Statistical Methods Quiz #3 (40 minutes)

'25. 12. 1.(Monday) Cadet number(교번): Name: Score:

<Instructions>

1. It is open-book, closed-web test.
2. You are not allowed to discuss on the problems during the exam.
3. Import Movie.csv data first.
4. Run the provided code block (visible below) to prepare data and train a baseline model then answer the questions. Do not change the given R code.

<Baseline Code> - Run This First (Do not change this R code)

```
library(keras3); set.seed(12)

review = Movie
colnames(review) <- c("score", "comment")
review$score    <- as.integer(review$score)
review <- na.omit(review) # Remove NAs
review$score = ifelse(review$score <= 8, 0, 1) # Recode to 0-1 data

# ----------------------------
# Train / valid / test split
# ----------------------------
n    <- nrow(review) %/% 10
idx <- sample.int(n); review <- review[idx, ]

n_test  <- max(1, round(0.20 * n))
test_set  <- review[seq_len(n_test), ]; train_set <- review[-seq_len(n_test), ]

x_train <- train_set$comment; x_test <- test_set$comment

y_train <- train_set$score; y_test  <- test_set$score

# ------------------------------------------------
# Text vectorization (minimal preprocessing)
# ------------------------------------------------
# - Keeps text as-is (no lowercasing/stripping) because it's Korean.
max_words <- 20000L
maxlen    <- 50L  # you can raise to reduce truncation

vec <- layer_text_vectorization(
  max_tokens = max_words,
  output_mode = "int",
  output_sequence_length = maxlen,
  standardize = NULL # keep Korean punctuation/spacing as-is
)

# Build the vocabulary from training text only
adapt(vec, train_set$comment)

# Save the whole vocabulary (not necessary)
vocab <- keras3::get_vocabulary(vec); head(vocab, 30)

# ------------------------------------------------
# Model: TextVectorization -> Embedding -> BiLSTM
# ------------------------------------------------
units <- 64L

model <- keras3::keras_model_sequential(
  layers = list(
    vec,
    keras3::layer_embedding(input_dim = max_words, output_dim = units, mask_zero = TRUE),
    keras3::layer_lstm(units = units),
    keras3::layer_dense(units = 1, activation = "sigmoid")
  )
)

model |> compile(
  optimizer = "adam",
  loss = "binary_crossentropy",
  metrics = "accuracy"
)

# ------------------------------------------------
# Train
# ------------------------------------------------
history <- model |>
  fit(
    x = x_train, y = y_train,
    epochs = 5,
    batch_size = 64,
    validation_split = 0.2,
    verbose = 2
  )
```

1. Print a readable model summary. How many parameters are there in total?[20 points]

<R-code>

<Output>

The number of total parameters:_____

2. Print the accuracy using test dataset.[20 points]

<R-code>

<Output>

Accuracy:_____

3. Compute predicted probabilities, convert to class labels with threshold 0.6, and report the confusion matrix.[30 points]

<R-code>

<Output>

4. Show 4 most confidently correct predictions using the predicted probabilities. For each, print true label, predicted probability, and the raw text.[30 points]

<R-code>

<Output>