# Define actual and predicted outcomes

# "N" = No Wolf, "P" = Wolf

actual <- c("N", "N", "P", "P") # Actual wolf presence

predicted <- c("P", "N", "P", "N") # Villagers' response

# Create an empty 2x2 matrix for confusion matrix

conf\_matrix <- matrix(0, nrow = 2, ncol = 2)

rownames(conf\_matrix) <- c("Pred\_No\_Wolf", "Pred\_Wolf")

colnames(conf\_matrix) <- c("Actual\_No\_Wolf", "Actual\_Wolf")

# Populate the confusion matrix manually

for (i in 1:length(actual)) {

if (actual[i] == "N" & predicted[i] == "N") {

conf\_matrix[1, 1] <- conf\_matrix[1, 1] + 1 # True Negative (TN)

} else if (actual[i] == "N" & predicted[i] == "P") {

conf\_matrix[2, 1] <- conf\_matrix[2, 1] + 1 # False Positive (FP)

} else if (actual[i] == "P" & predicted[i] == "P") {

conf\_matrix[2, 2] <- conf\_matrix[2, 2] + 1 # True Positive (TP)

} else if (actual[i] == "P" & predicted[i] == "N") {

conf\_matrix[1, 2] <- conf\_matrix[1, 2] + 1 # False Negative (FN)

}

}

# Print the Confusion Matrix

print("Confusion Matrix:")

print(conf\_matrix)

# Compute Accuracy, Sensitivity, and Specificity

TP <- conf\_matrix[2, 2] # True Positives

FP <- conf\_matrix[2, 1] # False Positives

TN <- conf\_matrix[1, 1] # True Negatives

FN <- conf\_matrix[1, 2] # False Negatives

accuracy <- (TP + TN) / sum(conf\_matrix)

sensitivity <- TP / (TP + FN) # True Positive Rate

specificity <- TN / (TN + FP) # True Negative Rate

# Print results

cat("\nAccuracy:", accuracy, "\n")

cat("Sensitivity (Recall):", sensitivity, "\n")

cat("Specificity:", specificity, "\n")