pMHC analysis

Léa ROGUE

2025-05-06

List of Figures

Contents

This scripts permit us to analyze pMHC from IEDB by 2 approaches. The first approach is based on the further transcriptomic analysis with CTA selection. With these genes, we searched the degree of validation of its peptides with the number of references. We explore the most referenced peptides (>= 15). The second approach constists of integrate the predict affinity of peptides link to cancer diseases with the expression of genes in healthy tissues and chondrosarcoma. With these 2 approaches, we can integer list of genes to see if genes with predict peptides are validated or not.

Load libraries

```r
library(ggplot2)
library(dplyr)
library(tidyr)
library(ggrepel)
library(plotly)
```

I. Explore IEDB references for selected CTA

From IEDB web interface, we searched for the number of reference, or the degree of validation in the literature of these peptides from selected genes. We selected peptides with >= 15 references.

Distribution of references number per peptide

```r
df_ref <- read.table("../data/table_ref.tsv", sep = "\t", header = TRUE)

# Associate Gene with peptide
df_ref$Gene_Peptide <- paste(df_ref$Genes, df_ref$main.peptide,
    sep = " - ")

# Create histogram
ggplot(df_ref, aes(x = reorder(Gene_Peptide, Number.of.IEDB.ref),
    y = Number.of.IEDB.ref, fill = HLA.A.02.01)) + geom_col() +
    scale_fill_manual(values = c(`HLA-A*02:01` = "#E74C3C", Others = "#3498DB")) +
    theme_classic() + theme(axis.text.x = element_text(angle = 90,
    hjust = 1, vjust = 0.5)) + labs(title = "Histogram of number of references per peptides",
    x = "Gene - Peptide", y = "IEDB reference number", fill = "HLA type")
```

This plot shows that the most reference genes is CTAG2 with the peptide SLLMWITQC associated with HLA-A0201.

II. Complete scatter plot with mean expression in tissues and in chondrosarcoma

1. Scatter plot GTEx with number of references

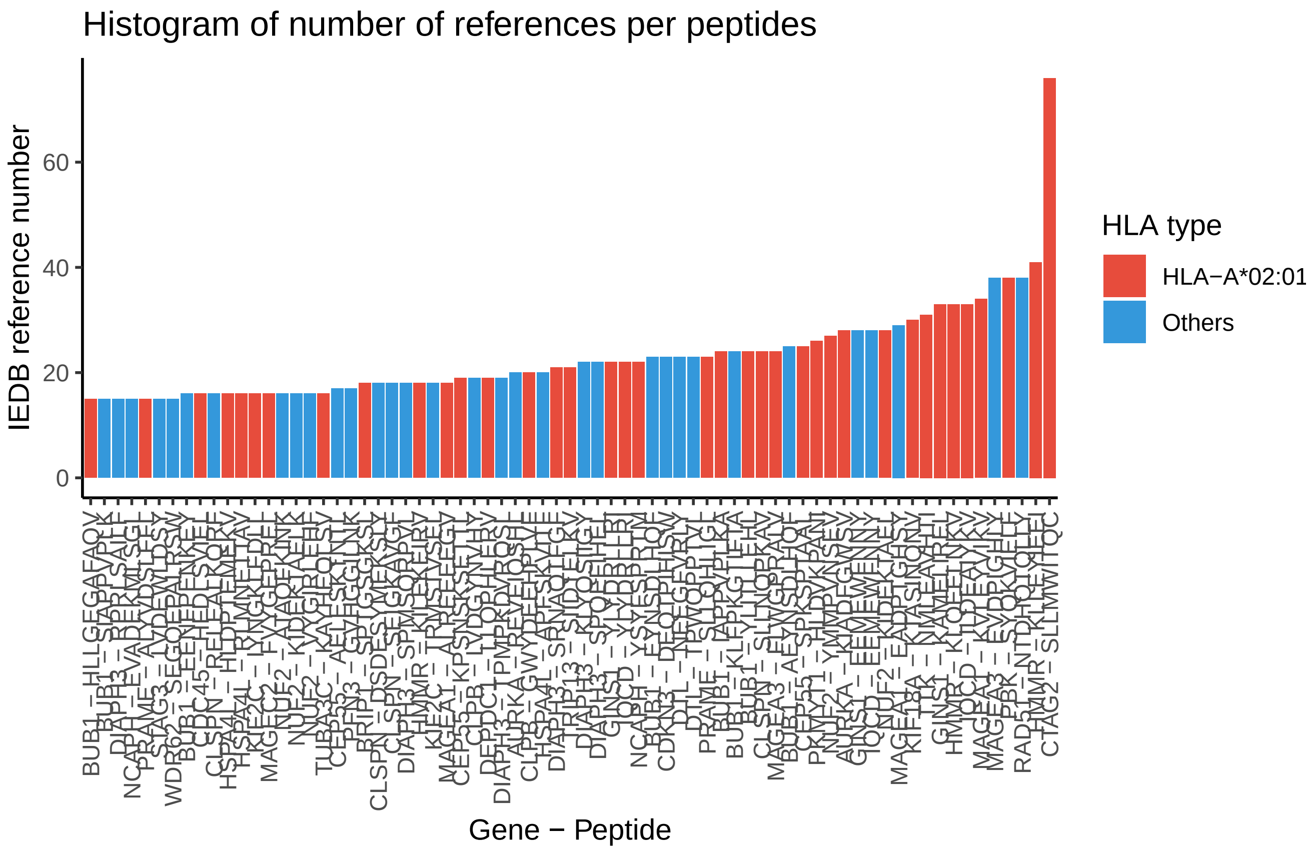

Figure 1: Histogram of number of references per peptides

```r
# Load and filter object to select genes
load("../../Chondrosarcoma/results/df_scatter_63.RData")
genes <- readLines("../../Chondrosarcoma/results/selected_cta_scatter.txt")
df_scatter_ref <- df_scatter_63[rownames(df_scatter_63) %in%
    genes, ]
df_scatter_ref$Genes <- rownames(df_scatter_ref)

# Merge with df_ref to have number of ref
df_scatter_ref <- merge(df_scatter_ref, df_ref, by = "Genes",
    all.x = TRUE)

# Replace NA with 0 to color points, if ref <15, color =
# grey
df_scatter_ref$Number.of.IEDB.ref[is.na(df_scatter_ref$Number.of.IEDB.ref)] <- 0
df <- df_scatter_ref %>%
    filter(Number.of.IEDB.ref >= 15) %>%
    select(Genes, Intensity, Mean_expression_tissues, main.peptide,
        HLA.A.02.01, Number.of.IEDB.ref)

# Filter to add points labels
df_labels <- df_scatter_ref %>%
    filter(Number.of.IEDB.ref >= 15) %>%
    group_by(Mean_expression_tissues, Intensity) %>%
    slice_max(order_by = Number.of.IEDB.ref, n = 1, with_ties = FALSE) %>%
    ungroup()

# Prepare color and legend
df_scatter_ref$color <- ifelse(df_scatter_ref$Number.of.IEDB.ref !=
    0 & df_scatter_ref$HLA.A.02.01 == "HLA-A*02:01", ">= 15 ref & HLA-A*02:01",
    ifelse(df_scatter_ref$Number.of.IEDB.ref != 0 & df_scatter_ref$HLA.A.02.01 !=
        "HLA-A*02:01", ">= 15 ref & others haplotypes", "< 15 ref"))

# Select the higher number of reference for genes with
# multiple peptides
df_scatter_ref <- df_scatter_ref %>%
    group_by(Mean_expression_tissues, Intensity) %>%
    slice_max(order_by = Number.of.IEDB.ref, n = 1, with_ties = FALSE) %>%
    ungroup()
```

```r
# Generate scatter plot
ggplot(df_scatter_ref, aes(x = Mean_expression_tissues, y = Intensity,
    size = Number.of.IEDB.ref, color = color)) + geom_point(alpha = 0.8) +
    geom_text_repel(data = df_labels, aes(label = Genes), color = "black",
        size = 2.5, max.overlaps = 100) + theme_minimal() + scale_x_log10() +
    scale_size_continuous(range = c(1, 6)) + scale_color_manual(values = c(`< 15 ref` = "#dadada",
    `>= 15 ref & others haplotypes` = "#3498DB", `>= 15 ref & HLA-A*02:01` = "#E74C3C")) +
    labs(title = "Scatter plot of tumor specificity of CTAs (n = 63)",
        x = "Normal tissues expression (Log10 TPM mean)", y = "Chondrosarcoma CTA expression (intensiti
        size = "IEDB References", color = "Number of references and HLA haplotypes")
```

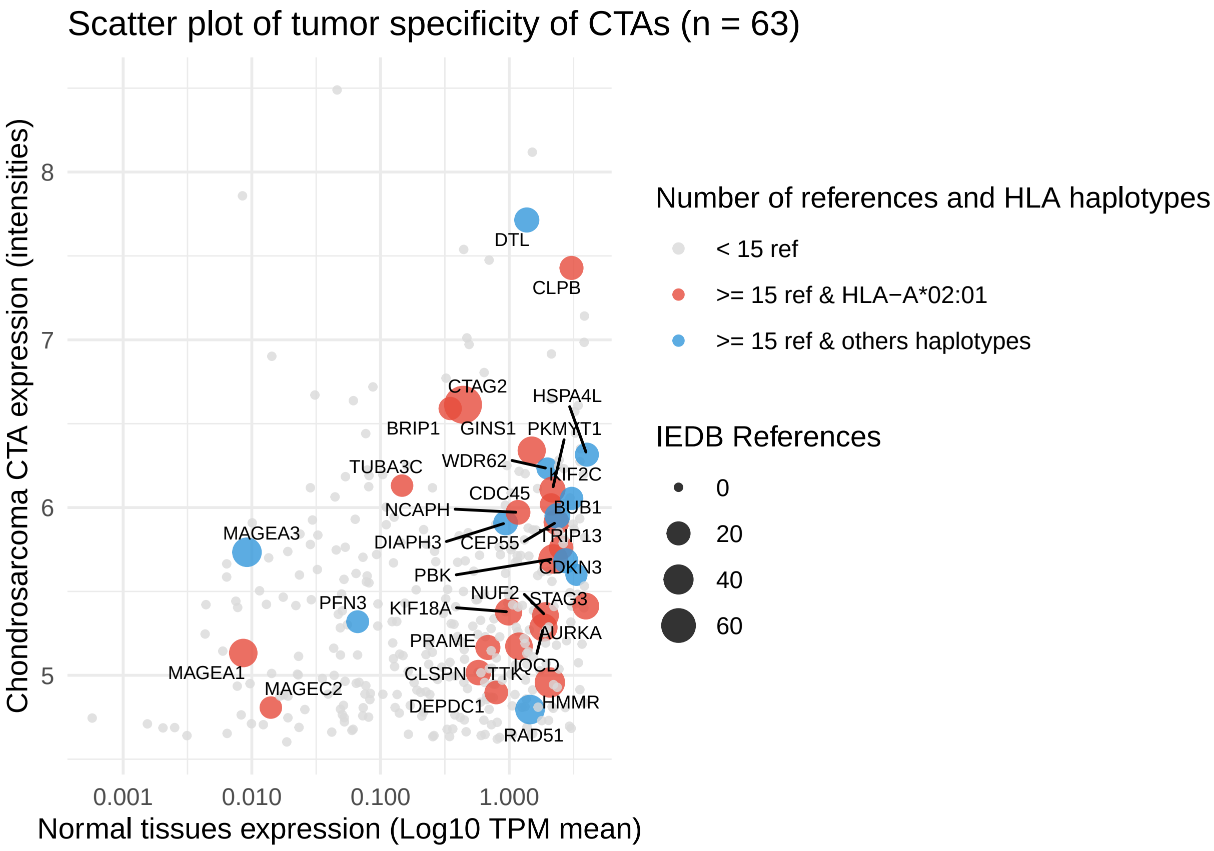

Figure 2: Scatter plot of tumor specificity of CTAs (n = 63) with peptides and number of ref

2. Scatter plot with predicted affinityfor HLA-A0201

Now, we want to integrate affinity prediction value (with netMHC tool) per gene on the scatter plot. ### a. Scatter plot with the minimum of affinity per genes

```
# Read affinity per peptide with strong binding prediction
df_affinity <- read.table("../results/df_min_affinity_expr_hla_a0201.tsv",
    sep = "\t", header = TRUE)

# Read mean expression with genes with affinity
df_scatter_aff <- df_scatter_63[rownames(df_scatter_63) %in%
    genes, ]
df_scatter_aff$SYMBOL <- rownames(df_scatter_aff)

# Merge
df_scatter_aff <- merge(df_scatter_aff, df_affinity, by = "SYMBOL",
    all.x = TRUE)
df_scatter_aff$Affinity <- as.numeric(df_scatter_aff$Affinity)

# Add a column to color point on the scatter
df_scatter_aff$color_group <- ifelse(is.na(df_scatter_aff$Affinity),
    "No SB peptides", "SB peptides")
df_scatter_aff$Affinity <- ifelse(is.na(df_scatter_aff$Affinity),
    40, df_scatter_aff$Affinity)
```

```
# Create plot
ggplot(df_scatter_aff, aes(x = Mean_expression_tissues, y = Intensity,
    size = Affinity, color = color_group)) + geom_point(alpha = 0.8) +
    theme_minimal() + scale_x_log10() + scale_size_continuous(range = c(5,
    1)) + scale_color_manual(values = c(`No SB peptides` = "#dadada",
    `SB peptides` = "#E74C3C")) + labs(title = "Scatter plot of tumor specificity of CTAs (n = 63)",
    x = "Normal tissues expression (Log10 TPM mean)", y = "Chondrosarcoma CTA expression (intensities)"
    size = "Inverse Affinity", color = "Binding")
```

We want to add labels with peptides information.

b. Scatter plot with all the information by genes

This part is to explore all the peptides per genes and see the best affinity.

```
# Df with all peptides and its affinity
df_affinity_pep <- read.table("../results/df_peptides_genes_aff_hla_a0201.tsv",
    sep = "\t", header = TRUE)

# Some peptides are aligned on multiple genes => separate
# on the ,
df_affinity_clean <- df_affinity_pep %>%
    separate_rows(Genes, sep = ",\\s*")

# Create legend of labels
df_affinity_clean <- df_affinity_clean %>%
    mutate(peptide_info = paste0(Peptide, " (", round(Affinity,
```

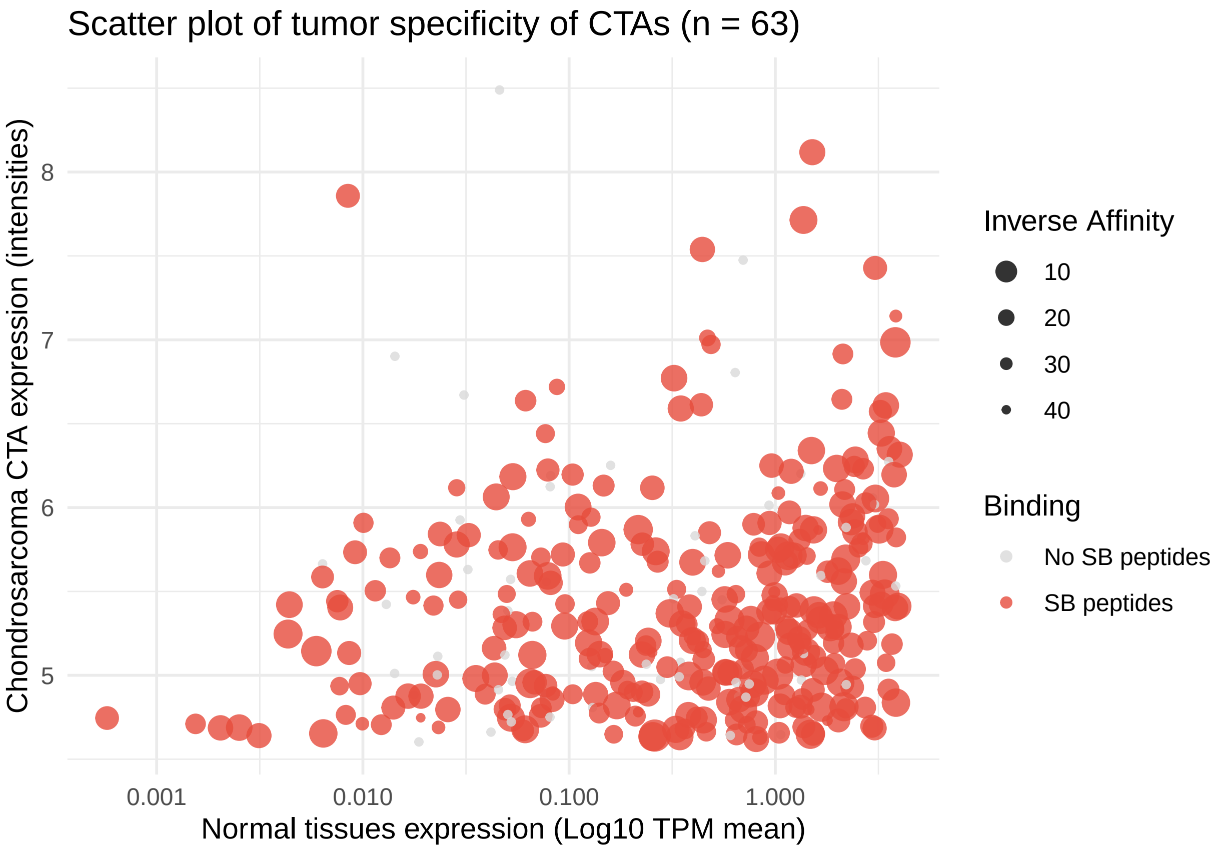

Figure 3: Scatter plot of tumor specificity of CTAs (n = 63) with predicted affinity

```r
        2), ")"))
df_affinity_grouped <- df_affinity_clean %>%
    group_by(Genes) %>%
    summarise(tooltip_peptides = paste(peptide_info, collapse = "<br>"),
        min_affinity = min(Affinity, na.rm = TRUE)) %>%
    rename(SYMBOL = Genes)


# Take and merges intensities and mean expression from
# selected genes with peptides
df_scatter_aff_pep <- df_scatter_63[rownames(df_scatter_63) %in%
    genes, ]
df_scatter_aff_pep$SYMBOL <- rownames(df_scatter_aff_pep)
df_scatter_aff_pep <- merge(df_scatter_aff_pep, df_affinity_grouped,
    by = "SYMBOL", all.x = TRUE)
df_scatter_aff_pep$color_group <- ifelse(is.na(df_scatter_aff_pep$min_affinity),
    "No SB peptides", "SB peptides")

# Add colors to differentiate genes with SB peptides and
# genes without SB peptides
df_scatter_aff_pep$tooltip_peptides[is.na(df_scatter_aff_pep$tooltip_peptides)] <- "No SB peptide"

# Assign a high value to plot point size
df_scatter_aff_pep$min_affinity[is.na(df_scatter_aff_pep$min_affinity)] <- 40
```

```r
# Create interactive plot
p <- ggplot(df_scatter_aff_pep, aes(x = Mean_expression_tissues,
    y = Intensity, size = min_affinity, color = color_group,
    text = paste0("Gene: ", SYMBOL, "<br>Mean expression: ",
        round(Mean_expression_tissues, 3), "<br>Intensity: ",
        round(Intensity, 2), "<br>Peptides (nM affinity):<br>",
        tooltip_peptides))) + geom_point(alpha = 0.8) + theme_minimal() +
    scale_x_log10() + scale_size_continuous(range = c(5, 1)) +
    scale_color_manual(values = c(`No SB peptides` = "#dadada",
        `SB peptides` = "#E74C3C")) + labs(title = "Scatter plot of tumor specificity of CTAs (n = 63)"
    x = "Normal tissues expression (Log10 TPM mean)", y = "Chondrosarcoma CTA expression (intensities)"
    size = "Inverse Affinity", color = "Binding")

# ggplotly(p, tooltip = 'text')
```

```r
# Create plot
p <- ggplot(df_scatter_aff_pep, aes(x = Mean_expression_tissues,
    y = Intensity, color = color_group, text = paste0("Gene: ",
        SYMBOL, "<br>Mean expression: ", round(Mean_expression_tissues,
            2), "<br>Intensity: ", round(Intensity, 2), "<br>Peptides (nM affinity):<br>",
        tooltip_peptides))) + geom_point(alpha = 0.8, size = 1) +
    theme_minimal() + scale_x_log10() + scale_color_manual(values = c(`No SB peptides` = "#dadada",
    `SB peptides` = "#E74C3C")) + labs(title = "Scatter plot of tumor specificity of CTAs (n = 63)",
    x = "Normal tissues expression (Log10 TPM mean)", y = "Chondrosarcoma CTA expression (intensities)"
    color = "Binding")

# ggplotly(p, tooltip = 'text')
```

This last plot is an hmtl output because it's an interactive plot. It is available in the html notebook.

c. Scatter plot with HR data

```r
# Adding data on pearson correlation
df_hot_cold <- read.table("../../Chondrosarcoma/results/clusters_indiv/clusters_cta_pearson_63.tsv",
    header = T, sep = "\t")
colnames(df_hot_cold) <- c("Associated_immunophenotype", "SYMBOL")
df_scatter_aff_pep <- merge(df_scatter_aff_pep, df_hot_cold,
    by = "SYMBOL", all = T)
```

```r
# Create plot
p <- ggplot(df_scatter_aff_pep[df_scatter_aff_pep$Cluster_conv !=
    "NA" & df_scatter_aff_pep$Cluster_conv != "c", ], aes(x = Mean_expression_tissues,
    y = Intensity, color = Cluster_conv, shape = Associated_immunophenotype,
    text = paste0("Gene: ", SYMBOL, "<br>Mean expression: ",
        round(Mean_expression_tissues, 2), "<br>Intensity: ",
        round(Intensity, 2), "<br>Peptides (nM affinity):<br>",
        tooltip_peptides))) + geom_point(alpha = 0.8) + scale_shape_manual(values = c(COLD = 15,
    HOT = 16)) + scale_color_manual(values = c(a = "#3498DB",
    b = "#E74C3C")) + theme_minimal() + scale_x_log10() + labs(title = "Scatter plot of tumor specifici
    x = "Normal tissues expression (Log10 TPM mean)", y = "Chondrosarcoma CTA expression (intensities)"
    color = "Clustering gene\nimpacting survival\n(conventional CHS)",
    shape = "Associated immunophenotype")
# ggplotly(p, tooltip = 'text')
```

This last plot is an hmtl output because it's an interactive plot. It is available in the html notebook.