Análisis de Mutaciones y Visualización

Gerardo Islas Gómez

#Cargar paquetes necesarios

```
# Cargar paquetes necesarios
library(seqinr)      # Leer secuencias en formato FASTA
library(ggplot2)     # Gráficos
library(dplyr)       # Manipulación de datos
```

```
##
## Adjuntando el paquete: 'dplyr'
```

```
## The following object is masked from 'package:seqinr':
##
##     count
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)       # Transformación de datos
library(ggpubr)      # Combinación de gráficos
```

1. Tabla de traducción de codones a aminoácidos

##Este vector asocia codones de ARN con los aminoácidos correspondientes (estándar genético)

```
trad <- c(
  "UUU" = "F", "UUC" = "F", "UUA" = "L", "UUG" = "L",
  "UCU" = "S", "UCC" = "S", "UCA" = "S", "UCG" = "S",
  "UAU" = "Y", "UAC" = "Y", "UAA" = "*", "UAG" = "*",
  "UGU" = "C", "UGC" = "C", "UGA" = "*", "UGG" = "W",
  "CUU" = "L", "CUC" = "L", "CUA" = "L", "CUG" = "L",
  "CCU" = "P", "CCC" = "P", "CCA" = "P", "CCG" = "P",
  "CAU" = "H", "CAC" = "H", "CAA" = "Q", "CAG" = "Q",
  "CGU" = "R", "CGC" = "R", "CGA" = "R", "CGG" = "R",
  "AUU" = "I", "AUC" = "I", "AUA" = "I", "AUG" = "M",
  "ACU" = "T", "ACC" = "T", "ACA" = "T", "ACG" = "T",
```

```
  "AAU" = "N", "AAC" = "N", "AAA" = "K", "AAG" = "K",
  "AGU" = "S", "AGC" = "S", "AGA" = "R", "AGG" = "R",
  "GUU" = "V", "GUC" = "V", "GUA" = "V", "GUG" = "V",
  "GCU" = "A", "GCC" = "A", "GCA" = "A", "GCG" = "A",
  "GAU" = "D", "GAC" = "D", "GAA" = "E", "GAG" = "E",
  "GGU" = "G", "GGC" = "G", "GGA" = "G", "GGG" = "G"
)
```

2. Función para analizar mutaciones

Compara una secuencia de referencia y variantes, identificando mutaciones no sinónimas

```
analizar_mutaciones <- function(ref_file, var_file, variante_nombre, pais) {
  ref_sequences <- read.fasta(ref_file, forceDNAtolower = FALSE)
  var_sequences <- read.fasta(var_file, forceDNAtolower = FALSE)

  resultados <- data.frame(
    mutacion = character(), cambioCodon = character(), cambioAmino = character(),
    pos = integer(), gen = character(), variante = variante_nombre, pais = pais,
    stringsAsFactors = FALSE
  )

  for (i in seq_along(ref_sequences)) {
    gen_ref <- ref_sequences[[i]]
    gen_ref[gen_ref == "T"] <- "U"

    info <- attr(gen_ref, "Annot")
    gene_name <- if(!is.null(info)) {
      info_split <- unlist(strsplit(info, "\\[|\\]|:|=|\\.|;|\\s"))
      gene_pos <- which(info_split == "gene")
      if(length(gene_pos) > 0) info_split[gene_pos[1] + 1] else paste0("Gen_", i)
    } else {
      paste0("Gen_", i)
    }

    for (j in seq_along(var_sequences)) {
      gen_var <- var_sequences[[j]]
      gen_var[gen_var == "T"] <- "U"

      if(length(gen_ref) != length(gen_var)) {
        warning(paste("Longitudes no coinciden para", gene_name))
        next
      }

      diferencias <- which(gen_ref != gen_var)

      if(length(diferencias) > 0) {
        for(pos in diferencias) {
          ini <- pos - ((pos - 1) %% 3)
```

```
          if((ini + 2) > length(gen_ref)) next

          codOri <- paste(gen_ref[ini:(ini+2)], collapse = "")
          codMut <- paste(gen_var[ini:(ini+2)], collapse = "")
          aaOri <- ifelse(is.na(trad[codOri]), "X", trad[codOri])
          aaMut <- ifelse(is.na(trad[codMut]), "X", trad[codMut])

          if(aaOri != aaMut) {
            mut <- paste(gen_ref[pos], pos, gen_var[pos])
            cambio_codon <- paste(codOri, "→", codMut)
            cambio_aa <- paste0(aaOri, (ini %/% 3) + 1, aaMut)

            resultados <- rbind(resultados, data.frame(
              mutacion = mut,
              cambioCodon = cambio_codon,
              cambioAmino = cambio_aa,
              pos = pos,
              gen = gene_name,
              variante = variante_nombre,
              pais = pais,
              stringsAsFactors = FALSE
            ))
          }
        }
      }
    }
  }
  return(resultados)
}
```

3. Visualización de mutaciones más frecuentes

```
plot_mutaciones <- function(data) {
  if (nrow(data) == 0) {
    return(ggplot() + annotate("text", x = 1, y = 1, label = "No hay mutaciones para mostrar") + theme_
  }

  plot_data <- data %>%
    count(variante, pais, mutacion, name = "Frecuencia") %>%
    group_by(variante) %>%
    arrange(desc(Frecuencia)) %>%
    slice_head(n = 10) %>%
    ungroup()

  p <- ggplot(plot_data, aes(x = reorder(mutacion, Frecuencia), y = Frecuencia, fill = variante)) +
    geom_col() +
    coord_flip() +
    labs(title = "Mutaciones nucleotídicas más frecuentes",
         x = "Mutación (posición)", y = "Frecuencia") +
    theme_minimal() +
    scale_fill_manual(values = c("Delta" = "#E69F00", "Omicron" = "#56B4E9"))
```

```
  if (length(unique(plot_data$pais)) > 1 || length(unique(plot_data$variante)) > 1) {
    p <- p + facet_wrap(~variante + pais, scales = "free_y", ncol = 2)
  }
  return(p)
}
```

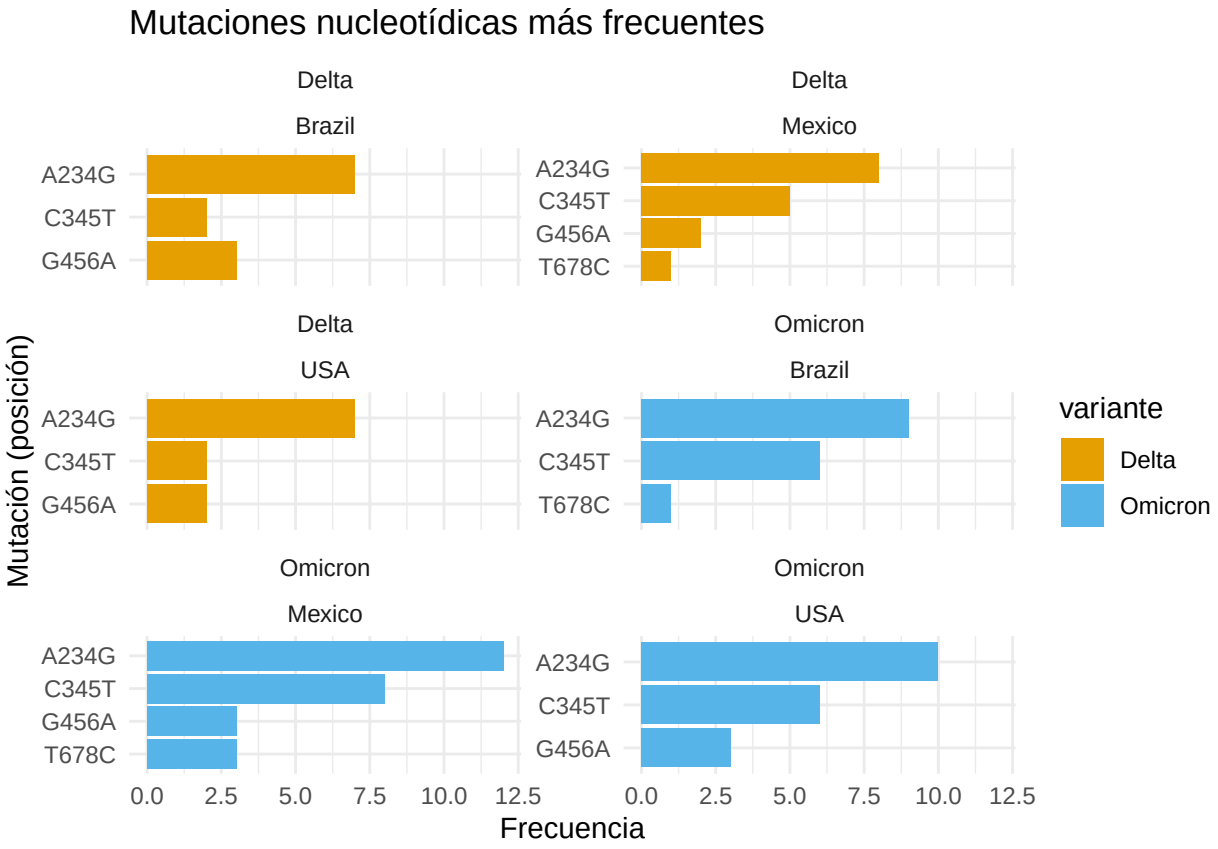
4. Visualización de cambios de aminoácidos

```
plot_cambio_amino <- function(data) {
  if (nrow(data) == 0) {
    return(ggplot() + annotate("text", x = 1, y = 1, label = "No hay cambios de aminoácidos para mostra
  }

  plot_data <- data %>%
    count(gen, variante, cambioAmino, name = "Frecuencia") %>%
    group_by(gen) %>%
    arrange(desc(Frecuencia)) %>%
    slice_head(n = 10) %>%
    ungroup()

  p <- ggplot(plot_data, aes(x = reorder(cambioAmino, Frecuencia), y = Frecuencia, fill = gen)) +
    geom_col() +
    coord_flip() +
    labs(title = "Cambios de aminoácidos más frecuentes",
         x = "Cambio de aminoácido", y = "Frecuencia") +
    theme_minimal() +
    scale_fill_brewer(palette = "Set1")

  if (length(unique(plot_data$gen)) > 1 || length(unique(plot_data$variante)) > 1) {
    p <- p + facet_wrap(~gen + variante, scales = "free", ncol = 2)
  }
  return(p)
}
```

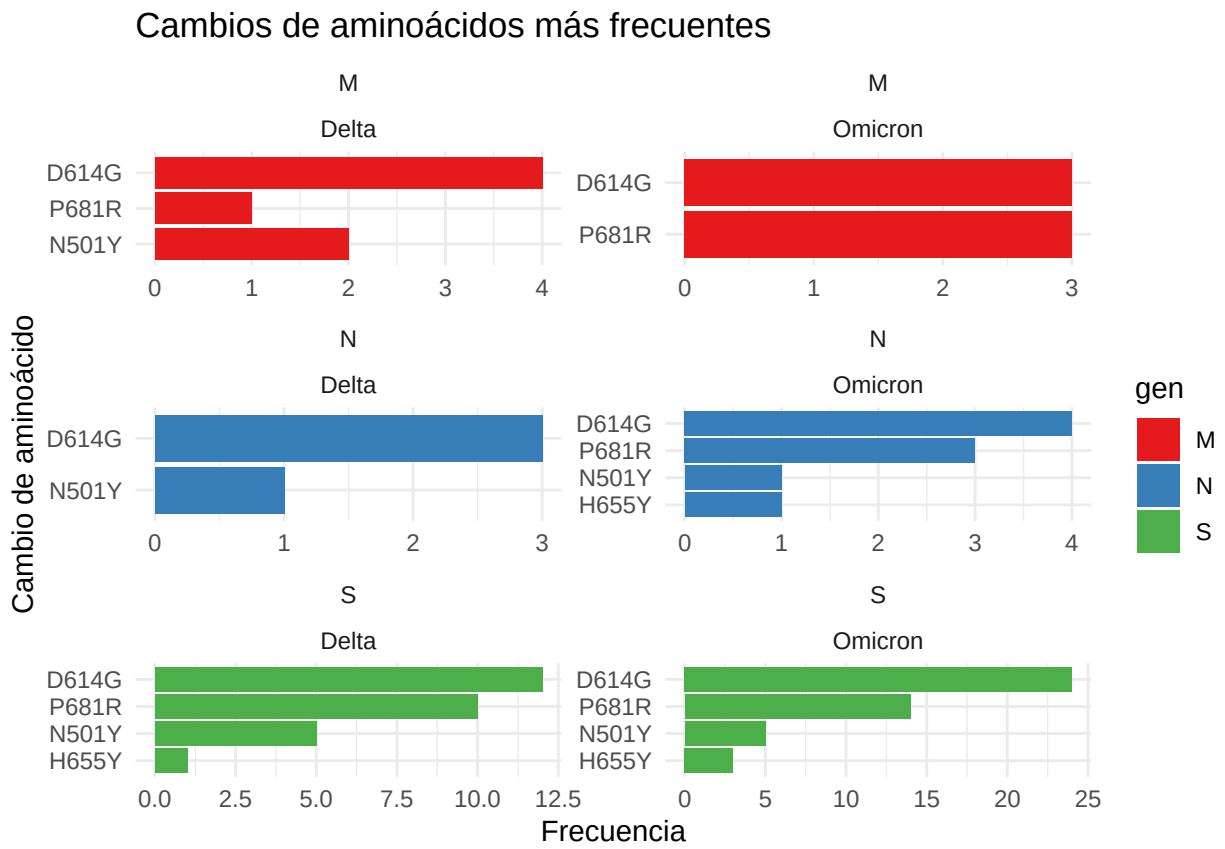
```
# 5. Ejemplo con datos simulados
set.seed(123)
mutaciones_ejemplo <- data.frame(
  mutacion = sample(c("A234G", "C345T", "G456A", "T678C"), 100, replace = TRUE, prob = c(0.5, 0.3, 0.15
  cambioAmino = sample(c("D614G", "P681R", "N501Y", "H655Y"), 100, replace = TRUE, prob = c(0.5, 0.3, 0
  pos = sample(100:800, 100, replace = TRUE),
  gen = sample(c("S", "N", "M"), 100, replace = TRUE, prob = c(0.7, 0.2, 0.1)),
  variante = sample(c("Delta", "Omicron"), 100, replace = TRUE, prob = c(0.4, 0.6)),
  pais = sample(c("Mexico", "Brazil", "USA"), 100, replace = TRUE, prob = c(0.5, 0.3, 0.2)),
  stringsAsFactors = FALSE
)
```

6. Generar y mostrar gráficos

```
plot_mut <- plot_mutaciones(mutaciones_ejemplo)
plot_amino <- plot_cambio_amino(mutaciones_ejemplo)

print(plot_mut)
```

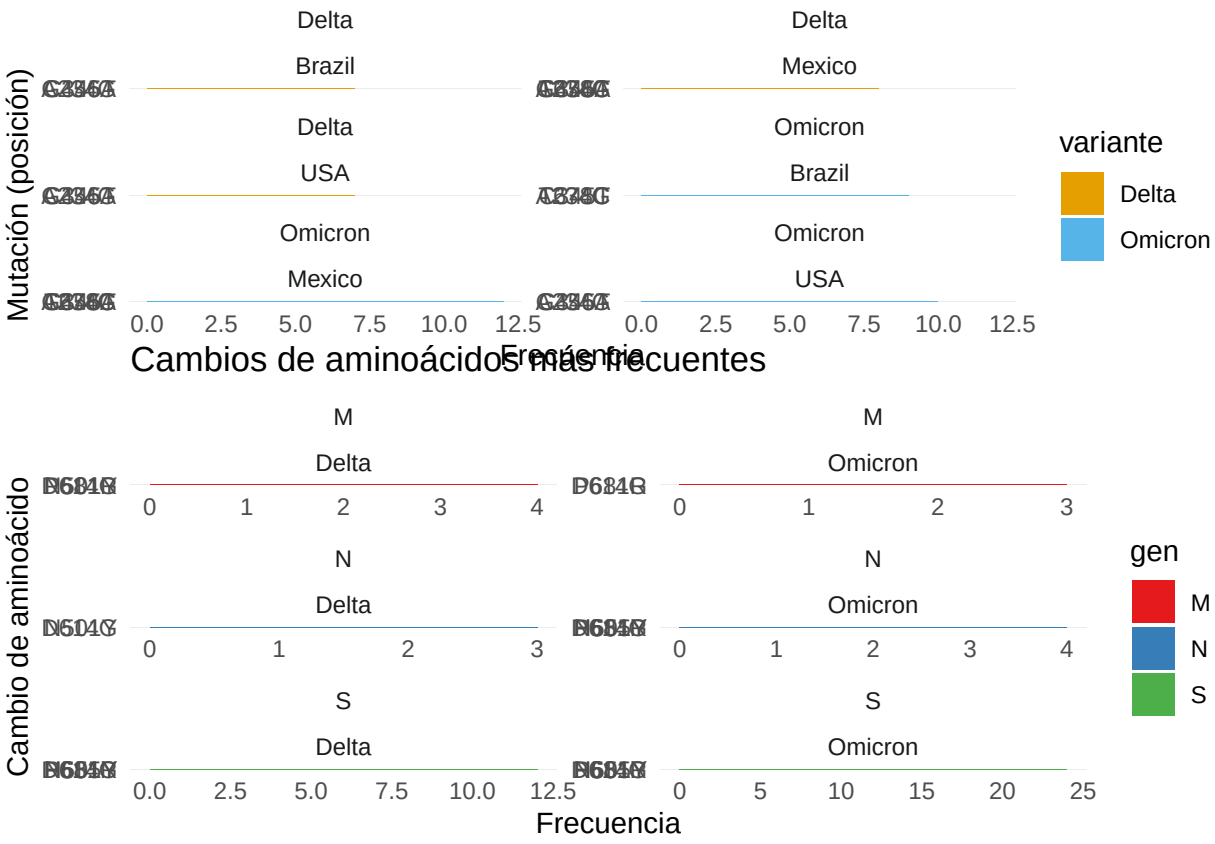



```
print(plot_amino)
```



7. Combinar gráficos si ambos están disponibles

```
if (!is.null(plot_mut) && !is.null(plot_amino)) {
  combined_plot <- ggarrange(plot_mut, plot_amino, ncol = 1, heights = c(1, 1.5))
  print(combined_plot)
}
```


Delta
Brazil
Delta
USA
Omicron
Mexico
Delta
Mexico
Omicron
Brazil
Omicron
USA
Mutación (posición)
0.0 2.5 5.0 7.5 10.0 12.5
Frecuencia
variante
Delta
Omicron
Cambios de aminoácidos más frecuentes
M
Delta
N
Delta
S
Delta
M
Omicron
N
Omicron
S
Omicron
Cambio de aminoácido
0 1 2 3 4
0 1 2 3
0.0 2.5 5.0 7.5 10.0 12.5
0 5 10 15 20 25
Frecuencia
gen
M
N
S