Online supplemental material, Oleson et al., "Essential Statistical Concepts for Research in Speech, Language, and Hearing Sciences," JSLHR, https://doi.org/10.1044/2018_JSLHR-S-ASTM-18-0239

Supplemental Material: Model Selection Simulation

Oleson, Brown, McCreery

The Effects of Model Selection

This brief simulation is designed to illustrate the effects of model selection on hypothesis testing, particularly in a regression setting. The examples considered here are quite simple; in real data analyses, the picture may be worse due to correlation among predictors or other regression assumption violations.

Problem Set Up

The function below simulates a data set with N observations and p covariates. The outcome, Y, follows the normal multiple linear regression model, with all β coefficients equal to zero. In other words, the global null hypothesis is true.

```
simulateData <- function(N=100,p = 10){
  X <- matrix(rnorm(N*p), nrow = N)
  Y <- rnorm(N)
  out <- data.frame(Y=Y, X=X)
  colnames(out) <- c("Y", paste("X", 1:ncol(X), sep = "."))
  out
}
```

The next function is used to conduct N simulations in which the first covariate, X_1 is of primary interest. $Nsim$ iterations are executed, at each of which a null data set is constructed with N observations and p covariates. Forward selection is conducted, starting with just the covariate of interest included. After concluding, the p-value for the variable of interest is compared to the chosen type 1 error rate.

```
simulation1 <- function(Nsim, N, p, alpha = 0.05){
  sum(replicate(Nsim, {
    dat <- simulateData(N, p)
    lm.start <- lm(Y~X.1, data = dat)
    lm.largest <- lm(Y~(.)^2, data = dat)
    finalMod <- step(lm.start, scope = list(upper=formula(lm.largest)),
         direction = "forward", trace=FALSE)
    summary(finalMod)$coefficients[2,4] < alpha
  }))
}
```

The next simulation function is provided for context. The simulation is the same as the previous one, except that only one very large model is fit; no model selection occurs. The model includes all p covariates, as well as all two-way interactions.

```
simulation2 <- function(Nsim, N, p, alpha = 0.05){
  sum(replicate(Nsim, {
    dat <- simulateData(N, p)
    lm.largest <- lm(Y~(.)^2, data = dat)
    summary(lm.largest)$coefficients[2,4] < alpha
  }))
}
```

The next two functions repeat the previous two, except that the overall F test is used, rather than a t-test for the variable of interest.

```
simulation3<- function(Nsim, N, p, alpha = 0.05){
  sum(replicate(Nsim, {
    dat <- simulateData(N, p)
    lm.start <- lm(Y~X.1, data = dat)
    lm.largest <- lm(Y~(.)^2, data = dat)
    finalMod <- summary(step(lm.start, scope = list(upper=formula(lm.largest)),
                             direction = "forward", trace=FALSE))
    (1-pf(finalMod$fstatistic[1],
          df1 = finalMod$fstatistic[2],
          df2 = finalMod$fstatistic[3])) < alpha
  }))
}

simulation4 <- function(Nsim, N, p, alpha = 0.05){
  sum(replicate(Nsim, {
    dat <- simulateData(N, p)
    lm.largest <- summary(lm(Y~(.)^2, data = dat))
    (1-pf(lm.largest$fstatistic[1],
          df1 = lm.largest$fstatistic[2],
          df2 = lm.largest$fstatistic[3])) < alpha
  }))
}
```

Run Simulations

The following code executes the simulations defined above

```
library(parallel)
simSize <- 50000
cl <- makeCluster(8)
clusterExport(cl,
              c("simSize",
                "simulateData",
                "simulation1",
                "simulation2",
                "simulation3",
                "simulation4"))

# Simulation with just one covariate (treatment)
results.1 <- Reduce("+", parLapply(cl, 1:length(cl),
                           function(x){
                             set.seed(123123+x)
                             simulation1(simSize, 100, 1)
                           }))


# Simulation with ten covariates, no model selection
results.2 <- Reduce("+", parLapply(cl, 1:length(cl),
                                   function(x){
                                     set.seed(223123+x)
```

```
                                  simulation2(simSize, 100, 10)
                                }))

# Simulation with ten covariates, stepwise selection
results.3 <- Reduce("+", parLapply(cl, 1:length(cl),
                                function(x){
                                  set.seed(323123+x)
                                  simulation1(simSize, 100, 10)
                                }))



# Overall F-test, single
results.4 <- Reduce("+", parLapply(cl, 1:length(cl),
                                   function(x){
                                     set.seed(423123+x)
                                     simulation4(simSize, 100, 10)
                                   }))



# Overall F-test, Step
results.5 <- Reduce("+", parLapply(cl, 1:length(cl),
                                   function(x){
                                     set.seed(523123+x)
                                     simulation3(simSize, 100, 10)
                                   }))

stopCluster(cl)
```

Results

Table 1: Type 1 Error Rates

Data	Selection	Test	Type1Err
One covariate model	None	Covariate	0.0502
Ten covariates	None	Covariate	0.0496
Ten covariates	Stepwise	Covariate	0.0645
Ten covariates	None	Overall-F	0.0498
Ten covariates	Stepwise	Overall-F	0.4314