```r
# ============ Section 1: MCMC MODELS ============
library(ggplot2)
library(tibble)
# ======== preprocess ========
dataset <- read.csv("diabetes.csv")
dataset_cleaned <- subset(dataset, select = -c(id, location, bp.2s, bp.2d, time.ppn,
ratio))
dataset_cleaned <- dataset_cleaned[dataset_cleaned$frame %in% c("small", "medium",
"large"), ]
df_clean <- dataset_cleaned[complete.cases(dataset_cleaned), ]
df_clean$gender <- ifelse(df_clean$gender == "male", 1, 0)
df_clean$group <- as.numeric(factor(df_clean$frame))
G_high <- length(unique(df_clean$group))
head(df_clean)
# ========== LOW DIMENSION MCMC (5 predictor) ==========
data_low_mcmc <- list(
  N = nrow(df_clean),
  age = as.vector(scale(df_clean$age)),
  weight = as.vector(scale(df_clean$weight)),
  waist = as.vector(scale(df_clean$waist)),
  bp_1d = as.vector(scale(df_clean$bp.1d)),
  bp_1s = as.vector(scale(df_clean$bp.1s)),
  y = df_clean$glyhb
)

fit_low_mcmc <- stan(
  file = "low_dimension.stan",
  data = data_low_mcmc,
  chains = 4, iter = 2000, seed = 123)
fit_low_mcmc

# ========== MID DIMENSION MCMC (9 predictor) ==========
data_mid_mcmc <- list(
  N = nrow(df_clean),
  age = as.vector(scale(df_clean$age)),
  weight = as.vector(scale(df_clean$weight)),
  waist = as.vector(scale(df_clean$waist)),
  bp_1d = as.vector(scale(df_clean$bp.1d)),
  bp_1s = as.vector(scale(df_clean$bp.1s)),
  chol = as.vector(scale(df_clean$chol)),
  stab_glu = as.vector(scale(df_clean$stab.glu)),
  hdl = as.vector(scale(df_clean$hdl)),
  height = as.vector(scale(df_clean$height)),
  y = df_clean$glyhb)
fit_mid_mcmc <- stan(
  file = "mid_dimension.stan",
  data = data_mid_mcmc,
  chains = 4, iter = 2000, seed = 123)
fit_mid_mcmc
# ========== HIGH DIMENSION MCMC (hierarchical models) ==========
df_clean$group <- as.numeric(factor(df_clean$frame))
G_high <- length(unique(df_clean$group))
data_high_mcmc <- list(
  N = nrow(df_clean),
  G = G_high,
  group = df_clean$group,
  weight = as.vector(scale(df_clean$weight)),
  waist = as.vector(scale(df_clean$waist)),
  age = as.vector(scale(df_clean$age)),
  bp_1d = as.vector(scale(df_clean$bp.1d)),
  bp_1s = as.vector(scale(df_clean$bp.1s)),
  chol = as.vector(scale(df_clean$chol)),
  stab_glu = as.vector(scale(df_clean$stab.glu)),
```

```r
  hdl = as.vector(scale(df_clean$hdl)),
  height = as.vector(scale(df_clean$height)),
  gender = df_clean$gender,
  y = df_clean$glyhb)
fit_high_mcmc <- stan(
  file = "high_model.stan",
  data = data_high_mcmc,
  chains = 4, iter = 2000, seed = 123)
fit_high_mcmc
# ============ Section 2: SNIS ============
# --------- SNIS LOW ---------
post_low <- as_draws_matrix(fit_low_mcmc)[, 1:6]
mu_low <- colMeans(post_low)
Sigma_low <- cov(post_low)

theta_low <- mvrnorm(10000, mu_low, Sigma_low)

X_low <- scale(df_clean[, c("age", "weight", "waist", "bp.1d", "bp.1s")])
y_low <- df_clean$glyhb

logpost_low <- function(theta, X, y) {
  beta <- theta[1:5]; sigma <- theta[6]
  if (sigma <= 0) return(-Inf)
  prior <- sum(dnorm(beta, 0, 5, log = TRUE)) + dcauchy(sigma, 0, 2.5, log = TRUE)
  loglik <- sum(dnorm(y, mean = X %*% beta, sd = sigma, log = TRUE))
  return(prior + loglik)
}

log_w_low <- apply(theta_low, 1, function(row) logpost_low(row, X_low, y_low))
log_q_low <- dmvnorm(theta_low, mean = mu_low, sigma = Sigma_low, log = TRUE)

w_low <- exp(log_w_low - log_q_low - max(log_w_low - log_q_low))
w_low <- w_low / sum(w_low)

ess_low_snis <- 1 / sum(w_low^2)
mean_low_snis <- colSums(theta_low * w_low)

ess_calc <- function(samples, weights) {
  weights <- weights / sum(weights); M <- length(weights)
  apply(samples, 2, function(p) {
    mu_w <- sum(p * weights)
    var_w <- sum(weights * (p - mu_w)^2)
    var_naive <- var(p)
    M * var_w / var_naive
  })
}
ess_low_params <- ess_calc(theta_low, w_low)
ess_low_params

# --------- SNIS MID ---------
post_mid <- as_draws_matrix(fit_mid_mcmc)[, 1:10]
mu_mid <- colMeans(post_mid)
Sigma_mid <- cov(post_mid)

theta_mid <- mvrnorm(10000, mu_mid, Sigma_mid)

X_mid <- scale(df_clean[, c("age", "weight", "waist",
                            "bp.1d", "bp.1s", "chol", "stab.glu", "hdl", "height")])
y_mid <- df_clean$glyhb

logpost_mid <- function(theta, X, y) {
  beta <- theta[1:9]; sigma <- theta[10]
  if (sigma <= 0 || !is.finite(sigma)) return(-Inf)
  prior <- sum(dnorm(beta, 0, 5, log = TRUE)) + dcauchy(sigma, 0, 2.5, log = TRUE)
```

```r
    loglik <- sum(dnorm(y, mean = X %*% beta, sd = sigma, log = TRUE))
    return(prior + loglik)
}

log_w_mid <- apply(theta_mid, 1, function(row) logpost_mid(row, X_mid, y_mid))
log_q_mid <- dmvnorm(theta_mid, mean = mu_mid, sigma = Sigma_mid, log = TRUE)

w_mid <- exp(log_w_mid - log_q_mid - max(log_w_mid - log_q_mid))
w_mid <- w_mid / sum(w_mid)

ess_mid_snis <- 1 / sum(w_mid^2)
mean_mid_snis <- colSums(theta_mid * w_mid)
ess_mid_params <- ess_calc(theta_mid, w_mid)
ess_mid_params

# --------- SNIS HIGH ---------
draws_high <- as_draws_matrix(fit_high_mcmc)
param_high <- c(
  paste0("alpha[", 1:G_high, "]"),
  paste0("beta_weight[", 1:G_high, "]"),
  paste0("beta_waist[", 1:G_high, "]"),
  "beta_age", "beta_bp_1d", "beta_bp_1s",
  "beta_chol", "beta_stab_glu", "beta_hdl", "beta_height", "beta_gender"
)

theta_high <- draws_high[, param_high]
mu_high <- colMeans(theta_high)
Sigma_high <- cov(theta_high)
samples_high <- mvrnorm(10000, mu_high, Sigma_high)
Xhigh <- list(
  age = scale(df_clean$age), bp_1d = scale(df_clean$bp.1d),
  bp_1s = scale(df_clean$bp.1s), chol = scale(df_clean$chol),
  stab_glu = scale(df_clean$stab.glu), hdl = scale(df_clean$hdl),
  height = scale(df_clean$height), gender = df_clean$gender,
  weight = scale(df_clean$weight), waist = scale(df_clean$waist),
  group = df_clean$group, y = df_clean$glyhb
)

logpost_high <- function(theta) {
  G <- G_high
  alpha <- theta[1:G]
  beta_weight <- theta[(G+1):(2*G)]
  beta_waist  <- theta[(2*G+1):(3*G)]
  beta_rest <- theta[(3*G+1):length(theta)]
  mu_i <- rep(0, length(Xhigh$y))
  for (i in seq_along(mu_i)) {
    g <- Xhigh$group[i]
    mu_i[i] <- alpha[g] + beta_weight[g]*Xhigh$weight[i] + beta_waist[g]*Xhigh$waist[i] +
      beta_rest[1]*Xhigh$age[i] + beta_rest[2]*Xhigh$bp_1d[i] +
beta_rest[3]*Xhigh$bp_1s[i] +
      beta_rest[4]*Xhigh$chol[i] + beta_rest[5]*Xhigh$stab_glu[i] +
beta_rest[6]*Xhigh$hdl[i] +
      beta_rest[7]*Xhigh$height[i] + beta_rest[8]*Xhigh$gender[i]
  }
  log_prior <- sum(dnorm(theta, 0, 5, log = TRUE))
  log_lik <- sum(dnorm(Xhigh$y, mean = mu_i, sd = 1, log = TRUE))
  return(log_prior + log_lik)
}
log_w_high <- apply(samples_high, 1, logpost_high)
log_q_high <- dmvnorm(samples_high, mean = mu_high, sigma = Sigma_high, log = TRUE)
w_high <- exp(log_w_high - log_q_high - max(log_w_high - log_q_high))
w_high <- w_high / sum(w_high)
ess_high_snis <- 1 / sum(w_high^2)
mean_high_snis <- colSums(samples_high * w_high)
```

```r
ess_high_params <- ess_calc(samples_high, w_high)
ess_high_params


# ============ Section 3: ESS Plot ============
# --------- LOW dim---------
library(ggplot2)
library(tibble)
params <- c("beta_age", "beta_weight", "beta_waist", "beta_bp_1d", "beta_bp_1s", "sigma")

# ESS value
ess_mcmc_raw <- c(2930, 2165, 2088, 2875, 2224, 3727)
ess_snis_raw <- c(9784.773, 9902.990, 9777.402, 9760.624, 9746.532, 9666.207)

# to percentage
ess_mcmc_pct <- ess_mcmc_raw / 4000 * 100
ess_snis_pct <- ess_snis_raw / 10000 * 100

# build tibble
ess_df <- tibble(
  parameter = rep(params, each = 2),
  method = rep(c("MCMC", "SNIS"), times = length(params)),
  ess_pct = as.vector(rbind(ess_mcmc_pct, ess_snis_pct))
)

# plot
ggplot(ess_df, aes(x = parameter, y = ess_pct, fill = method)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(
    title = "ESS (%) Comparison - Low Dimension",
    x = "Parameter", y = "ESS (%)"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5)
  )


# --------- MID dim---------
params <- c("beta_age", "beta_weight", "beta_waist", "beta_bp_1d", "beta_bp_1s",
            "beta_chol", "beta_stab_glu", "beta_hdl", "beta_height", "sigma")

# ESS value
ess_mcmc_raw <- c(3742, 3038, 3054, 3602, 3193, 3976, 4872, 4380, 4500, 4718)
ess_snis_raw <- c(10302.263, 9941.986, 9809.119, 9574.688, 9834.763,
                  10091.193, 9628.146, 10430.937, 10406.009, 9848.503)

# To percentage
ess_mcmc_pct <- pmin(ess_mcmc_raw / 4000 * 100, 100)
ess_snis_pct <- pmin(ess_snis_raw / 10000 * 100, 100)

# Build tibble
ess_df <- tibble(
  parameter = rep(params, each = 2),
  method = rep(c("MCMC", "SNIS"), times = length(params)),
  ess_pct = as.vector(rbind(ess_mcmc_pct, ess_snis_pct))
)

# Plot
ggplot(ess_df, aes(x = parameter, y = ess_pct, fill = method)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(
    title = "ESS (%) Comparison - Mid Dimension",
```

```
    x = "Parameter", y = "ESS (%)"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5)
  )




# ————————— HIGH dim—————————
params <- c(
  paste0("alpha[", 1:3, "]"),
  paste0("beta_weight[", 1:3, "]"),
  paste0("beta_waist[", 1:3, "]"),
  "beta_age", "beta_bp_1d", "beta_bp_1s",
  "beta_chol", "beta_stab_glu", "beta_hdl", "beta_height", "beta_gender", "sigma"
)

# ESS value
ess_mcmc_raw <- c(
  3214, 3642, 4810,
  3249, 3112, 3836,
  3014, 3142, 3827,
  4192, 3417, 3535,
  6654, 7019, 7027, 3159, 2681, 6580
)

ess_snis_raw <- c(
  4797.537, 4805.271, 4699.099,
  5310.169, 4846.996, 4837.563,
  5164.937, 4674.787, 4795.426,
  5017.904, 4494.935, 4691.733,
  5055.658, 4962.141, 4883.089,
  5127.854, 4967.354, 6634.882
)

# To percentage
ess_mcmc_pct <- pmin(ess_mcmc_raw / 4000 * 100, 100)
ess_snis_pct <- pmin(ess_snis_raw / 10000 * 100, 100)

# build tibble
ess_df <- tibble(
  parameter = rep(params, each = 2),
  method = rep(c("MCMC", "SNIS"), times = length(params)),
  ess_pct = as.vector(rbind(ess_mcmc_pct, ess_snis_pct))
)

# PLot
ggplot(ess_df, aes(x = parameter, y = ess_pct, fill = method)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(
    title = "ESS (%) Comparison — High Dimension",
    x = "Parameter", y = "ESS (%)"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1, size = 9),
    plot.title = element_text(hjust = 0.5)
  )
```

```r
# ============ Section 4: CI ============
# ========= weighted quantile function =========
weighted_quantile <- function(x, w, probs = c(0.025, 0.975)) {
  ord <- order(x)
  x_sorted <- x[ord]
  w_sorted <- w[ord]
  cum_w <- cumsum(w_sorted) / sum(w_sorted)
  approx(cum_w, x_sorted, xout = probs, method = "linear", ties = "ordered")$y
}

# =========   SNIS CI function =========
compute_snis_ci <- function(samples, weights, param_names = NULL) {
  weights <- weights / sum(weights)
  mean_vec <- colSums(samples * weights)
  ci_bounds <- t(apply(samples, 2, function(col) weighted_quantile(col, weights)))

  result <- tibble(
    parameter = if (!is.null(param_names)) param_names else colnames(samples),
    method = "SNIS",
    mean = mean_vec,
    lower = ci_bounds[, 1],
    upper = ci_bounds[, 2]
  )

  return(result)
}
# --------- LOW dim---------
ci_low_snis <- compute_snis_ci(theta_low, w_low,
                                 param_names = c("beta_age", "beta_weight", "beta_waist",
"bp_1d", "bp_1s", "sigma"))
print(ci_low_snis)



# --------- MID dim---------
ci_mid_snis <- compute_snis_ci(
  theta_mid, w_mid,
  param_names = c("beta_age", "beta_weight", "beta_waist",
                  "beta_bp_1d", "beta_bp_1s", "beta_chol",
                  "beta_stab_glu", "beta_hdl", "beta_height", "sigma")
)
print(ci_mid_snis)


# --------- HIGH dim---------
param_high <- c(
  paste0("alpha[", 1:G_high, "]"),
  paste0("beta_weight[", 1:G_high, "]"),
  paste0("beta_waist[", 1:G_high, "]"),
  "beta_age", "beta_bp_1d", "beta_bp_1s",
  "beta_chol", "beta_stab_glu", "beta_hdl", "beta_height", "beta_gender",
  "sigma"
)

theta_high <- draws_high[, param_high]
ci_high_snis <- compute_snis_ci(
  theta_high,
  w_high,
  param_names = param_high
)
```

```r
print(ci_high_snis)




# ============ Section 5: CI plot ============
# --------- Low dim---------
ci_low_mcmc <- tibble(
  parameter = c("beta_age", "beta_weight", "beta_waist", "beta_bp_1d", "beta_bp_1s",
"sigma"),
  method = "MCMC",
  lower = c(-0.12, -1.10, -1.00, -0.94, -0.76, 5.60),
  upper = c(1.39, 1.42, 1.56, 0.72, 1.07, 6.47)
)

ci_low_snis <- tibble(
  parameter = c("beta_age", "beta_weight", "beta_waist", "beta_bp_1d", "beta_bp_1s",
"sigma"),
  method = "SNIS",
  lower = c(-0.111, -1.09, -0.965, -0.939, -0.773, 5.59),
  upper = c(1.39, 1.41, 1.57, 0.718, 1.05, 6.48)
)

ci_low_df <- rbind(ci_low_mcmc, ci_low_snis)

ggplot(ci_low_df, aes(x = parameter, ymin = lower, ymax = upper, color = method)) +
  geom_errorbar(position = position_dodge(width = 0.6), width = 0.2, size = 1.1) +
  labs(
    title = "95% Credible Interval Comparison - Low Dimension",
    x = "Parameter", y = "Interval Range"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5)
  )

# --------- MID dim---------

ci_mid_mcmc <- tibble(
  parameter = c("beta_age", "beta_weight", "beta_waist", "beta_bp_1d", "beta_bp_1s",
                "beta_chol", "beta_stab_glu", "beta_hdl", "beta_height", "sigma"),
  method = "MCMC",
  lower = c(-0.55, -1.47, -1.13, -0.84, -0.84,
            -0.34, 0.81, -0.80, -0.60, 5.45),
  upper = c(0.93, 1.19, 1.49, 0.79, 0.95,
            0.96, 2.17, 0.49, 0.69, 6.31)
)

ci_mid_snis <- tibble(
  parameter = c("beta_age", "beta_weight", "beta_waist", "beta_bp_1d", "beta_bp_1s",
                "beta_chol", "beta_stab_glu", "beta_hdl", "beta_height", "sigma"),
  method = "SNIS",
  lower = c(-0.570, -1.47, -1.08, -0.822, -0.818,
            -0.338, 0.837, -0.798, -0.613, 5.45),
  upper = c(0.930, 1.14, 1.53, 0.770, 0.934,
            0.960, 2.13, 0.502, 0.711, 6.31)
)

ci_mid_df <- rbind(ci_mid_mcmc, ci_mid_snis)

ggplot(ci_mid_df, aes(x = parameter, ymin = lower, ymax = upper, color = method)) +
```

```r
  geom_errorbar(position = position_dodge(width = 0.6), width = 0.2, size = 1.1) +
  labs(
    title = "95% Credible Interval Comparison - Mid Dimension",
    x = "Parameter", y = "Interval Range"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5)
  )




# --------- HIGH dim---------
ci_high_mcmc <- tibble(
  parameter = c("alpha[1]", "alpha[2]", "alpha[3]",
                "beta_weight[1]", "beta_weight[2]", "beta_weight[3]",
                "beta_waist[1]", "beta_waist[2]", "beta_waist[3]",
                "beta_age", "beta_bp_1d", "beta_bp_1s",
                "beta_chol", "beta_stab_glu", "beta_hdl", "beta_height", "beta_gender",
"sigma"),
  method = "MCMC",
  lower = c(5.18, 5.45, 5.51,
            -0.75, -0.69, -0.24,
            -0.38, 0.06, -0.87,
            0.00, -0.23, -0.16,
            0.15, 1.34, -0.30, -0.14, -0.31, 1.35),
  upper = c(6.09, 6.02, 6.14,
            0.25, 0.21, 1.13,
            0.72, 0.81, 0.43,
            0.39, 0.18, 0.29,
            0.36, 1.67, 0.03, 0.32, 0.28, 1.56)
)

ci_high_snis <- tibble(
  parameter = c("alpha[1]", "alpha[2]", "alpha[3]",
                "beta_weight[1]", "beta_weight[2]", "beta_weight[3]",
                "beta_waist[1]", "beta_waist[2]", "beta_waist[3]",
                "beta_age", "beta_bp_1d", "beta_bp_1s",
                "beta_chol", "beta_stab_glu", "beta_hdl", "beta_height", "beta_gender",
"sigma"),
  method = "SNIS",
  lower = c(5.17, 5.45, 5.30,
            -0.742, -0.714, -0.248,
            -0.366, -0.00129, -0.854,
            0.000609, -0.241, -0.143,
            0.133, 1.35, -0.294, -0.105, -0.581, 1.36),
  upper = c(6.07, 6.02, 6.10,
            0.236, 0.169, 1.15,
            0.743, 0.810, 0.418,
            0.410, 0.208, 0.293,
            0.470, 1.65, 0.0290, 0.320, 0.259, 1.56)
)

ci_high_df <- rbind(ci_high_mcmc, ci_high_snis)

ggplot(ci_high_df, aes(x = parameter, ymin = lower, ymax = upper, color = method)) +
  geom_errorbar(position = position_dodge(width = 0.6), width = 0.3, size = 1.0) +
  labs(
    title = "95% Credible Interval Comparison - High Dimension",
    x = "Parameter", y = "Interval Range"
  ) +
  theme_minimal() +
```

```
theme(
  axis.text.x = element_text(angle = 45, hjust = 1, size = 9),
  plot.title = element_text(hjust = 0.5)
)
```