###########################################

# Statistical Analysis Script

# Purpose: Univariate and Multivariate Cox regression with FDR adjustment

# + Proportional hazards assumption test

###########################################

# Load required packages

library(survival) # For Cox models

library(forestmodel) # For forest plots

library(dplyr) # For data manipulation

library(readxl) # For reading Excel files

#-----------------------------------------

# 1. Data Import

#-----------------------------------------

# Replace with your actual data file name

my\_data <- read\_excel("my\_data.xlsx")

# Ensure time and status are numeric

my\_data$time <- as.numeric(my\_data$time)

my\_data$status <- as.numeric(my\_data$status)

#-----------------------------------------

# 2. Define categorical variables

#-----------------------------------------

factor\_vars <- c("Age", "Sex", "Smoking", "Alcohol", "Hypertension",

"Diabetes mellitus", "BMI", "T stage", "N stage",

"Tumor location", "Tumor length(cm)", "Induction therapy",

"Lymphopenia", "Clinical response")

# Convert to factor and set reference levels

my\_data$`Tumor length(cm)` <- factor(my\_data$`Tumor length(cm)`)

my\_data$`Tumor length(cm)` <- relevel(my\_data$`Tumor length(cm)`, ref = "≤5")

my\_data$Age <- factor(my\_data$Age)

my\_data$Age <- relevel(my\_data$Age, ref = "<60")

my\_data$BMI <- factor(my\_data$BMI)

my\_data$BMI <- relevel(my\_data$BMI, ref = "18.5-24.9")

my\_data$`Tumor location` <- factor(my\_data$`Tumor location`)

my\_data$`Tumor location` <- relevel(my\_data$`Tumor location`, ref = "Upper")

# Convert other variables to factor if not already

for (var in factor\_vars) {

if (!is.factor(my\_data[[var]])) {

my\_data[[var]] <- factor(my\_data[[var]])

}

}

# Function to safely handle variable names with spaces/special characters

safe\_var <- function(x) {

if (grepl("[^a-zA-Z0-9\_]", x)) paste0("`", x, "`") else x

}

#-----------------------------------------

# 3. Univariate Cox regression

#-----------------------------------------

univ\_results <- lapply(factor\_vars, function(var) {

var\_safe <- safe\_var(var)

f <- as.formula(paste("Surv(time, status) ~", var\_safe))

model <- coxph(f, data = my\_data)

s <- summary(model)

data.frame(

Variable = var,

HR = round(s$coefficients[,"exp(coef)"], 2),

Lower\_CI = round(s$conf.int[,"lower .95"], 2),

Upper\_CI = round(s$conf.int[,"upper .95"], 2),

P\_value = round(s$coefficients[,"Pr(>|z|)"], 4),

row.names = NULL

)

})

# Combine results

univ\_df <- do.call(rbind, univ\_results)

# Adjust p-values with Benjamini-Hochberg (FDR)

univ\_df$Adj\_P\_value <- p.adjust(univ\_df$P\_value, method = "BH")

# Print univariate results

print(univ\_df)

#-----------------------------------------

# 4. Multivariate Cox regression

#-----------------------------------------

# Select variables with adjusted p < 0.2 (threshold can be changed)

selected\_vars <- univ\_df %>% filter(Adj\_P\_value < 0.2) %>% pull(Variable)

selected\_vars\_safe <- sapply(selected\_vars, safe\_var)

if (length(selected\_vars\_safe) > 0) {

multiv\_formula <- as.formula(paste("Surv(time, status) ~", paste(selected\_vars\_safe, collapse = " + ")))

multiv\_model <- coxph(multiv\_formula, data = my\_data)

# Plot forest model

forest\_model(multiv\_model)

# Test proportional hazards assumption

ph\_test <- cox.zph(multiv\_model)

print(ph\_test)

# Plot Schoenfeld residuals

plot(ph\_test)

} else {

cat("No variables with adjusted p < 0.2. Multivariate model not performed.\n")

}