

```

## This R code reads a dataset, constructs empirical pmf
## generate Bootstrap resamples of the data
## calculate point estimate of mean
## uses Bootstrap resamples to calculate
## 90% confidence interval for mean
## utilizing percentile bootstrap method

dataset = as.matrix(read.table("folderpath/filename.extension", header = T or
F))

n = length(dataset) ## size of dataset

print("sample size")
print(n)

## empirical pmf construction

## only for values that occur in the data
## probability mass is no of occurrences divided by sample size n

empirical_pmf = function(x){length(which(dataset==x))/n}

## identify the unique values in the dataset
## arrange them from smallest to largest
empirical_pmf_points = sort(unique(dataset))

## how many unique values
no_of_unique_values = length(empirical_pmf_points)

## now determine the probability mass at those unique values
empirical_pmf_values = sapply(empirical_pmf_points,empirical_pmf)

## Now display it in a tabular form as we did in class
print("##### Empirical pmf #####")
epmf_table = data.frame(points = empirical_pmf_points, probability_mass =
empirical_pmf_values )
print(epmf_table, row.names=F)
print("#####")

## input how many bootstrap resamples of the data
## you like to generate

B =

## create a storage matrix for the bootstrap resamples
## about to be generated
## each row will contain one bootstrap resample
bootstrap_resamples = matrix(0,B,n)

## now generate the resamples
for (i in 1:B)

```

```

{
## a resample consists of n values drawn (with replacement) from the empirical
  pmf distribution
bootstrap_resamples[i, ] =
  sample(empirical_pmf_points,n,empirical_pmf_values,replace=TRUE)
}

## bootstrap resamples are ready
## now use them to construct 90% C.I. for mean

## first, point estimate of mean using original dataset
## sample mean is the point estimate

point_estimate_for_mean = mean(dataset)
print(paste("Point estimate for mean is =
  ",round(point_estimate_for_mean,2),sep=""))

## now construct 90% confidence interval using Percentile Bootstrap

sample_mean_for_each_bootstrap_resample = apply(bootstrap_resamples,1,mean)

## left boundary of CI is 5% quantile of the
## means of bootstrap resamples
CI_lower_limit = quantile(sample_mean_for_each_bootstrap_resample,p=0.05)

## right boundary of CI is 95% quantile of the
## means of bootstrap resamples
CI_upper_limit = quantile(sample_mean_for_each_bootstrap_resample,p=0.95)

print(paste("90% Bootstrap C.I. for mean is =
  (",round(CI_lower_limit,2),",",round(CI_upper_limit,2),")",sep=""))

```