bootstrap_code.R

March 3, 2019

```
# Load bootstrap data
bootstrap_data <- read.csv('bootstrap.csv', colClasses = c('NULL', 'integer', 'character'))

# find count for digits 1-9 per recipient
grouped_id <- split(x = bootstrap_data, f = bootstrap_data$parent_recipient_unique_id)
id_dist <- lapply(grouped_id, function(x)
  table(factor(x[,c('total_obligation')], levels = as.character(1:9))))

# Find Q(x) for the whole dataset
comb_dist <- do.call('rbind', id_dist)
comb_sum <- colSums(comb_dist)
Q_dist <- comb_sum / sum(comb_sum)

# Find P(x) per ID
id_table <- lapply(id_dist, function(x) x / sum(x))

# KLD function
KLD <- function(P,Q) { # Calculate the KLD for a given P, Q
  return(sum(P*log(P/Q), na.rm = TRUE))
}

# Find KLD per ID
id_KLD <- lapply(id_table, function(x) KLD(P = x, Q = Q_dist))
KLD_df <- do.call('rbind', id_KLD)

### Bootstrap
id_sizes <- rowSums(comb_dist) # Length of each ID
bootstrap_fn <- function(id_length, id_P, id_index, true_Q = Q_dist) {
  # id_length: the number of observations per ID
  # id_P: the P(x) for each ID
  # id_index: the current index of the ID from the dataset
  # true_Q: Q(x) for the dataset, set to Q_dist
  # Perform one bootstrap for a particular ID to find the KLD
  boot <- sample(x = 1:9, size = id_length, replace = TRUE, prob = id_P)
  boot_P <- table(factor(boot, levels = as.character(1:9))) / id_length
  boot_KLD <- KLD(P = boot_P, Q = true_Q)
  return(boot_KLD)
}

get_KLD_bounds = function(id_index, id_length, n_reps = 1000L, alpha = 0.05, curr_id_table = id_table) {
  # id_index: the current index of the ID from the dataset
  # id_length: the number of observations per ID
  # n_reps: the number of times to do a bootstrap
  # alpha: the confidence level for the quantile
  # curr_id_table: the table of probabilities for the id
  # Bootstrap 1000 times the KLD for each ID, and gather the quantile data
  KLD_1000 <- replicate(n = n_reps, bootstrap_fn(id_length = id_length,
```

```
    id_P = curr_id_table[[id_index]], id_index = id_index))
  bounds <- quantile(KLD_1000, c(alpha/2, 1-alpha/2))
  return(bounds)
}

# Gather quantiles from bootstrap for all recipients
KLD_quantile <- lapply(1:length(id_sizes),
                       function(x) get_KLD_bounds(id_index = x, id_length = id_sizes[x]))
KLD_quant_df <- do.call('rbind', KLD_quantile)

# Write to csv
write.csv(KLD_ci, 'KLD_ci.csv')
```