Chapter 3 - Fibroblast Stiffness & Heterogeneity, and YAP1 Activation Score

Methods

Tissue Stiffness Classification Based on Fibroblast Abundance Samples were classified into high- or low-stiffness groups using fibroblast abundance as a surrogate metric. The percentage of fibroblasts per sample was calculated relative to total cells. A median cutoff was applied to dichotomise samples into "high_stiff" (above median) or "low_stiff" (below median) categories. This classification was appended to metadata for downstream analyses.

YAP1 and Mutant Activity Quantification Gene signatures for wild-type YAP1, constitutively active YAP1-5SA, and transcriptionally deficient YAP1-S94A were curated from a predefined list (Excel file). Module scores for each signature were computed using Seurat's `AddModuleScore` with 40 control genes for normalization. Scores were rescaled to a 0–1 range to enable cross-sample comparisons.

Tumor Heterogeneity Analysis Intra-tumoral transcriptional heterogeneity was assessed using the Gini coefficient, a statistical measure traditionally applied in economics to quantify income inequality, which we adapted to evaluate the uneven distribution of YAP1 activity across cancer cells. The Gini coefficient ranges from 0 (perfect equality, where all cells exhibit identical expression) to 1 (maximal inequality, where a single cell dominates expression). This metric was selected over alternatives (e.g., Shannon entropy, variance) due to its:

1. Sensitivity to skewed distributions: The Gini coefficient robustly captures dominance of high- or low-YAP1 subpopulations, which are hallmarks of clonal selection in tumors.
2. Scale invariance: Unlike variance, it is unaffected by absolute expression levels, enabling comparison across samples with differing baseline YAP1 activity.
3. Biological interpretability: Higher Gini values reflect greater functional diversity within cancer cell populations, a feature linked to therapeutic resistance and metastasis.

To compute this, normalized YAP1 scores (rescaled 0–1) were subset to ovarian cancer cells, ensuring the metric specifically reflects tumor-intrinsic heterogeneity. The Gini coefficient was calculated per sample using the DescTools R package. This approach aligns with prior studies quantifying transcriptional heterogeneity in cancer (e.g., Patel et al., *Science*, 2014; Tirosh et al., *Nature*, 2016) and provides a parsimonious framework to test associations between mechanical stiffness and tumor plasticity **Spatial Data Integration and Visualization** Computed features (cell type annotations, stiffness class, YAP scores, heterogeneity metrics) were mapped back to individual Visium/Xenium spatial objects. Metadata harmonization ensured alignment between merged and sample-specific datasets. SpatialFeaturePlot (Seurat) visualized normalized YAP1-5SA activity across tissue coordinates, with point size scaled for clarity. Processed objects were archived for reproducibility.

Codes

Add Fibroblast Stiffness Score

```
fibro <- as.data.frame.matrix(table(merged$sample,
merged$cell.types))
fibro$All.Cells <- rowSums(fibro)
fibro$Fibroblast.Pct <- 100*(fibro$Fibroblast / fibro$All.Cells)
cutoff <- median(fibro$Fibroblast.Pct)
fibro$Stiff <- ifelse(fibro$Fibroblast.Pct > cutoff, "high_stiff",
"low_stiff")
merged$Stiffness <- ifelse(merged$sample %in% rownames(fibro)
[fibro$Stiff == "high_stiff"], "high_stiff", "low_stiff")
```

Add YAP1 and YAP1-mutant activation scores

```
# load in the gene features
YAP1 <- readxl::read_xlsx("~/Desktop/YAP1/data/YAP1.xlsx", col_names
= F)
YAP <- YAP1[1, 3:202] %>% unlist %>% unname %>%
intersect(rownames(merged))
YAP_S94A <- YAP1[2, 3:202] %>% unlist %>% unname %>%
intersect(rownames(merged))
YAP_5SA <- YAP1[3, 3:202] %>% unlist %>% unname %>%
intersect(rownames(merged))

# add original module scores
merged <- AddModuleScore(merged, features = list(YAP), name = "YAP1",
ctrl = 40)
merged <- AddModuleScore(merged, features = list(YAP_5SA), name =
"YAP1_5SA", ctrl = 40)
merged <- AddModuleScore(merged, features = list(YAP_S94A), name =
"YAP1_S94A", ctrl = 40)

# rename the columns
```

```
merged$YAP1 <- merged$YAP11; merged$YAP11 <- NULL
merged$YAP1_5SA <- merged$YAP1_5SA1; merged$YAP1_5SA1 <- NULL
merged$YAP1_S94A <- merged$YAP1_S94A1; merged$YAP1_S94A1 <- NULL

# scale the scores
merged$YAP1.norm <- rescale(merged$YAP1, to = c(0, 1))
merged$YAP1_5SA.norm <- rescale(merged$YAP1_5SA, to = c(0, 1))
merged$YAP1_S94A.norm <- rescale(merged$YAP1_S94A, to = c(0, 1))
```

Add Cancer Heterogeneity

```
meta <- merged@meta.data
meta$cells <- rownames(meta)
df <- data.frame(row.names = unique(meta$sample),
YAP1.hetero = rep(NA, length(unique(meta$sample))),
YAP1_5SA.hetero = rep(NA, length(unique(meta$sample))),
YAP1_S94A.hetero = rep(NA, length(unique(meta$sample))))
for(i in unique(meta$sample)){
meta_sub <- meta %>% dplyr::filter(sample == i) %>%
dplyr::filter(cell.types == "Ovarian.cancer.cell")
df[i, "YAP1.hetero"] <- DescTools::Gini(meta_sub$YAP1.norm)
df[i, "YAP1_5SA.hetero"] <- DescTools::Gini(meta_sub$YAP1.norm)
df[i, "YAP1_S94A.hetero"] <- DescTools::Gini(meta_sub$YAP1.norm)
}
meta <- merge(meta, df, by.x = "sample", by.y = "row.names") %>%
tibble::column_to_rownames(var = "cells")
merged@meta.data <- meta[colnames(merged), ]
```

Add the Metadata Back to each Sub-Sample

```
directories <- list.dirs("~/Desktop/YAP1/data/spatial/GSE211956",
recursive = F)
meta <- merged@meta.data
for(i in directories){
setwd(i)
spatial <- readRDS("output/seurat.rds")
spatial@meta.data$YAP1 <- NULL
spatial@meta.data$YAP1_S94A <- NULL
spatial@meta.data$YAP1_5SA <- NULL
sample_id <-
gsub("/Users/polly_hung/Desktop/YAP1/data/spatial/GSE211956/", "", i)
sub_meta <- meta %>% dplyr::filter(sample == sample_id) %>%
dplyr::select("cell.types", "Stiffness", "YAP1", "YAP1_5SA",
"YAP1_S94A", "YAP1.norm", "YAP1_5SA.norm", "YAP1_S94A.norm",
"YAP1.hetero")
rownames(sub_meta) <- gsub(sample_id, "", rownames(sub_meta))
rownames(sub_meta) <- gsub("_", "", rownames(sub_meta))
sub_meta2 <- merge(spatial@meta.data, sub_meta, by = "row.names") %>%
tibble::column_to_rownames(var = "Row.names")
sub_meta2 <- sub_meta2[rownames(spatial@meta.data), ]
spatial@meta.data <- sub_meta2
SpatialFeaturePlot(spatial, features = "YAP1_5SA.norm",
pt.size.factor = 3, image.alpha = 0)
saveRDS(spatial, "output/labelled2.rds")

}
```