

# Stat 425 HW5

Fritz Scholz

1. Anatomy of an R function. Execute the following function with the given default arguments.

```
Using.get=function(N=5, dist="norm") {  
  set.seed(26)  
  x=get(paste("r", dist, sep=""))(N)  
  x  
}
```

Follow this with the command lines

```
set.seed(26)  
rnorm(5)  
and again  
rnorm(5)
```

To understand the 3rd line in `Using.get` execute the following commands in the given progression and describe your understanding of what is happening.

```
paste("r", "norm")  
paste("r", "norm", sep="")  
get(paste("r", "norm"))  
get(paste("r", "norm", sep=""))  
rnorm  
set.seed(26)  
rnorm(5)  
set.seed(26)  
get(paste("r", "norm", sep=""))(5)
```

What is the usefulness of the argument `dist` in `Using.get`? Try the call

```
Using.get(N=5, dist="unif")
```

Possible values for `dist` are `unif`, `norm`, `lnorm`, `logis`, `exp`, `cauchy`. If you put an `r` before any of those strings you get the command for a random sample from that continuous distribution. For example, you get information on `logis` by searching in the web based help facility `help.start()` for `rlogis` or by executing `?rlogis` at the command line. All of these random sample generators have default arguments, but there are others (like `rf`, `rt`, `rweibull`, `rgamma`) that do not. If you use one of these latter random sample generating functions the above construct of `Using.get` would fail. To accommodate these the function `Using.get` would need some modification, but we will pass on that for now.

2. Via simulation you are to investigate the sampling distributions of the Hodges-Lehmann estimator  $\hat{\Delta} = \text{median}(Y_j - X_i)$  and of  $\bar{\Delta} = \bar{Y} - \bar{X}$  in the shift model for various distributions  $F$ . Write a function `Delta.est.sim(Nsim=10000, m=15, n=15, dist="norm", dist.name="normal", Delta=2) {...}` that uses a loop `for( i in 1:Nsim ) {...}` to simulate samples  $x$  of size  $m$  from  $F(x)$  (corresponding to `dist`, make use of what you learned from Problem 1.) and samples  $y$  of size  $n$  from  $F(x - \Delta)$  (i.e., shift  $\Delta = 2$  for the above default argument) and computes  $\hat{\Delta}$  and  $\bar{\Delta}$  for these two samples. Accumulate these estimates in respective vectors `Delta.hat` and `Delta.bar` of length `Nsim` each. While you develop `Delta.est.sim` use `Nsim=1000` but for the final plots use `Nsim=10000`.

You are to illustrate through plots (to be produced by `Delta.est.sim`) the following results from class, when they are known to be true and when they might be false.

The distributions of both estimators are symmetric around  $\Delta$  when  $m = n$  or when  $F$  is symmetric around some point  $\mu$ . The symmetry of the simulated distribution `y=Delta.hat` (and similarly `y=Delta.bar`) around  $\Delta$  is most effectively illustrated by using `x=y-Delta` and

```
qqplot(x[x>0], -x[x<0], pch=16, cex=.5)
abline(0, 1)
```

Here `x[x>0]` is the positive part of the  $x$ -distribution while `-x[x<0]` is the negative part reflected around zero. If both parts have equal length then `qqplot` plots the smallest of one part against the smallest of the other, then the second smallest ones are plotted against each other, ..., and finally the largest ones are plotted against each other. If the two parts are of different length then the sorted larger vector is interpolated to give the same length as the shorter vector and these shorter vectors are plotted against each other according to the above scheme. When the distribution of  $x$  is symmetric around zero, then the plotted point pattern should align well with the main diagonal. The extremes in any data set typically show greater fluctuations. Thus one should be more forgiving for the large observation (the tail values) when judging closeness to the main diagonal. `Delta.est.sim` should create annotated example plots (indicating `dist.name`, `m`, `n` and estimator type) that show

1. symmetry when  $m = n$  even when  $F$  is not symmetric.
2. asymmetry when  $m \neq n$  and  $F$  is not symmetric (make  $m$  and  $n$  sufficiently different so that the asymmetry becomes obvious).
3. symmetry when  $m \neq n$  and  $F$  is symmetric ( $m$  and  $n$  as different as in the previous illustration).

Do the above illustration for both the simulated  $\hat{\Delta}$  and  $\bar{\Delta}$  distribution as captured by `Delta.hat` and `Delta.bar`. Give the code for `Delta.est.sim` and these plots. If you write a function that produces several plots and you want to look at them at your own pace (also for cutting and pasting) you should add the following command after each plot is finished:

```
readline("hit return\n")
```

Optionally, for extra credit, illustrate the corresponding results or counter examples concerning unbiasedness and median unbiasedness. Here you would be looking at the mean of the generated distribution vectors, compare them with  $\Delta$  and look at the proportion of the distributions  $> \Delta$  and  $< \Delta$ .