

Survival analysis of patients with chondrosarcoma

Léa ROGUE

2025-02-14

List of Figures

1	Forest plot for categorical immune cell types expression (n = 82)	9
2	Forest plot for continuous immune cell types expression (n = 82)	10
3	Kaplan-Meier plots for immune cell types (n = 82)	12
4	Forest plot for Th2 cells (n = 82)	14
5	Forest plot for significative immune cells (n = 82)	16
6	Forest plot for categorical immune cell types expression (n = 63)	18
7	Forest plot for continuous immune cell types expression (n = 63)	19
8	Kaplan-Meier plots for immune cell types (n = 63)	21
9	Forest plot for Th2 cells (n = 63)	22
10	Forest plot for significative immune cells (n = 63)	24
11	Forest plot for significant CTA (n = 82)	26
12	Intensitites jitter plot for significant CTA (n = 82)	27
13	Intensitites jitter plot for significant CTA (n = 82)	29
14	Forest plot for significant CTA (n = 63)	32
15	Intensitites jitter plot for significant CTA (n = 63)	33
16	Kaplan-Meier plot with clusters from figure 31 (script 5) (n = 82)	34
17	Kaplan-Meier plot with clusters from figure 3 (script 5) (n = 82)	35
18	Kaplan-Meier plot with clusters from figure 4 (script 5) (n = 82)	36
19	Kaplan-Meier plot with clusters from figure 5 (script 5) (n = 82)	37
20	Kaplan-Meier plot merged clusters from figure 5 (script 5) (n = 82)	38
21	Kaplan-Meier plot with clusters from figure 31 (script 5) (n = 63)	39
22	Kaplan-Meier plot with clusters from figure 6 (script 5) (n = 63)	40
23	Kaplan-Meier plot with merged clusters from figure 6 (n = 63)	41
24	Kaplan-Meier plot with clusters from figure 7 (script 5) (n = 63)	42
25	Kaplan-Meier plot with clusters from figure 8 (script 5) (n = 63)	43
26	Kaplan-Meier plot merged clusters from figure 8 (script 5) (n = 63)	44

Contents

Load librairies	5
Functions	5
Load and format data	7
I. Immune cells survival analysis	8
1) All chondrosarcoma types	8
a- Categorical data	8
b- Continuous data	10
c- Kaplan-Meier plots	11
d- Survival analysis for Th2 cells	13
e- Significant immune cells genes survival analysis	14
2) Conventional chondrosarcomas	17
a- Categorical data	17
b- Continuous data	18
c- Kaplan-Meier plots	20
d- Survival analysis for Th2 cells	22
e- Significant immune cells genes survival analysis	23
II. CTA survival analysis	25
1) All chondrosarcoma types	25
a- Intensities jitter plot for significant CTA	27
b- Normalized intensities for CTA to see the differences	30
2) Conventional chondrosarcomas	30
Intensities jitter plot for significant CTA	33
III. Survival analysis for clusters from CTA heatmaps	34
1) All chondrosarcoma types	34
a- Clustering Kmeans k = 4 with all chondrosarcoma types	34
b- Significant coxph CTA genes clustering	35
c- Selected significant coxph CTA genes clustering	36
d- Non selected significant coxph CTA genes clustering	37
Merge C1 with C2 and C3 with C4	38
2) Conventional chondrosarcomas	39
a- Clustering Kmeans k = 2	39
b- Significant coxph CTA genes clustering	40

Merge C1 and C2	41
c- Selected significant coxph CTA genes clustering	42
d- Non selected significant coxph CTA genes clustering	43
Merge C1 with C2 and C3 with C4	44
3) Cluster 1 from CTA heatmaps analysis	45

This script performs a survival analysis of patients with chondrosarcoma within the E-MTAB-7264 dataset of chondrosarcoma tumors from 102 patients.

The 1st part consists of analyze the impact of immune cells types expression on the survival probabilities to see the types that have a positive or negative impact. The 2nd part is about the impact of CTA expression and finally the last part is about survival analysis from clustering of CTA expression that impact the survival probabilities (script 5) to see the impact of CTA and correlate CTA and immune cells.

These 2 analysis are separated by select all patients or conventional patients because dedifferentiated tumors are different and have an impact on the survival probabilities. The survival probabilities are computed thanks to Cox regression and log rank test.

Load librairies

```
library(dplyr)
library(survival)
library(forestplot)
library(survminer)
library(ggplot2)
library(ggsurvfit)
library(gridExtra)
library(tidyr)
```

Functions

```
# Function to apply coxph model per columns
apply_coxph_model <- function(df) {
  # Empty df
  results <- data.frame()

  # Loop on each columns to apply coxph
  for (col in colnames(df)[3:ncol(df)]) {
    model <- coxph(Surv(OS.delay, OS.event) ~ df[[col]],
                  data = df)

    # Extract results
    exp_coef <- summary(model)$coefficients[, "exp(coef)"]
    lower_ci <- summary(model)$conf.int[, "lower .95"]
    upper_ci <- summary(model)$conf.int[, "upper .95"]
    p_value <- summary(model)$coefficients[, "Pr(>|z|)"]
    results <- rbind(results, data.frame(Variable = col,
                                          HR = exp_coef, LowerCI = lower_ci, UpperCI = upper_ci,
                                          Pvalue = p_value))
  }
  return(results)
}

# Function to create Kaplan-Meier plot
plot_km <- function(mod, df) {
```

```

km_plot <- ggsurvplot(mod, data = df, pval = TRUE, conf.int = FALSE,
  ggtheme = theme_bw(), palette = c("#E7B800", "#2E9FDF",
    "#FF6F61", "#4EBB92"))
return(km_plot$plot)
}

# Function to generate forest plot from coxph analysis from
# the function apply_coxph_model
generate_forestplot <- function(data, Type) {
  if (Type == TRUE) {
    # Order by Signature and p-value
    data <- data[order(data$Signature, data$Pvalue), ]
    data$Pvalue <- sprintf("%.4f", data$Pvalue)
    data %>%
      forestplot(labeltext = c(Signature, Variable, Pvalue),
        mean = HR, lower = LowerCI, upper = UpperCI,
        grid = TRUE, zero = 1, col = fpColors(box = "black",
          line = "black"), hrzl_lines = TRUE, title = "Forest Plot for Cox Model",
        txt_gp = fpTxtGp(label = gpar(fontsize = 8)),
        ci.vertices = TRUE, boxsize = 0.1) %>%
      fp_set_zebra_style("#EFEFEF") %>%
      fp_add_header(Signature = c("", "Cell type"), Variable = c("",
        "Genes"), Pvalue = c("", "p-value"))
  } else {
    # Order by p-value
    data <- data[order(data$Pvalue), ]
    data$Pvalue <- sprintf("%.4f", data$Pvalue)
    data %>%
      forestplot(labeltext = c(Variable, Pvalue), mean = HR,
        lower = LowerCI, upper = UpperCI, grid = TRUE,
        zero = 1, col = fpColors(box = "black", line = "black"),
        hrzl_lines = TRUE, title = "Forest Plot for Cox Model",
        txt_gp = fpTxtGp(label = gpar(fontsize = 8)),
        ci.vertices = TRUE, boxsize = 0.1) %>%
      fp_set_zebra_style("#EFEFEF") %>%
      fp_add_header(Variable = c("", "Genes"), Pvalue = c("",
        "p-value"))
  }
}

# Function to calculate Z scores
calculate_z_scores <- function(df_input, col) {
  # Exclude col PROBEID SYMBOL and CTA
  data_values <- df_input[, -c(col)]

  # Calculate Z-scores
  z_scores_row <- t(scale(t(data_values)))

  # Add columns
  df_z_scores <- cbind(df_input[, c(col)], z_scores_row)

  # Return df
  return(df_z_scores)
}

```

```
}
```

Load and format data

```
# Read the metadata and select individuals that have
# survival data
df_metadata <- read.table("../results/metadata.tsv", sep = "\t",
  header = TRUE, check.names = FALSE, dec = ",")
df_metadata_surv <- df_metadata[, c("Patient", "OS.delay", "OS.event")]
df_metadata_surv <- na.omit(df_metadata_surv)

# Conventional chondrosarcoma
df_metadata_conv <- df_metadata[df_metadata$Histology != "N/A" &
  df_metadata$Histology != "benign" & df_metadata$Histology !=
  "dedifferentiated", ]
df_metadata_surv_conv <- df_metadata_conv[, c("Patient", "OS.delay",
  "OS.event")]
df_metadata_surv_conv <- na.omit(df_metadata_surv_conv)

# Read the data and select lines of the immune cells
# expression
df_imm_z_scores_82 <- read.table("../results/imm_sign_z_scores_82.tsv",
  sep = "\t", header = TRUE, row.names = 1)
df_imm_z_scores_82 <- as.data.frame(t(df_imm_z_scores_82))

# Read matrix
df_CTA_immune_whole_clean_avg_102 <- read.table("../results/whole_gene_int_CTA_sign_imm_clean.tsv",
  sep = "\t", header = TRUE, check.names = FALSE)
rownames(df_CTA_immune_whole_clean_avg_102) <- df_CTA_immune_whole_clean_avg_102$SYMBOL
df_CTA_immune_whole_clean_avg_82 <- df_CTA_immune_whole_clean_avg_102[,
  c("Signature", "CTA", colnames(df_CTA_immune_whole_clean_avg_102)[colnames(df_CTA_immune_whole_clean_avg_102) !=
  df_metadata_surv$Patient])]
df_z_scores_82 <- calculate_z_scores(df_CTA_immune_whole_clean_avg_82,
  c(1, 2))

# Matrix for conventional patients with survival metadata
df_CTA_immune_whole_clean_avg_63 <- df_CTA_immune_whole_clean_avg_102[,
  c("Signature", "CTA", colnames(df_CTA_immune_whole_clean_avg_102)[colnames(df_CTA_immune_whole_clean_avg_102) !=
  df_metadata_surv_conv$Patient])]
df_z_scores_63 <- calculate_z_scores(df_CTA_immune_whole_clean_avg_63,
  c(1, 2))

# Immune signature z scores for conv patients
df_imm_z_scores_63 <- read.table("../results/imm_sign_z_scores_63.tsv",
  sep = "\t", header = T)
df_imm_z_scores_63 <- as.data.frame(t(df_imm_z_scores_63))
```

I. Immune cells survival analysis

This part concern the survival analysis with the expression of the immune cells. This is to observe the impact of each cell types on the survival probabilities of the patients.

1) All chondrosarcoma types

a- Categorical data

Categorical data refer to HIGH or LOW expression based on the median of z-scores. If the score is $>$ than median, HIGH is affected contrary to $<$ than median, LOW is affected. In this section, we use Cox regression.

```
# Transform to categorical data and take continuous data
# Computing medians to have categorical data
medians_sign <- apply(df_imm_z_scores_82, 2, median)

# Add categorical data, LOW is for under expressed genes
# (under the median) and HIGH upreg
df_cat_all <- df_imm_z_scores_82
for (col in colnames(df_cat_all)) {
  new_column_name <- paste(col, "cat", sep = " ")
  df_cat_all[[new_column_name]] <- ifelse(df_imm_z_scores_82[[col]] >
    medians_sign[col], "HIGH", "LOW")
}

# Select categories
df_cat_all <- df_cat_all[, grep("cat", colnames(df_cat_all))]

# Replace space by _
colnames(df_cat_all) <- gsub(" ", "_", colnames(df_cat_all))

# Add col for merging
df_cat_all$Patient <- rownames(df_cat_all)

# Merge df
df_survival_all <- merge(df_metadata_surv, df_cat_all, by = "Patient")
rownames(df_survival_all) <- df_survival_all$Patient
df_survival_all <- df_survival_all[, -1]

# Continuous data
df_cont_all <- df_imm_z_scores_82

# Replace space by _
colnames(df_cont_all) <- gsub(" ", "_", colnames(df_cont_all))

# Merge df
df_cont_all$Patient <- rownames(df_cont_all)
df_survival_cont <- merge(df_metadata_surv, df_cont_all, by = "Patient")
rownames(df_survival_cont) <- df_survival_cont$Patient
df_survival_cont <- df_survival_cont[, -1]

# Apply coxph model on the categorical variables for the
# immune signatures
```



```

results_cat <- apply_coxph_model(df_survival_all)
# write.table(results_cat, '../results/results_coxhp.tsv',
# sep = '\\t', row.names = FALSE, quote = FALSE)
generate_forestplot(data = results_cat, Type = FALSE)

```

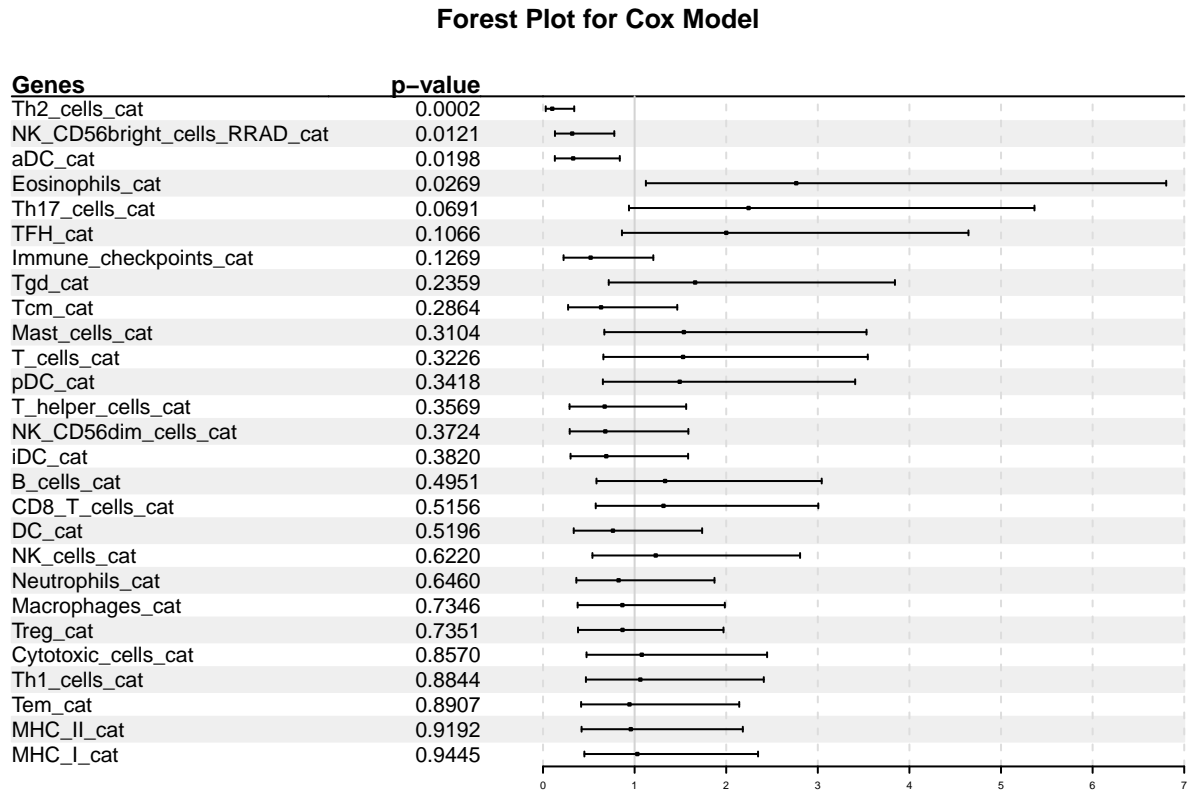


Figure 1: Forest plot for categorical immune cell types expression (n = 82)

We see that the p-value are high, maybe because of categorical data because Cox model rank the values.

b- Continuous data

In this section, we use z-scores data with Cox model.

```
# Apply coxph on continuous
results_cont <- apply_coxph_model(df_survival_cont)
# write.table(results_cont,
# '../results/results_coxph_var_cont.tsv', sep = '\t',
# row.names = FALSE, quote = FALSE)
# pdf('../results/figures/forest_plots/all_indiv/forest_plot_all_patients_var_cont.pdf')
generate_forestplot(results_cont, Type = FALSE)
```

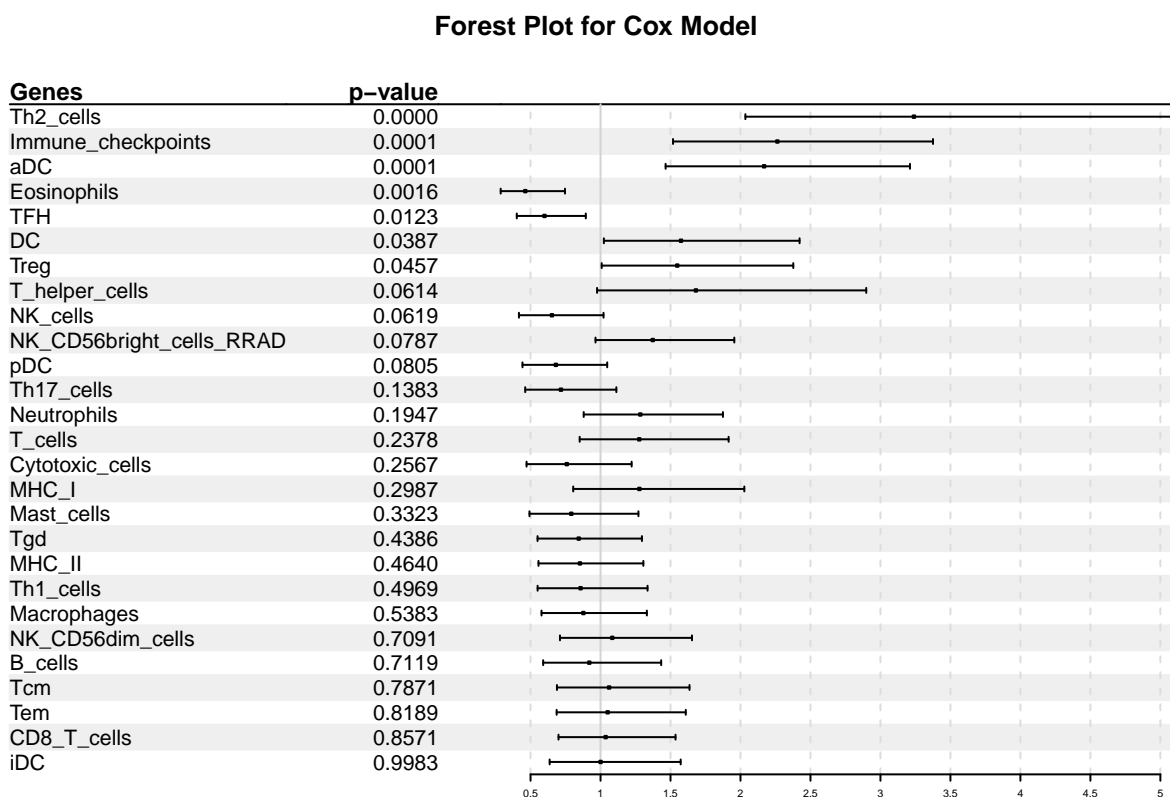


Figure 2: Forest plot for continuous immune cell types expression (n = 82)

```
# dev.off()
```

The p-values are smaller than the model with categorical data so it's seems to be better with continuous data.

For all patients, we see that the hazard ratio (survival probabilities) are increased for Th2 cells, Immune checkpoints, aDC and Treg. So, more these cell types are present in the tumors, more the probability of death is higher, these cell types have a bad impact on survival. Contrary to eosinophils and DC, the HR decreased so more these cells are present in the tumors, the death probability decreased so they have a good impact on prognostic.

c- Kaplan-Meier plots

This part compute Kaplan-Meier plot based on log rank test that use categorical data to see the trend of survival for the 2 groups (HIGH or LOW expression of the cell types) This shows the graph for each significant HR ratio previously computed with Cox model.

```
# Create Kaplan-Meier plots with categorical data
p1 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ DC_cat, data = df_survival_all),
  df_survival_all)
p2 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Eosinophils_cat,
  data = df_survival_all), df_survival_all)
p3 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Immune_checkpoints_cat,
  data = df_survival_all), df_survival_all)
p4 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ TFH_cat, data = df_survival_all),
  df_survival_all)
p5 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Th2_cells_cat,
  data = df_survival_all), df_survival_all)
p6 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Treg_cat, data = df_survival_all),
  df_survival_all)
p7 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ aDC_cat, data = df_survival_all),
  df_survival_all)
# pdf('../results/figures/other_plots/KM_plots_signif_cox_all_indiv.pdf',
# height = 20, width = 15)
grid.arrange(p1, p2, p3, p4, p5, p6, p7, ncol = 2)

# dev.off()
```

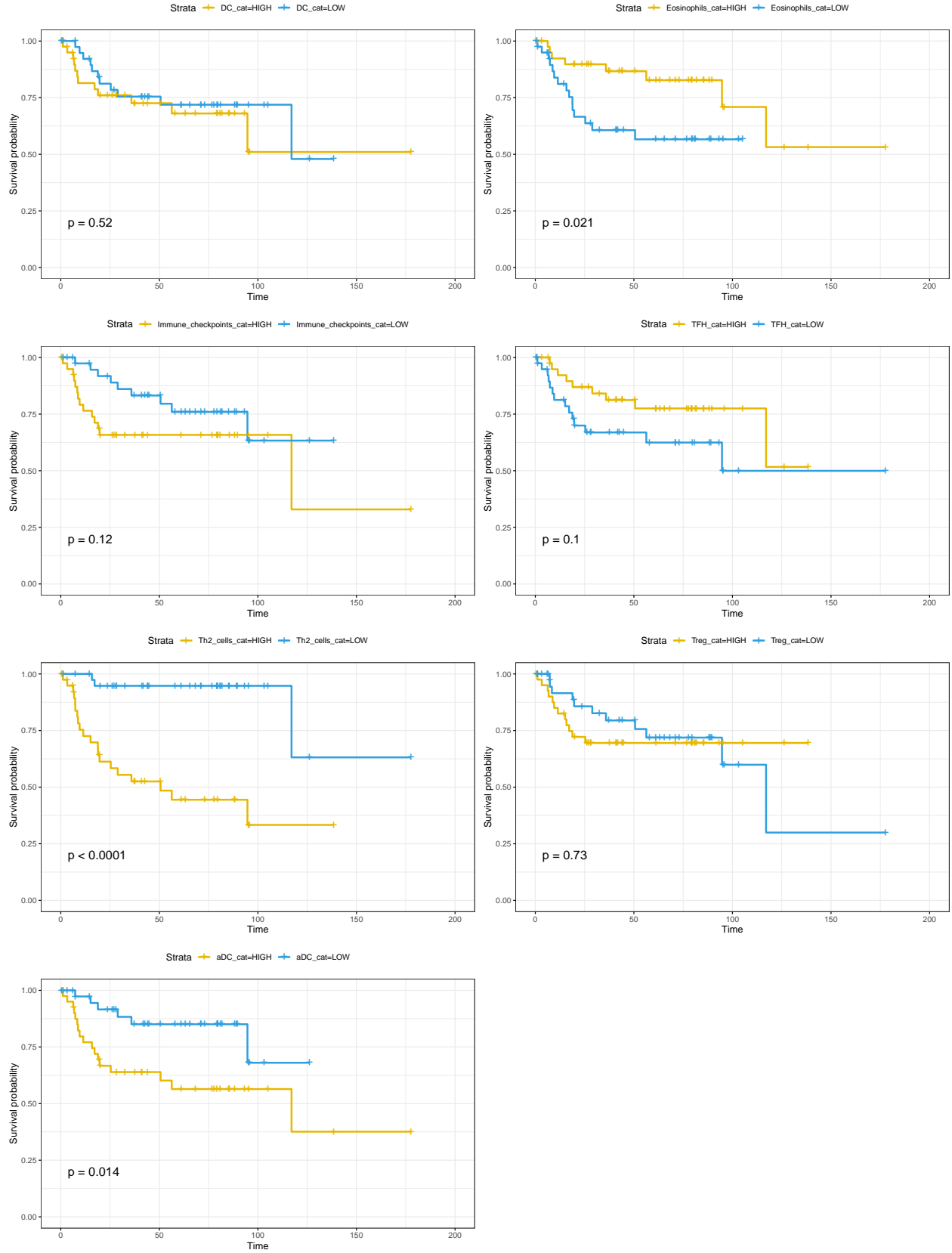


Figure 3: Kaplan-Meier plots for immune cell types (n = 82)

With log rank test, we see that only Th2 and eosinophils are significant.

d- Survival analysis for Th2 cells

In this part, we want to see the HR of the genes expressed by Th2 cells To determine if some genes have a dominant effect over others.

```
# List of genes expressed in th2 cells
l_genes_th2 <- c("ADCY1", "AHI1", "ANK1", "BIRC5", "CDC25C",
  "CDC7", "CENPF", "CXCR6", "DHFR", "EVI5", "GATA3", "GSTA4",
  "HELLS", "IL26", "LAIR2", "LIMA1", "MB", "MICAL2", "NEIL3",
  "PHEX", "PMCH", "PTGIS", "SLC39A14", "SMAD2", "SNRPD1", "WDHD1")

# Select rows in the expression matrix
df_expr_th2 <- df_z_scores_82[rownames(df_z_scores_82) %in% l_genes_th2,
]
df_expr_th2 <- df_expr_th2[, -c(1:2)]

# Apply coxph model
df_expr_th2 <- as.data.frame(t(df_expr_th2))
df_expr_th2 <- df_expr_th2[df_metadata_surv$Patient, ]
df_expr_th2$Patient <- rownames(df_expr_th2)
df_survival_th2 <- merge(df_metadata_surv, df_expr_th2, by = "Patient")
rownames(df_survival_th2) <- df_survival_th2$Patient
df_survival_th2 <- df_survival_th2[, -1]
results_th2 <- apply_coxph_model(df_survival_th2)

# Generate forest plot
# pdf('../results/figures/forest_plots/all_indiv/forest_plot_all_patients_th2.pdf')
generate_forestplot(results_th2, Type = FALSE)

# dev.off()
```

Forest Plot for Cox Model

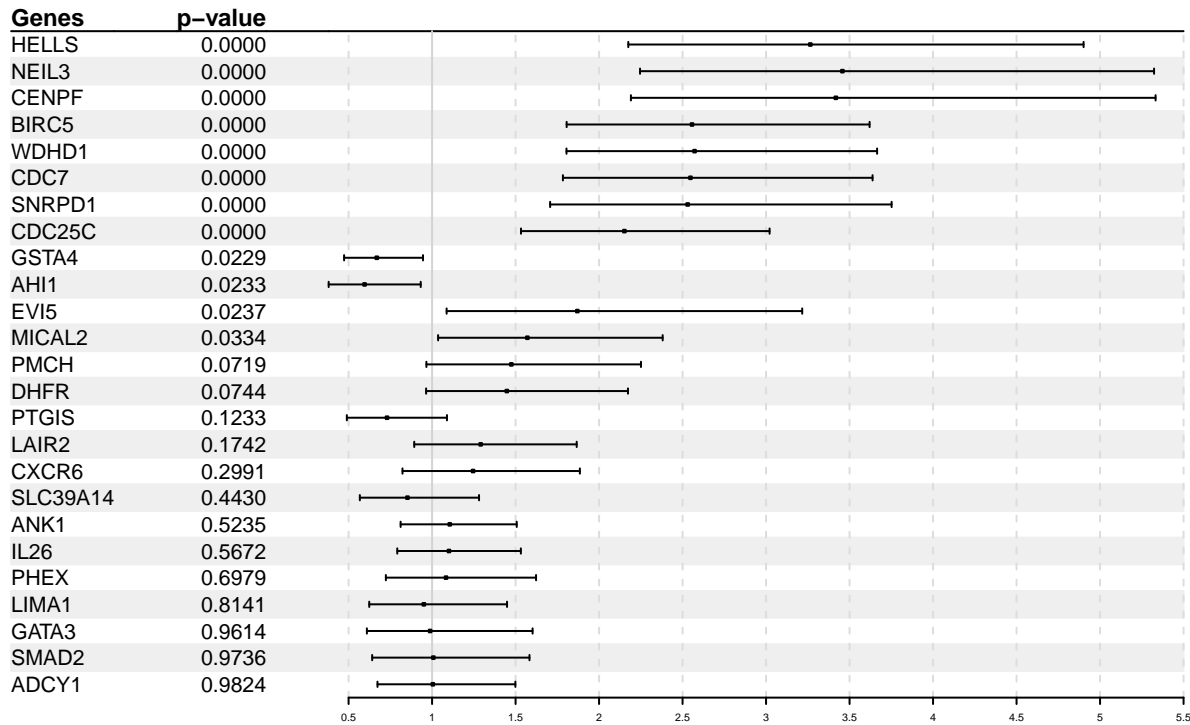


Figure 4: Forest plot for Th2 cells (n = 82)

So, we see that for the 12 significant genes, 10 have a ratio > 2 and only 2 genes have a small ratio. We now can confirm that the presence of Th2 cells has a bad impact on the prognosis.

e- Significant immune cells genes survival analysis

Here, we repeat the previous step for the genes of all the significant immune cells.

```
# List of immune cells type significant in coxph analysis
l <- c("DC", "Eosinophils", "Immune checkpoints", "TFH", "Th2 cells",
      "Treg", "aDC")

# Select the genes for cell types selected
df_selected_imm <- df_CTA_immune_whole_clean_avg_82[df_CTA_immune_whole_clean_avg_82$Signature %in%
  1, ] %>%
  select(Signature)
df_selected_imm$Variable <- rownames(df_selected_imm)
df_selected_imm_expr <- df_z_scores_82[rownames(df_z_scores_82) %in%
  rownames(df_selected_imm), -c(1, 2)]
df_selected_imm_expr <- as.data.frame(t(df_selected_imm_expr))
df_selected_imm_expr$Patient <- rownames(df_selected_imm_expr)

# Merge with metadata
df_selected_imm_expr <- merge(df_metadata_surv[, c("Patient",
  "OS.delay", "OS.event")], df_selected_imm_expr, by = "Patient")
rownames(df_selected_imm_expr) <- df_selected_imm_expr$Patient
```

```

df_selected_imm_expr <- df_selected_imm_expr[, -1]

# Apply coxph model and merge with cel types
results_selected_imm <- apply_coxph_model(df_selected_imm_expr)
results_selected_imm <- merge(results_selected_imm, df_selected_imm,
  by = "Variable")

# Create forest plot
# pdf('../results/figures/forest_plots/all_indiv/forest_plot_selected_imm_82.pdf',
# height = 15, width = 8)
generate_forestplot(results_selected_imm, Type = TRUE)

# dev.off()

```

Forest Plot for Cox Model

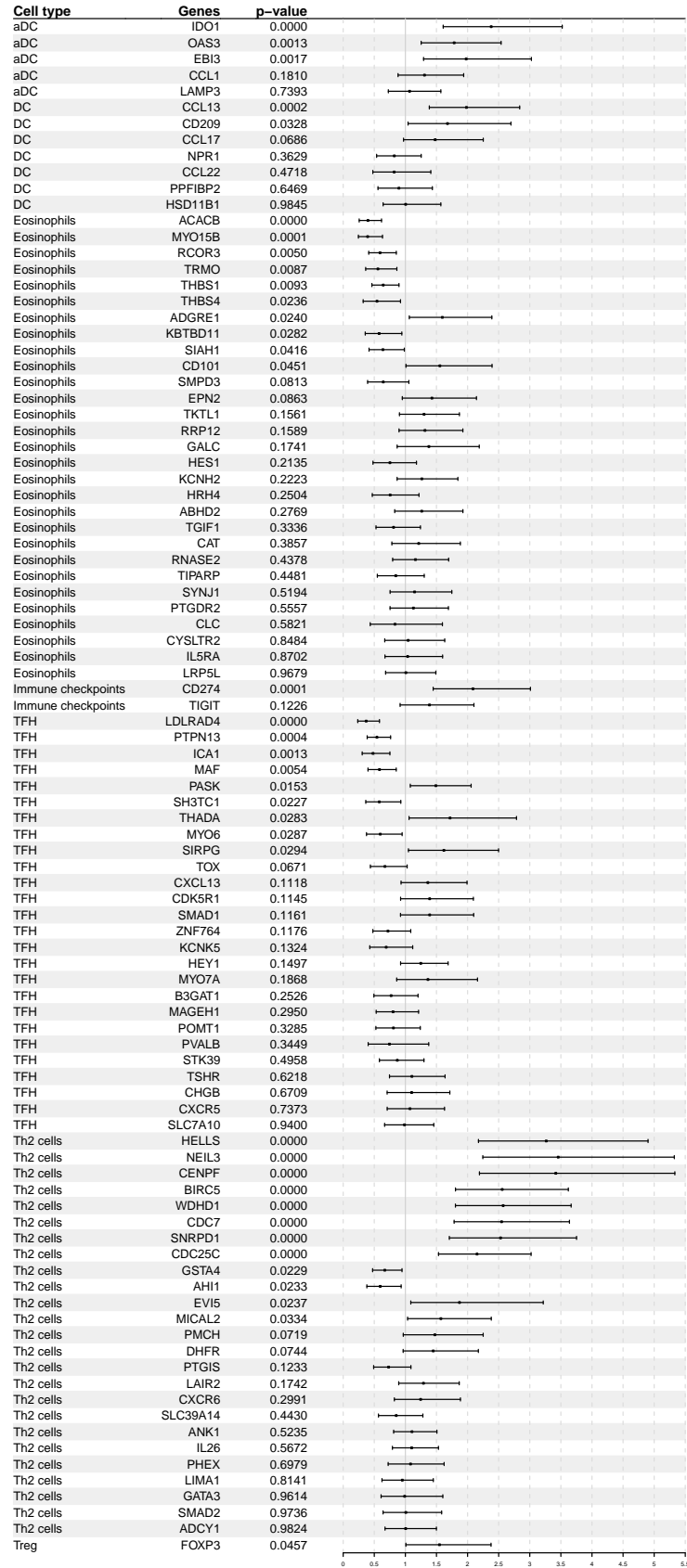


Figure 5: Forest plot for significant immune cells (n = 82)

This forest plot show the impact of each cell types on the survival probabilities. We can see the genes that are drive the results.

2) Conventional chondrosarcomas

This section concern the survival analysis for conventional chondrosarcomas. This is the same stpes than previously.

a- Categorical data

```
# Compute medians for each immune cell types
medians_sign <- apply(df_imm_z_scores_63, 2, median)

# Adding categorical column
df_cat_conv <- df_imm_z_scores_63
for (col in colnames(df_cat_conv)) {
  new_column_name <- paste(col, "cat", sep = " ")
  df_cat_conv[[new_column_name]] <- ifelse(df_imm_z_scores_63[[col]] >
    medians_sign[col], "HIGH", "LOW")
}

# Select categories
df_cat_conv <- df_cat_conv[, grep("cat", colnames(df_cat_conv))]

# Replace space by _
colnames(df_cat_conv) <- gsub(" ", "_", colnames(df_cat_conv))

# Merge df
df_cat_conv$Patient <- rownames(df_cat_conv)
df_survival_conv_cat <- merge(df_metadata_surv_conv, df_cat_conv,
  by = "Patient")
rownames(df_survival_conv_cat) <- df_survival_conv_cat$Patient
df_survival_conv_cat <- df_survival_conv_cat[, -1]

# Apply coxph model and create forest plot
results <- apply_coxph_model(df_survival_conv_cat)
# write.table(results, '../results/results_coxhp.tsv', sep
# = '\\t', row.names = FALSE, quote = FALSE)
generate_forestplot(results, Type = FALSE)
```

Forest Plot for Cox Model

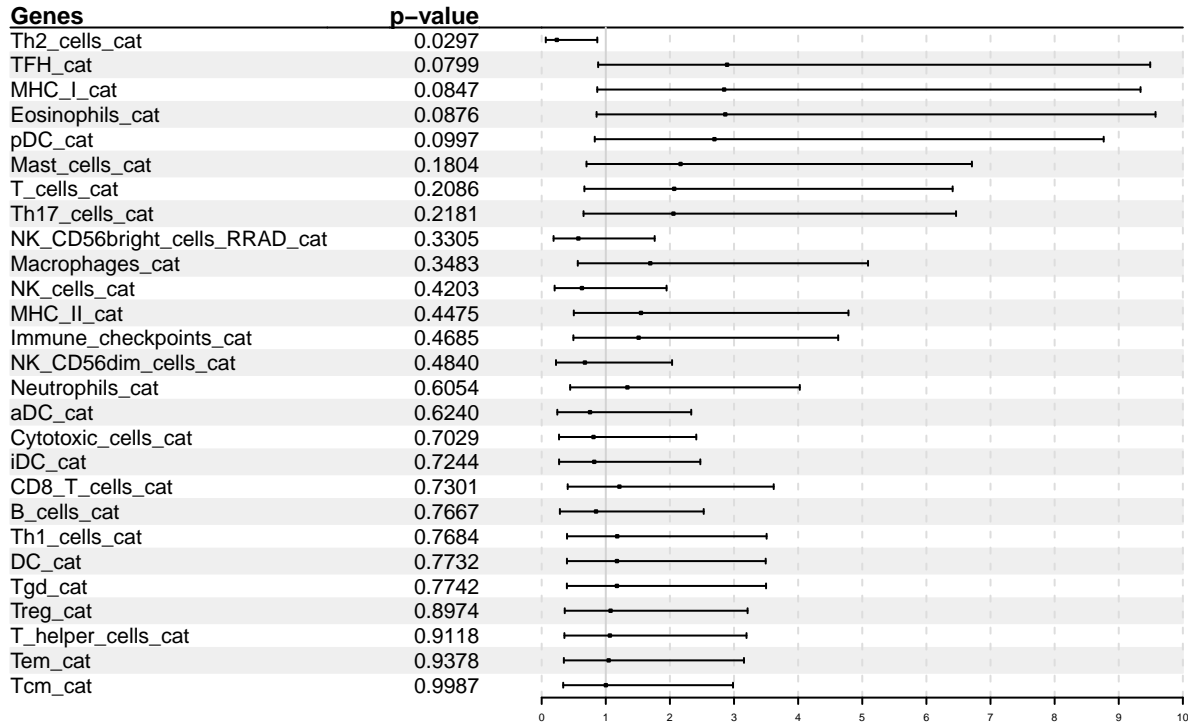


Figure 6: Forest plot for categorical immune cell types expression (n = 63)

By using categorical data and Cox model, there is only Th2 that are significant.

b- Continuous data

```
df_cont_conv <- df_imm_z_scores_63

# Replace space by _
colnames(df_cont_conv) <- gsub(" ", "_", colnames(df_cont_conv))

# Merge df
df_cont_conv$Patient <- rownames(df_cont_conv)
df_survival_conv <- merge(df_metadata_surv_conv, df_cont_conv,
  by = "Patient")
rownames(df_survival_conv) <- df_survival_conv$Patient
df_survival_conv <- df_survival_conv[, -1]

# Apply coxph model and generate plot
results_cont <- apply_coxph_model(df_survival_conv)
# write.table(results_cont,
#   './results/results_coxph_var_cont_patients_conv.tsv',
#   sep = '\t', row.names = FALSE, quote = FALSE)
# pdf('./results/figures/forest_plots/conv_chondro/forest_plot_conv_patients_var_cont.pdf')
generate_forestplot(results_cont, Type = FALSE)
```

Forest Plot for Cox Model

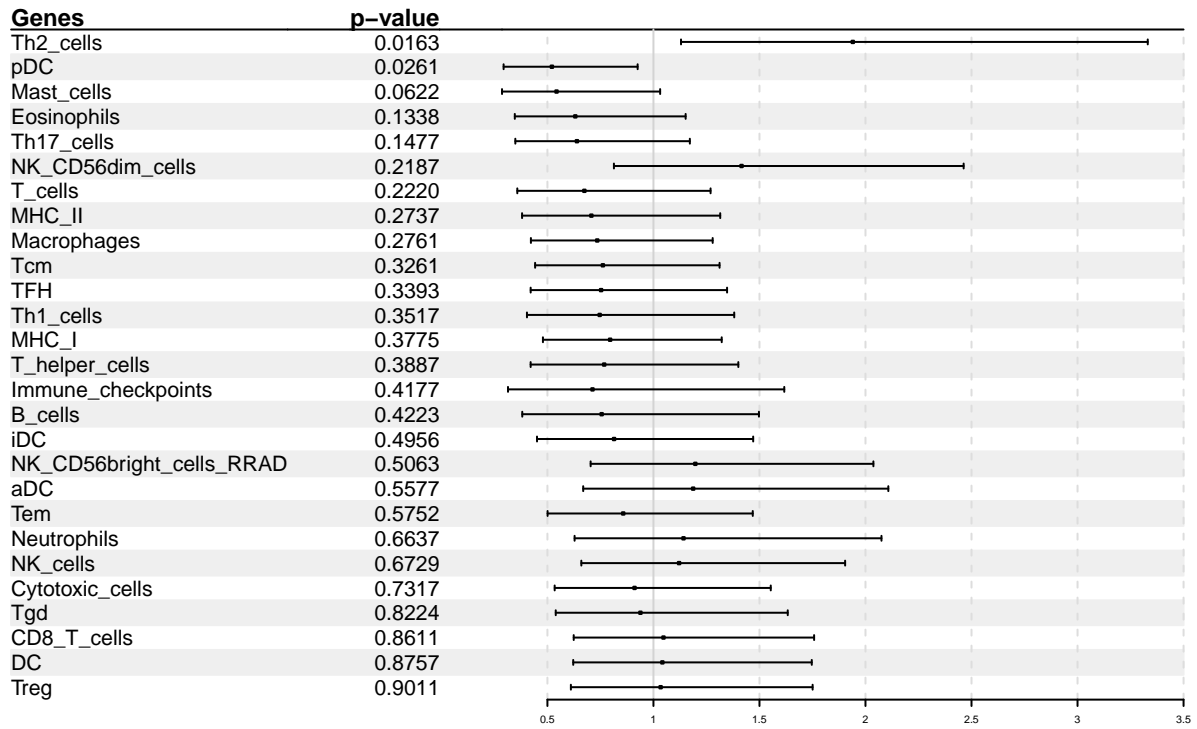


Figure 7: Forest plot for continuous immune cell types expression (n = 63)

```
# dev.off()
```

By using z-scores, pDC are now significant and have a good impact on survival probabilities.

c- Kaplan-Meier plots

This shows the graph for each significant HR ratio more than significant to all patients to observe the differences.

```
# Create Kaplan-Meier plots with categorical data
p1 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ DC_cat, data = df_survival_conv_cat),
  df_survival_conv_cat)
p2 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Eosinophils_cat,
  data = df_survival_conv_cat), df_survival_conv_cat)
p3 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Immune_checkpoints_cat,
  data = df_survival_conv_cat), df_survival_conv_cat)
p4 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ TFH_cat, data = df_survival_conv_cat),
  df_survival_conv_cat)
p5 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Th2_cells_cat,
  data = df_survival_conv_cat), df_survival_conv_cat)
p6 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Treg_cat, data = df_survival_conv_cat),
  df_survival_conv_cat)
p7 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ aDC_cat, data = df_survival_conv_cat),
  df_survival_conv_cat)
p8 <- plot_km(survfit(Surv(OS.delay, OS.event) ~ pDC_cat, data = df_survival_conv_cat),
  df_survival_conv_cat)
# pdf('../results/figures/other_plots/KM_plots_signif_cox_conv_indiv.pdf',
# height = 20, width = 15)
grid.arrange(p1, p2, p3, p4, p5, p6, p7, p8, ncol = 2)

# dev.off()
```

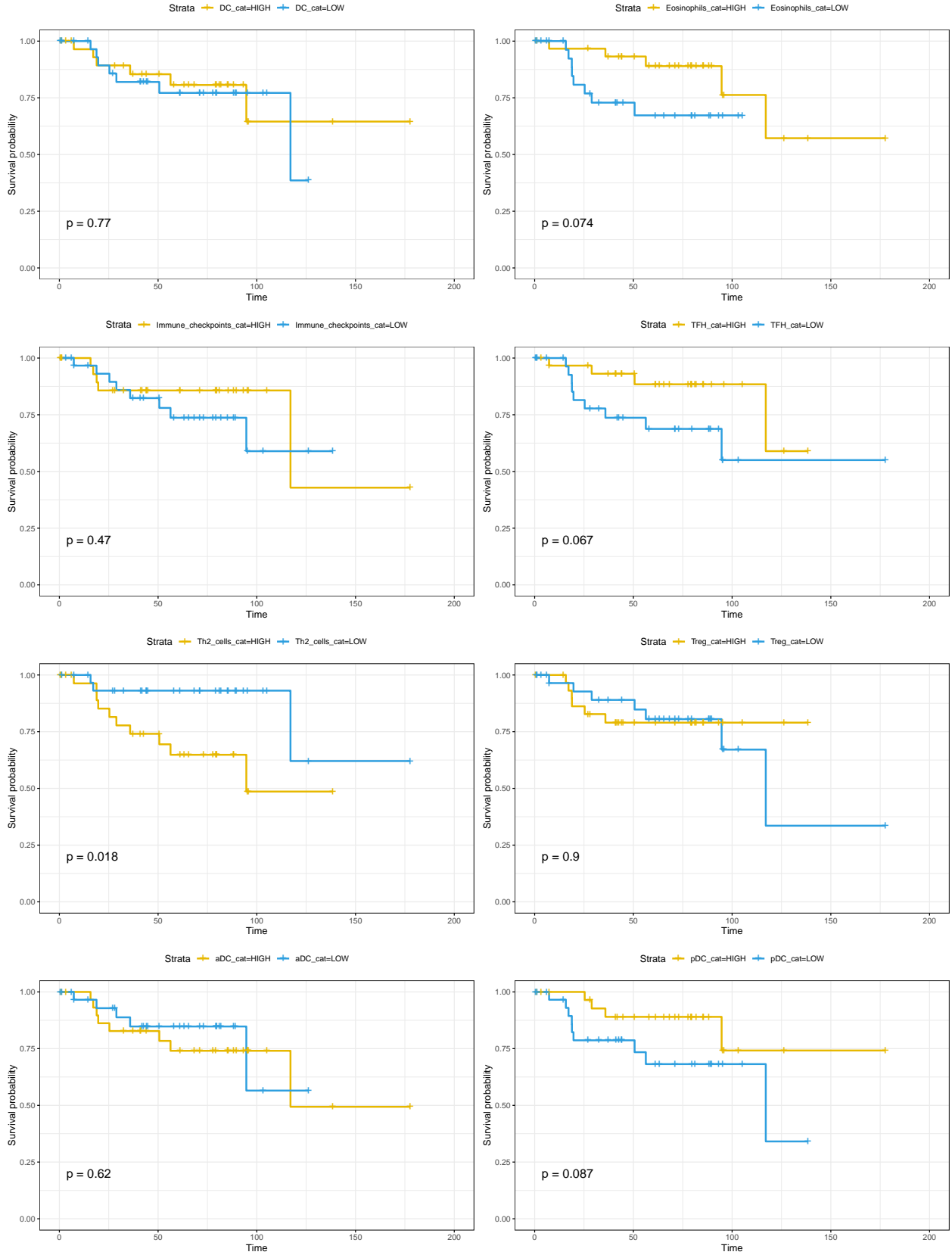


Figure 8: Kaplan-Meier plots for immune cell types ($n = 63$)

With this test, we see that only Th2 cells are significant. So we see that this type may be play an important role in the survival.

d- Survival analysis for Th2 cells

```
# Take conventional tumors
df_survival_th2 <- merge(df_metadata_surv_conv, df_expr_th2,
  by = "Patient")
rownames(df_survival_th2) <- df_survival_th2$Patient
df_survival_th2 <- df_survival_th2[, -1]

# Apply coxph model and create forest plot
results_th2 <- apply_coxph_model(df_survival_th2)
# pdf('../results/figures/forest_plots/conv_chondro/forest_plot_conv_th2.pdf')
generate_forestplot(results_th2, Type = FALSE)
```

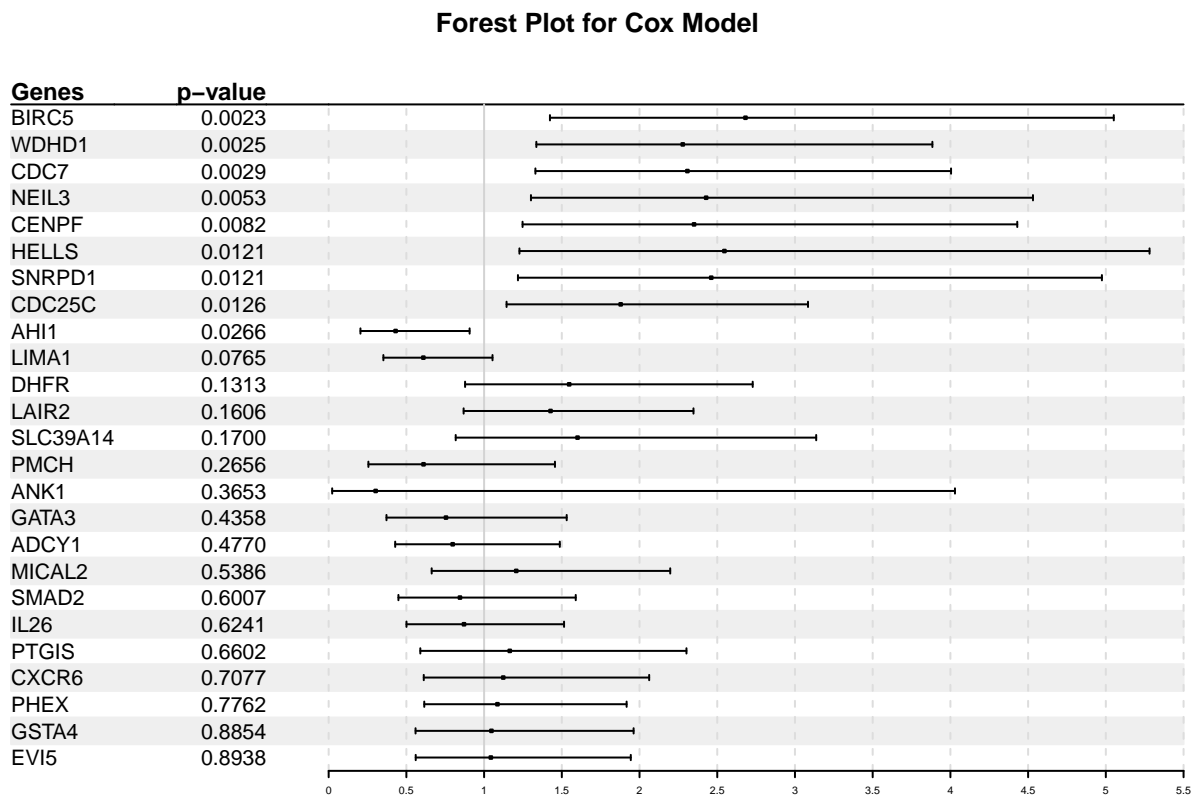


Figure 9: Forest plot for Th2 cells (n = 63)

```
# dev.off()
```

e- Significant immune cells genes survival analysis

```
# List of immune cell types significant
l_th2_pdc <- c("Th2 cells", "pDC")

# Select the genes for cell types selected
df_selected_imm <- df_CTA_immune_whole_clean_avg_63[df_CTA_immune_whole_clean_avg_63$Signature %in%
  l_th2_pdc, ] %>%
  select(Signature)
df_selected_imm$Variable <- rownames(df_selected_imm)
df_selected_imm_expr_conv <- df_z_scores_63[rownames(df_z_scores_63) %in%
  rownames(df_selected_imm), -c(1, 2)]
df_selected_imm_expr_conv <- as.data.frame(t(df_selected_imm_expr_conv))
df_selected_imm_expr_conv$Patient <- rownames(df_selected_imm_expr_conv)

# Merge with metadata
df_selected_imm_expr_conv <- merge(df_metadata_surv_conv[, c("Patient",
  "OS.delay", "OS.event")], df_selected_imm_expr_conv, by = "Patient")
rownames(df_selected_imm_expr_conv) <- df_selected_imm_expr_conv$Patient
df_selected_imm_expr_conv <- df_selected_imm_expr_conv[, -1]

# Apply coxph model and create forest plot
results_selected_imm_conv <- apply_coxph_model(df_selected_imm_expr_conv)
results_selected_imm_conv <- merge(results_selected_imm_conv,
  df_selected_imm, by = "Variable")
results_selected_imm_conv <- results_selected_imm_conv[order(results_selected_imm_conv$Signature,
  results_selected_imm_conv$Pvalue), ]
# pdf('./results/figures/forest_plots/conv_chondro/forest_plot_selected_imm_conv_indiv.pdf',
# height = 8, width = 8)
generate_forestplot(results_selected_imm_conv, Type = TRUE)

# dev.off()
```

Forest Plot for Cox Model

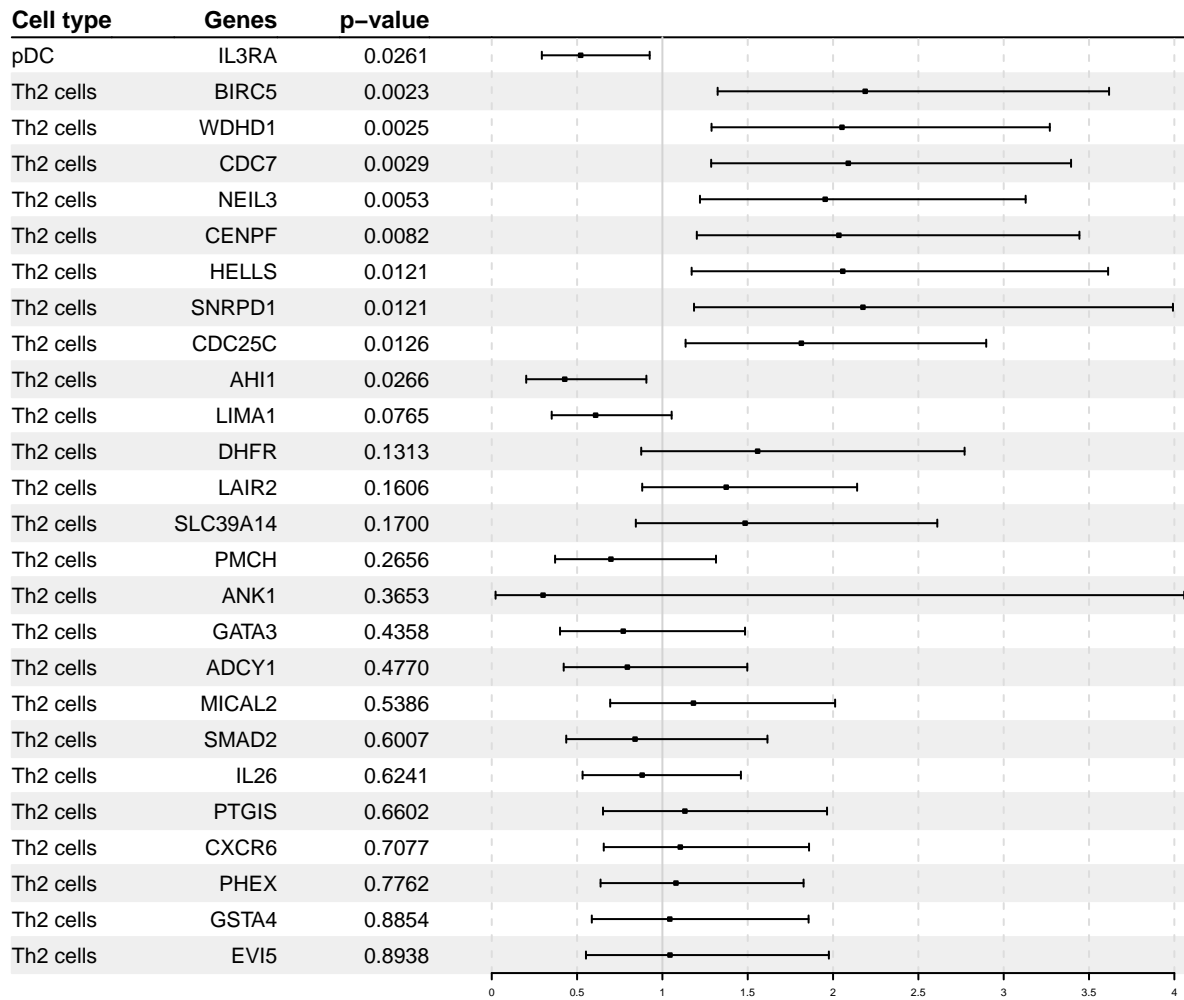


Figure 10: Forest plot for significant immune cells (n = 63)

II. CTA survival analysis

This section analyzes the impact of CTA on survival probabilities.

1) All chondrosarcoma types

This part concerns all the patients.

```
# CTA data
df_expr_cta_82 <- subset(df_z_scores_82, CTA == "CTA")
df_expr_cta_82 <- df_expr_cta_82[, !colnames(df_expr_cta_82) %in%
  c("Signature", "CTA", "SYMBOL")]
df_expr_cta_82 <- as.data.frame(t(df_expr_cta_82))
df_expr_cta_82$Patient <- rownames(df_expr_cta_82)
df_expr_cta_82 <- merge(df_metadata_surv, df_expr_cta_82, by = "Patient")
rownames(df_expr_cta_82) <- df_expr_cta_82$Patient
df_expr_cta_82 <- df_expr_cta_82[, -1]

res_cta <- apply_coxph_model(df_expr_cta_82)
# write.table(res_cta,
#   '../results/results_coxph_cta_all_indiv.tsv', sep = '\t',
#   row.names = FALSE, quote = FALSE)

# Forest plot on significative CTA
res_cta_signif <- res_cta[res_cta$Pvalue < 0.05, ]
# pdf('../results/figures/forest_plots/all_indiv/forest_plot__var_cont_cta_zscores_82.pdf',
#   width = 10, height = 30)
generate_forestplot(res_cta_signif, Type = FALSE)

# dev.off()
```

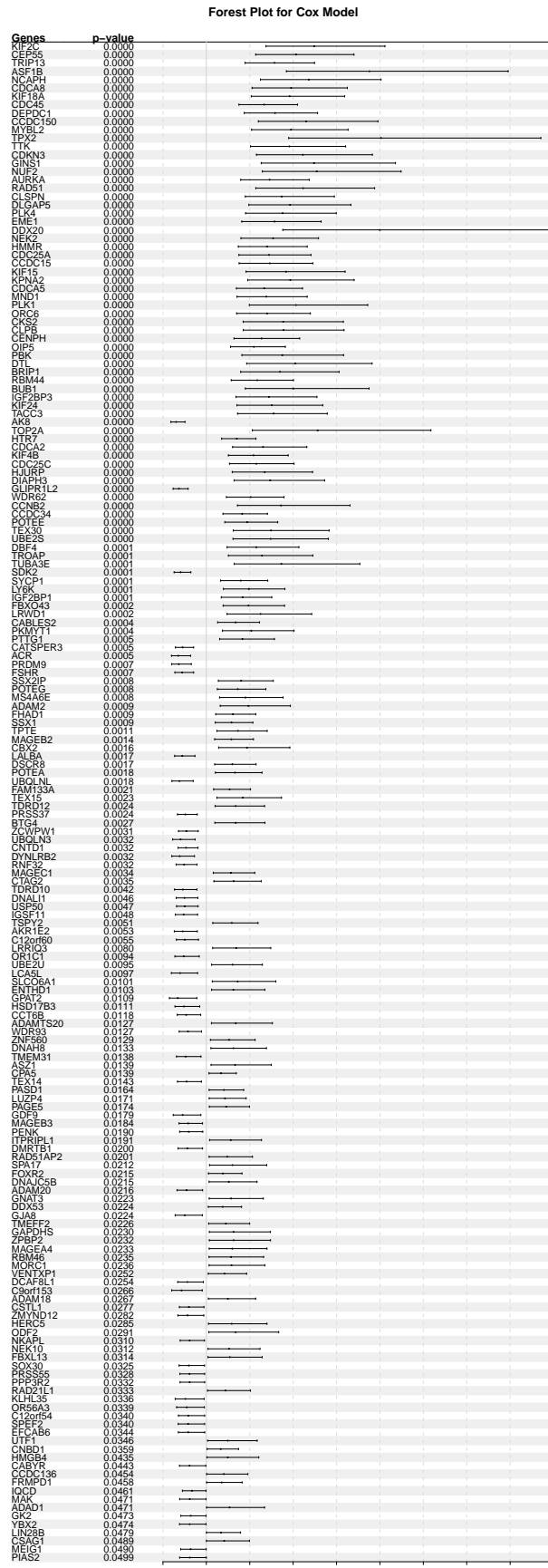


Figure 11: Forest plot for significant CTA (n = 82)

This forest plot illustrates only the significant HR. We see here that there is more bad impact CTA than good impact. We uses this CTA list in script 5 to analyze expression and make a clustering. This results illustrates that there is groups so we make survival analysis to see the difference between group (see III.)

a- Intensities jitter plot for significant CTA

In this section, we want to see the intensities per significant CTA to visualize.

```
# Read intensities
df <- df_CTA_immune_whole_clean_avg_82[, -c(1, 2)]
df <- df[rownames(df) %in% res_cta_signif[, 1], ]
df$SYMBOL <- rownames(df)

# Cluster data
cluster_data <- read.table("../results/clusters_indiv/clusters_cta_signif_coxph_all_indiv.tsv",
  sep = "\t", header = TRUE)
cluster_data$Cluster <- as.character(cluster_data$Cluster)

# Data transformation for the plot
df_long <- pivot_longer(df, cols = -SYMBOL, values_to = "value")

# Merge
df_long <- pivot_longer(df, cols = -SYMBOL, values_to = "value")
df_long <- merge(df_long, cluster_data, by = "SYMBOL")

# pdf(file =
# '../results/figures/other_plots/plot_cta_coxph_signif_all_indiv.pdf',
# height = 10, width = 50)
ggplot(df_long, aes(x = reorder(SYMBOL, -value), y = value, color = as.factor(Cluster))) +
  geom_jitter(width = 0.2, height = 0, size = 1, alpha = 0.3) +
  geom_boxplot(alpha = 0.4, outliers = FALSE, colour = "black",
    fill = NA) + labs(title = "Plot of intensities per significant coxph CTA",
  x = "CTA", y = "Intensities") + scale_color_manual(values = c(`1` = "#EA4343",
  `2` = "blue", `3` = "#75E05A")) + theme(axis.text.x = element_text(angle = 90,
  hjust = 1))
```

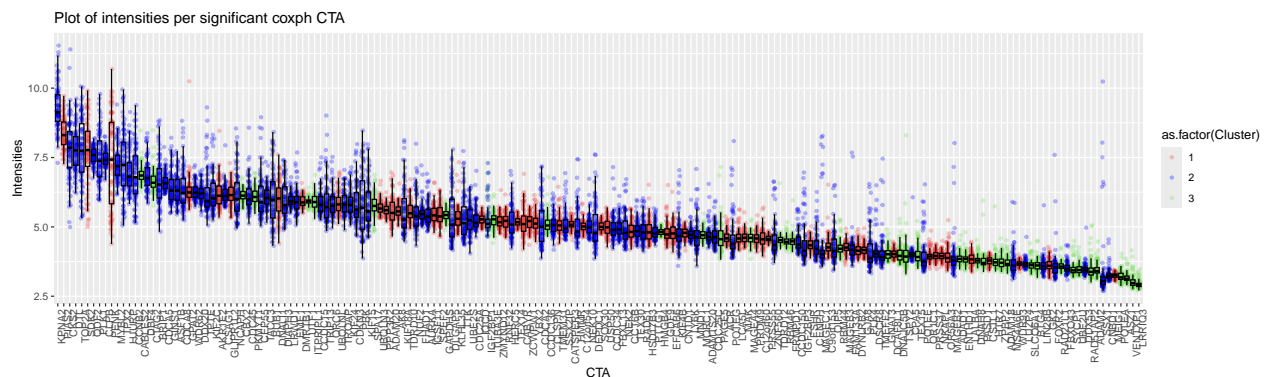


Figure 12: Intensities jitter plot for significant CTA (n = 82)

```
# dev.off()
```

```

# Cluster data
cluster_data <- read.table("../results/clusters_indiv/clusters_cta_signif_coxph_all_indiv.tsv", sep = "\t")
cluster_data$Cluster <- as.character(cluster_data$Cluster)

# Merge df_long avec les clusters
df_long <- pivot_longer(df,
                        cols = -SYMBOL,
                        values_to = "value")

# Ajouter l'information du cluster dans df_long
df_long <- merge(df_long, cluster_data, by = "SYMBOL")

# Maintenant on peut faire le graphique avec les couleurs par cluster
ggplot(df_long, aes(x = reorder(SYMBOL, -value), y = value, color = as.factor(Cluster))) +
  geom_jitter(width = 0.2, height = 0, size = 1, alpha = 0.3) + # Ajouter la couleur par cluster
  geom_boxplot(alpha = 0.4, outliers = FALSE, colour = "black", fill = NA) + # Boîtes sans remplissage
  labs(title = "Plot of intensities per significant coxph CTA", x = "CTA", y = "Intensities") +
  scale_color_manual(values = c("1" = "#EA4343", "2" = "blue", "3" = "#75E05A")) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))

```

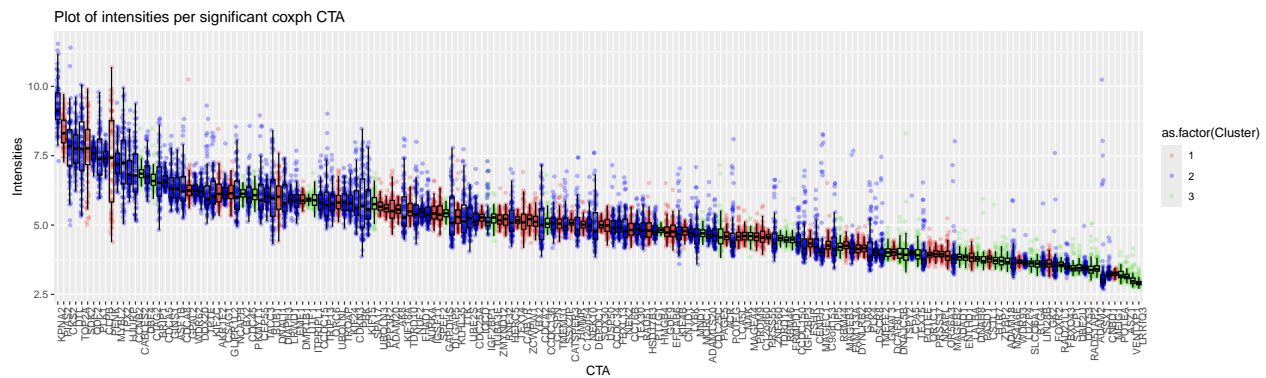


Figure 13: Intensities jitter plot for significant CTA (n = 82)

b- Normalized intensities for CTA to see the differences

Here, it is a test of survival analysis with normalized data and not z-scores to see if there are differences.

```
df_int <- df_CTA_immune_whole_clean_avg_82
df_int <- subset(df_int, CTA == "CTA")
df_int <- df_int[, !colnames(df_int) %in% c("Signature", "CTA",
      "SYMBOL")]
df_int <- as.data.frame(t(df_int))
df_int <- df_int[df_metadata_surv$Patient, ]
df_int$Patient <- rownames(df_int)
df_survival_int_cta <- merge(df_metadata_surv, df_int, by = "Patient")
rownames(df_survival_int_cta) <- df_survival_int_cta$Patient
df_survival_int_cta <- df_survival_int_cta[, -1]
res_cta_int <- apply_coxph_model(df_survival_int_cta)
res_cta_signif_int <- res_cta_int[res_cta_int$Pvalue < 0.05,
  ]
head(res_cta_signif_int)
```

##	Variable	HR	LowerCI	UpperCI	Pvalue
## 4	ACR	0.01276281	0.001090762	0.1493354	0.0005104523
## 14	ADAD1	7.75991984	1.026307431	58.6728246	0.0471308847
## 16	ADAM18	6.51130221	1.241794511	34.1417651	0.0266833243
## 17	ADAM2	15.85474040	3.102621206	81.0194917	0.0008989017
## 18	ADAM20	0.29768624	0.105880426	0.8369545	0.0215944351
## 22	ADAMTS20	4.33091460	1.367858121	13.7125489	0.0126789718

We see that with the intensities data that the p-val are the same as continuous data (in z-scores) but the HR are less clean than z score data.

2) Conventional chondrosarcomas

This section like part I concerns conventional chondrosarcomas

```
# CTA data CTA data
df_expr_cta_63 <- subset(df_z_scores_63, CTA == "CTA")
df_expr_cta_63 <- df_expr_cta_63[, !colnames(df_expr_cta_63) %in%
  c("Signature", "CTA", "SYMBOL")]
df_expr_cta_63 <- as.data.frame(t(df_expr_cta_63))
df_expr_cta_63$Patient <- rownames(df_expr_cta_63)
df_expr_cta_63 <- merge(df_metadata_surv, df_expr_cta_63, by = "Patient")
rownames(df_expr_cta_63) <- df_expr_cta_63$Patient
df_expr_cta_63 <- df_expr_cta_63[, -1]

# Apply coxph and generate forest plot with significant
# data
res_cta <- apply_coxph_model(df_expr_cta_63)
# write.table(res_cta,
#   '../results/results_coxph_cta_conv_indiv.tsv', sep =
#   # '\t', row.names = FALSE, quote = FALSE)

res_cta_signif <- res_cta[res_cta$Pvalue < 0.05, ]
```

```
# pdf('../results/figures/forest_plots/conv_chondro/forest_plot_var_cont_cta_zscores_63.pdf',  
# width = 10, height = 30)  
generate_forestplot(res_cta_signif, Type = FALSE)  
  
# dev.off()
```

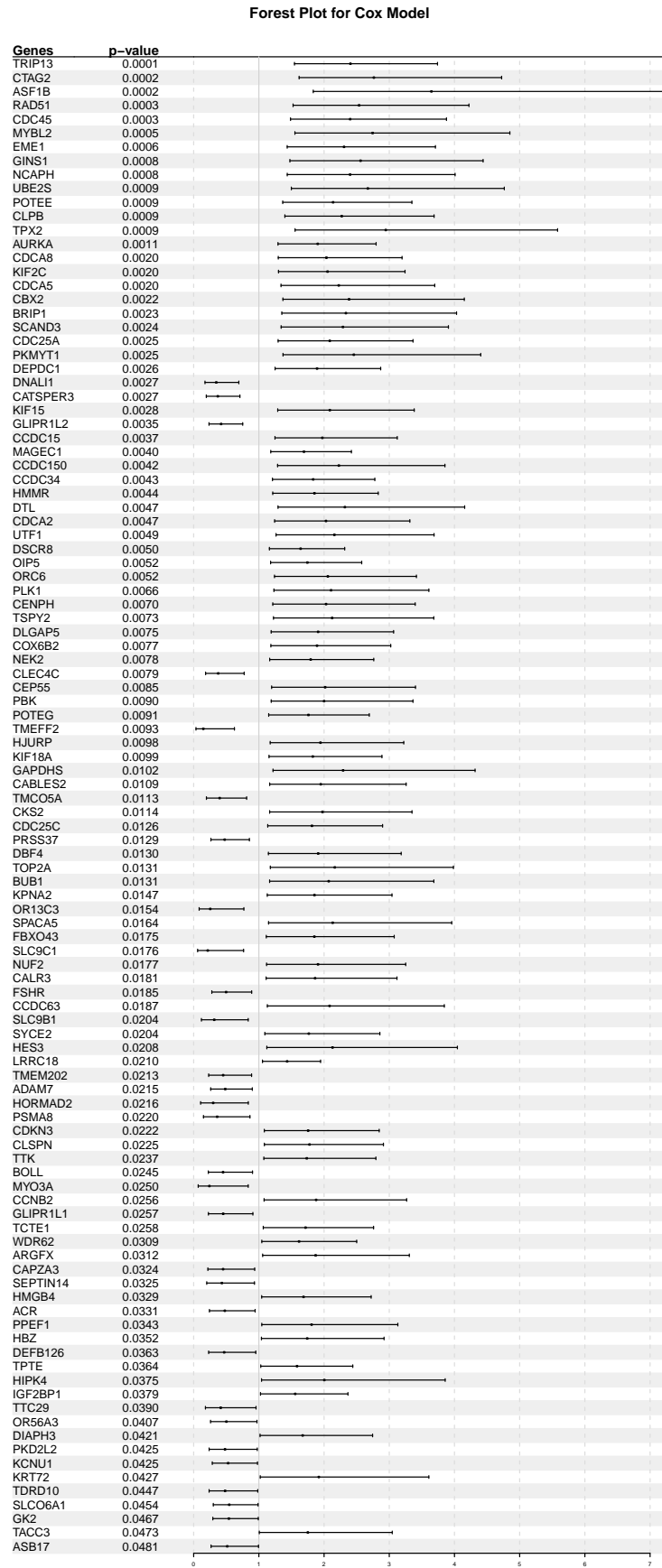


Figure 14: Forest plot for significant CTA (n = 63)

Intensities jitter plot for significant CTA

Plot of intensities per significant coxph CTA

Intensities

CTA

as.factor(Cluster)

- 1
- 2
- 3

```
# dev.off()
```

III. Survival analysis for clusters from CTA heatmaps

This section use clustering results from script 5 on expression analysis.

1) All chondrosarcoma types

This part compute with all the patients.

a- Clustering Kmeans k = 4 with all chondrosarcoma types

The clustering corresponds to the figure 32.

```
# Read clusters
l_anno_all <- read.table("../results/clusters_indiv/clusters_conv_indiv_mhc.tsv",
  header = TRUE, sep = "\t")
df_cluster_all <- merge(l_anno_all, df_metadata_surv, by = "Patient")
rownames(df_cluster_all) <- df_cluster_all$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_all),
  df_cluster_all)
p
```

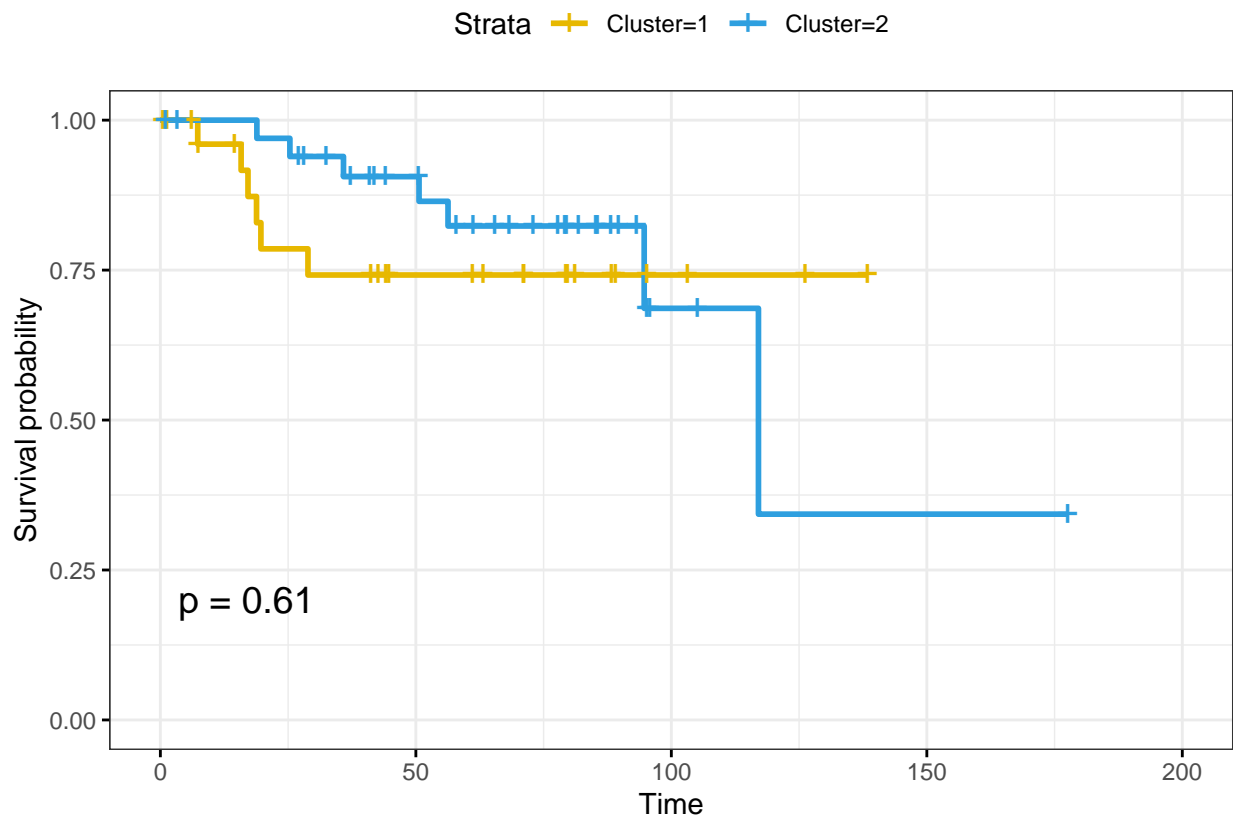


Figure 16: Kaplan-Meier plot with clusters from figure 31 (script 5) (n = 82)

We see that there is no significant differences between the 4 clusters.

b- Significant coxph CTA genes clustering

This clustering is from heatmap 2.

```
# Read clusters
l_cta_all <- read.table("../results/clusters_indiv/clusters_all_indiv_mhc_cta_signif_coxph.tsv",
  header = TRUE, sep = "\t")

df_cluster_cta_all <- merge(l_cta_all, df_metadata_surv, by = "Patient")
rownames(df_cluster_cta_all) <- df_cluster_cta_all$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_cta_all),
  df_cluster_cta_all)
p
```

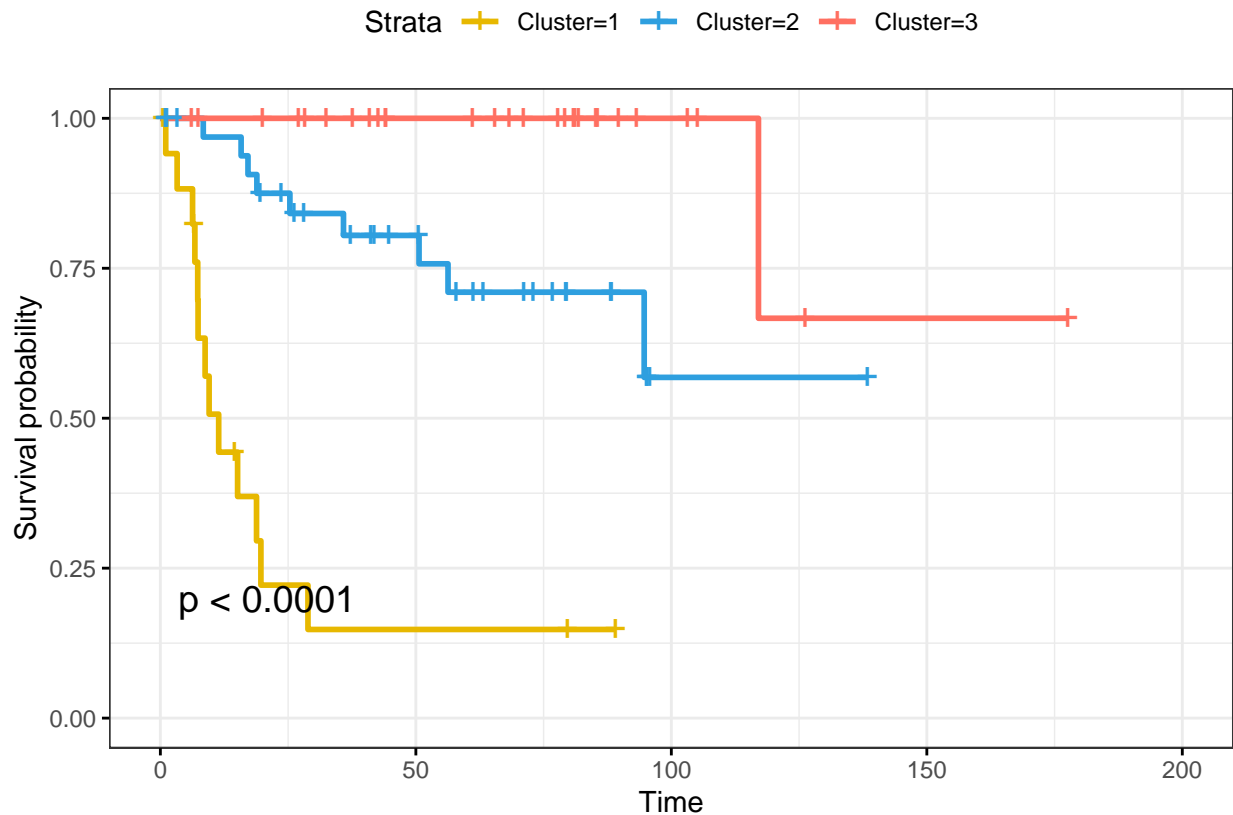


Figure 17: Kaplan-Meier plot with clusters from figure 3 (script 5) (n = 82)

Here we see that the C3 have bad survival and this cluster correspond to an over expression of some CTA. So, we can sat that these CTA could be markers.

c- Selected significant coxph CTA genes clustering

```
# Prepare table
l <- read.table("../results/clusters_indiv/clusters_indiv_signif_selected_cta_all.tsv",
  sep = "\t", header = TRUE)

df_cluster_all <- merge(l, df_metadata_surv, by = "Patient")
rownames(df_cluster_all) <- df_cluster_all$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_all),
  df_cluster_all)
p
```

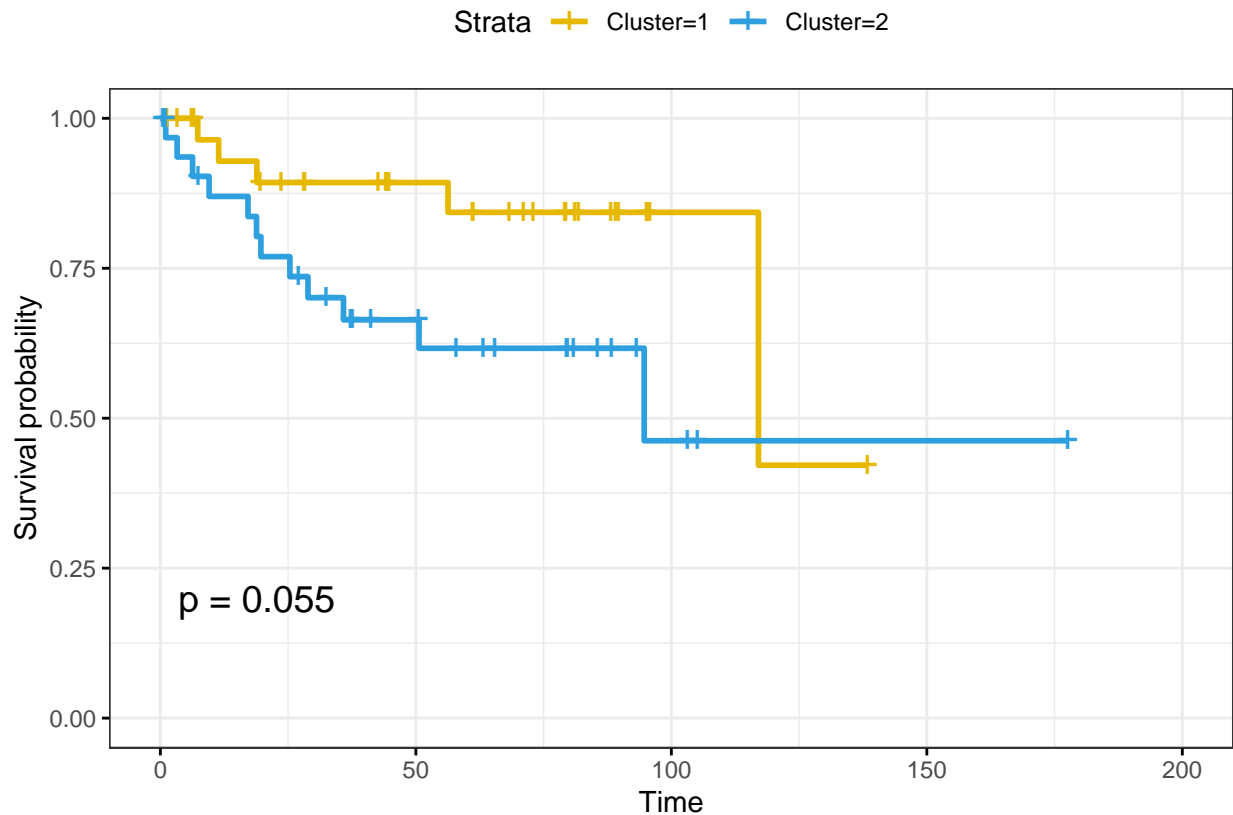


Figure 18: Kaplan-Meier plot with clusters from figure 4 (script 5) (n = 82)

d- Non selected significant coxph CTA genes clustering

```
# Prepare table
l <- read.table("../results/clusters_indiv/clusters_indiv_signif_non_selected_cta_all.tsv",
  sep = "\t", header = TRUE)

df_cluster_all <- merge(l, df_metadata_surv_conv, by = "Patient")
rownames(df_cluster_all) <- df_cluster_all$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_all),
  df_cluster_all)
p
```

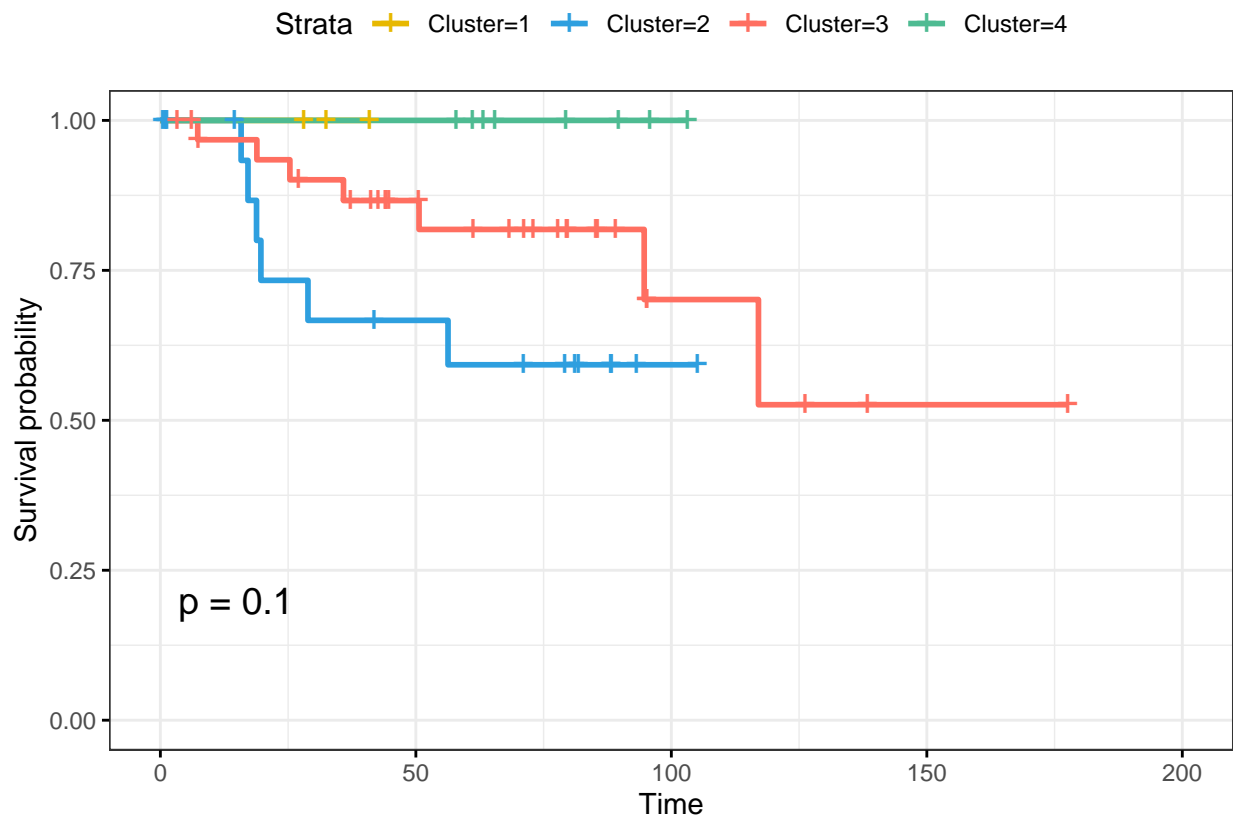


Figure 19: Kaplan-Meier plot with clusters from figure 5 (script 5) (n = 82)

```

# Prepare table
df_cluster_conv_merge <- merge(1, df_metadata_surv_conv, by = "Patient")
df_cluster_conv_merge$Cluster <- ifelse(df_cluster_conv_merge$Cluster ==
  2, 1, df_cluster_conv_merge$Cluster)
df_cluster_conv_merge$Cluster <- ifelse(df_cluster_conv_merge$Cluster ==
  4, 3, df_cluster_conv_merge$Cluster)
df_cluster_conv_merge$Cluster <- ifelse(df_cluster_conv_merge$Cluster ==
  3, 2, df_cluster_conv_merge$Cluster)
rownames(df_cluster_conv_merge) <- df_cluster_conv_merge$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_conv_merge),
  df_cluster_conv_merge)
p

```

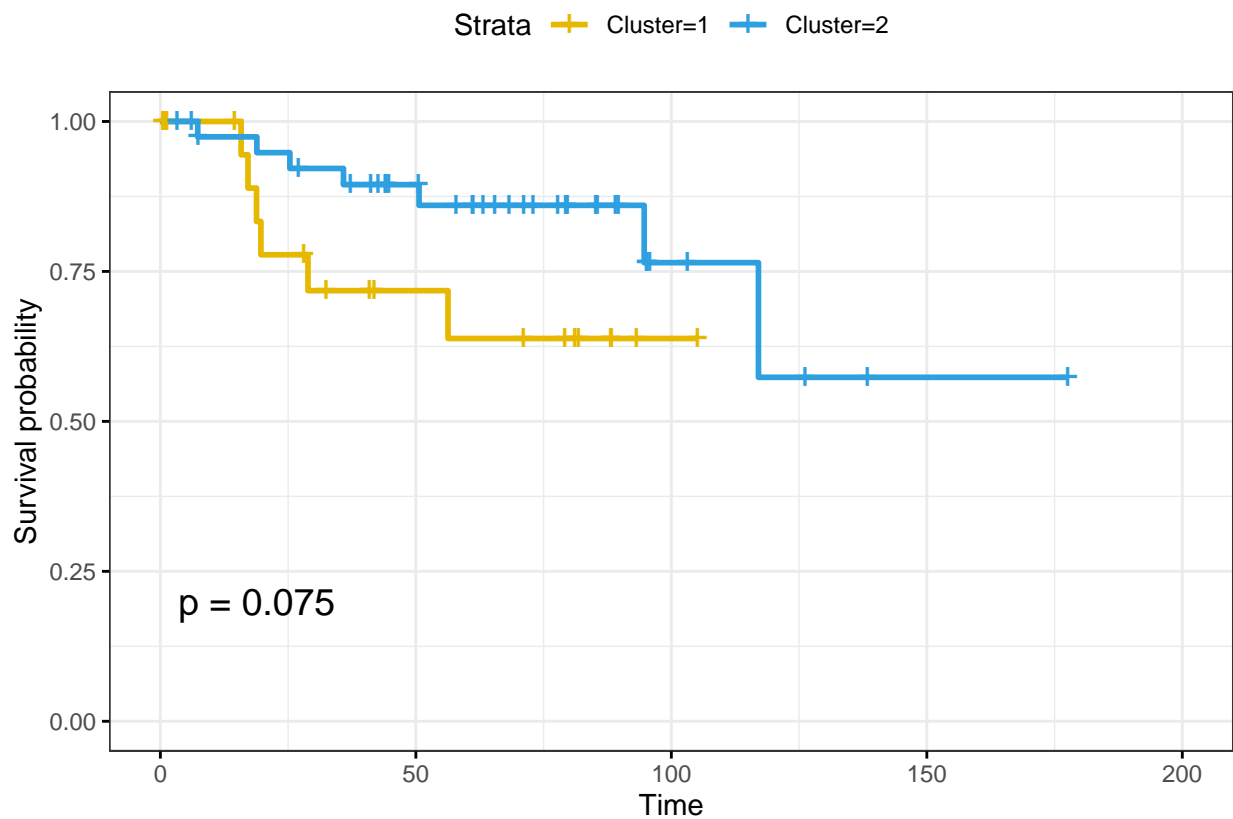


Figure 20: Kaplan-Meier plot merged clusters from figure 5 (script 5) (n = 82)

Merge C1 with C2 and C3 with C4

2) Conventional chondrosarcomas

This part concerns conventional types.

a- Clustering Kmeans $k = 2$

This clustering is from figure 33.

```
# Read clusters
l_anno_conv <- read.table("../results/clusters_indiv/clusters_conv_indiv_mhc.tsv",
  header = TRUE, sep = "\t")
df_cluster_conv <- merge(l_anno_conv, df_metadata_surv_conv,
  by = "Patient")
rownames(df_cluster_conv) <- df_cluster_conv$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_conv),
  df_cluster_conv)
p
```

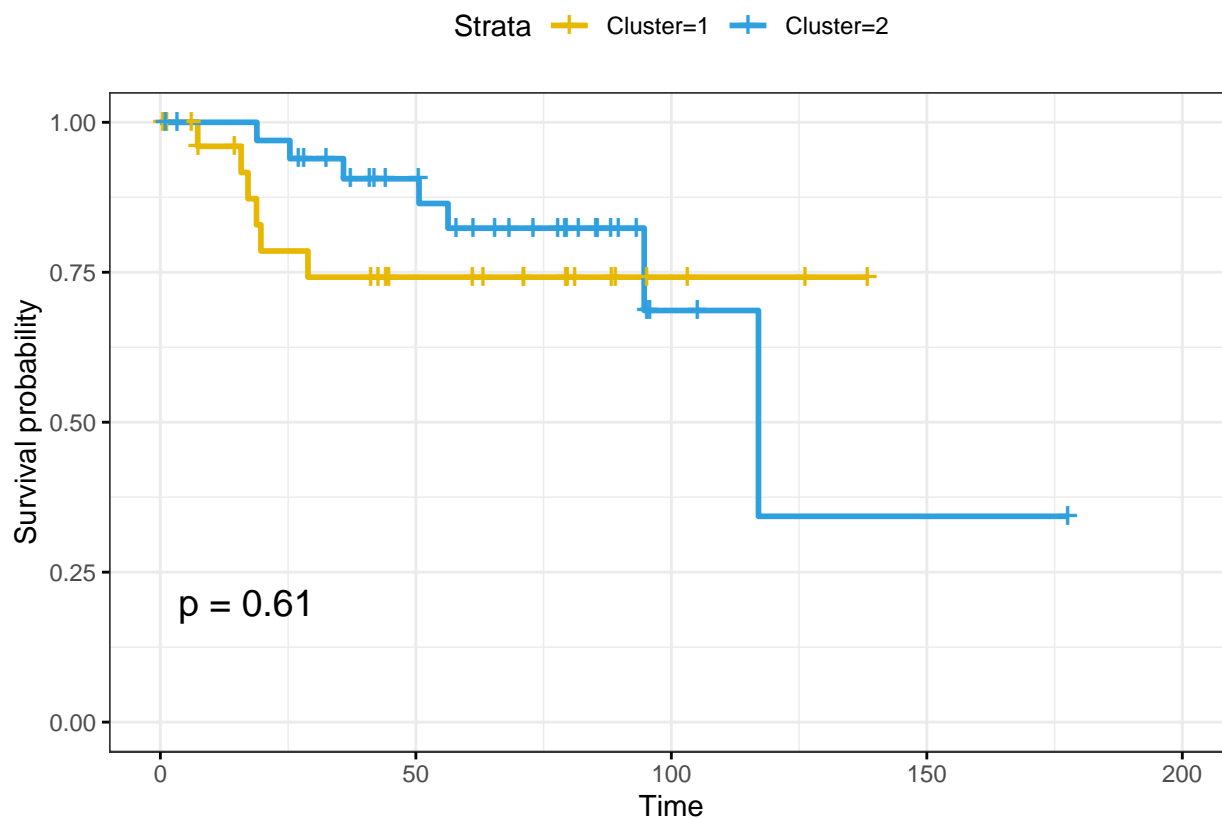


Figure 21: Kaplan-Meier plot with clusters from figure 31 (script 5) ($n = 63$)

We see that there is not differences.

b- Significant coxph CTA genes clustering

This clustering is from heatmap 3.

```
# Read clusters
l_cta_conv <- read.table("../results/clusters_indiv/clusters_conv_indiv_mhc_cta_signif_coxph.tsv",
  header = TRUE, sep = "\t")

# Prepare table
df_cluster_cta_all <- merge(l_cta_conv, df_metadata_surv, by = "Patient")
rownames(df_cluster_cta_all) <- df_cluster_cta_all$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_cta_all),
  df_cluster_cta_all)
p
```

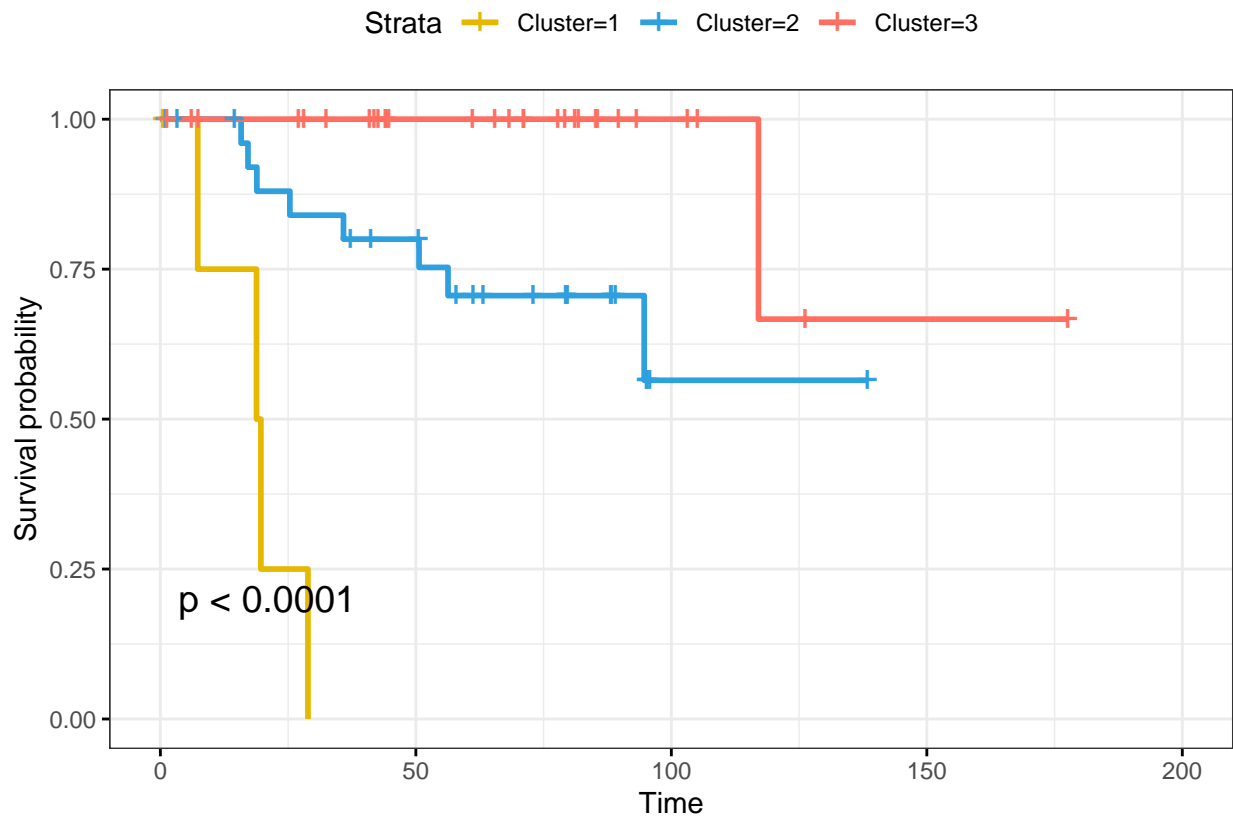


Figure 22: Kaplan-Meier plot with clusters from figure 6 (script 5) (n = 63)

This confirm the KM plot with all patients. We clearly see that some CTA have bad impact on survival.

Merge C1 and C2 Here, we want to merge the 2 clusters to see the differences than with 3.

```
# Prepare table
df_cluster_cta_all_merge <- merge(l_cta_conv, df_metadata_surv,
  by = "Patient")
df_cluster_cta_all_merge$Cluster <- ifelse(df_cluster_cta_all_merge$Cluster ==
  2, 1, df_cluster_cta_all_merge$Cluster)
rownames(df_cluster_cta_all_merge) <- df_cluster_cta_all_merge$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_cta_all_merge),
  df_cluster_cta_all_merge)
p
```

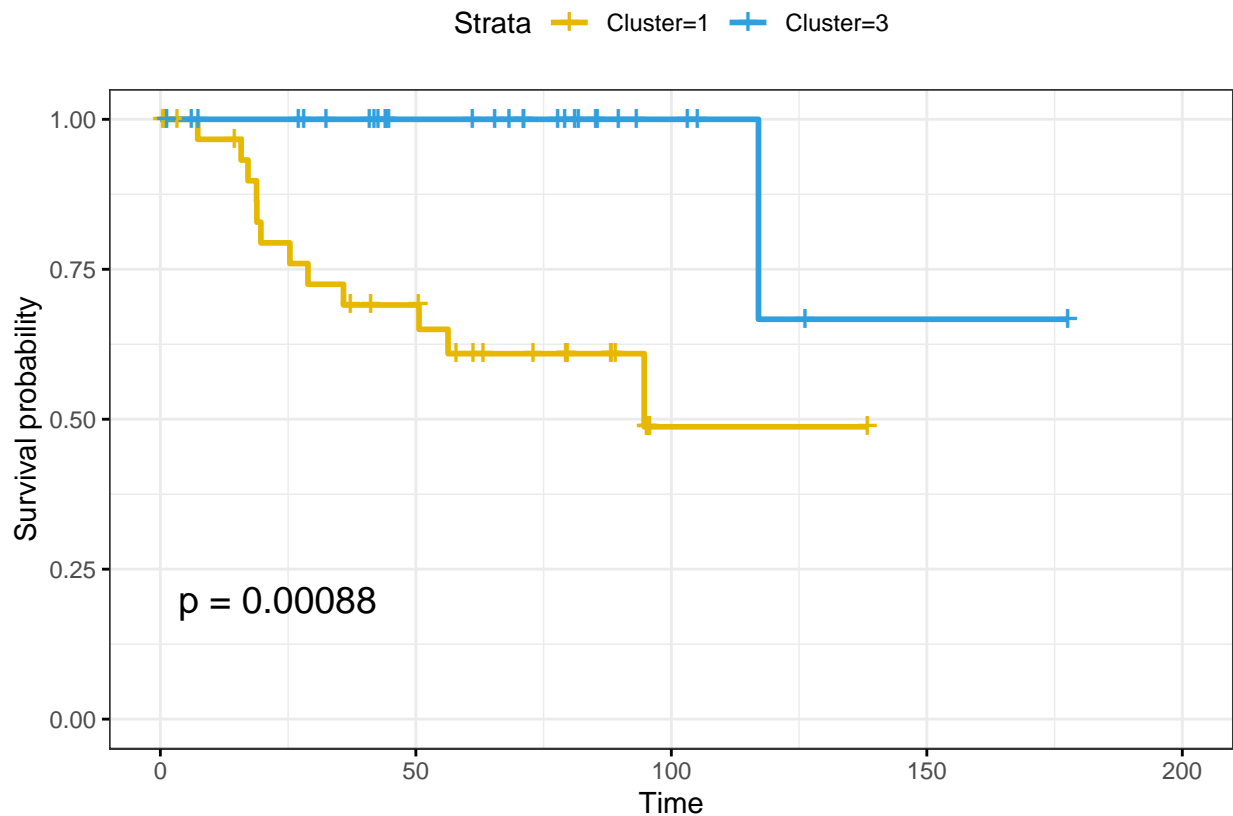


Figure 23: Kaplan-Meier plot with merged clusters from figure 6 (n = 63)

We see that there is always significant.

c- Selected significant coxph CTA genes clustering

```
# Prepare table
l <- read.table("../results/clusters_indiv/clusters_indiv_signif_selected_cta_conv.tsv",
  sep = "\t", header = TRUE)

df_cluster_conv <- merge(l, df_metadata_surv_conv, by = "Patient")
rownames(df_cluster_conv) <- df_cluster_conv$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_conv),
  df_cluster_conv)
p
```

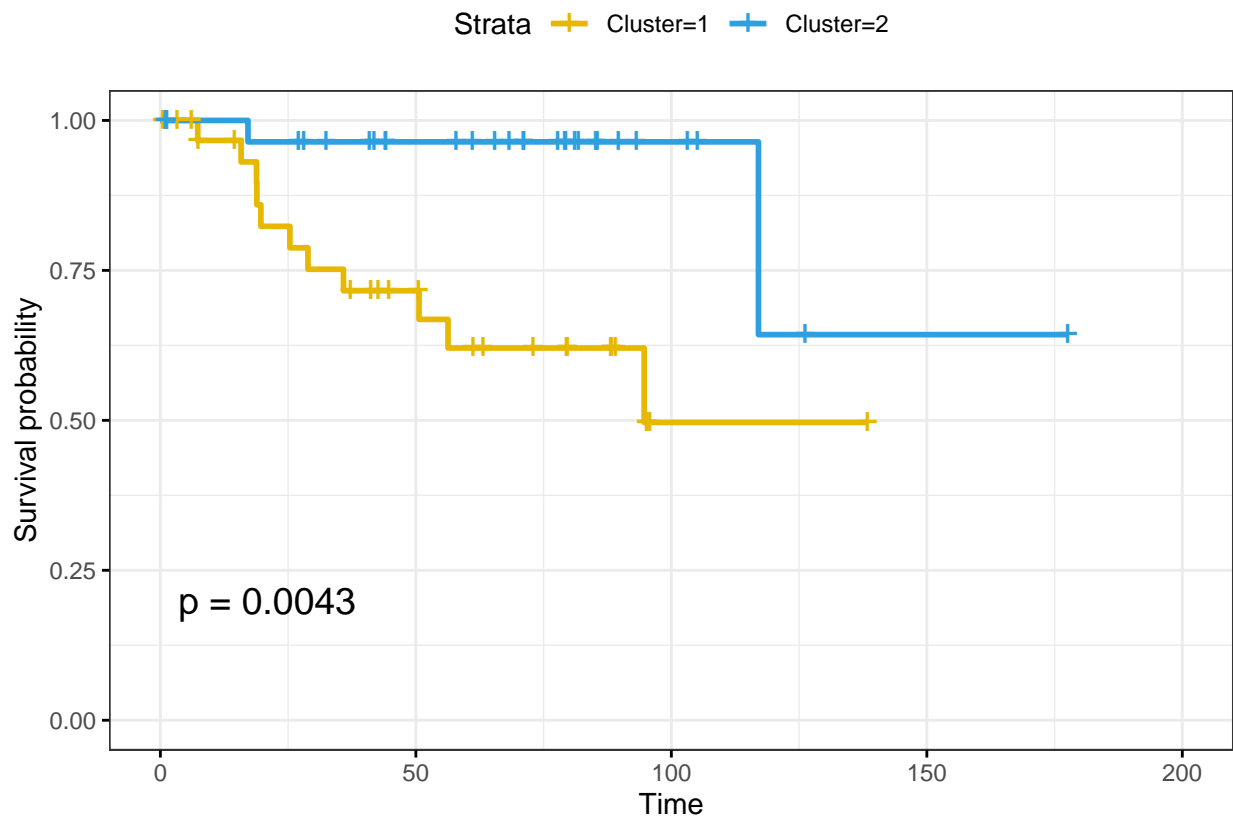


Figure 24: Kaplan-Meier plot with clusters from figure 7 (script 5) (n = 63)

d- Non selected significant coxph CTA genes clustering

```
# Prepare table
l <- read.table("../results/clusters_indiv/clusters_indiv_signif_non_selected_cta_conv.tsv",
  sep = "\t", header = TRUE)

df_cluster_conv <- merge(l, df_metadata_surv_conv, by = "Patient")
rownames(df_cluster_conv) <- df_cluster_conv$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_conv),
  df_cluster_conv)
p
```

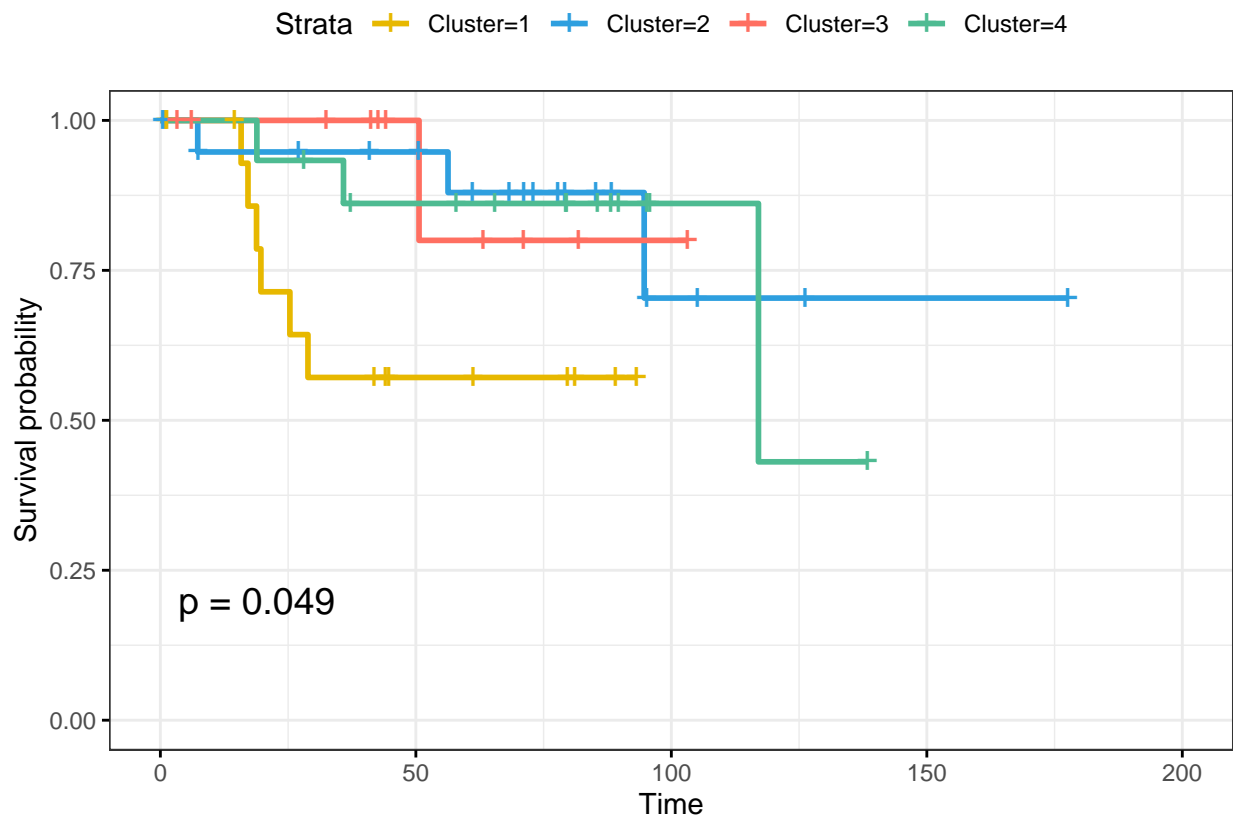


Figure 25: Kaplan-Meier plot with clusters from figure 8 (script 5) (n = 63)

```

# Prepare table
df_cluster_conv_merge <- merge(1, df_metadata_surv_conv, by = "Patient")
df_cluster_conv_merge$Cluster <- ifelse(df_cluster_conv_merge$Cluster ==
  2, 1, df_cluster_conv_merge$Cluster)
df_cluster_conv_merge$Cluster <- ifelse(df_cluster_conv_merge$Cluster ==
  4, 3, df_cluster_conv_merge$Cluster)
df_cluster_conv_merge$Cluster <- ifelse(df_cluster_conv_merge$Cluster ==
  3, 2, df_cluster_conv_merge$Cluster)
rownames(df_cluster_conv_merge) <- df_cluster_conv_merge$Patient
p <- plot_km(survfit(Surv(OS.delay, OS.event) ~ Cluster, data = df_cluster_conv_merge),
  df_cluster_conv_merge)
p

```

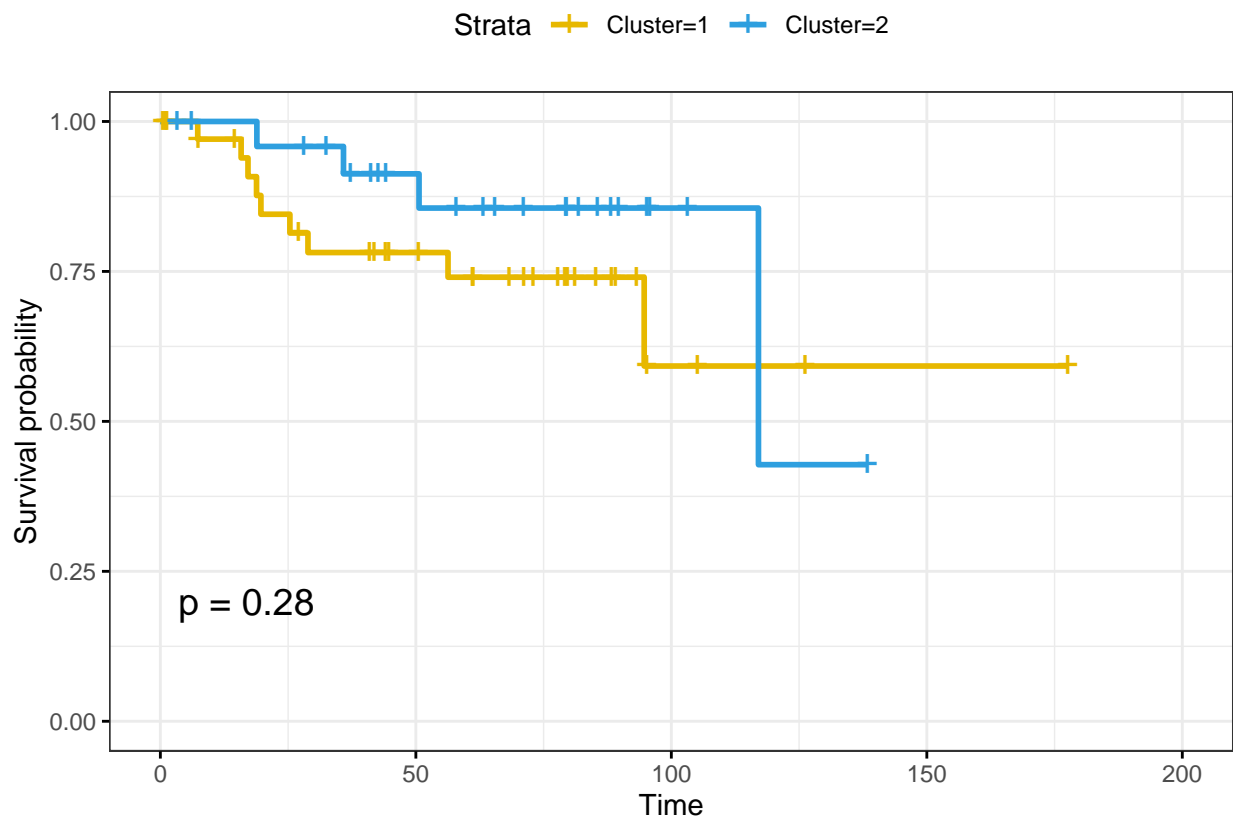


Figure 26: Kaplan-Meier plot merged clusters from figure 8 (script 5) (n = 63)

Merge C1 with C2 and C3 with C4

3) Cluster 1 from CTA heatmaps analysis

This section is the analysis of the previous results. We want to see where the patients clustered in other heatmaps.

```
# Take indiv from cluster 1
l_c1_cta_all <- l_cta_all[l_cta_all$Cluster == 1, ]

# Merge to see where the indiv are in the other heatmap
df_analysis_cold_all <- l_c1_cta_all %>%
  left_join(df_metadata %>%
    select(Patient, Histology), by = "Patient") %>%
  left_join(l_anno_all, by = "Patient")
df_analysis_cold_all
```

##	Cluster.x	Patient	Histology	Cluster.y
## 1	1	CN027	dedifferentiated	NA
## 2	1	TR031	dedifferentiated	NA
## 3	1	LY040	dedifferentiated	NA
## 4	1	CN012	dedifferentiated	NA
## 5	1	TR062	dedifferentiated	NA
## 6	1	TE063	dedifferentiated	NA
## 7	1	Li010	dedifferentiated	NA
## 8	1	LY016	G3	1
## 9	1	TE015	G3	1
## 10	1	TR016	dedifferentiated	NA
## 11	1	TR022	G3	1
## 12	1	CN019	G3	1
## 13	1	TE017	G3	1
## 14	1	LY019	G2	1
## 15	1	CN033	G3	1
## 16	1	CN008	G3	1
## 17	1	TR042	dedifferentiated	NA
## 18	1	TR004	dedifferentiated	NA

We see that there is a lot of dediff type that expressed CTA involved in bad survival events but retrieve in cluster 4 (HOT) for the immune cells. One dediff is in the cold cluster. So the differentiated cancers are really different than conventional and are infiltrated by immune cells or not but the most is infiltrated.

```
# Take indiv from cluster 1
l_c1_cta_conv <- l_cta_conv[l_cta_conv$Cluster == 1, ]
df_analysis_cold_conv <- l_c1_cta_conv %>%
  left_join(df_metadata %>%
    select(Patient, Histology), by = "Patient") %>%
  left_join(l_anno_all, by = "Patient") %>%
  left_join(l_anno_conv, by = "Patient")
df_analysis_cold_conv
```

##	Cluster.x	Patient	Histology	Cluster.y	Cluster
## 1	1	LY016	G3	1	1
## 2	1	TE015	G3	1	1
## 3	1	CN033	G3	1	1
## 4	1	CN019	G3	1	1
## 5	1	CN008	G3	1	1

For the conventional chondrosarcomas, we see that all the patient have G3 chondrosarcoma and can be in the cluster 2.