GSoC: Periodic Time Changepoint Detection Medium Test

Haoting Tan

2024-04-01

Medium Test

Goal:

write a function to take the (Bayesian) implementation from the easy task and identify the best location for a global changepoint in the Bayesian periodic changepoint process. Hint: use the original implementation and check each possible location for a changepoint. Return the maximum difference in the fit before and after the change.

Function Body

```
evaluate_segment <- function(data_segment) {
  # Estimate the parameter of the Bernoulli distribution
  p_hat <- mean(data_segment)

  # Calculate the log-likelihood of the data segment under the Bernoulli model
  log_likelihood <- sum(dbinom(data_segment, size = 1, prob = p_hat, log = TRUE))

  return(log_likelihood)
}

find_best_changepoint <- function(binary_data) {
  best_location <- NULL
  max_diff <- -Inf

  # Iterate over all possible changepoint locations
  for (cp_location in 2:(length(binary_data) - 1)) {
    # Split the data at the current candidate location
    data_before <- binary_data[1:(cp_location - 1)]
    data_after <- binary_data[cp_location:length(binary_data)]

    # Compute the fit for data before and after the changepoint
    fit_before <- evaluate_segment(data_before)
    fit_after <- evaluate_segment(data_after)

    diff_fit <- fit_after - fit_before

    # Update
    if (diff_fit > max_diff) {
```

```
      max_diff <- diff_fit
      best_location <- cp_location
    }
  }

  print(paste("Best location in the Data:", as.character(best_location)))
  print(paste("Max difference is:", as.character(max_diff)))
}
```

Run the function using generated binaray data from easy task

```
set.seed(217) # For reproducibility
binary_data = ts( rbinom(90, size = 1, prob = rep(c(0.2, 0.8), each=6)), freq = 12)
find_best_changepoint(binary_data)
```

```
## [1] "Best location in the Data: 89"
## [1] "Max difference is: 60.7922476588328"
```