Project Report 1: Mutation Analysis

Group 15 - Liver Carcinoma

2025-11-21

1. Key Findings

Should summarize the key findings of the analysis in up to 5 bullet points.

2. Overall Analysis Workflow

Give a description of your analysis workflow including the algorithms (approaches) you have tried (or will try) and the patient variables you have investigated. This is the most important part of this report. You are required to include a flowchart of all the analysis.

From The Cancer Genome Atlas (TCGA), we imported the following datasets:

```
clinical <- read.delim("data_clinical_patient.txt", header = TRUE,
                       sep = "\t", stringsAsFactors = FALSE)
mutations <- read.delim("data_mutations.txt", header = TRUE,
                       sep = "\t", stringsAsFactors = FALSE)
```

2.1. Data Preprocessing

a. Standardize Patient IDs

In clinical dataset, patient IDs have the form of "TCGA-2V-A95S" while mutation dataset contains tumour sample barcode which has the form of "TCGA-2V-A95S-01." As a result, we will standardized patient IDs such that it is consistent by taking only the first 12 characters. We will add a column with the standardized patient IDs to each of the dataset. Then take only patients that are present in both datasets.

```
clinical$Patient_ID <- clinical$X.Patient.Identifier
mutations$Patient_ID <- substr(mutations$Tumor_Sample_Barcode, 1, 12)

common_ids <- intersect(clinical$Patient_ID, mutations$Patient_ID)
clinical <- clinical[clinical$Patient_ID %in% common_ids, ]
mutations <- mutations[mutations$Patient_ID %in% common_ids, ]
```

b. Filter for Non-Synonymous Mutations

We will use Variant_Classification columns to see the type of mutations presented in the dataset:

```
unique(mutations$Variant_Classification)
```

```
##  [1] "Missense_Mutation"      "5'UTR"                  "Silent"
##  [4] "RNA"                    "Intron"                 "In_Frame_Del"
##  [7] "Frame_Shift_Del"        "Splice_Site"            "Nonsense_Mutation"
## [10] "Splice_Region"          "Frame_Shift_Ins"        "Translation_Start_Site"
## [13] "Nonstop_Mutation"       "In_Frame_Ins"           "3'UTR"
## [16] "3'Flank"                "5'Flank"
```

We will keep the non-synonymous mutations which are those that mutate the coding regions.

```
nonsynonymous <- c(
  "Missense_Mutation", "Nonsense_Mutation",
  "Frame_Shift_Del", "Frame_Shift_Ins",
  "In_Frame_Del", "In_Frame_Ins",
  "Splice_Site", "Splice_Region", "Nonstop_Mutation",
  "Translation_Start_Site"
)
mut_filtered <- mutations[mutations$Variant_Classification %in% nonsynonymous, ]
```

c. Gene x Patient Matrix

The Gene x Patient matrix turns raw muation calls into a form that allows clustering and other downstream data analysis methods. The matrix's row will be genes and its column is patient. If a mutation is present in a patient, then that cell will contain the value of one. Note that we are building a matrix for non-synonymous genes only.

```
genes <- unique(mut_filtered$Hugo_Symbol)
patients <- unique(mut_filtered$Patient_ID)

mut_matrix <- matrix(0, nrow = length(genes), ncol = length(patients))
rownames(mut_matrix) <- genes
colnames(mut_matrix) <- patients

for (i in seq_len(nrow(mut_filtered))) {
    g <- mut_filtered$Hugo_Symbol[i]
    p <- mut_filtered$Patient_ID[i]
    mut_matrix[g, p] <- 1
}
```

d. Tumor Mutation Burden (TMB)

We will compute TMB which is the total number of mutated genes per patient. This can give us more insights into survival in later in our analysis as TMB can associate with mutation-driven cancer, immune response or prognosis.

```
tmb <- colSums(mut_matrix)
```

e. Transform Clinical Variables

Many clinical variables are in string form when we want them to be in numeric form. We will transform these variables to help with downstream data analysis. We will create a function to remove labels like 1:Progressed and keep only number, then convert into numeric form.

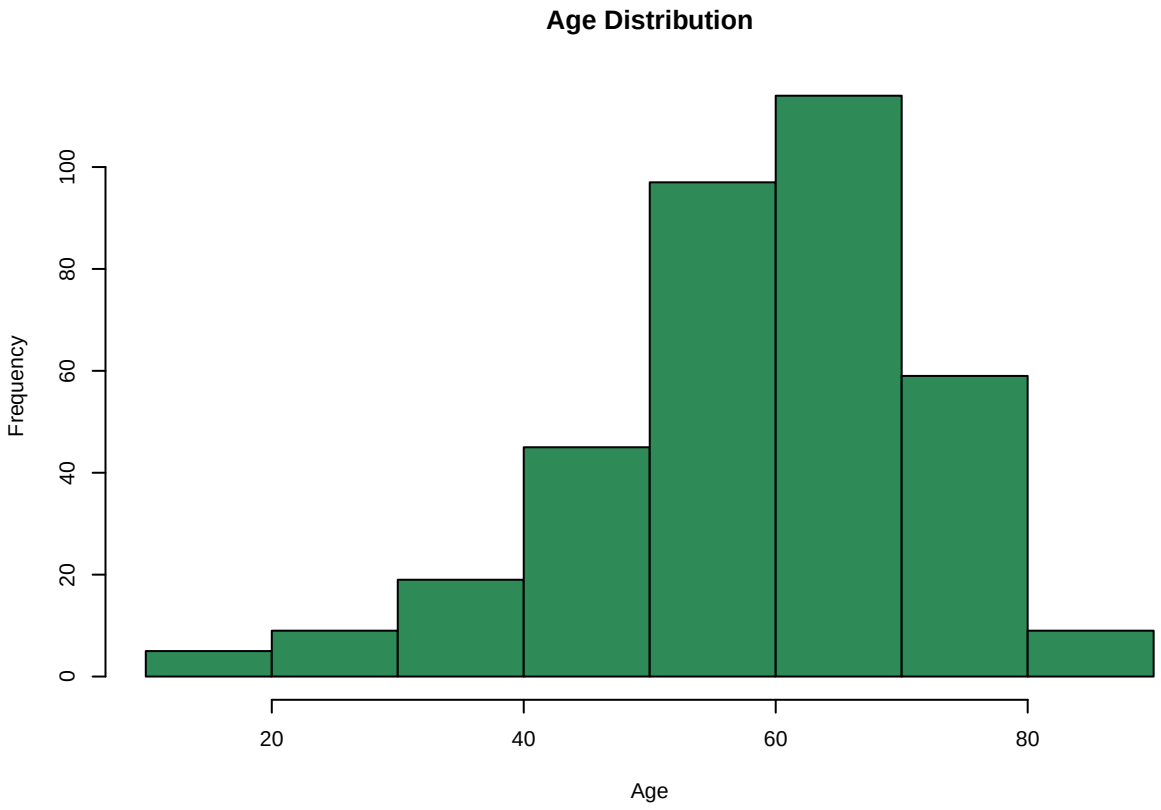
```r
parse_status <- function(x) {
    as.numeric(sub(":.*", "", x))
}

# Convert time strings ("63.72") to numeric
parse_time <- function(x) {
    as.numeric(trimws(x))
}

# Age
clinical$Diagnosis.Age <- parse_time(clinical$Diagnosis.Age)
# Survival Times
clinical$Overall_Time  <- parse_time(clinical$Overall.Survival..Months.)
clinical$DSS_Time      <- parse_time(clinical$Months.of.disease.specific.survival)
clinical$DFS_Time      <- parse_time(clinical$Disease.Free..Months.)
clinical$PFS_Time      <- parse_time(clinical$Progress.Free.Survival..Months.)
# Survival Status
clinical$Overall_Status <- parse_status(clinical$Overall.Survival.Status)
clinical$DSS_Status     <- parse_status(clinical$Disease.specific.Survival.status)
clinical$DFS_Status     <- parse_status(clinical$Disease.Free.Status)
clinical$PFS_Status     <- parse_status(clinical$Progression.Free.Status)
```

2.2. Clinical Exploration

a. Age Distribution

We will use Diagnosis.Age to contruct a histogram on the distribution of patient age.

```r
age_data <- clinical[!is.na(clinical$Diagnosis.Age), ]
par(cex = 0.67)
hist(age_data$Diagnosis.Age,
     main = "Age Distribution",
     xlab = "Age",
     col = "seagreen", border = "black")
```

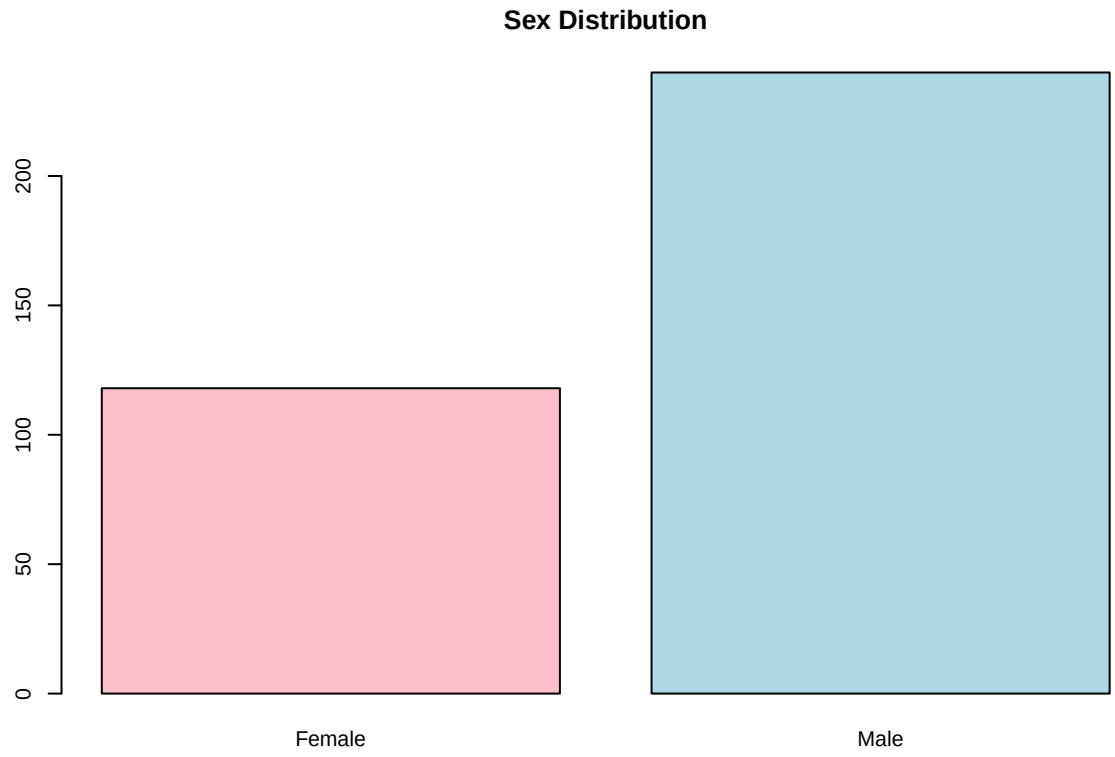


b. Sex Distribution We will contruct a barplot on sex distribution.

```
table(clinical$Sex)
```

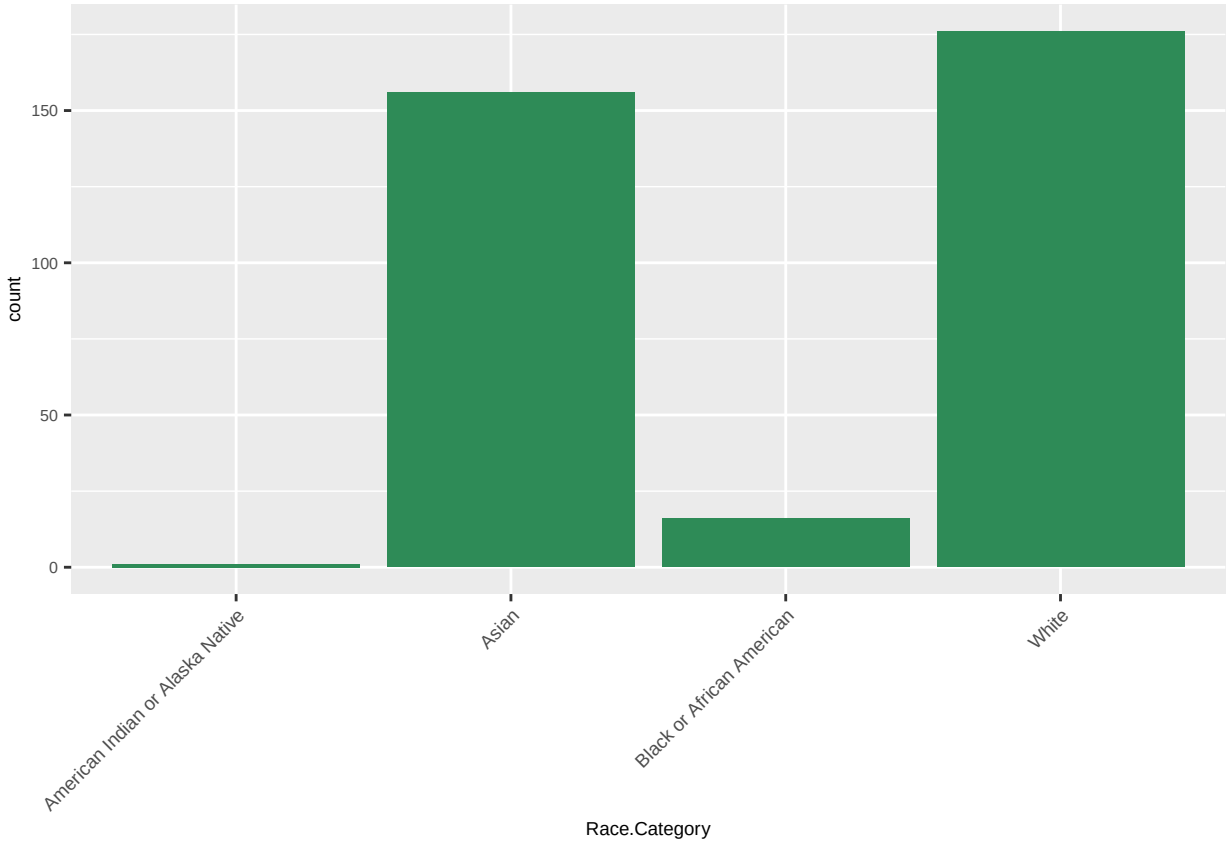
```
##
## Female   Male
##    118    240
```

```
par(cex = 0.67)
barplot(table(clinical$Sex),
        main = "Sex Distribution",
        col = c("pink", "lightblue"))
```



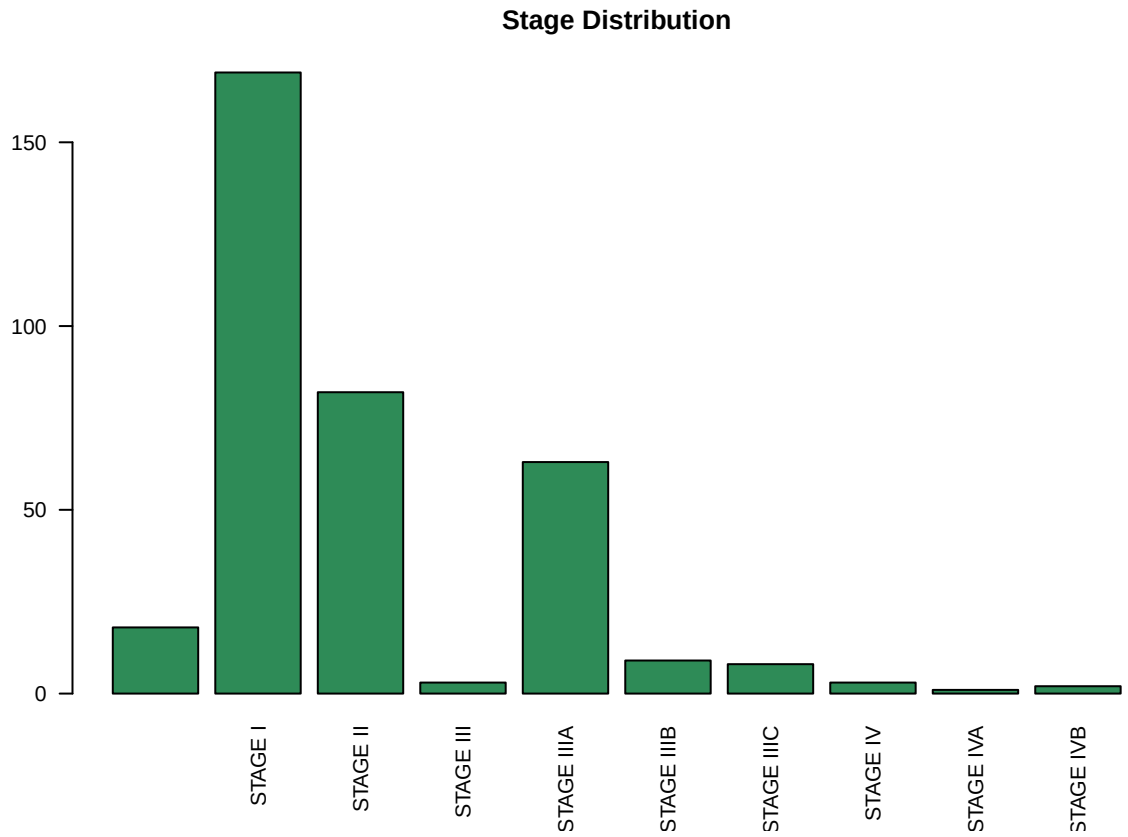
c. Race Distribution

```
library(ggplot2)

ggplot(subset(clinical, Race.Category != "" & !is.na(Race.Category)),
       aes(x = Race.Category)) +
  geom_bar(fill = "seagreen") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 7.3),
    text = element_text(size = 7.3))
```



d. Stage Distribution

```
par(cex = 0.67)
barplot(table(clinical$Neoplasm.Disease.Stage.American.Joint.Committee.on.Cancer.Code),
        main = "Stage Distribution", las = 2, col = "seagreen")
```



e. Survival Summaries

```
summary(clinical$Overall_Time)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   11.21   19.33   26.50   35.67  120.82       1
```

```
summary(clinical$DFS_Time)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   5.638  13.085  19.571  25.085 120.821      51
```

```
summary(clinical$PFS_Time)
```

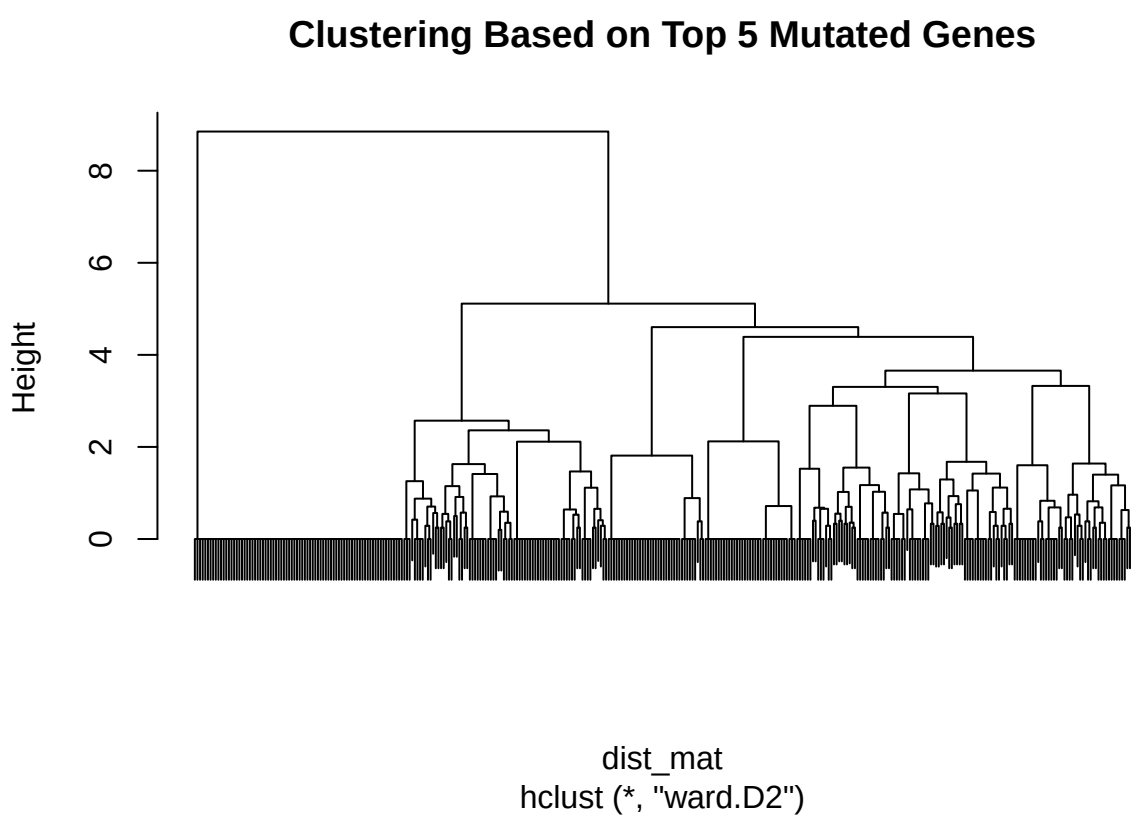
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   4.373  12.033  18.017  22.619 120.821       1
```

```
summary(clinical$DSS_Time)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   11.21   19.33   26.50   35.67  120.82       1
```

2.3. Mutation-Based Clusters

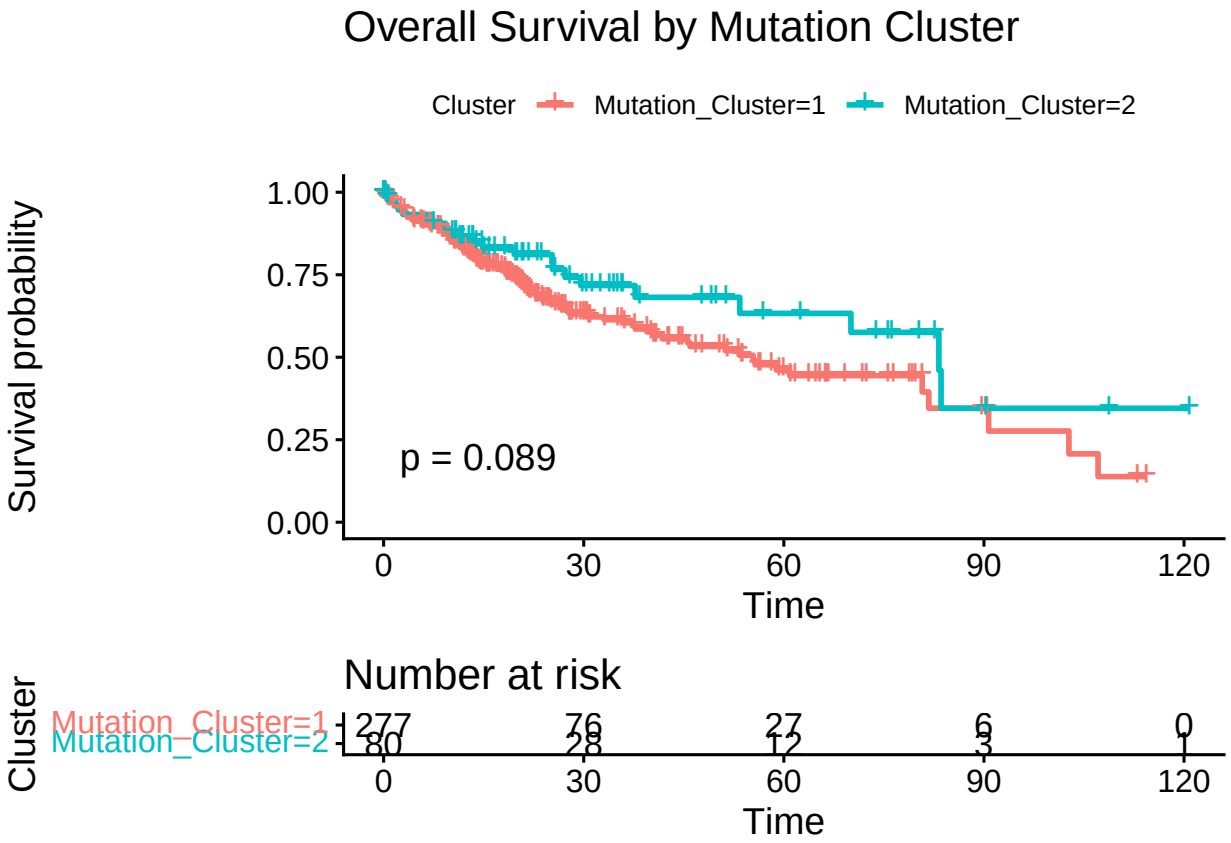
```r
# Compute mutation frequency per gene
gene_counts <- rowSums(mut_matrix)

# Select top 20 most mutated genes
top_genes <- names(sort(gene_counts, decreasing = TRUE))[1:10]

# Subset mutation matrix
mut_top <- mut_matrix[top_genes, ]

# Match clinical rows to mutation matrix columns
clinical2 <- clinical[match(colnames(mut_top), clinical$Patient_ID), ]

# Check alignment
stopifnot(all(clinical2$Patient_ID == colnames(mut_top)))

dist_mat <- dist(t(mut_top), method = "binary")
hc <- hclust(dist_mat, method = "ward.D2")

plot(hc, labels = FALSE,
     main = "Clustering Based on Top 5 Mutated Genes")
```



```r
clusters <- cutree(hc, k = 2)
clinical2$Mutation_Cluster <- clusters
```

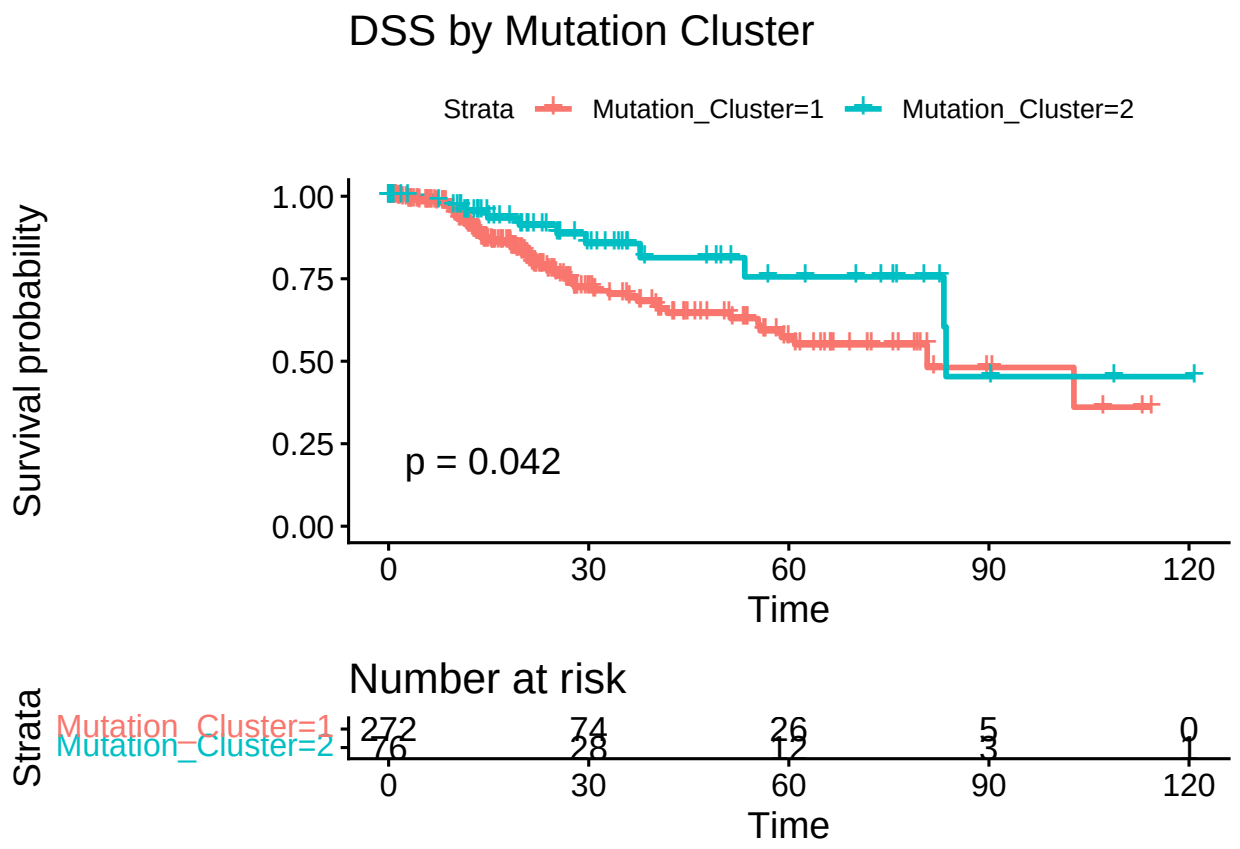
```r
os_data <- clinical2[, c("Overall_Time", "Overall_Status", "Mutation_Cluster")]
os_data <- na.omit(os_data)

fit_os <- survfit(
  Surv(os_data$Overall_Time, os_data$Overall_Status) ~ Mutation_Cluster,
  data = os_data
)

ggsurvplot(fit_os, data = os_data,
           pval = TRUE, risk.table = TRUE,
           title = "Overall Survival by Mutation Cluster",
           legend.title = "Cluster")
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## i The deprecated feature was likely used in the ggpubr package.
##   Please report the issue at <https://github.com/kassambara/ggpubr/issues>.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.

## Ignoring unknown labels:
## * colour : "Cluster"
```



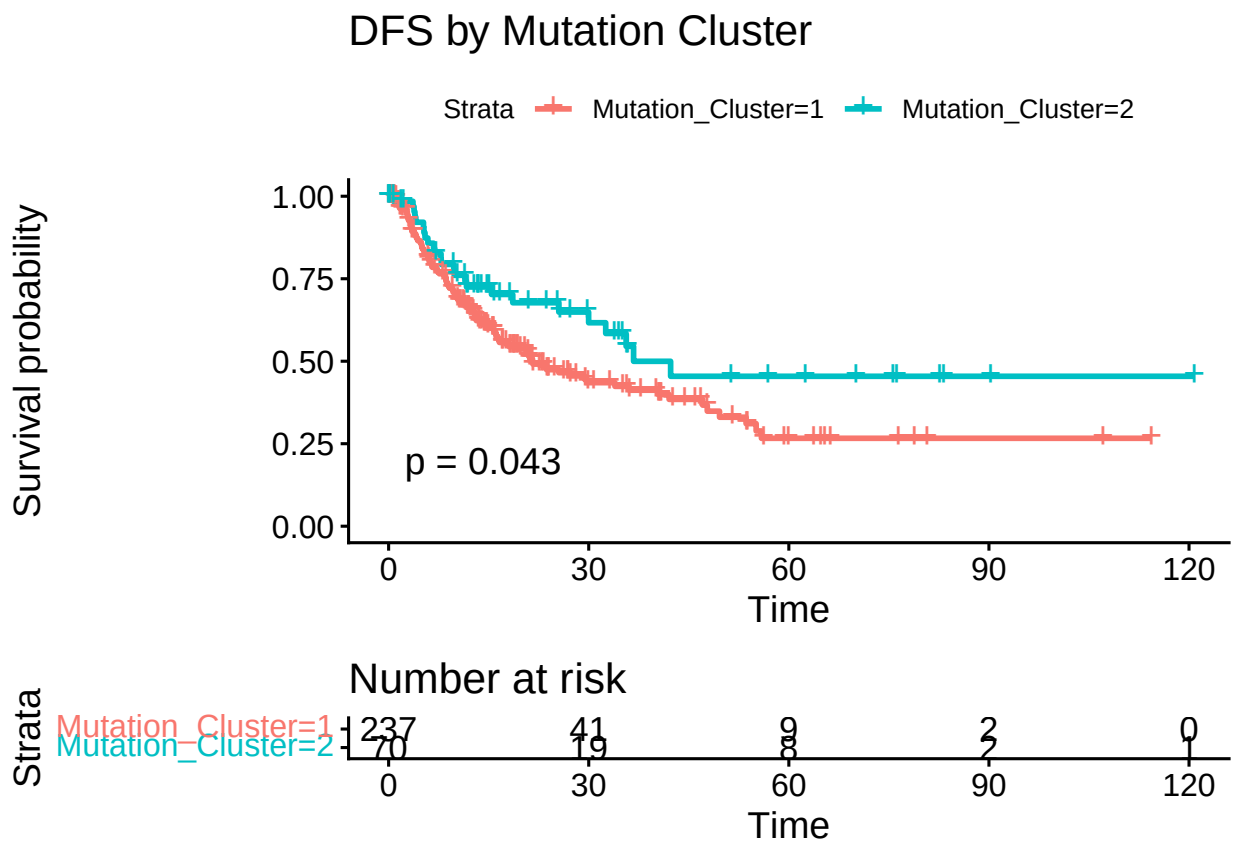

```r
dss_data <- clinical2[, c("DSS_Time", "DSS_Status", "Mutation_Cluster")]
dss_data <- na.omit(dss_data)

fit_dss <- survfit(
  Surv(DSS_Time, DSS_Status) ~ Mutation_Cluster,
  data = dss_data
)

ggsurvplot(fit_dss, data = dss_data,
           pval = TRUE, risk.table = TRUE,
           title = "DSS by Mutation Cluster")
```

```
## Ignoring unknown labels:
## * colour : "Strata"
```



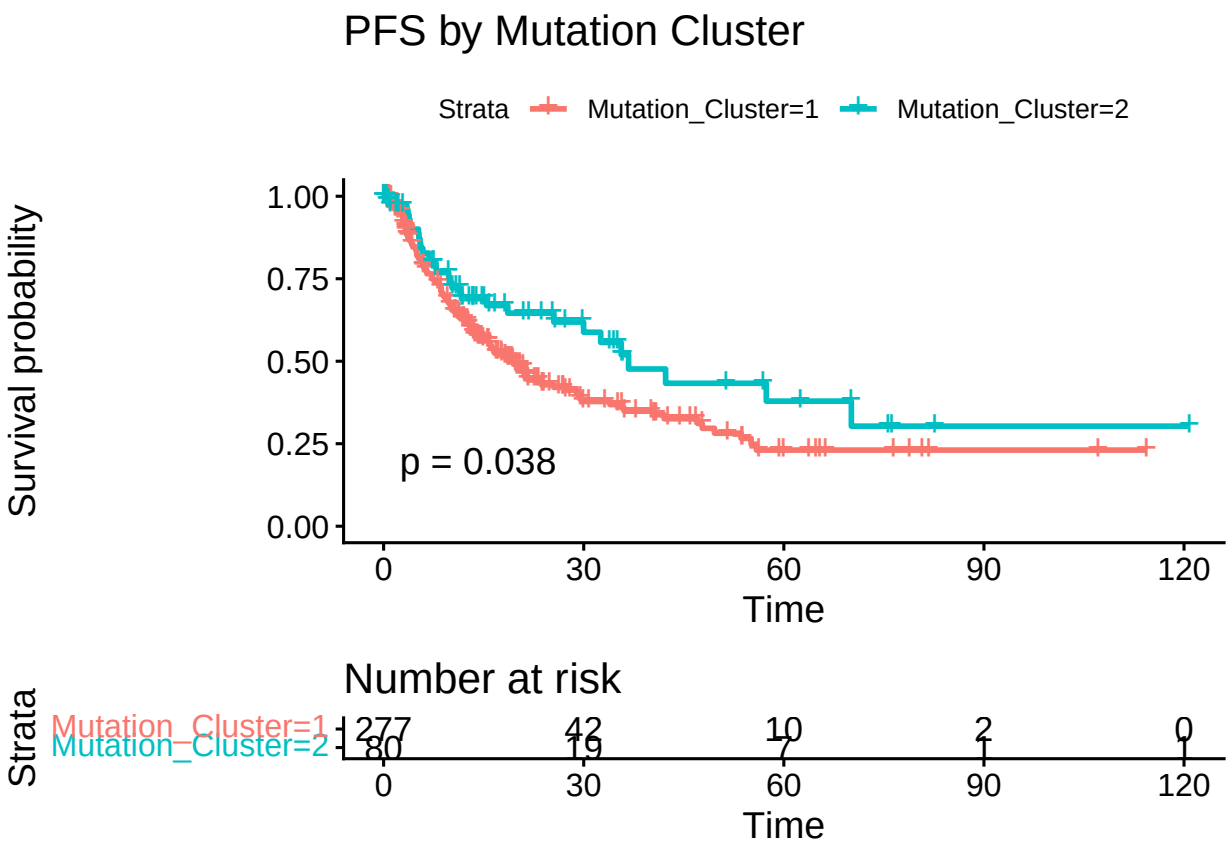

```
dfs_data <- clinical2[, c("DFS_Time", "DFS_Status", "Mutation_Cluster")]
dfs_data <- na.omit(dfs_data)

fit_dfs <- survfit(
  Surv(DFS_Time, DFS_Status) ~ Mutation_Cluster,
  data = dfs_data
)

ggsurvplot(fit_dfs, data = dfs_data,
           pval = TRUE, risk.table = TRUE,
           title = "DFS by Mutation Cluster")
```

```
## Ignoring unknown labels:
## * colour : "Strata"
```




```r
pfs_data <- clinical2[, c("PFS_Time", "PFS_Status", "Mutation_Cluster")]
pfs_data <- na.omit(pfs_data)

fit_pfs <- survfit(
  Surv(PFS_Time, PFS_Status) ~ Mutation_Cluster,
  data = pfs_data
)

ggsurvplot(fit_pfs, data = pfs_data,
           pval = TRUE, risk.table = TRUE,
           title = "PFS by Mutation Cluster")
```

```
## Ignoring unknown labels:
## * colour : "Strata"
```



3. Results

Should provide an itemized list of investigations and analysis you have performed, along with the findings of such an analysis. Every result or claim needs to be backed up by R markdown code as well as accompanying outputs and figures/flowcharts. Organize every investigation into a sub-section in the results section. You would need to include at least 5 investigations in the results section. The quality (rather than quantity) of the analysis will be key in grading.

4. Challenges

Use this section to address any difficulties you are encountering, or expect you will encounter as you make progress with the project. This may include limitations of a current approach or a lack of meaningful results. Please explain why you believe this problem is occurring, and highlight any limitations in your data you have found.