36-401 Class Notes and Exercise for 12/5/00

Computer Intensive Methods I (RwG 303-326)

Howard Seltman

I. Monte Carlo Simulation

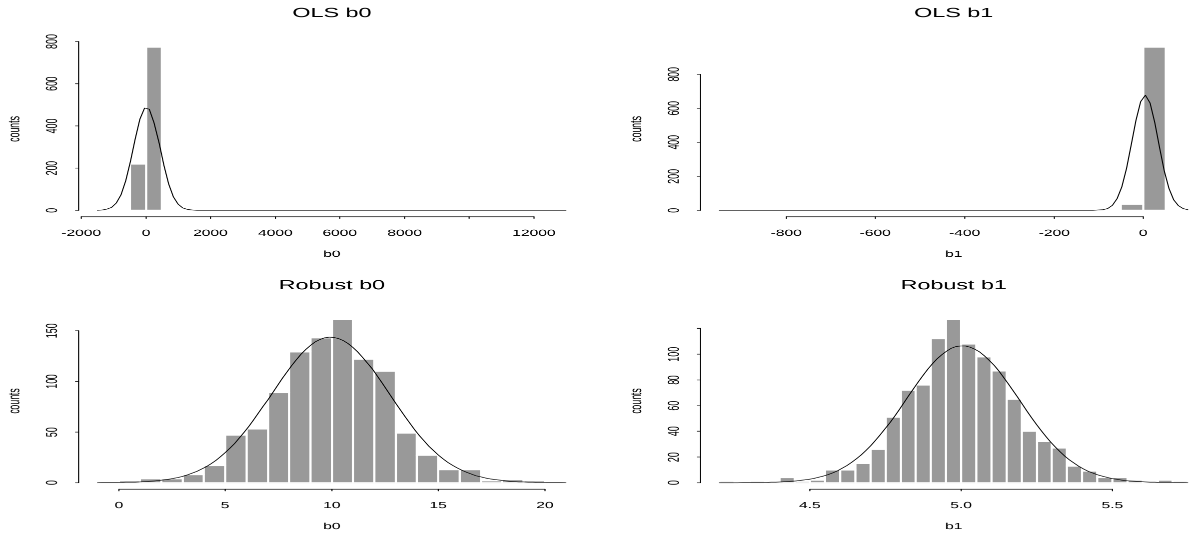
Main Idea: Sampling distributions are difficult (or impossible) to calculate when errors are non-normal, especially if sample sizes are small.

```
n _ 50
X _ round(rnorm(n, 15, 2), 2)
sim _ matrix(NA, 1000, 8)
b0 _ 10;    b1 _ 5
library("MASS")

for (i in 1:1000) {
  Y _ b0 + b1*X + rt(n, 1)
  rslt _ lm(Y ~ X)
  sim[i, 1:4] _ as.vector(summary(rslt)$coef[,1:2])
  rslt _ rlm(Y ~ X)
  sim[i, 5:8] _ as.vector(summary(rslt)$coef[,1:2])
  synchronize()  # prevents memory overusage
  cat(i,"\r")
}
cat("\n")

round(apply(sim, 2, mean),2) #  23.83  3.99 48.41  3.26  9.94  5.00  2.59  0.17
round(apply(sim, 2, sd),2)   # 407.12  29.45 332.33  22.35   2.78   0.19   0.57   0.04
```

```
apply(sim[,c(1,2,5,6)],2,summary)
#            OLS b0   OLS b1   RR b0 RR b1
#    Min. -1064.000 -908.900  -0.521 4.248
# 1st Qu.     1.421    4.447   8.207 4.890
# Median      9.966    5.009  10.010 4.996
#   Mean     23.830    3.987   9.941 5.004
# 3rd Qu.    17.980    5.574  11.690 5.118
#    Max. 12580.000   69.850  20.930 5.717
```



Cosinor regression example

Breakout Session for 12/5/00

```
n _ 50;  X _ round(rnorm(n, 15, 2), 2)
sim _ matrix(NA, 1000, 8)
b0 _ 10;  b1 _ 5;  residvar _ 2

for (i in 1:1000) {
  Xe _ X+rnorm(n, 0, sqrt(0.5))
  Y _ b0 + b1*X + rnorm(n, 0, 1)
  rslt _ lm(Y ~ Xe)
  sim[i, 1:4] _ as.vector(summary(rslt)$coef[,1:2])
  rslt _ rlm(Y ~ Xe)
  sim[i, 5:8] _ as.vector(summary(rslt)$coef[,1:2])
  synchronize()
  cat(i,"\r")
}
cat("\n")

print(round(apply(sim, 2, mean),2))
# 18.00  4.47  3.56  0.23 17.97  4.47  3.64  0.24
print(round(apply(sim, 2, sd),2))
# 3.41 0.22 0.44 0.03 3.44 0.23 0.50 0.03

cat("Efficiencies:",
    round(100*var(sim[,1])/var(sim[,5]), 1), "%   ",
    round(100*var(sim[,2])/var(sim[,6]), 1), "%\n")
# Efficiencies: 98.6 %    98.7 %

tval_qt(0.975, 48)
round( sum(sim[,1]-tval*sim[,3] <=10 & sim[,1]+tval*sim[,3] >=10)/1000*100, 1)
round( sum(sim[,2]-tval*sim[,4] <=5 & sim[,2]+tval*sim[,4] >=5)/1000*100, 1)
round( sum(sim[,5]-tval*sim[,7] <=10 & sim[,5]+tval*sim[,7] >=10)/1000*100, 1)
round( sum(sim[,6]-tval*sim[,8] <=5 & sim[,6]+tval*sim[,8] >=5)/1000*100, 1)
# 39.7  38.7  42.3  40.7
```

1) What is being simulated? What are the conclusions? Which conclusions are expected and which are surprising?