Statistics 406 Lab 3
GSI: Adam Rothman
Tuesday September 26

How to make a scatter plot in R

Use the `plot(x,y)` command where `x` is your x axis list of values and `y` is your corresponding y axis list of values. After you generate the plot, you can right click the plot and copy it to the clipboard as a bitmap and paste it into MSWORD. You can also save the plot as a post-script file.

Some important terminology

- $\hat{\theta}$ estimates a population parameter θ, where $\hat{\theta}$ is a function of our data

  For one example, let our data be $X_1, \ldots, X_n$ iid N(μ, σ^2). Let $\hat{\theta} = \overline{X}$, which is a function of our data, which we know is an estimator for $\theta = \mu$.

  For a second example, let our data be $X_1, \ldots, X_n$ iid U(0,θ). Let $\hat{\theta} = \max(X_1, \ldots, X_n)$, which is a function of our data, which is an estimator for θ.

- $\mathrm{var}(\hat{\theta})$ is the variance of this estimator
  For example, let our data be $X_1, \ldots, X_n$ iid N(μ, σ^2). Let $\hat{\theta} = \overline{X}$,

$$\mathrm{var}(\hat{\theta}) = \mathrm{var}(\overline{X}) = \frac{\sigma^2}{n}$$

- $\mathrm{bias}(\hat{\theta})$ is the bias of this estimator

$$\mathrm{bias}(\hat{\theta}) = E(\hat{\theta}) - \theta$$

  For example, let our data be $X_1, \ldots, X_n$ iid N(μ, σ^2). Let $\hat{\theta} = \overline{X}$,

$$\mathrm{bias}(\hat{\theta}) = \mathrm{bias}(\overline{X}) = E(\overline{X}) - \mu = 0$$

- $\mathrm{MSE}(\hat{\theta})$ is the mean-squared error of this estimator

$$\mathrm{MSE}(\hat{\theta}) = E[(\hat{\theta} - \theta)^2]$$
$$\mathrm{MSE}(\hat{\theta}) = [\mathrm{bias}(\hat{\theta})]^2 + \mathrm{var}(\hat{\theta})$$

  For example, let our data be $X_1, \ldots, X_n$ iid N(μ, σ^2). Let $\hat{\theta} = \overline{X}$,

$$\mathrm{MSE}(\hat{\theta}) = \mathrm{MSE}(\overline{X}) = E[(\overline{X} - \mu)^2]$$

$$\mathrm{MSE}(\overline{X}) = [\mathrm{bias}(\overline{X})]^2 + \mathrm{var}(\overline{X})$$

$$\text{MSE}(\overline{X}) = [0]^2 + \frac{\sigma^2}{n}$$

$$\text{MSE}(\overline{X}) = \frac{\sigma^2}{n}$$

Exercises

1. Use simulation to estimate the bias, variance and MSE when using the sample median to estimate the population **mean** for a uniformly distributed population between 0 and 1. Consider sample sizes 10,20,40, and 80.

```
reps <- 1000

bias_med <- NULL
Var_med <- NULL
MSE_med <- NULL

bias_mean <- NULL
Var_mean <- NULL
MSE_mean <- NULL

populationMean <- 1/2

for (n in c(10,20,40,80))
{
  Z <- array(runif(n*reps), c(n,reps))
  Med <- apply(Z, 2, median)
  Mean <- apply(Z, 2, mean)

  bias_med <- c(bias_med, mean(Med) - populationMean)
  Var_med <- c(Var_med, var(Med))
  MSE_med <- c(MSE_med, mean((Med-populationMean)^2))

  bias_mean <- c(bias_mean, mean(Mean) - populationMean)
  Var_mean <- c(Var_mean, var(Mean))
  MSE_mean <- c(MSE_mean, mean((Mean-populationMean)^2))
}
```

2. Use simulation to estimate the bias, variance and MSE when using the sample median to estimate the population **median** for a uniformly distributed population between 0 and 1. Consider sample sizes 10,20,40, and 80.

```
reps <- 1000

bias_med <- NULL
Var_med <- NULL
MSE_med <- NULL

bias_mean <- NULL
Var_mean <- NULL
MSE_mean <- NULL

populationMedian <- 1/2

for (n in c(10,20,40,80))
{
  Z <- array(runif(n*reps), c(n,reps))
  Med <- apply(Z, 2, median)
  Mean <- apply(Z, 2, mean)

  bias_med <- c(bias_med, mean(Med) - populationMedian)
  Var_med <- c(Var_med, var(Med))
  MSE_med <- c(MSE_med, mean((Med-populationMedian)^2))

  bias_mean <- c(bias_mean, mean(Mean) - populationMedian)
  Var_mean <- c(Var_mean, var(Mean))
  MSE_mean <- c(MSE_mean, mean((Mean-populationMedian)^2))
}
```

3. If $X_1, X_2, \ldots, X_n$ are iid $U(0, \theta)$, analyze the performance of the estimator $\hat{\theta} = \max(X_1, X_2, \ldots, X_n)$. Disuss the variance, bias, and MSE of $\hat{\theta}$, then provide some simulations to verify your results.

Use the hints in the comments to help you with your modification of the cluter mean code for the homework.

```
## From the lecture notes, here are some hints
##  for modification of the code

## Number of clusters.
nc <- 100
## Number of observations per cluster.
cs <- 5
## Shrinkage factors.
F <- c(0.5, 0.6, 0.7, 0.8, 0.9, 1)    # Modification Needed here
                                      # We want F to vary from 0 to 1 by 0.05
  # consider using the seq() command

## Save all the estimates.
Estimate <- array(0, c(1000,6))       # Modification
  # Want to save an estimate for
  # for each cluster, for each lambda,
  # and each repetition
  # Consider adding a dimension to the array
  # that corresponds to each cluster
## True cluster means.
CM <- rnorm(nc)

## Simulation replications.
for (r in 1:1000)
{
  ## Estimated cluster means.
  Z <- array(0, nc)
  for (k in 1:nc)
  {
    z <- CM[k] + rnorm(cs)
    Z[k] <- mean(z)
  }

  ## The grand mean
  GM <- mean(Z)
  for (k in 1:length(F))
  {
    ## Modify the cluster 1 mean estimate by shrinking toward   ## change this
```

```
    ## the grand mean.
    S <- GM + F[k]*(Z[1]-GM)         ## Modification Needed here
    Estimate[r,k] <- S  ## Modification Needed here (especially if
  }  ## Estimate has more dimensions)
}

## Need to use the Estimate array to find the
## bias, variance, and mse for each cluster
## conside looping here to get the Bias, Var, MSE for each cluster

Bias <- apply(Estimate, 2, mean) - CM[1]  ## Modification needed here
Var <- apply(Estimate, 2, var)            ## Modification needed here
MSE <- apply((Estimate-CM[1])^2, 2, mean)  ## Modification needed here

## After you update the code, you will now want to calculate the average
## MSE of the clusters, one aveMSE for each value of lambda

## You will also want to examine the Bias, Bias^2, and Variance similarly
```