HW4 Question 3C

Gifty Osei

2024-09-27

Question 3c:

```r
# Set Seed
set.seed(123)

# Parameters
n <- 100  # Sample size
num_sim <- 1000  # Number of simulations
fwer_alpha <- 0.1  # Nominal family-wise error rate

# Storage vector
reject_F_test <- logical(num_sim)
reject_pairwise <- logical(num_sim)

# fixed x-values from N(0,1)
x <- matrix(rnorm(n * 4), n, 4)

# Function for the F-test
F_test_function <- function(y, x) {
  # Fit the full model
  full_model <- lm(y ~ x)

  # Fit the reduced model under H0: beta1 = beta2 = beta3 = beta4
  reduced_model <- lm(y ~ rowMeans(x))

  #  F-statistic
  F_stat <- anova(reduced_model, full_model, test = "F")$`F`[2]

  # p-value of the F-test
  p_value <- anova(reduced_model, full_model, test = "F")$`Pr(>F)`[2]

  return(p_value < fwer_alpha) ## return
}

# Function for pairwise t-tests with adjustment
pairwise_tests_function <- function(y, x) {
  #  full model
  model <- lm(y ~ x)

  # Extract estimated coefficients and covariance
  beta_hat <- coef(model)[-1]  # no intercept
```

```r
  cov_matrix <- vcov(model)[-1, -1]  # no intercept

  # variances for pairwise differences
  var_diff_12 <- cov_matrix[1, 1] + cov_matrix[2, 2] - 2 * cov_matrix[1, 2]
  var_diff_23 <- cov_matrix[2, 2] + cov_matrix[3, 3] - 2 * cov_matrix[2, 3]
  var_diff_34 <- cov_matrix[3, 3] + cov_matrix[4, 4] - 2 * cov_matrix[3, 4]

  #  t-statistics for each pairwise test
  t_12 <- (beta_hat[1] - beta_hat[2]) / sqrt(var_diff_12)
  t_23 <- (beta_hat[2] - beta_hat[3]) / sqrt(var_diff_23)
  t_34 <- (beta_hat[3] - beta_hat[4]) / sqrt(var_diff_34)

  # p-values - each t-test
  p_values <- 2 * pt(-abs(c(t_12, t_23, t_34)), df = n - 5)

  # Apply Bonferroni and compare
  reject <- any(p_values < fwer_alpha / 3)

  return(reject)
}

# Simulation
for (i in 1:num_sim) {
  # Generate y-values based on the model
  epsilon <- rnorm(n)
  y <- 1 + x[, 1] + x[, 2] + x[, 3] + x[, 4] + epsilon

  #  F-test
  reject_F_test[i] <- F_test_function(y, x)

  #  pairwise t-tests with Bonferroni adjustment
  reject_pairwise[i] <- pairwise_tests_function(y, x)
}

# empirical family-wise error rate (FWER)
FWER_F_test <- mean(reject_F_test)
FWER_pairwise <- mean(reject_pairwise)


## Result data
result_data <- data.frame("Simultaneous" = FWER_F_test,
                          "Bonferroni" = FWER_pairwise,
                          "Nominal FWER" = fwer_alpha)
# Display results

kable(result_data, caption = "Comparison between the 2 Methods to FWER")
```

Table 1: Comparison between the 2 Methods to FWER

Simultaneous	Bonferroni	Nominal.FWER
0.089	0.079	0.1

We can see from Table 1 that, simultaneous test gives a value that is closer to the true nominal family-wise error rate value.