Agentic AI for Adaptive Pharmacogenomic Biomarker Discovery in Cancer

```r
suppressPackageStartupMessages({
  library(SummarizedExperiment)
  library(tidyverse)
  library(ComplexHeatmap)
  library(RColorBrewer)
  library(circlize)
  library(here)
})

set.seed(101)
```

Applications in Data Exploration

Curated clinical genomic datasets provide an opportunity for exploratory analysis to assist in experiment and analysis design. While SummarizedExperiment objects are ideal for storing molecular profiles and associated metadata, navitgating these objets can be time-consuming and challenging. An agentic framework can be leveraged to facilitate data extraction and perform quick exploratory analyses to identify preliminary associations between variables of interest.

Case scenario 1:

High tumour cellularity can be associated with more aggressive cancers. Gene signatures that are surrogate markers of tumour cellularity may be predictive of response to immunotherapies.

The following dataset was downloaded from orcestra.ca. The dataset is stored as a `RangedSummarizedExperiment`.
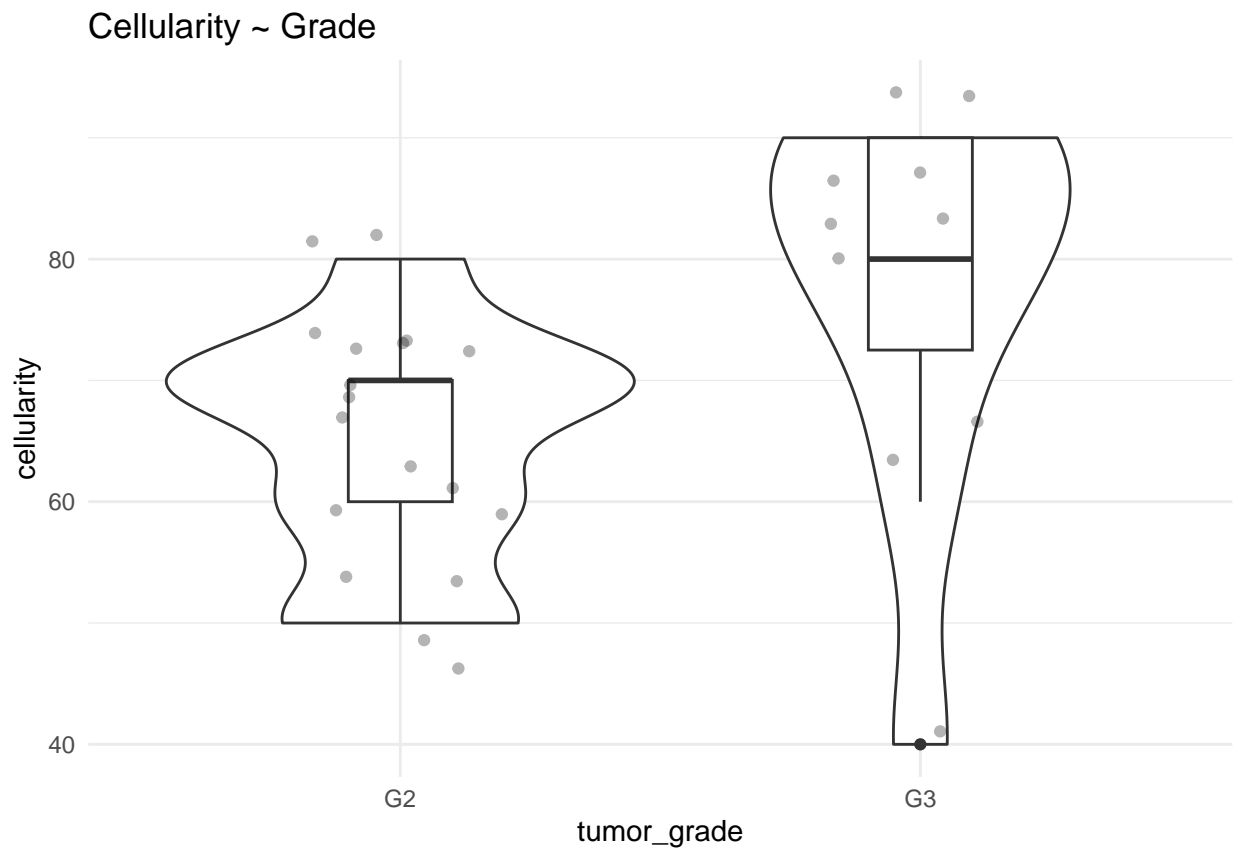
```r
pdac <- readRDS(here("./data/rawdata/PDAC_Haider.rds"))
pdac
```

```
## class: RangedSummarizedExperiment
## dim: 16171 28
## metadata(3): '' annotation protocolData
## assays(1): ''
## rownames(16171): ENSG00000237613 ENSG00000273547 ... ENSG00000240450
##   ENSG00000172288
## rowData names(6): gene_id gene_name ... symbol entrezid
## colnames(28): GSM1363848 GSM1363849 ... GSM1363878 GSM1363879
## colData names(8): sample_id cellularity ... sample_type
##   unique_patient_id
```

`. . .`

Task: show the associations between tumour cellularity and tumour grade

```
# get clinical metadata of samples
sample_meta <- pdac@colData
sample_meta$cellularity <- as.numeric(sample_meta$cellularity)
sample_meta$age <- as.numeric(sample_meta$age)

# plot distribution of tumour cellularity by tumour grade
ggplot(sample_meta, aes(x = tumor_grade, y = cellularity)) +
  geom_violin() +
  geom_boxplot(width = 0.2) +
  geom_jitter(width = 0.2, alpha = 0.3) +
  theme_minimal() + labs(title = "Cellularity ~ Grade")
```


. . .

Task: identify a list of genes that demonstrate association with cellularity

```
# get gene expression
rna <- assay(pdac) |> t() |> as.data.frame()

# compute pearson's correlation coefficient
results <- sapply(rna, function(gene) {
  res <- cor.test(gene, sample_meta$cellularity)
  c(estimate = res$estimate,
    p_value = res$p.value)
}) |> t() |> as.data.frame()
cellularity_genes <- results[abs(results$estimate.cor) > 0.6 & results$p_value < 0.05,]
```

Task: visualize distribution of identified genes and tumour cellularity

```r
# keep cellularity genes
cellularity_rna <- rna[,rownames(cellularity_genes)] |> log2()

# plot heatmap of cellularity ~ gene expression
row_ha <- rowAnnotation(
  cellularity = sample_meta$cellularity,
  annotation_name_gp = gpar(fontsize = 8),
  col = list(cellularity = colorRamp2(
    breaks = seq(min(sample_meta$cellularity),
                 max(sample_meta$cellularity),
                 length.out = 9),
    colors = brewer.pal(9, "Purples")
  ))
)
ht <- Heatmap(
    cellularity_rna, name = "Gene\nExpression",
    right_annotation = row_ha,
    show_column_names = FALSE,
    col = c("#BEC5D1", "#5B618A"),
    rect_gp = gpar(col = "grey80", lwd = 0.5),
    row_names_gp = gpar(fontsize = 8)
) |> suppressWarnings()
print(ht)
```