Untitled

2025-07-29

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
gene_expr <- read_csv("QBS103_GSE157103_genes.csv")
```

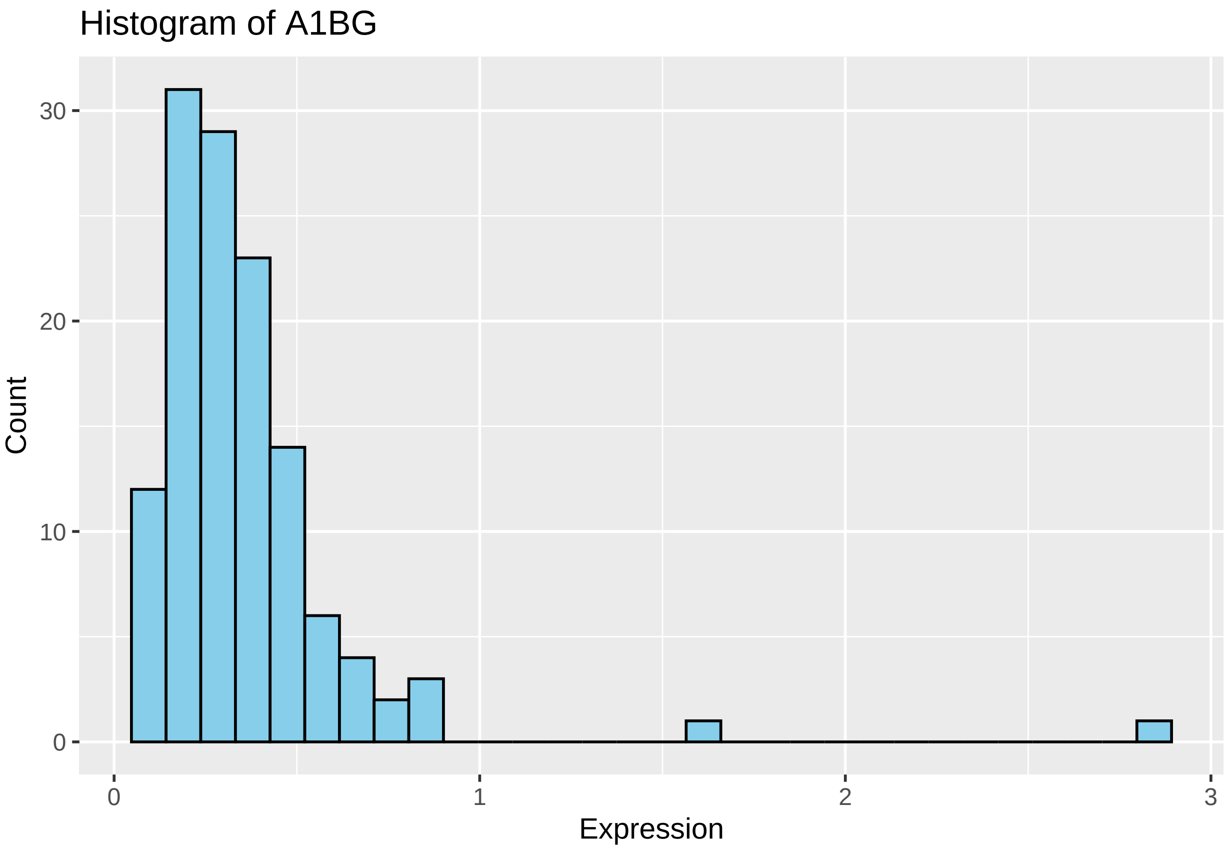
```
## New names:
## Rows: 100 Columns: 127
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (1): ...1 dbl (126): COVID_01_39y_male_NonICU, COVID_02_63y_male_NonICU,
## COVID_03_33y_...
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

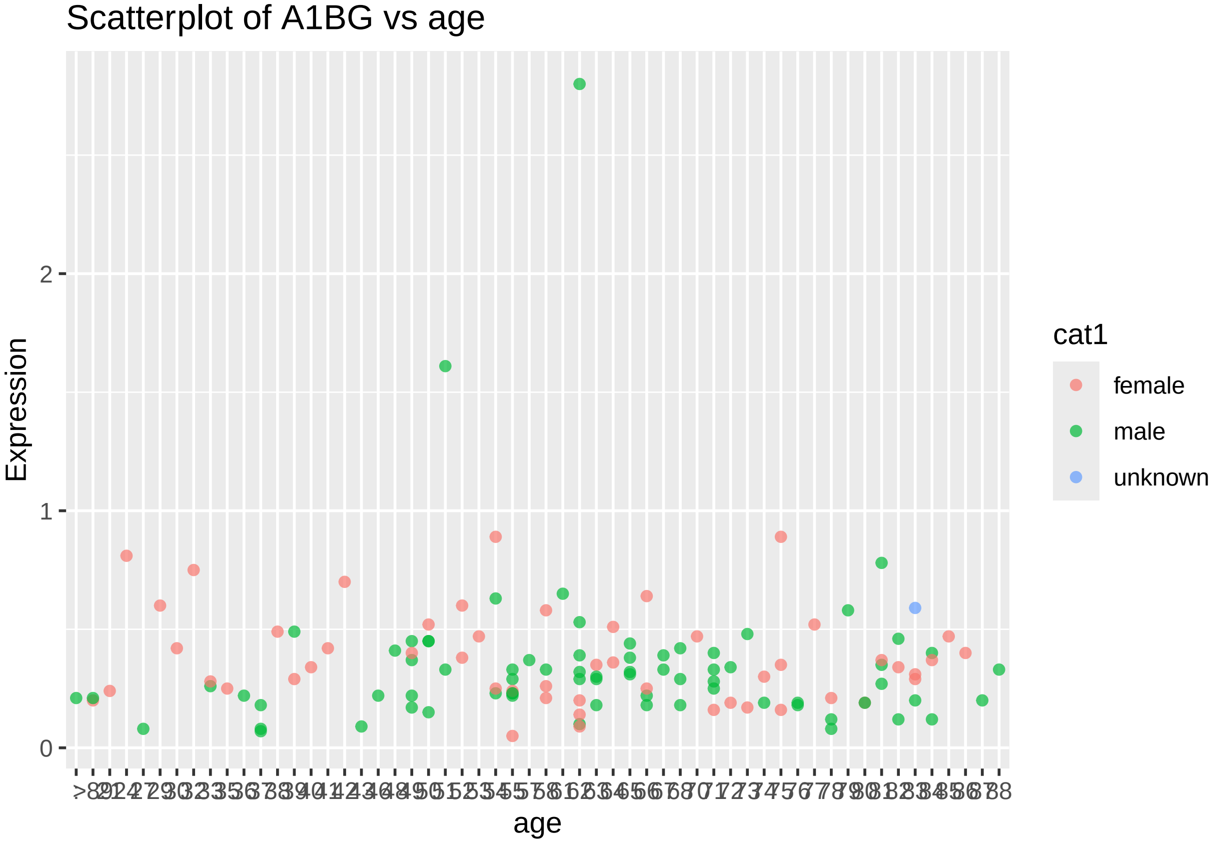
```
metadata <- read_csv("QBS103_GSE157103_series_matrix-1.csv")
```

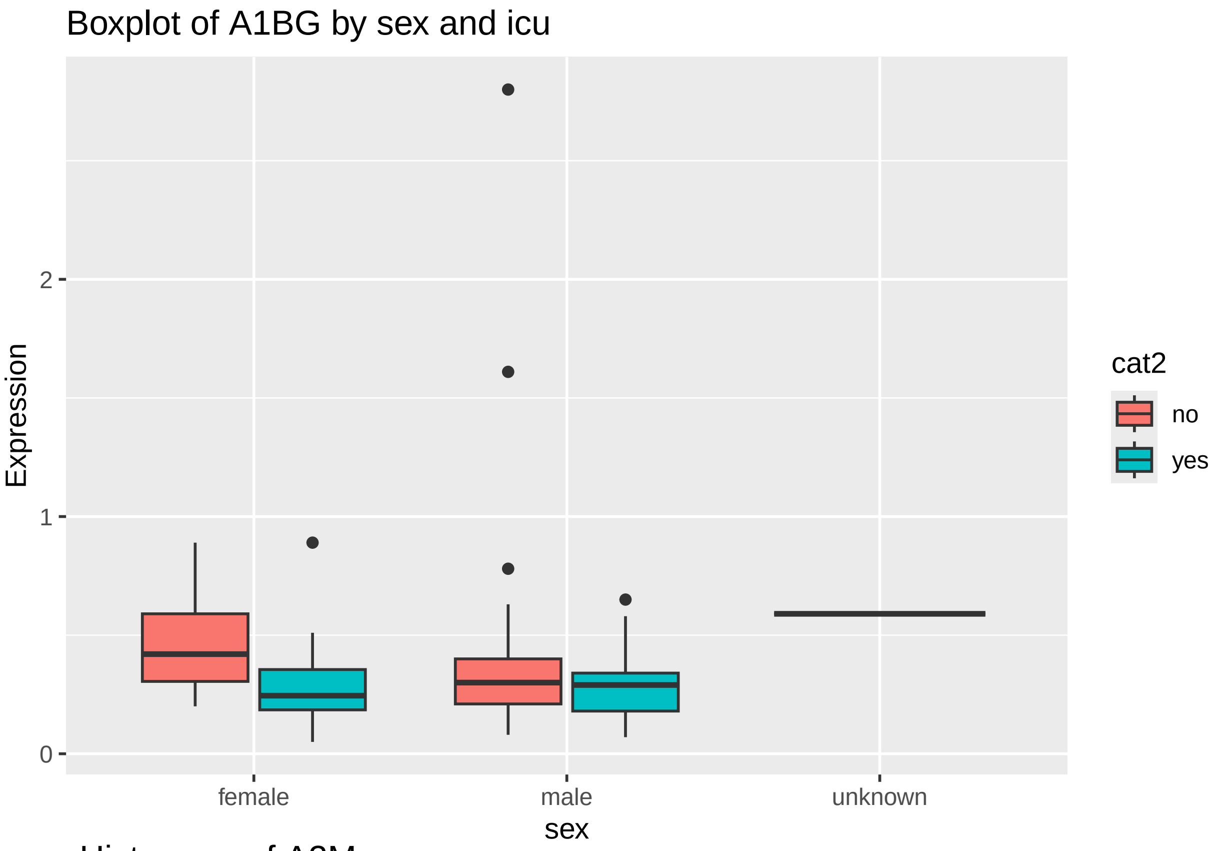
```
## Rows: 126 Columns: 25
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (21): participant_id, geo_accession, status, !Sample_submission_date, la...
## dbl  (4): channel_count, charlson_score, ventilator-free_days, hospital-free...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

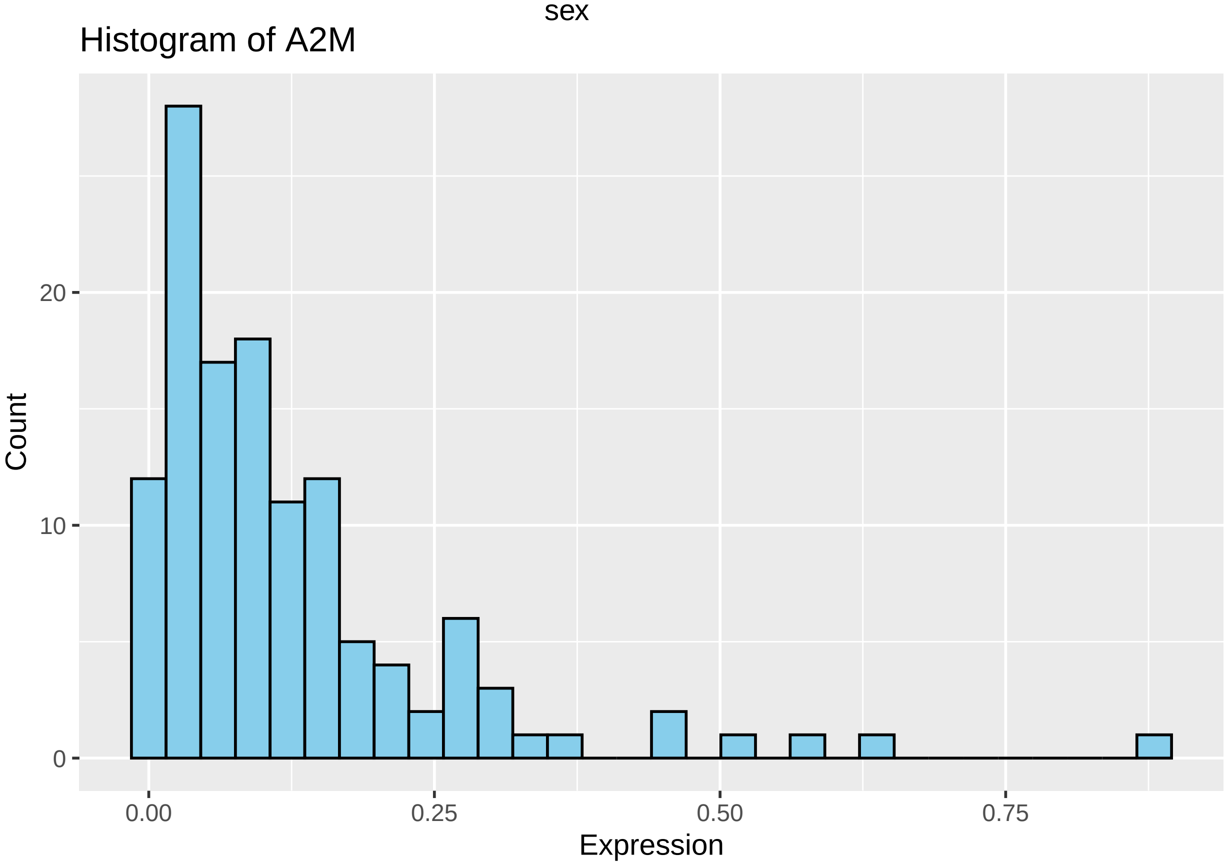
```
# ---- Standardize Column Names ----
metadata <- metadata %>%
  rename(icu = icu_status)
```

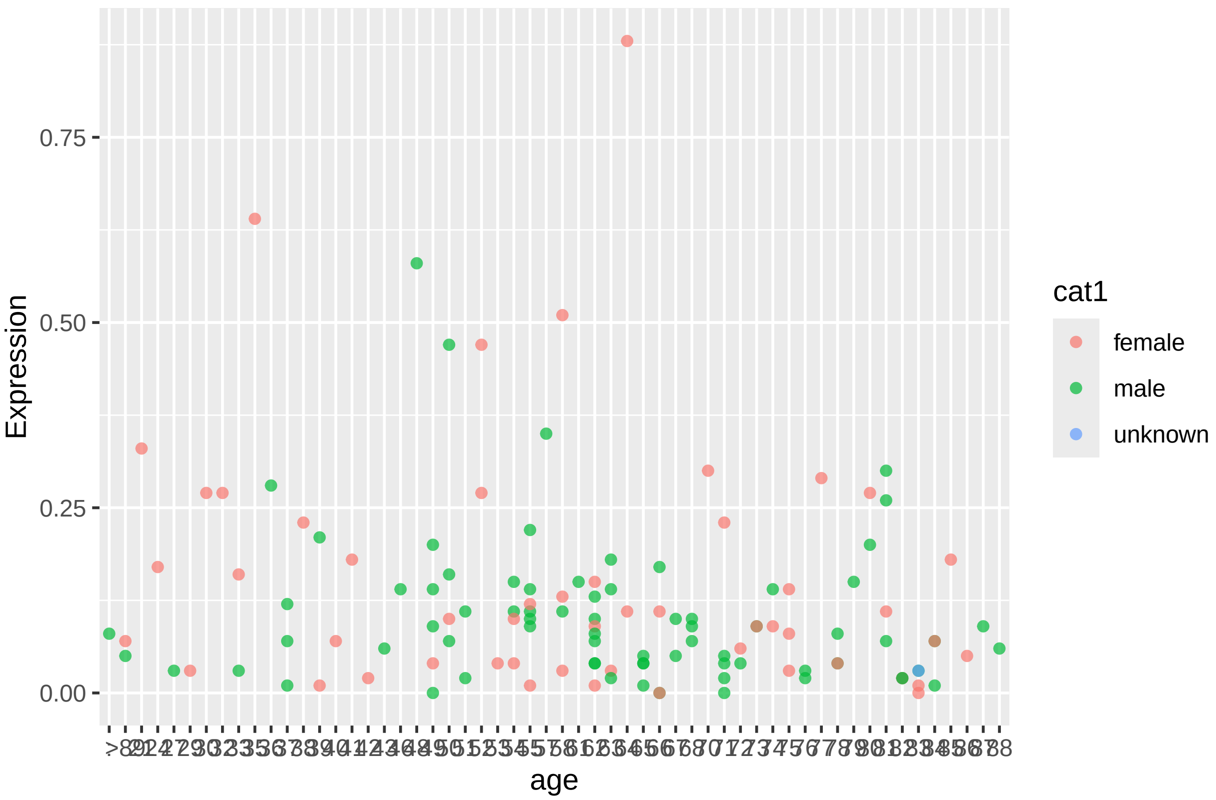
```
# ---- Reshape gene expression ----
gene_expr_long <- gene_expr %>%
  rename(Gene = `...1`) %>%
  pivot_longer(-Gene, names_to = "participant_id", values_to = "expression") %>%
  pivot_wider(names_from = Gene, values_from = expression)

# ---- Merge with metadata ----
full_data <- left_join(metadata, gene_expr_long, by = "participant_id")

# ---- Define plotting function ----
create_gene_plots <- function(df, genes, cont_cov, cat_cov1, cat_cov2) {
  for (gene in genes) {
    if (!(gene %in% colnames(df))) {
      warning(paste("Gene", gene, "not found."))
      next
    }

    plot_df <- df %>%
      select(participant_id, gene_expression = all_of(gene),
             cont = all_of(cont_cov),
             cat1 = all_of(cat_cov1),
             cat2 = all_of(cat_cov2)) %>%
      drop_na()

    # Histogram
    p1 <- ggplot(plot_df, aes(x = gene_expression)) +
      geom_histogram(bins = 30, fill = "skyblue", color = "black") +
      labs(title = paste("Histogram of", gene), x = "Expression", y = "Count")
    print(p1)

    # Scatterplot
    p2 <- ggplot(plot_df, aes(x = cont, y = gene_expression)) +
      geom_point(aes(color = cat1), alpha = 0.7) +
      labs(title = paste("Scatterplot of", gene, "vs", cont_cov),
           x = cont_cov, y = "Expression")
    print(p2)

    # Boxplot
    p3 <- ggplot(plot_df, aes(x = cat1, y = gene_expression, fill = cat2)) +
      geom_boxplot() +
      labs(title = paste("Boxplot of", gene, "by", cat_cov1, "and", cat_cov2),
           x = cat_cov1, y = "Expression")
    print(p3)
  }
}

# ---- Run the function for selected genes ----
selected_genes <- c("A1BG", "A2M", "A4GALT")
create_gene_plots(full_data,
                  genes = selected_genes,
                  cont_cov = "age",
                  cat_cov1 = "sex",
                  cat_cov2 = "icu")
```


Histogram of A1BG

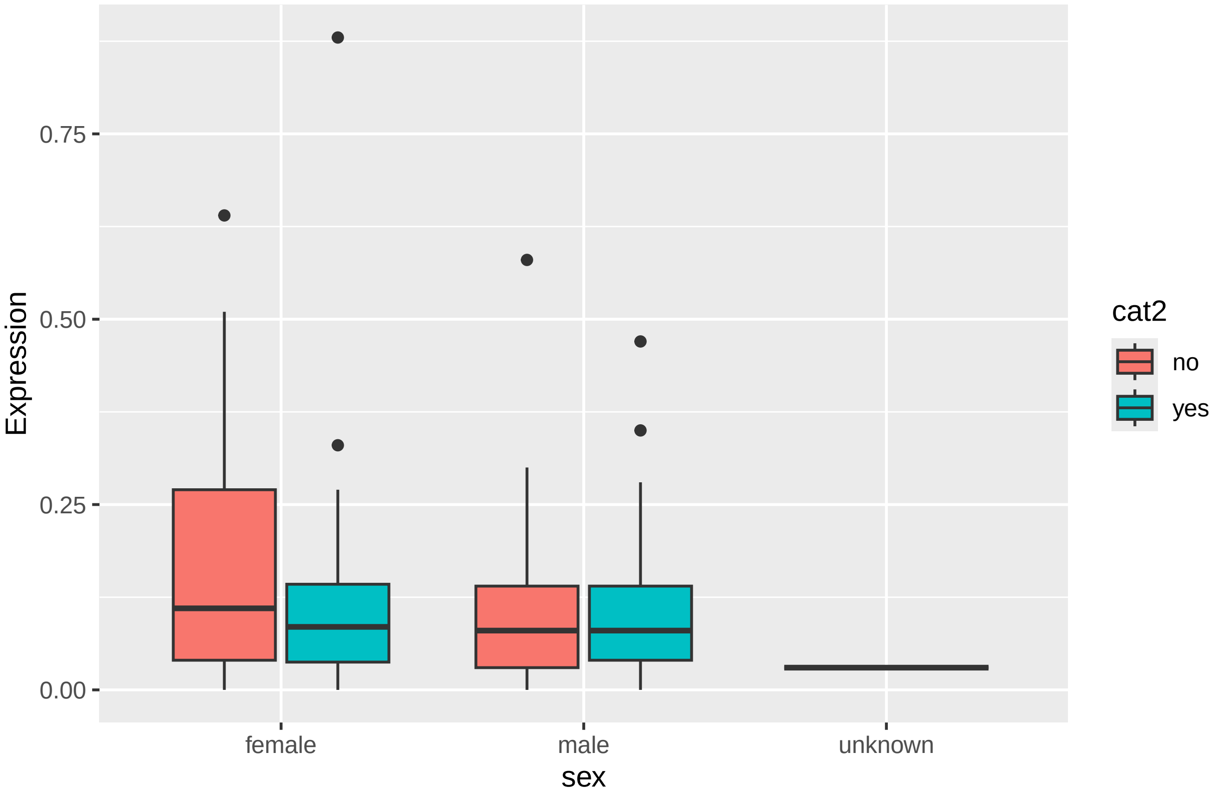

Scatterplot of A1BG vs age


Boxplot of A1BG by sex and icu
Expression
0
1
2
female
male
unknown
sex
cat2
no
yes

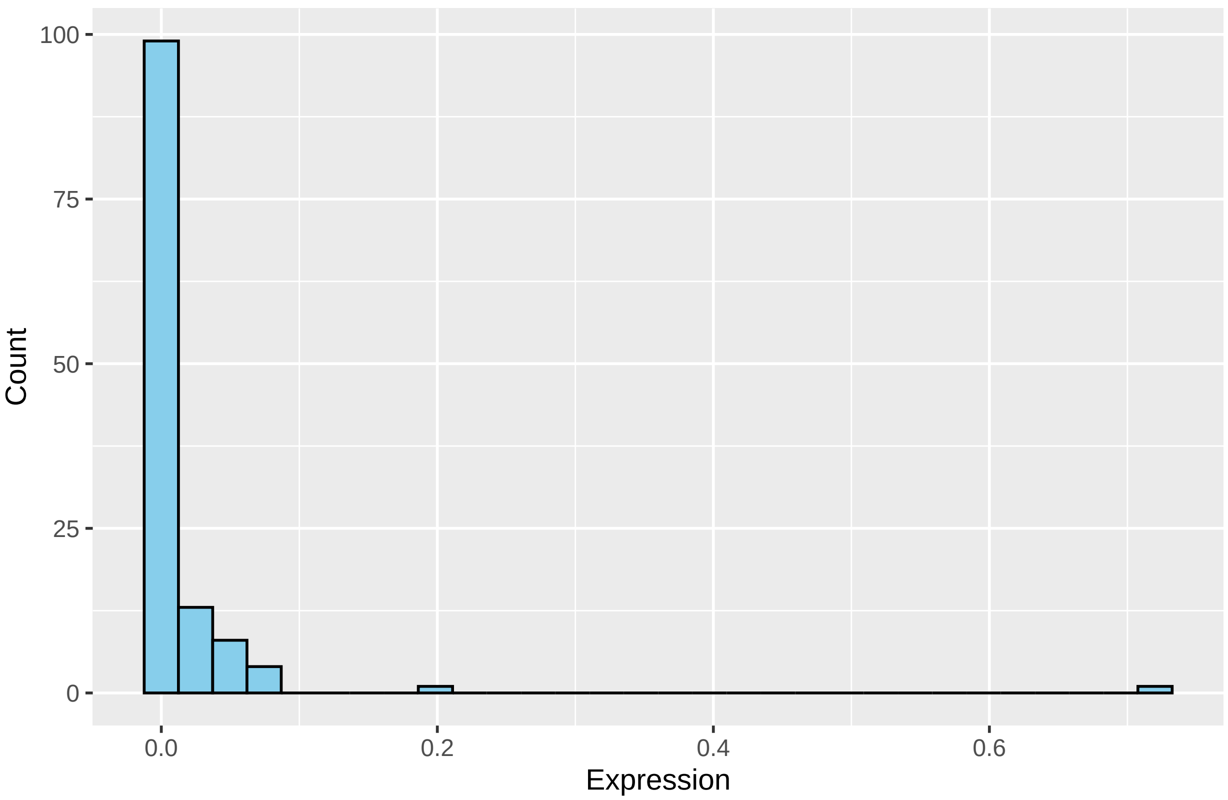

Histogram of A2M
Count
0
10
20
0.00
0.25
0.50
0.75
Expression

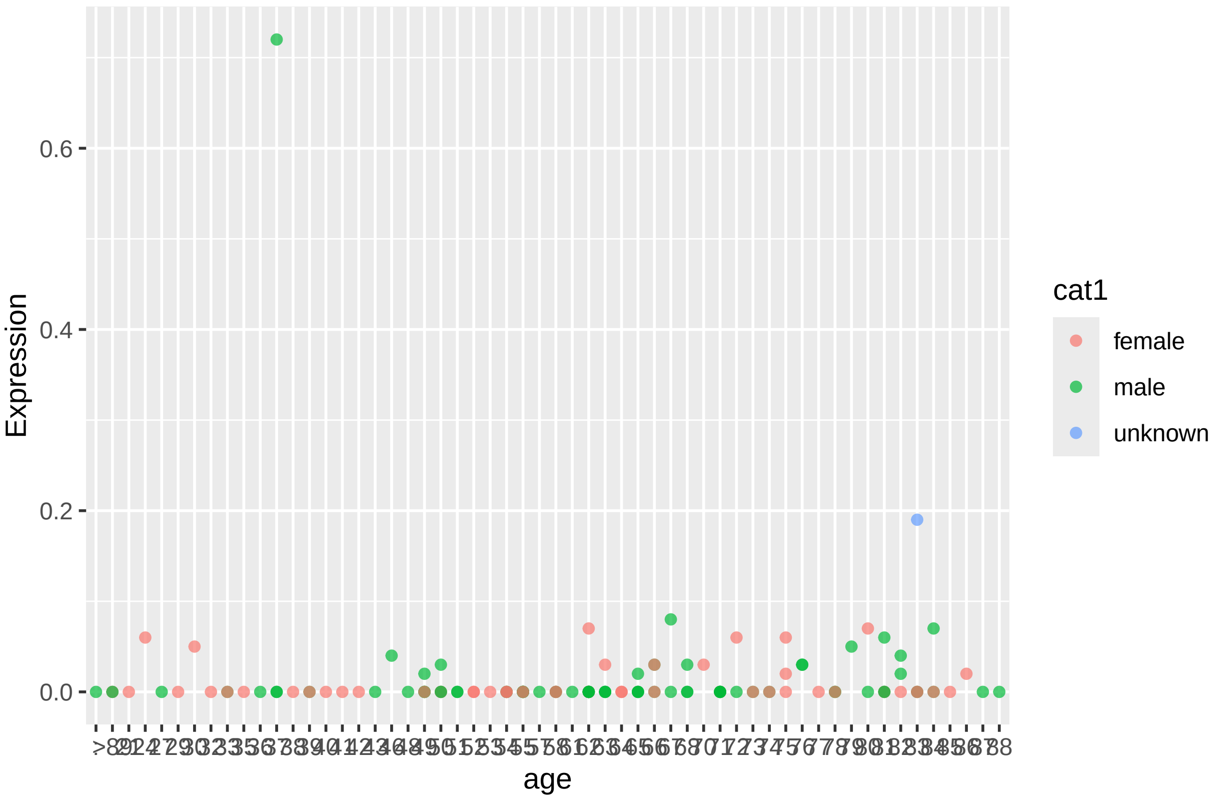


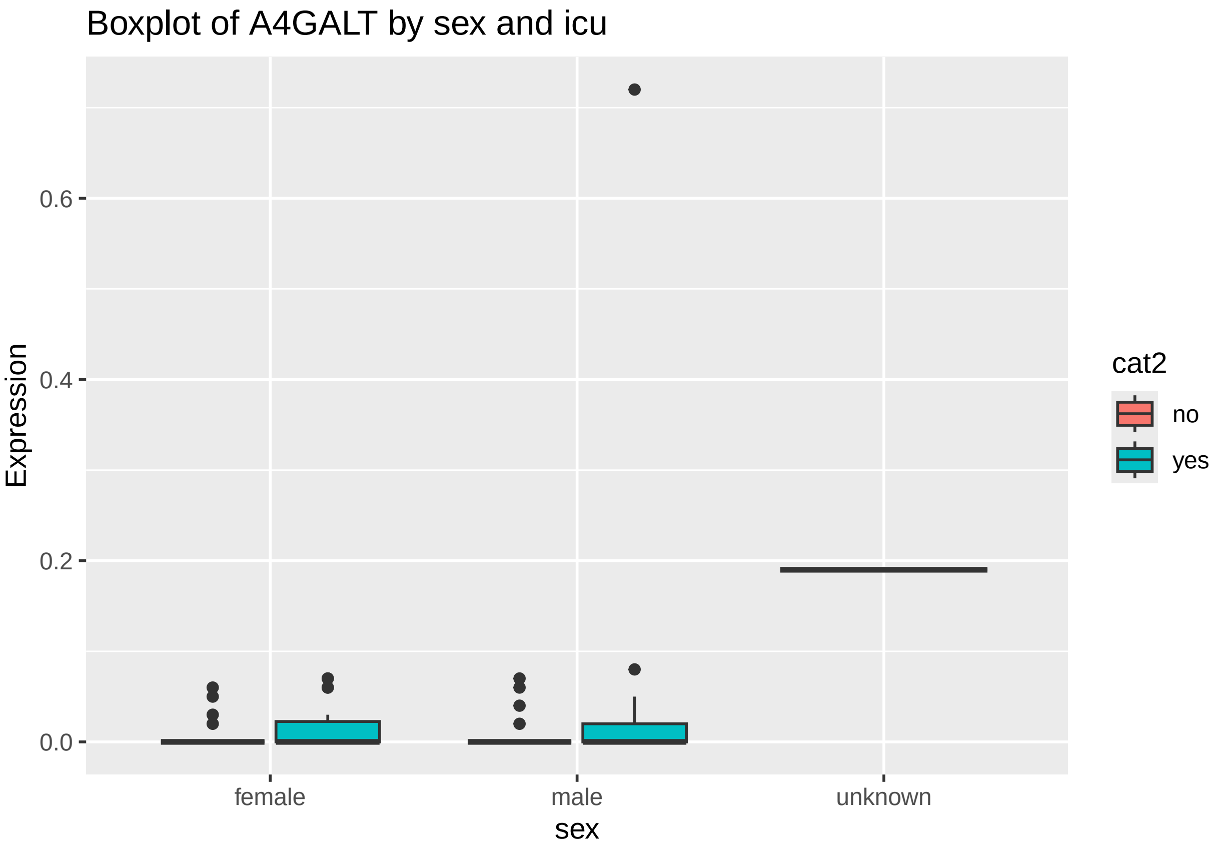
Scatterplot of A2M vs age



Boxplot of A2M by sex and icu


Histogram of A4GALT


Scatterplot of A4GALT vs age


Boxplot of A4GALT by sex and icu
Expression
0.6
0.4
0.2
0.0
female
male
unknown
sex
cat2
no
yes