c( "#6883bc", "#56B4E9", "#E69F00")

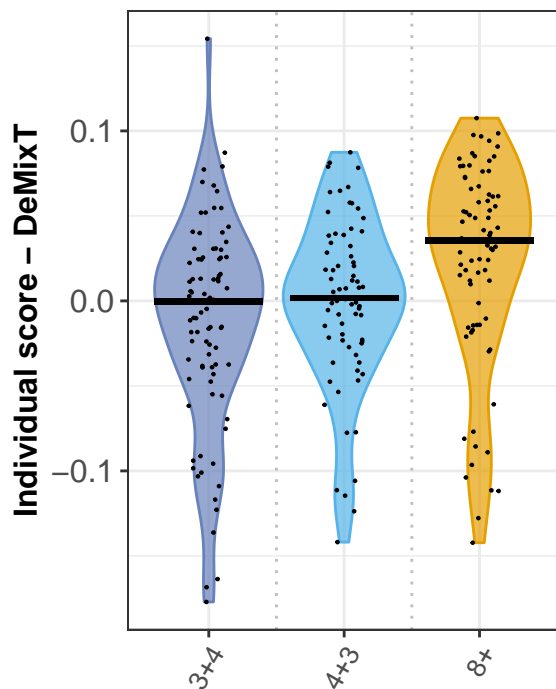
Med.ECCA= data.frame(Cancer.type=unique(as.character(data$Gleason_score)),
                     med = c(by(data = data$Individual_score_DeMixT, INDICES = as.character(data$Gleason_score),
                                mad = c(by(data = data$Individual_score_DeMixT, INDICES = as.character(data$Gleason_score),
Med.ECCA$Gleason_score=rownames(Med.ECCA)
Med.ECCA[order(Med.ECCA$med, decreasing = F), "order"] = 1:3
Med.ECCA.reorder=Med.ECCA[order(Med.ECCA$order),]
data.plot=merge(data, Med.ECCA.reorder, by="Gleason_score")

violin_plot =
ggplot(data.plot, aes(x=reorder(Gleason_score, Individual_score_DeMixT, FUN=median, decreasing = F), y=
  geom_violin(aes(fill=Gleason_score, color=Gleason_score), alpha = 0.7, position=position_dodge(0.8)) +
  labs(title="", x="", y = "Individual score - DeMixT") + guides(fill=FALSE) +
  geom_vline(xintercept = 1:(3-1)+0.5, col="grey", lty=3) +
  geom_jitter(width = 0.15, size=0.15) +
  scale_color_manual(values=cls) +
  scale_fill_manual(values=cls) +
  geom_segment(aes(x=order-0.4, xend=order+0.4, y=med, yend=med), color="black", size=1) +
  theme(title=element_text(size=6), axis.text.x=element_text(size=10, angle=60, hjust=1),
        axis.title.x=element_text(size=6), axis.text.y=element_text(size=12),
        axis.title.y=element_text(size=12, face="bold"), legend.position = "none",
        legend.text=element_text(size=6), legend.title=element_text(size=6))

## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.

print(violin_plot)

```



```

stat.test <- data %>%
  t_test(Individual_score_DeMixT ~ Gleason_score) %>%

```

```

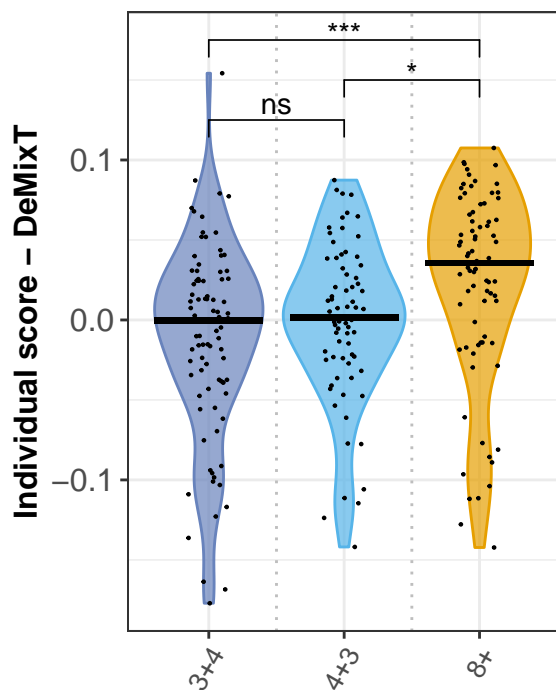
adjust_pvalue(method = "BH") %>%
add_significance()

stat.test

## # A tibble: 3 x 10
##   .y.      group1 group2    n1    n2 statistic    df      p    p.adj p.adj.signif
##   <chr>   <chr>   <chr>  <int> <int>    <dbl> <dbl>   <dbl>   <dbl>   <chr>
## 1 Indivi~ 3+4    4+3     89    72    -1.29  159.  1.98e-1  1.98e-1 ns
## 2 Indivi~ 3+4     8+     89    78    -3.71  162.  2.89e-4  8.67e-4 ***
## 3 Indivi~ 4+3     8+     72    78    -2.60  146.  1.5e-2    1.5e-2 *

stat.test <- stat.test %>% add_xy_position(x = "Gleason_score")
p = violin_plot + stat_pvalue_manual(stat.test, label = "p.adj.signif", y.position = c(0.125, 0.175, 0.225))
print(p)

```



```

ggsave(plot = p, filename = "../Image/2_Ind_DeMixT_vs_Gleason.pdf", width = 3, height = 4, dpi = 300)

# Fig4A - 3 Individual TIMER 1 and Gleason score
cls = c( "#6883bc", "#56B4E9", "#E69F00")

Med.ECCA= data.frame(Cancer.type=unique(as.character(data$Gleason_score)),
                     med = c(by(data = data$Individual_score_TIMER_1, INDICES = as.character(data$Gleason_score), FUN=median, decreasing = F), y.position=position_dodge(0.8)),
                     mad = c(by(data = data$Individual_score_TIMER_1, INDICES = as.character(data$Gleason_score), FUN=mad, decreasing = F), y.position=position_dodge(0.8)))
Med.ECCA$Gleason_score=rownames(Med.ECCA)
Med.ECCA[order(Med.ECCA$med, decreasing = F), "order"] = 1:3
Med.ECCA.reorder=Med.ECCA[order(Med.ECCA$order),]
data.plot=merge(data, Med.ECCA.reorder, by="Gleason_score")

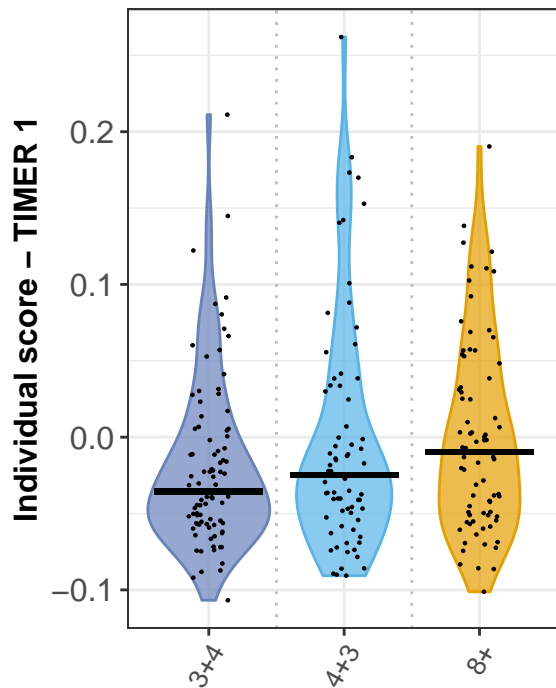
violin_plot =
ggplot(data.plot, aes(x=reorder(Gleason_score, Individual_score_TIMER_1, FUN=median, decreasing = F), y=Individual_score_TIMER_1, fill=Gleason_score, color=Gleason_score), alpha = 0.7, position=position_dodge(0.8)) +
  geom_violin(aes(fill=Gleason_score, color=Gleason_score), alpha = 0.7, position=position_dodge(0.8)) +
  labs(title="", x="", y = "Individual score - TIMER 1") + guides(fill=FALSE) +

```

```
geom_vline(xintercept = 1:(3-1)+0.5, col="grey", lty=3) +
geom_jitter(width = 0.15, size=0.15) +
scale_color_manual(values=cls) +
scale_fill_manual(values=cls) +
geom_segment(aes(x=order-0.4, xend=order+0.4, y=med, yend=med), color="black", size=1) +
theme(title=element_text(size=6), axis.text.x=element_text(size=10, angle=60, hjust=1),
      axis.title.x=element_text(size=6), axis.text.y=element_text(size=12),
      axis.title.y=element_text(size=12, face="bold"), legend.position = "none",
      legend.text=element_text(size=6), legend.title=element_text(size=6))
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

```
print(violin_plot)
```

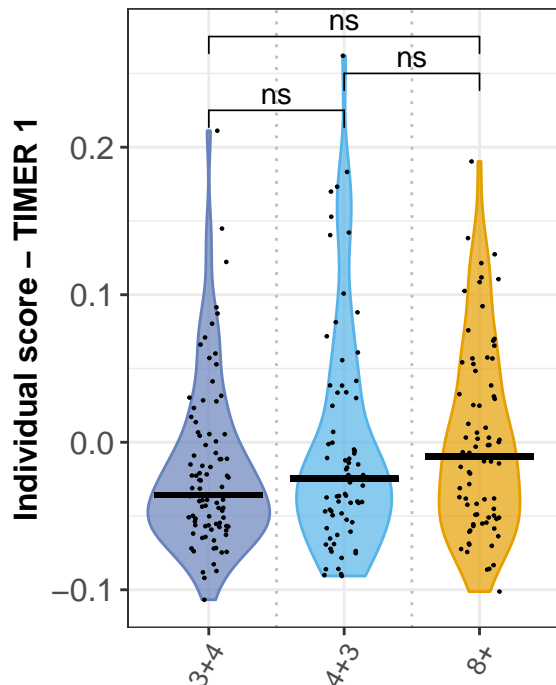


```
stat.test <- data %>%
  t_test(Individual_score_TIMER_1 ~ Gleason_score) %>%
  adjust_pvalue(method = "BH") %>%
  add_significance()
```

```
stat.test
```

```
## # A tibble: 3 x 10
##   .y.      group1 group2   n1    n2 statistic    df      p p.adj p.adj.signif
##   <chr>    <chr>  <chr> <int> <int>    <dbl> <dbl> <dbl> <dbl> <chr>
## 1 Individual~ 3+4    4+3     89    72    -1.47   127. 0.143 0.214 ns
## 2 Individual~ 3+4    8+     89    78    -2.07   155. 0.04  0.12 ns
## 3 Individual~ 4+3    8+     72    78    -0.299  139. 0.765 0.765 ns
```

```
stat.test <- stat.test %>% add_xy_position(x = "Gleason_score")
p = violin_plot + stat_pvalue_manual(stat.test, label = "p.adj.signif", y.position = c(0.225, 0.275, 0.325))
print(p)
```



```
ggsave(plot = p, filename = "./Image/3_Ind_TIMER_vs_Gleason.pdf", width = 3, height = 4, dpi = 300)
```

```
# Fig4A - 3 Individual TIMER 2 and Gleason score
```

```
cls = c( "#6883bc", "#56B4E9", "#E69F00")
```

```
Med.ECCA= data.frame(Cancer.type=unique(as.character(data$Gleason_score)),
```

```
                      med = c(by(data = data$Individual_score_TIMER_2, INDICES = as.character(data$Gleason_score),
```

```
                      mad = c(by(data = data$Individual_score_TIMER_2, INDICES = as.character(data$Gleason_score),
```

```
Med.ECCA$Gleason_score=rownames(Med.ECCA)
```

```
Med.ECCA[order(Med.ECCA$med, decreasing = F), "order"] = 1:3
```

```
Med.ECCA.reorder=Med.ECCA[order(Med.ECCA$order),]
```

```
data.plot=merge(data, Med.ECCA.reorder, by="Gleason_score")
```

```
violin_plot =
```

```
ggplot(data.plot, aes(x=reorder(Gleason_score, Individual_score_TIMER_2, FUN=median, decreasing = F), y=
```

```
  geom_violin(aes(fill=Gleason_score, color=Gleason_score), alpha = 0.7, position=position_dodge(0.8)) +
```

```
  labs(title="", x="", y = "Individual score - TIMER 2") + guides(fill=FALSE) +
```

```
  geom_vline(xintercept = 1:(3-1)+0.5, col="grey", lty=3) +
```

```
  geom_jitter(width = 0.15, size=0.15) +
```

```
  scale_color_manual(values=cls) +
```

```
  scale_fill_manual(values=cls) +
```

```
  geom_segment(aes(x=order-0.4, xend=order+0.4, y=med, yend=med), color="black", size=1) +
```

```
  theme(title=element_text(size=6), axis.text.x=element_text(size=10, angle=60, hjust=1),
```

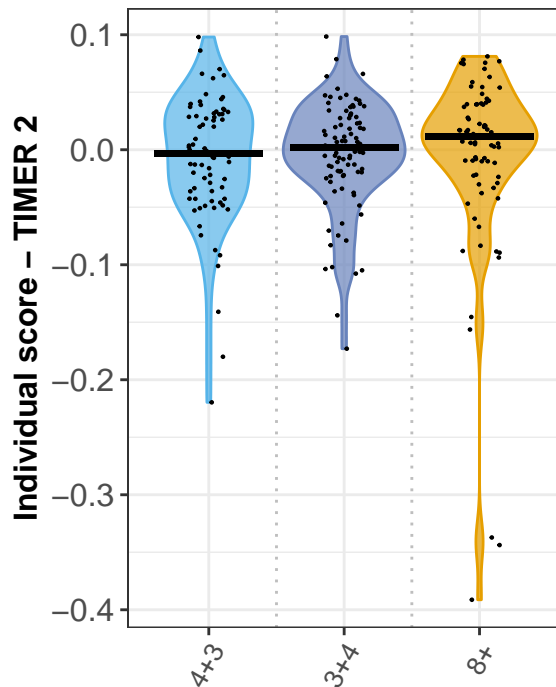
```
        axis.title.x=element_text(size=6), axis.text.y=element_text(size=12),
```

```
        axis.title.y=element_text(size=12, face="bold"), legend.position = "none",
```

```
        legend.text=element_text(size=6), legend.title=element_text(size=6))
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =`  
## "none")` instead.
```

```
print(violin_plot)
```

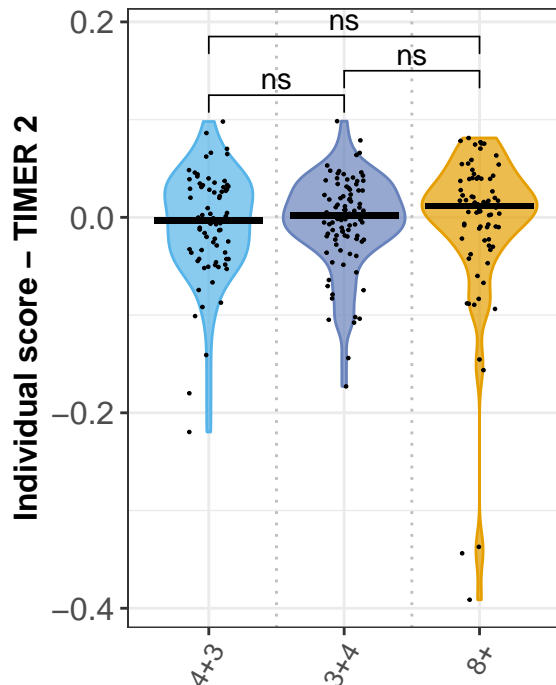



```
stat.test <- data %>%
  # group_by(Gleason_score) %>%
  t_test(Individual_score_TIMER_2 ~ Gleason_score) %>%
  adjust_pvalue(method = "BH") %>%
  add_significance()
```

```
stat.test
```

```
## # A tibble: 3 x 10
##   .y.      group1 group2    n1    n2 statistic    df      p p.adj p.adj.signif
##   <chr>    <chr>  <chr>  <int> <int>    <dbl> <dbl> <dbl> <dbl>  <chr>
## 1 Individual~ 3+4    4+3     89    72    0.458   139. 0.647 0.953 ns
## 2 Individual~ 3+4    8+     89    78    0.414   117. 0.68  0.953 ns
## 3 Individual~ 4+3    8+     72    78    0.0588  134. 0.953 0.953 ns
```

```
stat.test <- stat.test %>% add_xy_position(x = "Gleason_score")
p = violin_plot + stat_pvalue_manual(stat.test, label = "p.adj.signif", y.position = c(0.125, 0.185, 0.245))
print(p)
```



```
ggsave(plot = p, filename = "./Image/4_Ind_TIMER_vs_Gleason.pdf", width = 3, height = 4, dpi = 300)
```

CoxPh for joint and individual scores with and without adjusting for Gleason scores #####

```
# Standardize the age into range 0-1
range01 <- function(x){(x-mean(x))/(sd(x))}
data$Age_scale = range01(as.numeric(data$Age))

# Create the grouping vector based on the gleason score
gleason_group = rep(NA, length(data$Gleason_score))
for (i in 1:length(data$Gleason_score)){
  if (data[i,"Gleason_score"] == "3+4"){
    gleason_group[i] = "7"
  } else if (data[i,"Gleason_score"] == "4+3"){
    gleason_group[i] = "7"
  } else if (data[i,"Gleason_score"] == "8+"){
    gleason_group[i] = "8+"
  }
}
data$Gleason_score_scale = gleason_group

# Note that there are scaling issues looking at coefficient magnitude, so rescale the scores
data$Joint_score_scale = 100 * data$Joint_score
data$Individual_score_DeMixT_scale = 10 * data$Individual_score_DeMixT
data$Individual_score_TIMER_1_scale = 10 * data$Individual_score_TIMER_1
data$Individual_score_TIMER_2_scale = 10 * data$Individual_score_TIMER_2

var = c("PFI", "PFI.time", "Age_scale", "Gleason_score_scale",
        "Joint_score_scale", "Individual_score_DeMixT_scale",
        "Individual_score_TIMER_1_scale", "Individual_score_TIMER_2_scale")

Coxph_input = data[,var]
```

```
##### Check correspondence between gleason score and various structure #####
t.test(data$Joint_score ~ data$Gleason_score_scale)$p.value # significant difference in Ujoint 2.320089e-05

## [1] 2.320089e-05
t.test(data$Individual_score_DeMixT ~ data$Gleason_score_scale)$p.value # very significant difference o

## [1] 0.00037857
t.test(data$Individual_score_TIMER_1 ~ data$Gleason_score_scale)$p.value # not significant for individu

## [1] 0.1707365
t.test(data$Individual_score_TIMER_2 ~ data$Gleason_score_scale)$p.value # not significant for individu

## [1] 0.7898687
# [1] 2.320089e-05
# [1] 0.00037857
# [1] 0.1707365
# [1] 0.7898687
```

Model_1, without Gleason scores, to test the prognostic predictive ability of Joint scores #####

```
modZero <- coxph(Surv(PFI.time/365, PFI, type = "right") ~ Age_scale + Joint_score_scale ,
  data = Coxph_input)
modAll <- coxph(Surv(PFI.time/365, PFI, type = "right") ~
  Age_scale +
  Joint_score_scale + Individual_score_DeMixT_scale + Individual_score_TIMER_1_scale + In
  data = Coxph_input)
model_1 = MASS::stepAIC(modZero, direction = "both", scope = list(lower = modZero, upper = modAll))

## Start: AIC=471.74
## Surv(PFI.time/365, PFI, type = "right") ~ Age_scale + Joint_score_scale
##
##              Df    AIC
## + Individual_score_DeMixT_scale  1 468.69
## + Individual_score_TIMER_1_scale  1 471.16
## <none>                                471.74
## + Individual_score_TIMER_2_scale  1 473.42
##
## Step: AIC=468.69
## Surv(PFI.time/365, PFI, type = "right") ~ Age_scale + Joint_score_scale +
##   Individual_score_DeMixT_scale
##
##              Df    AIC
## + Individual_score_TIMER_1_scale  1 468.02
## <none>                                468.69
## + Individual_score_TIMER_2_scale  1 470.41
## - Individual_score_DeMixT_scale  1 471.74
##
## Step: AIC=468.02
## Surv(PFI.time/365, PFI, type = "right") ~ Age_scale + Joint_score_scale +
##   Individual_score_DeMixT_scale + Individual_score_TIMER_1_scale
##
##              Df    AIC
```

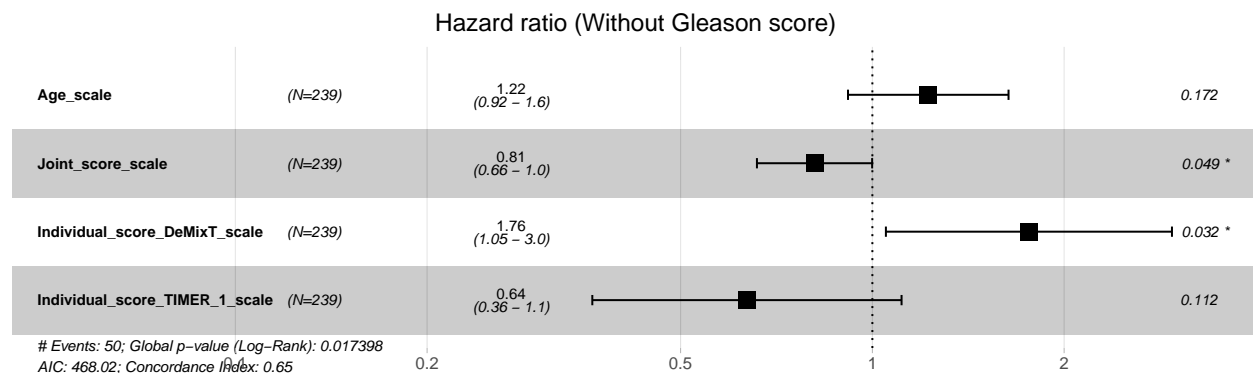
```
## <none> 468.02
## - Individual_score_TIMER_1_scale 1 468.69
## + Individual_score_TIMER_2_scale 1 469.87
## - Individual_score_DeMixT_scale 1 471.16

summary(model_1)

## Call:
## coxph(formula = Surv(PFI.time/365, PFI, type = "right") ~ Age_scale +
## Joint_score_scale + Individual_score_DeMixT_scale + Individual_score_TIMER_1_scale,
## data = Coxph_input)
##
## n= 239, number of events= 50
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## Age_scale      0.2021   1.2240  0.1480  1.365  0.1721
## Joint_score_scale -0.2090   0.8114  0.1062 -1.968  0.0491 *
## Individual_score_DeMixT_scale  0.5659   1.7610  0.2638  2.145  0.0319 *
## Individual_score_TIMER_1_scale -0.4534   0.6355  0.2852 -1.590  0.1119
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##               exp(coef) exp(-coef) lower .95 upper .95
## Age_scale      1.2240   0.8170   0.9158   1.6359
## Joint_score_scale  0.8114   1.2324   0.6589   0.9992
## Individual_score_DeMixT_scale  1.7610   0.5678   1.0501   2.9532
## Individual_score_TIMER_1_scale  0.6355   1.5736   0.3634   1.1113
##
## Concordance= 0.65 (se = 0.038 )
## Likelihood ratio test= 11.99 on 4 df,  p=0.02
## Wald test            = 12.06 on 4 df,  p=0.02
## Score (logrank) test = 12.23 on 4 df,  p=0.02
```

```
ggforest(model_1,main = "Hazard ratio (Without Gleason score)")
```

```
## Warning in .get_data(model, data = data): The `data` argument is not provided.
## Data will be extracted from model fit.
```



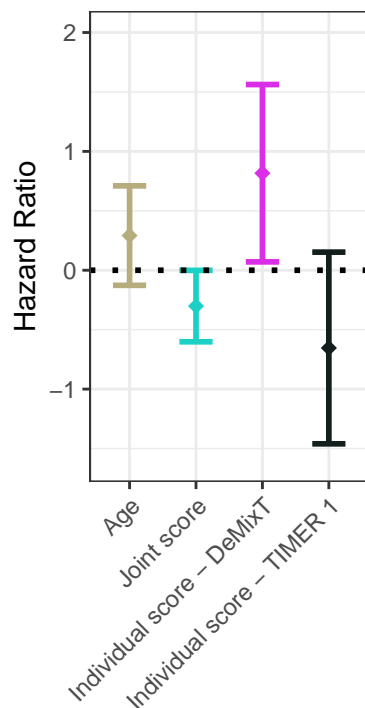
```
input = summary(model_1)[[8]][,c(1,3:4)]
colnames(input) = c("HR", "HR.Lower", "HR.Upper")
rownames(input) = c("Age", "Joint score", "Individual score - DeMixT", "Individual score - TIMER 1")
Variable.name = Order.new = factor(x = c("Age", "Joint score", "Individual score - DeMixT", "Individual score - TIMER 1"),
levels = c("Age", "Joint score", "Individual score - DeMixT", "Individual score - TIMER 1"))
HR_valid_plot <-
```

```

ggplot(input, aes(x = (Order.new), y = 0)) +
  geom_errorbar(aes(x = Order.new, ymin = log2(HR.Lower), ymax = log2(HR.Upper), color = Variable.name),
    width = 0.5, size = 1, position = position_dodge(0.8)) +
  geom_point(aes(x = Order.new, y = log2(HR), color = Variable.name), position = position_dodge(0.8), size = 1) +
  scale_color_manual(values=c("#B7AC7D", "#1CD0C6", "#DA2DE7", "#121F1C")) +
  scale_y_continuous(limits=c(-1.6, 2)) +
  geom_hline(yintercept = 0, linetype = "dotted", color = "black", size = 1) +
  theme_bw() +
  ylab(expression("Hazard Ratio")) +
  xlab(" ") +
  # ggtitle("Without Gleason score") +
  theme(legend.position="none",
    axis.text.x = element_text(angle = 45, hjust=1))

print(HR_valid_plot)

```



```

ggsave(plot = (HR_valid_plot), filename = "./Image/5_HR_valid_plot.pdf", width = 2, height = 4, dpi = 300)

Model_2, with Gleason scores, to validate the clinical importance of joint score #####
modZero <- coxph(Surv(PFI.time/365, PFI, type = "right") ~ Age_scale + Gleason_score_scale + Joint_score_scale,
  data = Coxph_input)
modAll <- coxph(Surv(PFI.time/365, PFI, type = "right") ~
  Age_scale + Gleason_score_scale + Joint_score_scale +
  Individual_score_DeMixT_scale + Individual_score_TIMER_1_scale + Individual_score_TIMER_2_scale,
  data = Coxph_input)
model_2 = MASS::stepAIC(modZero, direction = "both", scope = list(lower = modZero, upper = modAll))

## Start: AIC=465.64
## Surv(PFI.time/365, PFI, type = "right") ~ Age_scale + Gleason_score_scale +
## Joint_score_scale

```

```

##
##              Df      AIC
## + Individual_score_DeMixT_scale  1 465.09
## <none>                          465.64
## + Individual_score_TIMER_1_scale  1 465.92
## + Individual_score_TIMER_2_scale  1 467.48
##
## Step:  AIC=465.09
## Surv(PFI.time/365, PFI, type = "right") ~ Age_scale + Gleason_score_scale +
##      Joint_score_scale + Individual_score_DeMixT_scale
##
##              Df      AIC
## + Individual_score_TIMER_1_scale  1 465.06
## <none>                          465.09
## - Individual_score_DeMixT_scale  1 465.64
## + Individual_score_TIMER_2_scale  1 466.93
##
## Step:  AIC=465.06
## Surv(PFI.time/365, PFI, type = "right") ~ Age_scale + Gleason_score_scale +
##      Joint_score_scale + Individual_score_DeMixT_scale + Individual_score_TIMER_1_scale
##
##              Df      AIC
## <none>                          465.06
## - Individual_score_TIMER_1_scale  1 465.09
## - Individual_score_DeMixT_scale  1 465.92
## + Individual_score_TIMER_2_scale  1 466.96
summary(model_2)

## Call:
## coxph(formula = Surv(PFI.time/365, PFI, type = "right") ~ Age_scale +
##      Gleason_score_scale + Joint_score_scale + Individual_score_DeMixT_scale +
##      Individual_score_TIMER_1_scale, data = Coxph_input)
##
##      n= 239, number of events= 50
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## Age_scale      0.1849   1.2031  0.1487  1.243   0.214
## Gleason_score_scale8+ 0.6749   1.9639  0.3032  2.226   0.026 *
## Joint_score_scale -0.1521   0.8589  0.1092 -1.393   0.164
## Individual_score_DeMixT_scale 0.4436   1.5583  0.2719  1.632   0.103
## Individual_score_TIMER_1_scale -0.4038   0.6678  0.2897 -1.394   0.163
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## Age_scale      1.2031   0.8312   0.8989   1.610
## Gleason_score_scale8+ 1.9639   0.5092   1.0840   3.558
## Joint_score_scale 0.8589   1.1643   0.6934   1.064
## Individual_score_DeMixT_scale 1.5583   0.6417   0.9146   2.655
## Individual_score_TIMER_1_scale 0.6678   1.4975   0.3785   1.178
##
## Concordance= 0.668 (se = 0.037 )
## Likelihood ratio test= 16.96 on 5 df,  p=0.005
## Wald test              = 16.77 on 5 df,  p=0.005

```

```
## Score (logrank) test = 17.61 on 5 df, p=0.003
```

```
# ggforest(model_2, main = "Hazard ratio (With Gleason score)")
```

```
Heatmap show joint/individual loading vector #####
```

```
V1 = outECCA$V1
```

```
A1 = outECCA$A1 # corresponding individual loading, this is primarily immune
```

```
V2 = outECCA$V2
```

```
A2 = outECCA$A2 # corresponding individual loading of TIMER
```

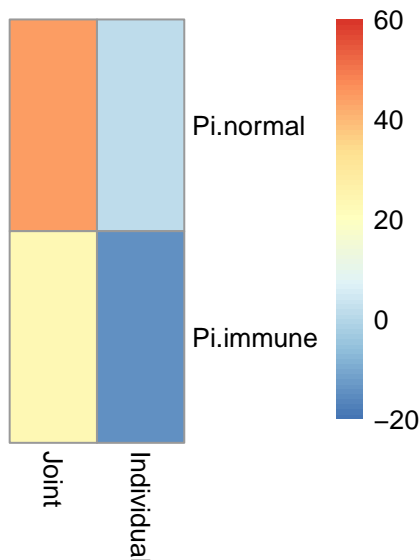
```
breaksList = seq(-20, 60, by = 1)
```

```
demixt.input = cbind(V1,A1)
```

```
colnames(demixt.input) = c("Joint", "Individual")
```

```
demixt.input = demixt.input[c("Pi.normal", "Pi.immune"),]
```

```
p1 = pheatmap(demixt.input, cluster_rows = F, cluster_cols = F, show_rownames = T, show_colnames = T, #
              color = colorRampPalette(rev(brewer.pal(n = 7, name = "RdYlBu")))(length(breaksList)),
              breaks = breaksList)
```



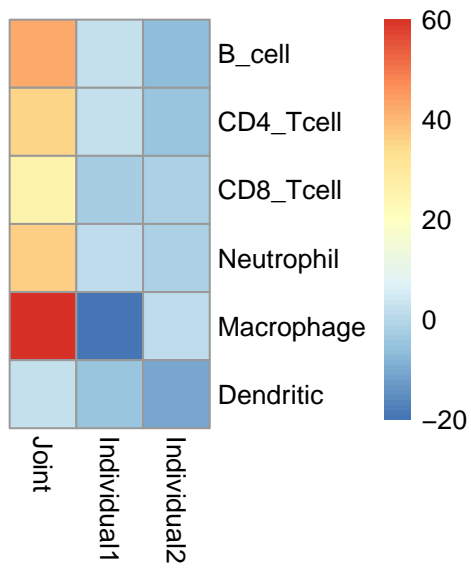
```
ggsave(plot = p1, filename = "./Image/5_loading_vactor_DeMixT.pdf", width = 1.25, height = 3, dpi = 300)
```

```
timer.input = cbind(V2,A2)
```

```
colnames(timer.input) = c("Joint", "Individual1", "Individual2")
```

```
# demixt.input = demixt.input[c("Pi.normal", "Pi.immune"),]
```

```
p2 = pheatmap(timer.input, cluster_rows = F, cluster_cols = F, show_rownames = T, show_colnames = T,
              color = colorRampPalette(rev(brewer.pal(n = 7, name = "RdYlBu")))(length(breaksList)),
              breaks = breaksList)
```



```
ggsave(plot = p2, filename = "./Image/6_loading_vector_TIMER.pdf", width = 1.5, height = 3, dpi = 300)
```