Sorbonne University

Master 2 BMC - Systems Immunology

Characterization of the inflammatory status in osteosarcoma by *in silico* RNA-Seq analysis

By

HUYNH Minh-Anh

September 2022

Table of Contents

List of Abbreviations

- DEG : Differentially Expressed Genes
- FDR : False Discovery Rate
- FC : Fold Change
- HIR : Hallmark Inflammatory Response
- MDS : MultiDimensional Scaling
- MSigDB : Molecular Signature Database
- scRNA-seq : single-cell RNA sequencing
- TANs: Tumor-associated Neutrophils
- TME: Tumor MicroEnvironment
- TPM : Transcripts Per Million
- UMAP : Uniform Manifold Approximation and Projection for Dimension Reduction

Acknowledgements

Thank you for following this tutorial!

I hope you’ll find it useful to write a very professional dissertation.

1 Introduction

- introduce the reader to the subject area and clarify the knowledge gap that the dissertation research will fill.
- set the context for the dissertation by reviewing the relevant literature.
- include relevant references to general (theoretical papers and reviews) and specific (specific to the particular question addressed) literature, to justify the research that has been undertaken and define the questions being addressed.
- state the primary research questions and hypotheses in the final paragraph.
- follow an 'inverted triangle' format, progressing from general scientific ideas and why they matter to the specific research questions addressed in the dissertation project.

The introduction should not be just a 'Literature Review'.

Osteosarcomas are one of the most aggressive diseases, characterized by high tumor heterogeneity and numerous genomic instabilities. They usually occur in long bones and represent one-fifth of all primary malignant bone tumors and 2.4% of pediatric cancers (Fu et al., 2022). Osteosarcomas occur most frequently in children aged 14-18 years, and more than 30% of cases, within adults over 40 years of age (**Mirabello.2009?**). The combination of surgery and chemotherapy has increased the 5-year survival rate of patients to over 70%, however metastatic osteosarcoma, frequently occurring in the lungs, drops the survival rate to less than 20%. (Fu et al., 2022).

The difficulty in treating these tumors is largely due to the genomic instability that characterizes them and that leads to a great inter and intra tumor heterogeneity. Thus, the cancer cells can have various phenotypes. Some tumors are osteoblastic, chondroblastic or fibroblastic dominant.

It is established that tumor cells recruit numerous accessory cell types that will form a stroma that supports tumor growth, metastatic progression and resistance to treatment. Cellular (immune cells) and molecular (cytokines, metabolites) inflammatory mediators play a key role in the tumor stroma. The main immune component has been shown to be M2 macrophages. However, scRNA-seq analyses have revealed other immune cell populations such as neutrophils that are also important mediators of inflammation. Bone tumors are a source of inflammatory factors that serve to recruit and educate the stroma.

An issue that remains poorly documented concerns the heterogeneity of the inflammatory landscape of osteosarcomas and the possible relationships between tumor inflammatory status (TIS) and phenotypic and/or clinical characteristics of these tumors. Similarly, the characteristics of neutrophils associated with bone tumors and their role in tumor inflammation have not been explored to date.

Recently, immunotherapy has proven to be particularly effective in treating previously difficult and deadly cancers, and has been promoted as a medical care therapy in many of them. However, osteosarcomas have not withstood immunotherapy and have instead turned to conventional chemotherapy, which is not very effective and has a plethora of side effects. In order to find a cure for these difficult-to-treat cancers, a better understanding of the tumor microenvironment (TME) could potentially lead to effective and novel targeted TME therapies, especially since osteosarcomas exhibit high heterogeneity and the immunological mechanisms of immune resistance are still unknown. Current advances in bulk RNA sequencing (RNA-Seq) and single-cell RNA sequencing (scRNA-seq) have shown their potential in exploring the tumor microenvironment (TME) to explore intra-tumor heterogeneity and cellular dialogue between tumor and inflammatory cells.

Neutrophils are the primary innate immune cells recruited during an inflammation and multiple papers have already elucidated that tumor-associated neutrophils (TANs) or circulating neutrophils are related to worse patient survival therapy and chemoresistance (Faget et al., 2021; Long et al., 2021). Neutrophils support cancer development through 3 pathways : they're able to promote cancer initiation, assistance of metastasis and increase of tumor growth (Faget et al., 2021).

- Tumeur ne répondent pas à l'immunothérapie
- Tumeur source d'inflammation
- Inflammation et TME non étudié dans ostéosarcomes

2 Objective and experimental principle

The study aims to analyze public dataset and to make a custom analysis using bioinformatic tools in order to understand the cellular crosstalk and interactions in the TME, between neutrophils and tumor cells and their inflammatory status.

Collaborating with Dr Jean-Marc Schwartz's team in Manchester and Sophie Khakoo, a Master student in Systems Biology, collecting data from a scRNA-Seq dataset, a customized downstream analysis will be performed on GSE87686 and Target-OS, and a scRNAseq dataset in order to identify the relationships between neutrophils and osteosarcoma through the use of genetic signatures available from MSigDB and customized gene lists.

3 Methods

3.1 Data collection

Analyses were performed on two bulk RNA-seq osteosarcoma datasets, TARGET-OS Osteosarcoma and GSE87686. Data was collected from TARGET-OS using GDC Data Transfer Tool UI (v1.0.0), returning 19493 protein coding genes and 88 samples for TARGET-OS, containing both raw data and TPM data. GSE87686 data was obtained through the lab's previously pre-processed kallisto files, downloaded through SRA Run Selector to obtain SRA run files. Data was then imported from kallisto files via *tximport (v1.22.0)* R package. Genes were converted to ENST and ENSG and finally HUGO gene symbols through *biomaRt (v2.50.3)*.

3.2 Normalization

Raw data was converted to TPM as it is the best performing normalization method than FPKM or RPKM, based on its preservation of biological signal as compared to other methods (Abrams et al., 2019). Calculation was performed using counts and lengths for each gene, returned from the tximport to kallisto process.

Normalization by Z-score with the *scale* function was used to normalize the data for each gene, in order to be able to visualize the data in corresponding heatmaps.

Z-score of the mean of TPM data was used to construct grouped heatmaps for a given gene or gene signature.

Inflammatory groups characterizing the intensity of inflammatory status in tumors were created by choosing the lowest and highest mean of Z-score of the Hallmark Inflammatory Response signature from MSigDB, containing 200 genes. ICAM4 was notably not present in the dataset in TARGET-OS cohort. The groups were cut off evenly using the *ntile* function in *dplyr (v1.0.8)* R package.

3.3 Construction of gene signatures

Gene signatures relevant to the research topic were obtained from MSigDB via *msigdbr* (v7.5.1)
HAY_BONE_MARROW_NEUTROPHIL.v7.5.1

Canonical markers for osteosarcoma markers were adapted from Zhou et al. (2020) 's analysis and their canonical markers generated from the literature in their Supplementary Table 2.

3.4 Construction of inflammatory groups

The groups were cut off evenly using the *ntile* function in *dplyr* R package.

3.5 MDS visualization and k-means clustering

MDS visualization was performed with *vegan*'s metaMDS function, and clustering was effected with k-means clustering

3.6 Determination of cell population abundance

Estimation of tumor immune infiltration using bulk RNA-seq can be estimated using Microenvironment Cell Populations-counter (MCP-counter) (Becht et al., 2016).

CIBERSORTx algorithm was also used for immune cell deconvolution, from the CIBERSORTx platform (https://cibersortx.stanford.edu/) to estimate the abundance for 22 immune cell populations. CIBERSORTx is run in absolute mode, with batch correction, no quantile normalization.

Similarly, xCell algorithm via *xCell* and *immunedeconv* packages was used to obtain abundance for 36 immune cell populations.

3.7 Identification of Differentially Expressed Genes (DEG)

Using *DESeq2 (v1.34.0)* (Love et al., 2014), standard DEG pipeline using raw data was performed between inflammatory groups (Low, Medium, High). Fold Change (FC) is calculated as $FC = High/Low$. DEGs were identified by using a cut-off on adjusted *P* value ≤0.05 and $log_2(FC) \geq 1; log_2(FC) \leq -1$. An interpretation of the FDR/Benjamini-Hochberg method for controlling the FDR is implemented in DESeq2 in which we rank the genes by p-value, then multiply each ranked *P* value by m/rank (m = total number of tests).

3.7.1 Over Representation Analysis (ORA)

enrichR v(3.0) was used (Xie et al., 2021; **R-enrichR?**) for functional gene enrichment/pathway analysis. Following databases were queried : “GO Molecular Function 2021”, “GO Biological Process 2021”, “GO Cellular Component 2021”, “Human Gene Atlas”, “BioPlanet 2019”, “KEGG 2021 Human”, “MSigDB Hallmark 2020”, “Reactome 2016”

3.8 Gene Set Enrichment Analysis (GSEA)

GSEA was computed through *clusterprofiler* which uses *fgsea* as a backend. $P < 0.10$ was considered statistically significant. *P* adjusted is calculated using the False Discovery Rate from Benjamini-Hochberg. Pathways in the MSigDB databases (<https:// www.gsea-msigdb.org/gsea/msigdb>) were used for the GSEA analysis.

3.9 Statistical Testing

Statistical testing was performed using *R software 4.2.1 (2022-06-23)* and the *rstatix* and *car* package. ANOVA was performed for multiple comparison testing, only if the data is normally distributed and has homoscedasticity, verified through shapiro’s test and levene’s test, respectively. Tukey’s honestly significant different testing was effected when appropriate, with *P* values adjusted with Holm’s method. Kruskal-Wallis testing was performed for non-metric comparative analysis between groups when. Post-hoc analysis was performed using Dunn’s test as opposed to Wilcoxon due to the test taking into account Kruskal-Wallis’s rank. Post-hoc dunn’s test was done when appropriate.

$P < 0.05$ and *P* adjusted < 0.05 was considered statistically significant.

4 Results

Some more guidelines from the School of Geosciences.

This section should summarize the findings of the research referring to all Figure s, tables and statistical results (some of which may be placed in appendices).

- include the primary results, ordered logically - it is often useful to follow the same order as presented in the methods.
- alternatively, you may find that ordering the results from the most important to the least important works better for your project.
- data should only be presented in the main text once, either in tables or Figure s; if presented in Figure s, data can be tabulated in appendices and referred to at the appropriate point in the main text.

Often, it is recommended that you write the results section first, so that you can write the methods that are appropriate to describe the results presented. Then you can write the discussion next, then the introduction which includes the relevant literature for the scientific story that you are telling and finally the conclusions and abstract – this approach is called writing backwards.

4.1 Determination of inflammatory groups representing intratumor inflammatory status

In order to see if inflammatory groups associated to inflammation response can be obtained, first an MDS visualization of the dataset on the Hallmark Inflammatory Response signature (HIR) containing 200 genes, from MSigDB **(Figure** 1**.A)**. The 88 osteosarcoma samples are overall clustered in the middle, while other samples are scattered outwardly. The elbow method **(Figure** 1**.B)** seems to indicate that 2 clusters seems to be the optimal number of clusters after which the wss decreases the least. Visually, 8 clusters is the minimum amount of clusters for which each point visually belongs to its nearest cluster. The gene *ICAM4* was notably not detected in the dataset in the TARGET-OS cohort. Thus, 199 genes from the signature were used.

However, due to the high amount of clusters and combined with a low number of samples and the heterogeneity of the samples, another method based on the intensity of the inflammation has been tried. By doing the mean of the Z-score of the hallmark inflammatory response signature, three groups describing the inflammatory status can be obtained, classified as low, medium and high groups **(Figure** 1**.C)**.

Furthermore, the resulting groups are also partially functionally relevant as they seem to correspond fairly well to the functional groups, defined bythe k-means clustering based on MDS visualization **(Figures** 1**.C, D)**. Each sample is thus attributed to its inflammatory status and this group will be subsequently used for the following results.

4.2 Characterization of osteosarcomas associated to inflammatory status

In order to characterize osteosarcomas, the *Hp Osteosarcoma* gene signature from MSigDB was used to see whether the inflammatory groups can be related to gene expression from this signature. Visually, the heatmap representing the gene signature, annotated with the inflammatory groups, does seem to indicate that the samples express different genes between Low, Medium and High inflammation group **(Figure 2A)**. However, the dendrogram clustering the samples

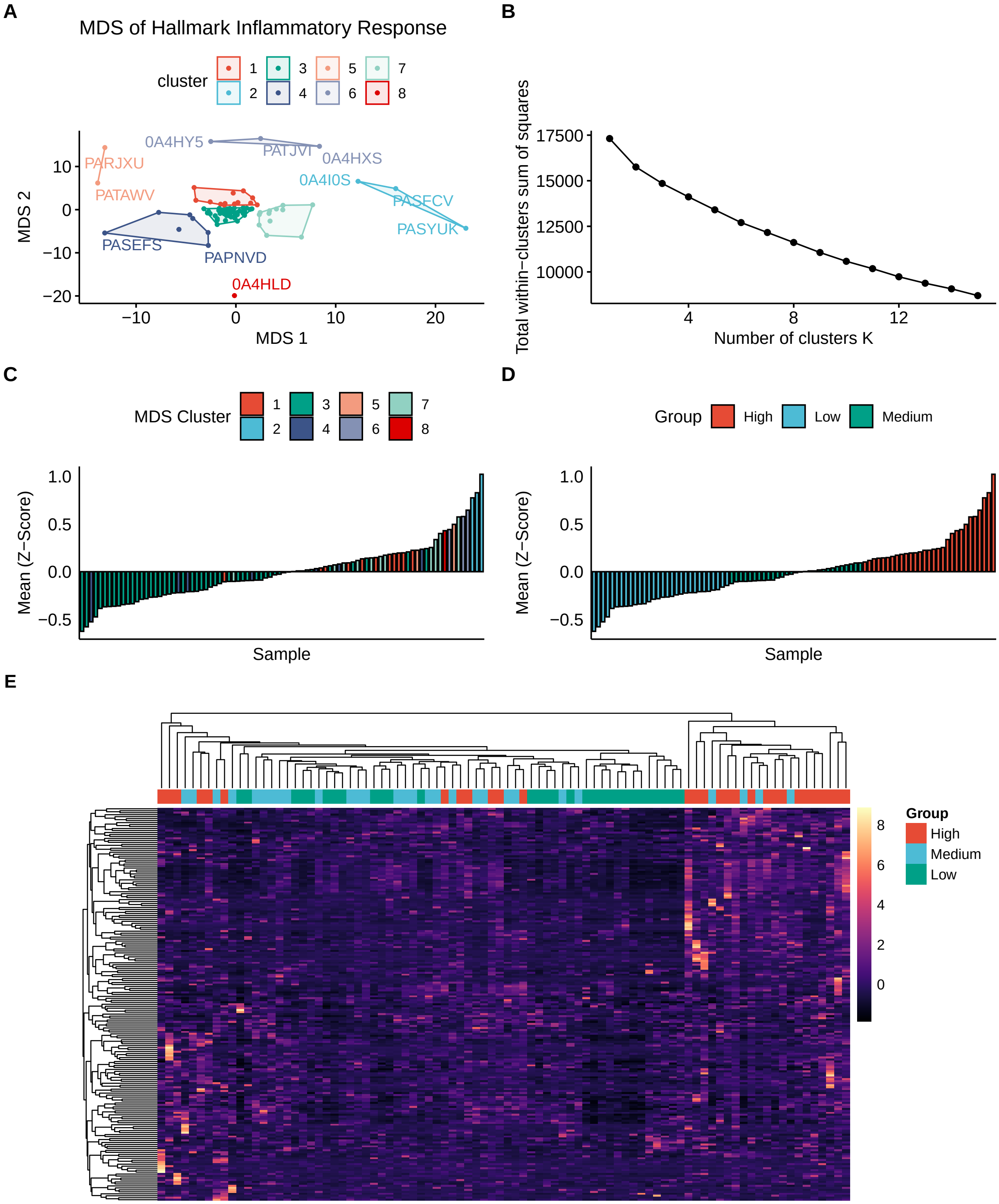

Figure 1: Construction of inflammatory groups for the 88 samples in TARGET-OS dataset based on the HIR signature from MSigDB.

(A) MDS visualization with k-means clustering of the HIR signature.

(B) Elbow graph determining the optimal number of k-clusters.

(C) Histogram of the mean of Z-score of the HIR signature, annotated with the 8 k-means clusters.

(D) Histogram of the mean of Z-score of the HIR signature, annotated with the 3 inflammatory status groups.

(E) Heatmap of the HIR signature with samples annotated to their corresponding inflammatory groups. Rows correspond to genes and columns correspond to samples. HIR = Hallmark Inflammatory Response. MDS = Multidimensional Scaling

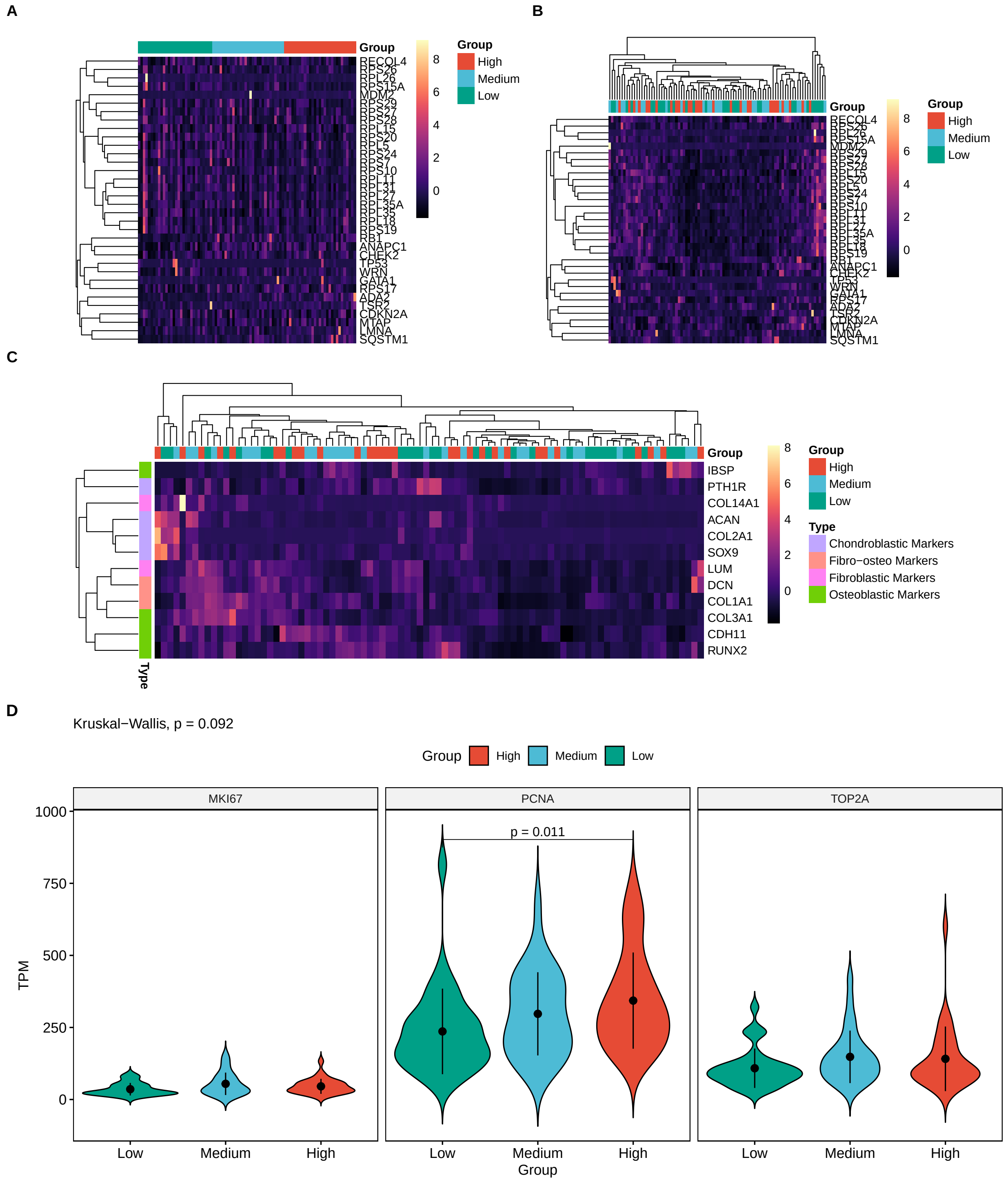

Figure 2: Displays of various osteosarcoma related genes with samples annotated to their respective inflammatory groups. Values are Z-scaled for heatmaps.

(A) Heatmap *Hp Osteosarcoma* signature with unclustered samples.

(B) Heatmap of *Hp Osteosarcoma* with clustered samples.

(C) Heatmap of osteosarcoma type markers.

(D) Violin plot of proliferation markers associated with their inflammatory group.

indicates that the gene signature does not cluster well with the inflammatory annotations **(Figure 2B)**. However, despite high heterogeneity between samples and inflammatory status, there seems to be an overall distinct expression of genes in each inflammatory group.

To further characterize subtypes of osteosarcomas, comparison of the expression of specific osteosarcoma markers relating to osteoblastic, chondroblastic, fibroblastic markers has also been done, through a heatmap representation. Hierarchical clustering of the samples does not appear to be associated with corresponding inflammatory status. However it does reveal that there are groups of osteoblastic, chondroblastic and fibroblastic osteosarcomas which is expected **(Figure 2C)**.

To assess the proliferation status of the samples related to the inflammation status, the mean of three markers of proliferation (MKI67, PCNA, TOP2A) have been compared, along with the mean of the three individual markers **(Figure 2D)**. Statistical Kruskal-Wallis testing is significant ($P = 0.00968$) and post-hoc Dunn analysis reveals that the mean of the proliferation markers between low and high group is significantly different ($P = 0.016$). The data suggests that proliferation is hindered when inflammatory status is high in the osteosarcoma samples.

4.3 Characterization of intra-tumor inflammation associated to inflammatory status

4.3.1 Relationship with ESTIMATE and inflammatory signatures

Using the ESTIMATE algorithm, an immune score has been calculated for each sample which reflects the immune infiltration in a given tumor sample. The violin plot represents the values obtained for each inflammatory group **(Figure 3)**. Because the data is normally distributed and has homoscedasticity, ANOVA has been performed ($P = 0.002$) followed by Tukey's test, revealing that there is a statistically significant difference between Low and High group ($P = 0.002$), but not for the other conditions.

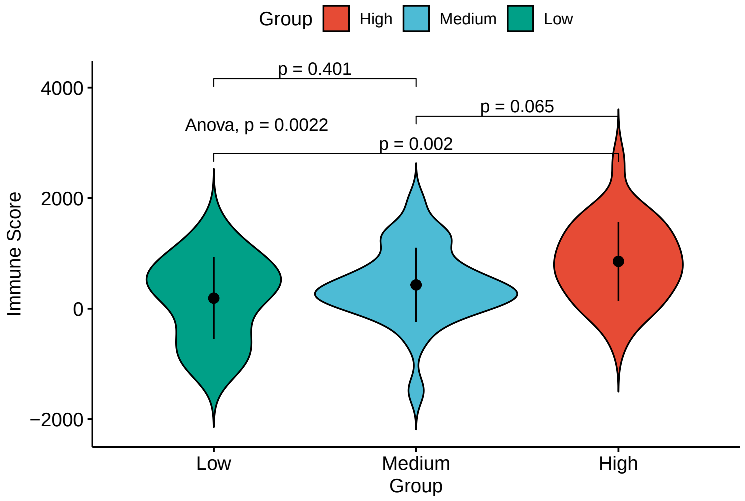

Figure 3: **Violin plot of ESTIMATE score for each inflammatory group.** ANOVA was performed followed by Tukey's post-hoc analysis. *P* $\leq$ 0.05 is considered statistically significant.

4.3.2 Immune abundance by immune deconvolution algorithm

Immune cell abundance can be determined by various immune deconvolution algorithm on TPM normalized bulk-RNASeq. Here, MCP-counter, CIBERSORTx and xCell have been tried. In MCP-counter, the mean (SD) of Z-score reflecting the abundances of neutrophils is statistically different between all groups, it is increased when the inflammation status gets higher **(Figure 4A)**

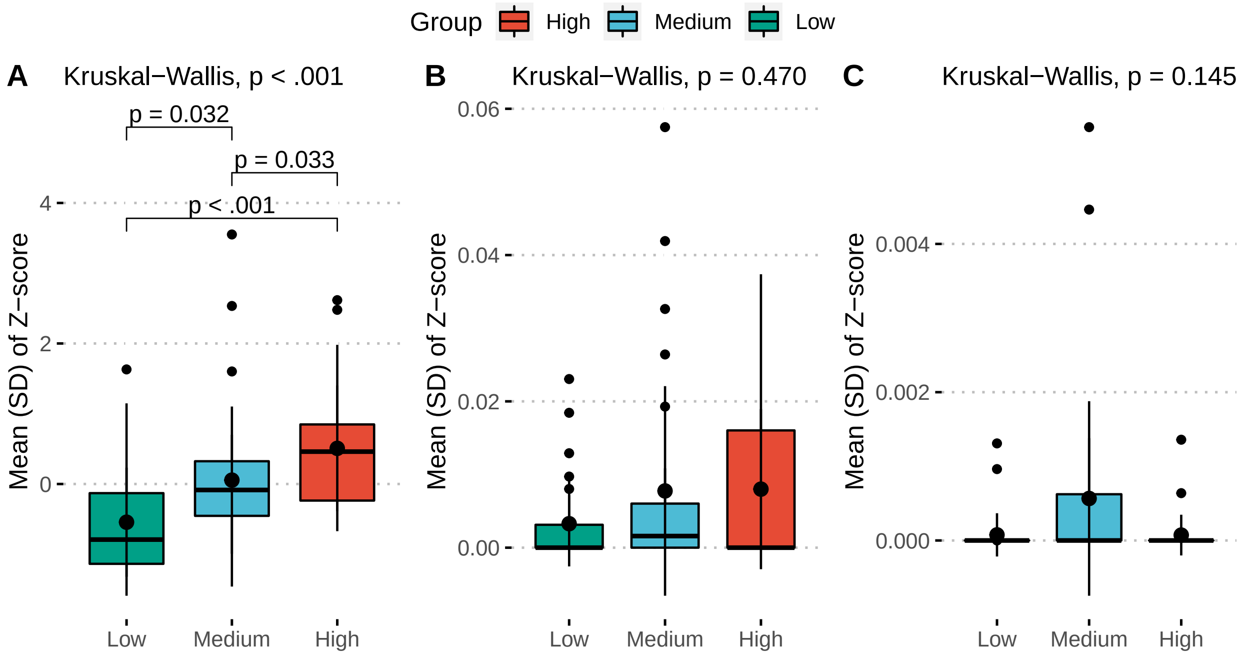

Figure 4: Immune deconvolution plot of neutrophils with various immune deconvolution algorithm.
(A) MCP-counter.
(B) CIBERSORTx.
(C) xCell.
Kruskal-Wallis testing is performed on non-parametric data and a post-hoc dunn's test was performed when appropriate. $P \leq 0.05$ is considered statistically significant.

4.3.3 Survival curve

In order to determine if inflammation groups are linked with increased survival odds, a Kaplan-Meier survival analysis was performed between low and high group of inflammation **(Figure 5)**. Log-rank analysis shows that the survival curve between low and high group of inflammation are not statistically different ($P = 0.16$). However the P value is low enough that it could be considered a trend. Estimation considers 29 patients in the low group and 28 patients in the high group, and estimates that only 9 patients remain alive in the low group while 20 patients are alive in the high group.

4.3.4 Similarity of inflammatory signatures

In order to see if the inflammatory signatures used have overlapping genes or not, a Venn diagram was constructed from the signatures **(Figure 6**. Overall, the three immune signatures depicting immune infiltration (immune score),

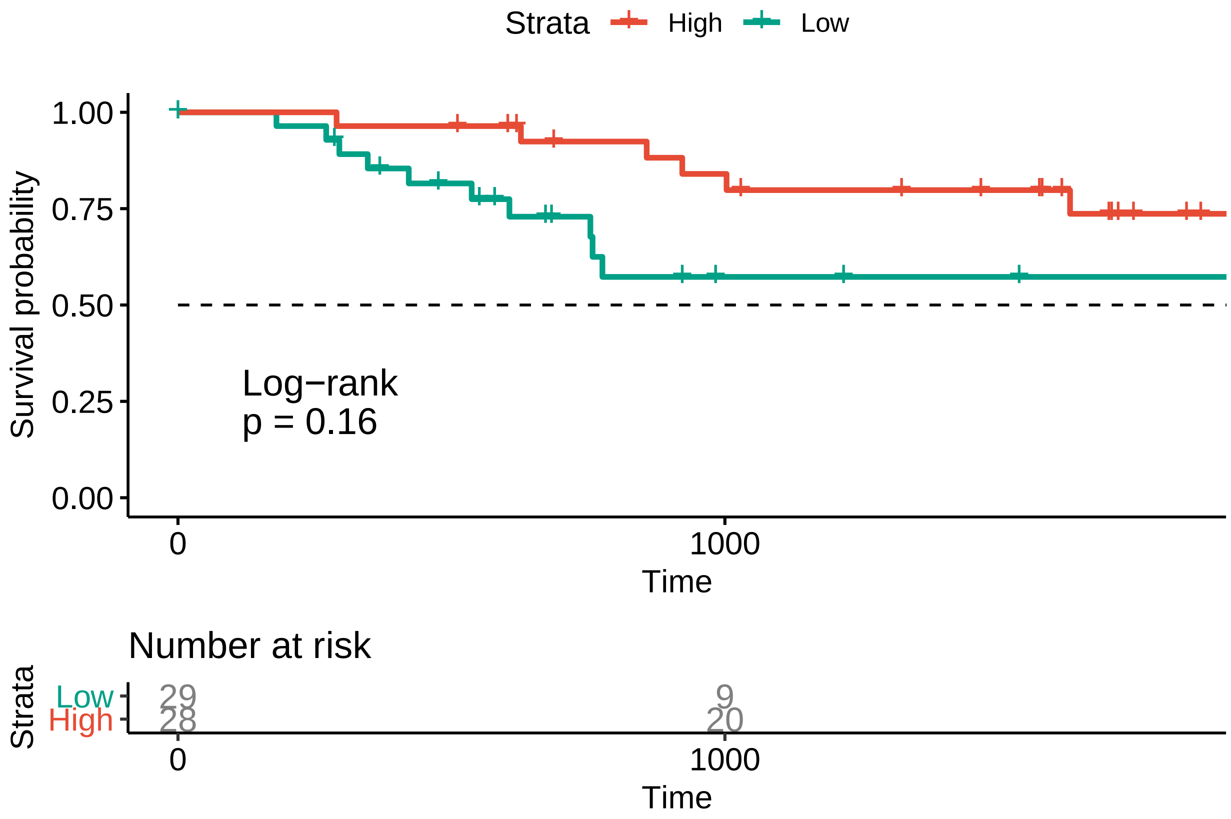

Figure 5: Kaplan-Meier survival analysis plot of low vs high group indicate a trend associating inflammatory status with survival prognostic. Horizontal and vertical axes represent survival times and rates, respectively. Blue and red colors represents low and high inflammation group respectively. Plus signs indicate censored values. Depicted P-values were obtained by the log-rank test. The diagram is cut at 1825 days (5 years). $P \leq 0.05$ is considered statistically significant.

inflammation response and TNF-α pathway have low overlapping genes and can be considered different.

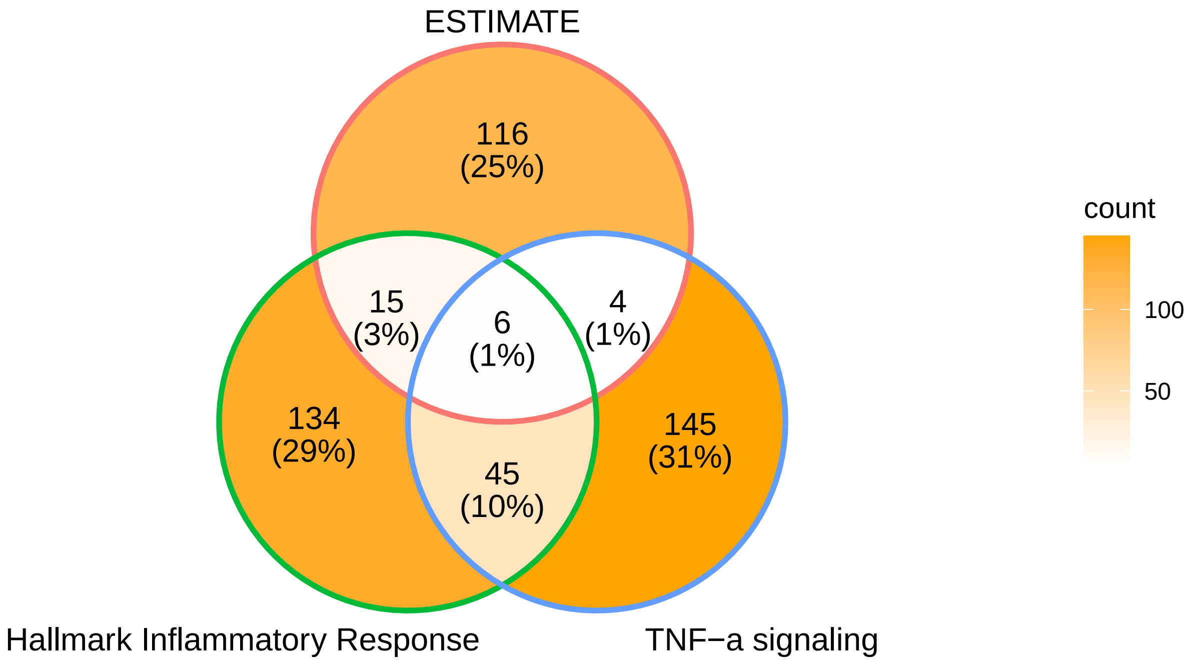

Figure 6: Venn diagram comparing three inflammatory signatures.

4.4 Biological mechanisms underlying the inflammatory groups

4.4.1 Differential Gene Expression Analysis and Enrichment

A differential gene expression analysis was performed on the inflammatory status condition (Low vs High) on the total 19114 genes. DEGs are enriched through multiple databases in order to identify potentially interesting pathways relating to a high level of inflammation **(Figure 7)**. The combined enrichment barplot **(Figure 7A)** returned multiple pathways associated with chemokines, PD-1 signaling, and T cell lymphocyte associated mechanisms. Interestingly, enrichment of downregulated genes in GO Biological Process database returned a lot of pathways linked to immune cell chemotaxis, and particularly neutrophil chemotaxis.

4.4.2 Gene Set Enrichment Analysis (GSEA)

GSEA was performed on 95 selected signatures from MSigDB, based on the theme of osteosarcoma, angiogenesis, neutrophils, migration, hypoxia and inflammation. Only significant pathways are reported in the **Table 1**. Certain pathways are not significant but relevant, such as "Gobp Positive Regulation of Vascular Associated Smooth Muscle Cell Proliferation" ($P = 0.13$), "Gobp Positive Regulation of Inflammatory Response", and all other pathways concerning dendritic, lymphocyte, monocyte, macrophage chemotaxis and migration.

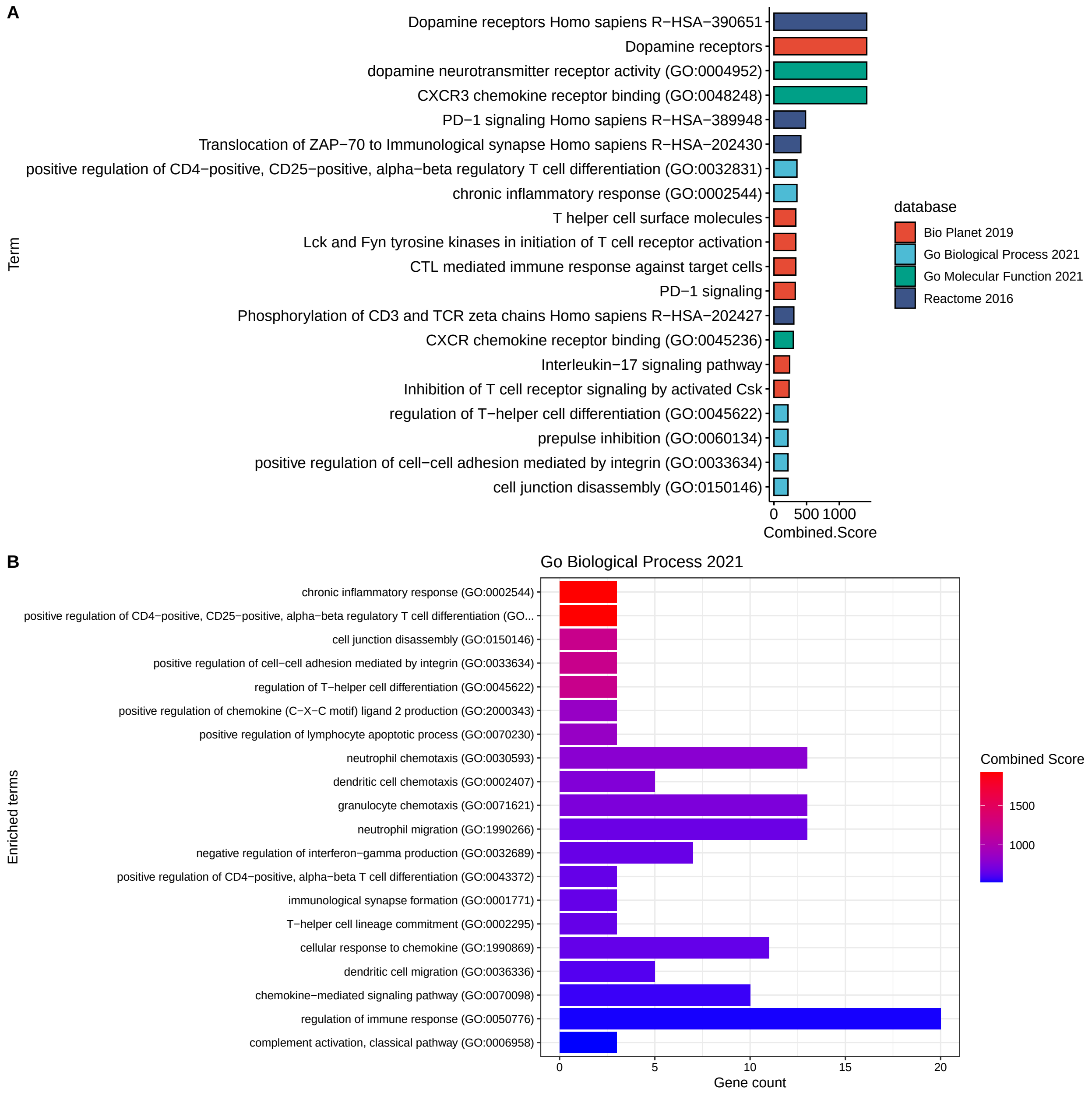

Figure 7: Top 20 results of enrichment using enrichR for Low vs High condition.
(A) Combined enrichment for 8 databases.
(B) Enrichment for downregulated genes for GO Biological Process 2021 database.

Table 1: Results of GSEA by clusterProfiler, pre-ranked by descending order of log2 of fold change for genes differentially expressed in low vs high condition. Genes at the top of the list are thus differentially expressed in the high group, and vice versa. Only significant pathways are shown, out of 95 selected pathways. P adjust ≤ 0.10 is considered significant. NES = Normalized Enrichment Score

Pathway	Set Size	NES	p.adjust	p.sign
Gobp Cell Migration	1431	1.401	p < .001	***
Reactome Neutrophil Degranulation	475	1.488	p < .001	***
Hallmark Hypoxia	199	1.557	p < .001	***
Theilgaard Neutrophil at Skin Wound Dn	226	1.560	p < .001	***
Hay Bone Marrow Neutrophil	439	1.385	p < .001	***
Hallmark Inflammatory Response	199	1.504	p < .001	***
Hp Abnormality of Neutrophils	267	1.461	p < .001	***
Gobp Negative Regulation of Inflammatory Response	144	1.512	p = 0.001	**
Hp Abnormal Neutrophil Count	189	1.460	p = 0.001	**
Gobp Endothelial Cell Migration	216	1.425	p = 0.002	**
Kegg Leukocyte Transendothelial Migration	113	1.500	p = 0.004	**
Gobp Negative Regulation of Neuroinflammatory Response	14	1.663	p = 0.007	**
Wp Neovascularisation Processes	37	1.571	p = 0.023	*
Gobp Sprouting Angiogenesis	127	1.416	p = 0.023	*
Hallmark Tnfa Signaling via Nfkb	200	1.344	p = 0.023	*
Theilgaard Neutrophil at Skin Wound Up	75	1.485	p = 0.026	*
Travaglini Lung Neutrophil Cell	332	1.230	p = 0.038	*
Gobp Negative Regulation of Neutrophil Activation	5	1.497	p = 0.038	*
Sig Chemotaxis	45	1.482	p = 0.047	*
Reactome Cellular Response to Hypoxia	75	1.437	p = 0.055	*
Hay Bone Marrow Immature Neutrophil	189	1.276	p = 0.067	*
Gobp Positive Regulation of Neutrophil Activation	6	-1.687	p = 0.093	*
Wp Angiogenesis	24	1.455	p = 0.093	*

Interestingly, there is no found significant pathway relating to positive regulation of inflammation, but there are significant pathways relating to negative regulation of inflammation, indicating that in the high group, there is negative regulation of inflammation. Similarly, the only signature with a negative enrichment score is “Gobp Positive Regulation of Neutrophil Activation”. It is also notable that the signature *Hp Osteosarcoma* in GSEA is not significantly different (P = 0.38).

5 Discussion

the purpose of the discussion is to summarize your major findings and place them in the context of the current state of knowledge in the literature. When you discuss your own work and that of others, back up your statements with evidence and citations.

- The first part of the discussion should contain a summary of your major findings (usually 2 – 4 points) and a brief summary of the implications of your findings. Ideally, it should make reference to whether you found support for your hypotheses or answered your questions that were placed at the end of the introduction.
- The following paragraphs will then usually describe each of these findings in greater detail, making reference to previous studies.
- Often the discussion will include one or a few paragraphs describing the limitations of your study and the potential for future research.
- Subheadings within the discussion can be useful for orienting the reader to the major themes that are addressed.

5.1 Limitations

- Immune deconvolutions methods do not give the same results, thus they are unclear. Only MCP Counter reports significantly different neutrophil abundance between low and high group.
- Immune deconv paper has 7 limitations (Sturm et al., 2019).
- Z-score and equal mean of groups may not be the best normalization method and way of making groups. Geometric mean (inspired from a package) could have been used but returned 0 value despite all values being positive.
- Literature analysis comparing normalization methods reveals library size normalization (TPM, FPKM, RPKM) methods perform worse than distribution normalization methods (DESeq2, TMM), which are recommended by Zhao et al. (2021). Notably, library size normalization methods assume that the total amount of mRNA per cell is identical in every condition.
- DESeq2's normalized count could have then been used, as it is possible to get it from rawcounts after performing DESeq2 analysis, using two functions, `estimateSizeFactors` and `counts(normalized = TRUE)`. However some immune deconvolution methods still expect TPM as input.
- Using DESeq2's normalized raw count as input, it is inconsistent to then use TPM in order to perform other analysis.
- Perhaps some outliers could have been removed in order to better identify clusters.
- In single-cell, one potential reason that we do not find neutrophils is because chemotherapy induces aplasia which kills off all short lives immune cells.
- Données de littérature peu nombreuses sur ostéosarcomes car cancer rare, cohorte petite et plus petite dans GSE87686
- Nouvelle cohorte de IGR
- Using mean could have resulted in a less precise formation of groups as the distribution of our samples regarding the inflammatory score was not normal. Thus groups based on the median or the quartiles of the median values of inflammatory samples could have been more appropriate.

6 Bibliography

Abrams, Z.B., Johnson, T.S., Huang, K., Payne, P.R.O., and Coombes, K. (2019). A protocol to evaluate RNA sequencing normalization methods. BMC Bioinformatics *20*, 679.

Becht, E., Giraldo, N.A., Lacroix, L., Buttard, B., Elarouci, N., Petitprez, F., Selves, J., Laurent-Puig, P., Sautès-Fridman, C., Fridman, W.H., et al. (2016). Estimating the population abundance of tissue-infiltrating immune and stromal cell populations using gene expression. Genome Biology *17*, 218.

Faget, J., Peters, S., Quantin, X., Meylan, E., and Bonnefoy, N. (2021). Neutrophils in the era of immune checkpoint blockade. Journal for Immunotherapy of Cancer *9*, e002242.

Fu, Y., Jin, Z., Shen, Y., Zhang, Z., Li, M., Liu, Z., He, G., Wu, J., Wen, J., Bao, Q., et al. (2022). Construction and validation of a novel apoptosis-associated prognostic signature related to osteosarcoma metastasis and immune infiltration. Translational Oncology *22*, 101452.

Long, W., Chen, J., Gao, C., Lin, Z., Xie, X., and Dai, H. (2021). Brief review on the roles of neutrophils in cancer development. Journal of Leukocyte Biology *109*, 407–413.

Love, M.I., Huber, W., and Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology *15*, 550.

Sturm, G., Finotello, F., Petitprez, F., Zhang, J.D., Baumbach, J., Fridman, W.H., List, M., and Aneichyk, T. (2019). Comprehensive evaluation of transcriptome-based cell-type quantification methods for immuno-oncology. Bioinformatics *35*, i436–i445.

Xie, Z., Bailey, A., Kuleshov, M.V., Clarke, D.J.B., Evangelista, J.E., Jenkins, S.L., Lachmann, A., Wojciechowicz, M.L., Kropiwnicki, E., Jagodnik, K.M., et al. (2021). Gene Set Knowledge Discovery with Enrichr. Current Protocols *1*, e90.

Zhao, Y., Li, M.-C., Konaté, M.M., Chen, L., Das, B., Karlovich, C., Williams, P.M., Evrard, Y.A., Doroshow, J.H., and McShane, L.M. (2021). TPM, FPKM, or Normalized Counts? A Comparative Study of Quantification Measures for the Analysis of RNA-seq Data from the NCI Patient-Derived Models Repository. Journal of Translational Medicine *19*, 269.

Zhou, Y., Yang, D., Yang, Q., Lv, X., Huang, W., Zhou, Z., Wang, Y., Zhang, Z., Yuan, T., Ding, X., et al. (2020). Single-cell RNA landscape of intratumoral heterogeneity and immunosuppressive microenvironment in advanced osteosarcoma. Nature Communications *11*, 6322.

7 Appendix: code

Analyses were conducted using the R Statistical language (version 4.2.1; R Core Team, 2021) on Windows 11 x64. Code is available at *https://github.com/Minh-AnhHuynh/Osteosarcoma-Project*.

```r
# Load imported data
data_path <- "../TargetOS-Osteosarcoma/03_results/TARGET-OS_GDC_protein_data.tsv"
global_data <- vroom(data_path) %>%
  column_to_rownames("hgnc_symbol") %>%
  t() %>%
  as.data.frame()


# Transform data in Z-scale
global_data_scaled <- global_data %>%
  rownames_to_column(var = "sample") %>%
  dplyr::select(-sample) %>%
  scale(center = TRUE, scale = TRUE) %>%
  bind_cols("sample" = row.names(global_data)) %>%
  column_to_rownames("sample")
global_data_scaled %<>% .[, colSums(is.na(.)) < nrow(.)] # Should be obsolete

remove_genes <- remove_unknown_genes(get_list_unknown_genes = TRUE) %>%
  extract2("Hallmark_Inflammatory_Response")


# Read data and remove unknown genes
hallmark_inflammatory <- vroom("02_data/msigdb_signature/msigdb_signature.tsv") %>%
  filter(
    gs_name == "hallmark_inflammatory_response",
    !gene_symbol %in% remove_genes
  )
hallmark_inflammatory <- global_data_scaled[, hallmark_inflammatory$gene_symbol]


# Construction of groups by mean of Z-score
hallmark_mean_groups <- hallmark_inflammatory %>%
  rownames_to_column("sample") %>%
  pivot_longer(!1, names_to = "markers") %>%
  group_by(sample) %>%
  summarise(
    median = median(value),
    mean = mean(value)
  ) %>%
  mutate(group_mean = as.factor(ntile(mean, 3))) %>%
  arrange(group_mean)
# dplyr::ntile splits input vector into n even groups


# Rename numbers to Low, Medium and High group
hallmark_mean_groups %<>%
  mutate(
    group_mean = str_replace(group_mean, "1", "Low"),
    group_mean = str_replace(group_mean, "2", "Medium"),
    group_mean = str_replace(group_mean, "3", "High"),
  ) %>% write_tsv("03_results/hallmark_mean_groups.tsv")
```

```r
# Compute hallmark_mean_groups for heatmap
mds_data <- hallmark_heatmap(k_cluster = 8, scale = TRUE)
hallmark_cluster <- mds_data %>%
  dplyr::select(sample, cluster) %>%
  arrange(cluster) %>%
  dplyr::rename(cluster_8 = cluster)
hallmark_mean_groups %<>% left_join(hallmark_cluster) %>% arrange(mean)


# Histogram of hallmark mean with MDS clusters
hmg_mds_8 <- ggbarplot(hallmark_mean_groups,
  x = "sample",
  y = "mean",
  xlab = "Sample",
  ylab = "Mean (Z-Score)",
  fill = "cluster_8",
  palette = "npg",
  legend.title = "MDS Cluster",
) +
  theme(
    axis.ticks = element_blank(),
    axis.text.x = element_blank()
  )
ggsave("04_figures/ggplot/hallmark_mean_vs_sample+MDS_8_clusters.png",
  dpi = "retina"
)


## Histogram of hallmark groups of inflammation ====
hmg <- ggbarplot(hallmark_mean_groups,
  x = "sample",
  y = "mean",
  xlab = "Sample",
  ylab = "Mean (Z-Score)",
  fill = "group_mean",
  palette = "npg",
  legend.title = "Group"
) +
  theme(
    axis.ticks = element_blank(),
    axis.text.x = element_blank()
  )
ggsave("04_figures/ggplot/hallmark_mean_vs_sample+group_mean.png",
  dpi = "retina"
)



### Compute kruskal test ####
stat_table <- make_mean_comparison(hallmark_mean_groups, "mean", "group_mean")

# Compute elbow, mds and heatmap
elbow_hi <- elbow_method(hallmark_inflammatory)
```

```r
mds_8 <- hallmark_heatmap(k_cluster = 8, scale = TRUE, return_plot = TRUE)


# Compute heatmap
hallmark_annotation <- hallmark_mean_groups %>%
  column_to_rownames("sample") %>%
  select("Group" = "group_mean") %>%
  `levels<-`(c("Low", "Medium", "High"))

hallmark_inflammatory <- vroom("02_data/msigdb_signature/msigdb_signature.tsv") %>%
  filter(
    gs_name == "hallmark_inflammatory_response",
    !gene_symbol %in% remove_genes
  )
hallmark_inflammatory <- global_data_scaled[, hallmark_inflammatory$gene_symbol] %>%
  t()
hallmark_inflammatory <- hallmark_inflammatory[, match(
  row.names(hallmark_annotation),
  colnames(hallmark_inflammatory)
)]



### Colors for pheatmap group mean annotation ####
gm_colors <- list("Group" = c(
  "High" = ggsci::pal_npg()(3)[1],
  "Medium" = ggsci::pal_npg()(3)[2],
  "Low" = ggsci::pal_npg()(3)[3]
))

ph <- pheatmap(
  hallmark_inflammatory,
  annotation_col = hallmark_annotation,
  main = " ",
  annotation_names_col = FALSE,
  annotation_colors = gm_colors,
  labels_row = "Gene",
  angle_col = "0",
  color = magma(256),
  cluster_cols = TRUE,
  show_colnames = FALSE,
  show_rownames = FALSE,
  cluster_rows = TRUE,
  border_color = NA,
  width = 15,
  filename = glue(
    "04_figures/pheatmap/pheatmap-hallmark_group_mean_cluster_cols.png"
  )
)


ph <- as.ggplot(ph) # Transform pheatmap intp a ggplot object
# p <- ggarrange(elbow_hi, mds_8[[2]], hmg, hmg_mds_8, labels = "AUTO")
#
# suppressWarnings(print(p)) # Used to suppress ggrepel warnings
# ggarrange(ph, labels = ("E"))
```

```r
patchwork <- mds_8[[2]] + elbow_hi + hmg_mds_8 + hmg + ph
layout <- "
AAABBB
CCCDDD
EEEEEE"
patchwork <- patchwork +
  plot_layout(design = layout, heights = c(1, 1, 3)) +
  plot_annotation(tag_levels = "A") &
  theme(plot.tag = element_text(face = "bold"))
# <<hp_osteosarcoma>>
# <<mcp_counter>>
# <<survival_code>>
# <<load_deseq2>>
# <<generate_raw_count>>
# <<deseq2>>


# 1. Setup ----------------------------------------------------------------

librarian::shelf(
  tidyverse,
  glue,
  magrittr,
  msigdbr,
  fgsea,
  clusterProfiler,
  enrichplot,
  snakecase,
  ggridges,
  vroom
)


# Read data
data <-
  vroom("03_results/DESeq2_genes/DEG_results/DESeq2_inflammation_TARGET-OS.tsv")
data_low_vs_high <-
  data %>% filter(df_name == "DESeq2_group_mean_Low_vs_High_df") # %>%
# filter(gene %in% hallmark_inflammatory)
# Note that ICAM4 doesn't appear thus we have a gene list of 199 genes
gene_list_inflam <- data_low_vs_high %>%
  drop_na(log2FoldChange) %>%
  select(gene, log2FoldChange)

# The log2FC of all genes is sorted from high to low, in order to input a pre-ranked in a way that upreg
gene_list_inflam %<>%
  pull(log2FoldChange) %>%
  sort(decreasing = TRUE) %>%
  set_names(gene_list_inflam$gene)


gene_list_diap <- vroom("03_results/DESeq2_genes/DEG_results/DESeq2_diapedesis_TARGET-OS.tsv") %>%
  dplyr::select(gene, log2FoldChange)
gene_list_diap %<>%
  pull(log2FoldChange) %>%
  sort(decreasing = TRUE) %>%
  set_names(gene_list_diap$gene)



# 2. msigdbr -------------------------------------------------------------


# Get pathway from source "01_script/get_msigdb_signature.R"
msigdb_files <-
  list.files("02_data/msigdb_signature/", full.names = TRUE)
all_signatures <- msigdb_files %>%
  map(vroom) %>%
  set_names(basename(tools::file_path_sans_ext(msigdb_files)))


# Make GSEA and GSEA Table
make_gsea_clusterProfiler <-
  function(gene_list,
           msigdb_signature,
           dir_save = "04_figures/GSEA/clusterProfiler/",
           get_gsea_plot = FALSE) {
    # Traditional running enrichment score

    msigdb_signature_df <- bind_rows(msigdb_signature)

    set.seed(123)
    gsea_msig <-
      GSEA(
        gene_list,
        TERM2GENE = msigdb_signature_df,
        minGSSize = 1,
        maxGSSize = 10000,
        verbose = TRUE,
        seed = TRUE,
        by = "fgsea",
        pvalueCutoff = 1
        # adjusted pvalue cutoff to allow every signatures in @result tab
      )

    if (get_gsea_plot == TRUE) {
      # Single GSEA plot for each gene set
      gsea_msig@result[["ID"]] %>%
        map2(seq_along(.), ~ {
          gseaplot2(
            gsea_msig,
            geneSetID = .y,
            title = glue("{.x}"),
            pvalue_table = TRUE
          )
          dir.create(dir_save, showWarnings = FALSE)
          ggsave(glue("{dir_save}GSEA-{.x}.png"), width = 10)
          print(glue("Single GSEA plot saved for {.x}"))
        })
    }
    # Combined GSEA plot for each separate element in msigdb_signature list
    iwalk(msigdb_signature, ~ {
```

```
  signature_group <- distinct(., gs_name) %>% pull()
  signature_index_id <- which(gsea_msig@result[["ID"]] %in% signature_group)
  gseaplot2(
    gsea_msig,
    geneSetID = signature_index_id,
    title = glue("{.y}"),
    pvalue_table = TRUE
  )

  dir.create(glue("{dir_save}combined_GSEA/"), showWarnings = FALSE)
  ggsave(glue("{dir_save}combined_GSEA/GSEA_combined-{.y}.png"),
    width = 12
  )
  print(glue("Combined GSEA plot saved for {.y}"))
})

# Ridgeplot
ridgeplot(gsea_msig, orderBy = "NES", decreasing = TRUE)
dir.create(glue("{dir_save}ridgeplot/"), showWarnings = FALSE)
ggsave(glue("{dir_save}ridgeplot/GSEA-ridgeplot.png"))
print(glue("Ridge plot"))

# Make p significant table for easy visualization
gsea_msig_sign <- gsea_msig@result %>%
  rownames_to_column("Pathway") %>%
  mutate(
    p.sign = case_when(
      p.adjust <= 0.001 ~ "***",
      p.adjust <= 0.01 ~ "**",
      p.adjust <= 0.1 ~ "*",
      TRUE ~ "ns"
    ),
    Pathway = to_title_case(Pathway)
  ) %>%
  dplyr::select(Pathway,
    "Set Size" = setSize,
    NES,
    p.adjust,
    qvalues,
    p.sign
  )
dir.create(glue("{dir_save}GSEA_table/"), showWarnings = FALSE)

# Do a separate table for each sublist with p.sign
iwalk(msigdb_signature, ~ {
  signature_group <- distinct(.x, gs_name) %>%
    pull() %>%
    to_title_case()
  html_file <- glue("{dir_save}GSEA_table/GSEA_table-{.y}.html")
  gsea_msig_sign %>%
    filter(Pathway %in% signature_group) %>%
    kableExtra::kbl("html") %>%
    kableExtra::kable_styling() %>%
    kableExtra::save_kable(html_file)

  # Instead of using save_kable to png directly, which doesn't work, we use a
```

```
      # workaround by saving html first and using webshot to save png, which is what
      # the function is supposed to do automatically anyway
      webshot::webshot(
        html_file,
        glue("{dir_save}GSEA_table/GSEA_table_sign-{.y}.png")
      )
      print(glue("GSEA Table saved for {.y} in {dir_save}GSEA_table/."))
    })
    return(gsea_msig_sign)
  }



gsea_df <- make_gsea_clusterProfiler(gene_list_inflam, all_signatures,
  dir_save = "04_figures/GSEA/inflammation/clusterProfiler/"
)
```

```
## - Session info ---------------------------------------------------------------
##  setting  value
##  version  R version 4.2.1 (2022-06-23 ucrt)
##  os       Windows 10 x64 (build 22621)
##  system   x86_64, mingw32
##  ui       RTerm
##  language (EN)
##  collate  French_France.utf8
##  ctype    French_France.utf8
##  tz       Europe/Paris
##  date     2022-07-19
##  pandoc   2.17.1.1 @ C:/Program Files/RStudio/bin/quarto/bin/ (via rmarkdown)
##
## - Packages -------------------------------------------------------------------
##  package              * version date (UTC) lib source
##  assertthat           * 0.2.1   2019-03-21 [1] CRAN (R 4.2.1)
##  Biobase              * 2.56.0  2022-04-26 [1] Bioconductor
##  BiocGenerics         * 0.42.0  2022-04-26 [1] Bioconductor
##  BiocManager          * 1.30.18 2022-05-18 [1] CRAN (R 4.2.0)
##  bookdown             * 0.27    2022-06-14 [1] CRAN (R 4.2.1)
##  car                  * 3.1-0   2022-06-15 [1] CRAN (R 4.2.1)
##  carData              * 3.0-5   2022-01-06 [1] CRAN (R 4.2.1)
##  data.table           * 1.14.2  2021-09-27 [1] CRAN (R 4.2.1)
##  DESeq2               * 1.36.0  2022-04-26 [1] Bioconductor
##  dplyr                * 1.0.9   2022-04-28 [1] CRAN (R 4.2.0)
##  EnhancedVolcano      * 1.14.0  2022-04-26 [1] Bioconductor
##  forcats              * 0.5.1   2021-01-27 [1] CRAN (R 4.2.1)
##  GenomeInfoDb         * 1.32.2  2022-05-15 [1] Bioconductor
##  GenomicRanges        * 1.48.0  2022-04-26 [1] Bioconductor
##  ggplot2              * 3.3.6   2022-05-03 [1] CRAN (R 4.2.1)
##  ggplotify            * 0.1.0   2021-09-02 [1] CRAN (R 4.2.1)
##  ggpmisc              * 0.4.7   2022-06-15 [1] CRAN (R 4.2.1)
##  ggpp                 * 0.4.4   2022-04-10 [1] CRAN (R 4.2.1)
##  ggpubr               * 0.4.0   2020-06-27 [1] CRAN (R 4.2.1)
##  ggrepel              * 0.9.1   2021-01-15 [1] CRAN (R 4.2.1)
##  ggVennDiagram        * 1.2.0   2021-10-22 [1] CRAN (R 4.2.1)
##  glue                 * 1.6.2   2022-02-24 [1] CRAN (R 4.2.0)
##  gridExtra            * 2.3     2017-09-09 [1] CRAN (R 4.2.1)
##  IRanges              * 2.30.0  2022-04-26 [1] Bioconductor
##  kableExtra           * 1.3.4   2021-02-20 [1] CRAN (R 4.2.1)
##  knitr                * 1.39    2022-04-26 [1] CRAN (R 4.2.0)
##  magrittr             * 2.0.3   2022-03-30 [1] CRAN (R 4.2.0)
##  Matrix               * 1.4-1   2022-03-23 [2] CRAN (R 4.2.1)
##  MatrixGenerics       * 1.8.1   2022-06-30 [1] Bioconductor
##  matrixStats          * 0.62.0  2022-04-19 [1] CRAN (R 4.2.1)
##  moments              * 0.14.1  2022-05-02 [1] CRAN (R 4.2.0)
##  msigdbr              * 7.5.1   2022-03-30 [1] CRAN (R 4.2.1)
##  patchwork            * 1.1.1   2020-12-17 [1] CRAN (R 4.2.1)
##  pheatmap             * 1.0.12  2019-01-04 [1] CRAN (R 4.2.1)
##  purrr                * 0.3.4   2020-04-17 [1] CRAN (R 4.2.0)
##  ragg                 * 1.2.2   2022-02-21 [1] CRAN (R 4.2.1)
##  readr                * 2.1.2   2022-01-30 [1] CRAN (R 4.2.1)
##  rmarkdown            * 2.14    2022-04-25 [1] CRAN (R 4.2.0)
##  RSQLite              * 2.2.14  2022-05-07 [1] CRAN (R 4.2.1)
##  rstatix              * 0.7.0   2021-02-13 [1] CRAN (R 4.2.1)
##  S4Vectors            * 0.34.0  2022-04-26 [1] Bioconductor
##  sessioninfo          * 1.2.2   2021-12-06 [1] CRAN (R 4.2.1)
##  snakecase            * 0.11.0  2019-05-25 [1] CRAN (R 4.2.1)
##  stringr              * 1.4.0   2019-02-10 [1] CRAN (R 4.2.0)
##  SummarizedExperiment * 1.26.1  2022-04-29 [1] Bioconductor
##  tibble               * 3.1.7   2022-05-03 [1] CRAN (R 4.2.0)
##  tidyestimate         * 1.1.0   2021-09-09 [1] CRAN (R 4.2.1)
##  tidyr                * 1.2.0   2022-02-01 [1] CRAN (R 4.2.1)
##  tidyverse            * 1.3.1   2021-04-15 [1] CRAN (R 4.2.1)
##  uwot                 * 0.1.11  2021-12-02 [1] CRAN (R 4.2.1)
##  viridis              * 0.6.2   2021-10-13 [1] CRAN (R 4.2.1)
##  viridisLite          * 0.4.0   2021-04-13 [1] CRAN (R 4.2.1)
##  vroom                * 1.5.7   2021-11-30 [1] CRAN (R 4.2.1)
##
##  [1] C:/Users/Minh-Anh/AppData/Local/R/win-library/4.2
##  [2] C:/Program Files/R/R-4.2.1/library
##
## ------------------------------------------------------------------------------
```

Package	Version	Source
assertthat	0.2.1	CRAN (R 4.2.1)
Biobase	2.56.0	Bioconductor
BiocGenerics	0.42.0	Bioconductor
BiocManager	1.30.18	CRAN (R 4.2.0)
bookdown	0.27	CRAN (R 4.2.1)
car	3.1.0	CRAN (R 4.2.1)
carData	3.0.5	CRAN (R 4.2.1)
data.table	1.14.2	CRAN (R 4.2.1)
DESeq2	1.36.0	Bioconductor
dplyr	1.0.9	CRAN (R 4.2.0)
EnhancedVolcano	1.14.0	Bioconductor
forcats	0.5.1	CRAN (R 4.2.1)
GenomeInfoDb	1.32.2	Bioconductor
GenomicRanges	1.48.0	Bioconductor
ggplot2	3.3.6	CRAN (R 4.2.1)
ggplotify	0.1.0	CRAN (R 4.2.1)
ggpmisc	0.4.7	CRAN (R 4.2.1)
ggpp	0.4.4	CRAN (R 4.2.1)
ggpubr	0.4.0	CRAN (R 4.2.1)

Package	Version	Source
ggrepel	0.9.1	CRAN (R 4.2.1)
ggVennDiagram	1.2.0	CRAN (R 4.2.1)
glue	1.6.2	CRAN (R 4.2.0)
gridExtra	2.3	CRAN (R 4.2.1)
IRanges	2.30.0	Bioconductor
kableExtra	1.3.4	CRAN (R 4.2.1)
knitr	1.39	CRAN (R 4.2.0)
magrittr	2.0.3	CRAN (R 4.2.0)
Matrix	1.4.1	CRAN (R 4.2.1)
MatrixGenerics	1.8.1	Bioconductor
matrixStats	0.62.0	CRAN (R 4.2.1)
moments	0.14.1	CRAN (R 4.2.0)
msigdbr	7.5.1	CRAN (R 4.2.1)
patchwork	1.1.1	CRAN (R 4.2.1)
pheatmap	1.0.12	CRAN (R 4.2.1)
purrr	0.3.4	CRAN (R 4.2.0)
ragg	1.2.2	CRAN (R 4.2.1)
readr	2.1.2	CRAN (R 4.2.1)
rmarkdown	2.14	CRAN (R 4.2.0)
RSQLite	2.2.14	CRAN (R 4.2.1)
rstatix	0.7.0	CRAN (R 4.2.1)
S4Vectors	0.34.0	Bioconductor
sessioninfo	1.2.2	CRAN (R 4.2.1)
snakecase	0.11.0	CRAN (R 4.2.1)
stringr	1.4.0	CRAN (R 4.2.0)
SummarizedExperiment	1.26.1	Bioconductor
tibble	3.1.7	CRAN (R 4.2.0)
tidyestimate	1.1.0	CRAN (R 4.2.1)
tidyr	1.2.0	CRAN (R 4.2.1)
tidyverse	1.3.1	CRAN (R 4.2.1)
uwot	0.1.11	CRAN (R 4.2.1)
viridis	0.6.2	CRAN (R 4.2.1)
viridisLite	0.4.0	CRAN (R 4.2.1)
vroom	1.5.7	CRAN (R 4.2.1)

Summary

English Summary

no more than 250 words for the abstract

- a description of the research question/knowledge gap – what we know and what we don’t know
- how your research has attempted to fill this gap
- a brief description of the methods
- brief results
- key conclusions that put the research into a larger context