R Code for the Paper:

Classroom Simulation: Understanding One-Way Random-Effect ANOVA

Table of contents:

Classroom Simulation: Understanding One-Way Random-Effect ANOVA
Eric A. Suess, Bruce E. Trumbo, and Yun Jiang, CSU, East Bay
Poster: Section on Statistics Education, JSM 2005, Minneapolis, MN

 # Fig. 1. Power (left) and P{MME(σ_A^2) < 0} for g = 6, 30; r = 5.

R code. Generic method, but Legends specific to the given DFs and full screen size.

```
par(mfrow=c(1,2))
x.label <- "psi = Ratio of Batch to Error Variance"

# Left graph
alpha <- .05
head <- "Power Curves: g Batches and r Reps"

g <- 30; r <- 5
dfB <- g - 1; dfE <-g*(r-1)
f.star <- qf(1-alpha,dfB,dfE)
psi <- seq(0,1.5,by=.01)
Power <- 1 - pf(f.star/(r*psi +1), dfB, dfE)
plot(psi, Power, type="l", xlab=x.label, main=head)

g <- 6; r <- 5
dfB <- g - 1; dfE <-g*(r-1)
f.star <- qf(1-alpha,dfB,dfE)
Power <- 1 - pf(f.star/(r*psi +1), dfB, dfE)
lines(psi, Power, lty=2, lwd=2, col="blue")

# Legend
lines(c(1.3, 1.5), c(.32, .32), lty=2, lwd=2, col="blue")
text(1.10, .32, "g=6,  r=5:")
lines(c(1.3, 1.5), c(.4, .4))
text(1.10, .4, "g=30, r=5:")

# Right graph
alpha <- .05
head <- "P{Negative MME of Batch Variance}"

g <- 30; r <- 5
dfB <- g - 1; dfE <-g*(r-1)
psi <- seq(0,1.5,by=.01)
Probability <- pf(1/(r*psi +1), dfB, dfE)
plot(psi, Probability, type="l", xlab=x.label, main=head)

g <- 6; r <- 5
```

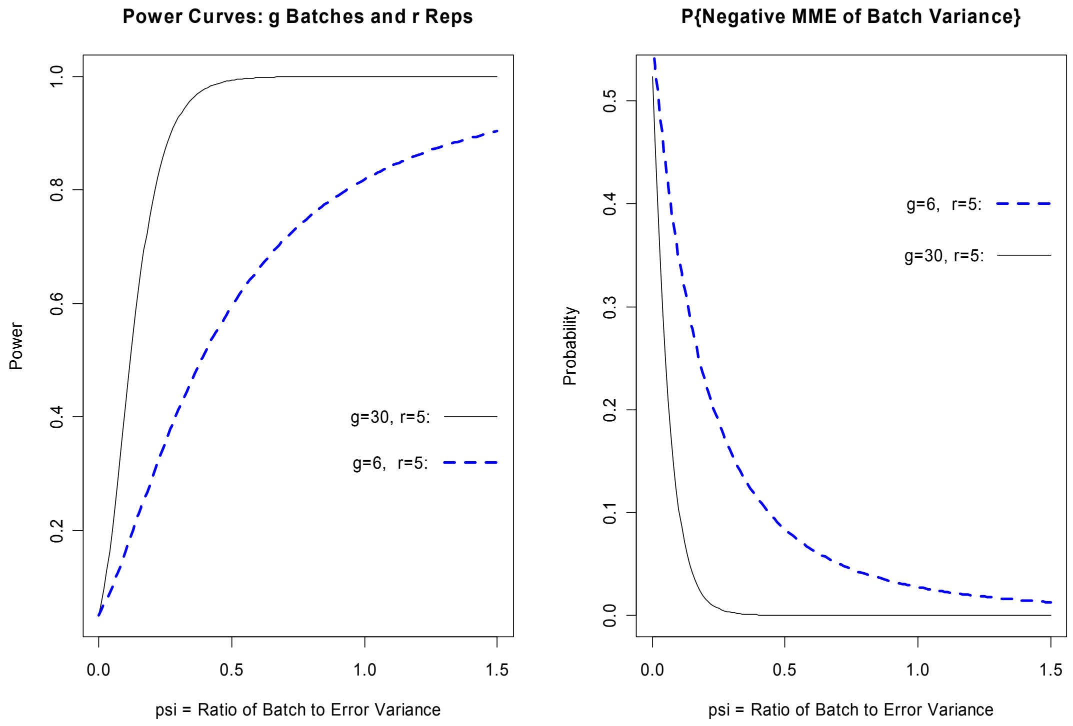
```
dfB <- g - 1; dfE <-g*(r-1)
Probability <- pf(1/(r*psi +1), dfB, dfE)
lines(psi, Probability, lty=2, lwd=2, col="blue")

 # Legend
 lines(c(1.3, 1.5), c(.40, .40), lty=2, lwd=2, col="blue")
 text(1.10, .40, "g=6,  r=5:")
 lines(c(1.3, 1.5), c(.35, .35))
 text(1.10, .35, "g=30, r=5:")

par(mfrow=c(1,1))
```

☆ Output Plot

Fig. 1. Power (left) and P{MME(σ_A^2) < 0} for g = 6, 30; r = 5.



Classroom Simulation: Understanding One-Way Random-Effect ANOVA
Eric A. Suess, Bruce E. Trumbo, and Yun Jiang, CSU, East Bay
Poster: Section on Statistics Education, JSM 2005, Minneapolis, MN

 # Fig. 2. Boxplots for 30 batches of Dataset 1: σ_A = 25, σ = 15.

To simulate different data set, just change the value of grand mean, batch sd. and error sd.

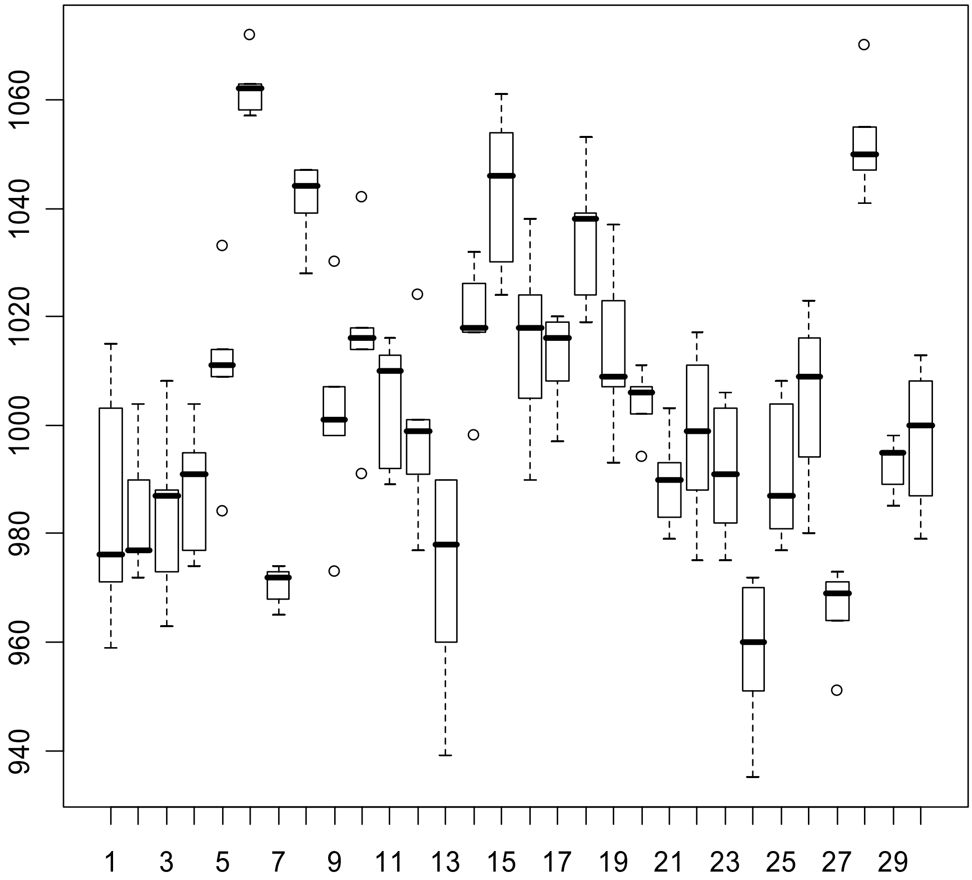
```
#simulate dataset 1
set.seed(1237) # R
g <- 30     # number of batches
r <- 5      # number of replications per batch
mu.grand <- 1000   # the grand mean
sd.Bat <- 25       # the batch stand deviation
sd.Err <- 15       # the error stand deviation
Y <- matrix(0, nrow=r, ncol=g)
mu <- numeric(g)
for (i in 1:g)
 {mu[i] <- rnorm(1, mu.grand, sd.Bat)
   for (j in 1:r)
      {Y[j,i] <- rnorm(1, mu[i], sd.Err)
      }
  }
x <- round(as.vector(Y))
Bat <- as.factor(rep(1:g, each = r))
FRAM <- data.frame(x, Bat)
FRAM
X <- round(t(Y))

### Notes:
### In code, batches are simulated as columns of Y
### In vector x, 5 obs. of each batch are adjacent
### In matrix X, batches are as rows

#boxplot
plot(Bat,x)
```

☆ Output Plot

Fig. 2. Boxplots for 30 batches of Dataset 1: $\sigma_A = 25$, $\sigma = 15$.



Classroom Simulation: Understanding One-Way Random-Effect ANOVA
Eric A. Suess, Bruce E. Trumbo, and Yun Jiang, CSU, East Bay
Poster: Section on Statistics Education, JSM 2005, Minneapolis, MN

 # Fig. 3. ANOVA table for Dataset 1: F significant.
 # MME of grand mean, batch sd., error sd. and ICC

```
#simulate dataset 1
set.seed(1237) # R
g <- 30     # number of batches
r <- 5      # number of replications per batch
mu.grand <- 1000   # the grand mean
sd.Bat <- 25       # the batch stand deviation
sd.Err <- 15       # the error stand deviation
Y <- matrix(0, nrow=r, ncol=g)
mu <- numeric(g)
for (i in 1:g)
 {mu[i] <- rnorm(1, mu.grand, sd.Bat)
   for (j in 1:r)
     {Y[j,i] <- rnorm(1, mu[i], sd.Err)
     }
  }
x <- round(as.vector(Y))
Bat <- as.factor(rep(1:g, each = r))
FRAM <- data.frame(x, Bat)
X <- round(t(Y))

X.bar <- apply(X, 1, mean)  #batch sample means
A <- sum(X^2)
B <- sum(apply(X, 1, sum)^2)/r
C <- (sum(X))^2/(r * g)
dfB <- g-1
dfE <- g*(r-1)
crit <- 0.05          # critical value
MS.Batch <- (B - C)/(g - 1)
MS.Error <- (A - B)/(g * (r - 1))
F.ratio <- MS.Batch/ MS.Error
L <- qf(crit/2, dfB, dfE)
U <- qf(1-crit/2, dfB, dfE)
Est.va <- (MS.Batch - MS.Error)/r   # MME of among group variance
Est.ve <- MS.Error                  # MME of with group variance
CI.ve <- sqrt((A-B)/qchisq(c(.975, .025),dfE))
Est.ICC <- Est.va/(Est.va+Est.ve)   # MME of IIC
CI.ICC <-c((F.ratio-U)/(F.ratio+(r-1)*U),(F.ratio-L)/(F.ratio+(r-1)*L))
```

```
# ANOVA Procedure
anova(lm(x~Bat))

# MME
sqrt(Est.va)
sqrt(Est.ve)
CI.ve
Est.ICC
CI.ICC
```

☆ Output

ANOVA table for Dataset 1: F significant.
& MME

```
> anova(lm(x~Bat))
Analysis of Variance Table

Response: x
           Df Sum Sq  Mean Sq   F value     Pr(>F)
Bat        29   91365     3151    14.993  < 2.2e-16 ***
Residuals 120  25215      210
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
>
> # MME
> sqrt(Est.va)
[1] 24.25035
> sqrt(Est.ve)
[1] 14.49575
> CI.ve
[1] 12.87086 16.59389
> Est.ICC
[1] 0.7367513
> CI.ICC
[1] 0.6099924 0.8451011
>
```

Classroom Simulation: Understanding One-Way Random-Effect ANOVA
Eric A. Suess, Bruce E. Trumbo, and Yun Jiang, CSU, East Bay
Poster: Section on Statistics Education, JSM 2005, Minneapolis, MN

 # Fig. 4. Boxplots for 30 batches of Dataset 2: σ_A = 1, σ = 25.

To simulate different data set, just change the value of grand mean, batch sd. and error sd.

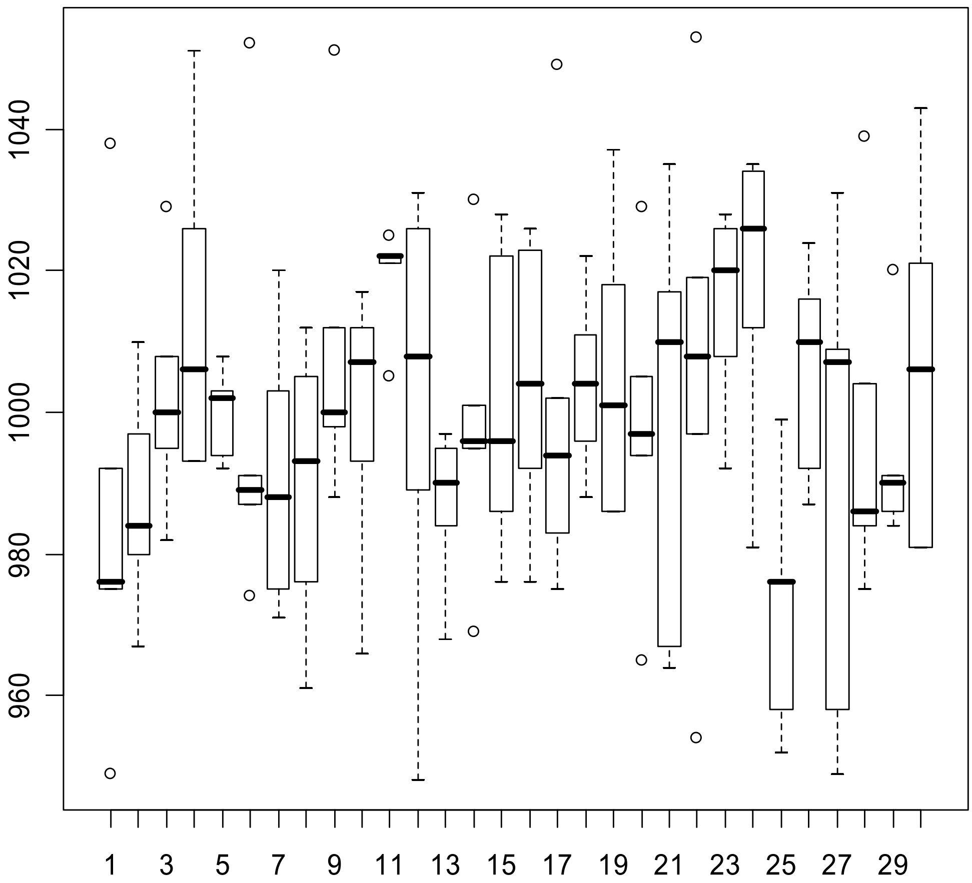
```
#simulate dataset 2
set.seed(12) # R
g <- 30     # number of batches
r <- 5      # number of replications per batch
mu.grand <- 1000   # the grand mean
sd.Bat <- 1        # the batch stand deviation
sd.Err <- 25       # the error stand deviation
Y <- matrix(0, nrow=r, ncol=g)
mu <- numeric(g)
for (i in 1:g)
 {mu[i] <- rnorm(1, mu.grand, sd.Bat)
   for (j in 1:r)
     {Y[j,i] <- rnorm(1, mu[i], sd.Err)
     }
  }
x <- round(as.vector(Y))
Bat <- as.factor(rep(1:g, each = r))
FRAM <- data.frame(x, Bat)
FRAM
X <- round(t(Y))

### Notes:
### In code, batches are simulated as columns of Y
### In vector x, 5 obs. of each batch are adjacent
### In matrix X, batches are as rows

#boxplot
plot(Bat,x)
```

☆ Output Plot

Fig. 4. Boxplots for 30 batches of Dataset 1: σ_A = 1, σ = 25.



Classroom Simulation: Understanding One-Way Random-Effect ANOVA
Eric A. Suess, Bruce E. Trumbo, and Yun Jiang, CSU, East Bay
Poster: Section on Statistics Education, JSM 2005, Minneapolis, MN

 # Fig. 5. ANOVA table for Dataset 2: F not significant.
 # MME of grand mean, batch sd., error sd. and ICC

```
#simulate dataset 2
set.seed(12) # R
g <- 30     # number of batches
r <- 5      # number of replications per batch
mu.grand <- 1000    # the grand mean
sd.Bat <- 1        # the batch stand deviation
sd.Err <- 25        # the error stand deviation
Y <- matrix(0, nrow=r, ncol=g)
mu <- numeric(g)
for (i in 1:g)
 {mu[i] <- rnorm(1, mu.grand, sd.Bat)
    for (j in 1:r)
      {Y[j,i] <- rnorm(1, mu[i], sd.Err)
      }
  }
x <- round(as.vector(Y))
Bat <- as.factor(rep(1:g, each = r))
FRAM <- data.frame(x, Bat)
X <- round(t(Y))

X.bar <- apply(X, 1, mean)   #batch sample means
A <- sum(X^2)
B <- sum(apply(X, 1, sum)^2)/r
C <- (sum(X))^2/(r * g)
dfB <- g-1
dfE <- g*(r-1)
crit <- 0.05          # critical value
MS.Batch <- (B - C)/(g - 1)
MS.Error <- (A - B)/(g * (r - 1))
F.ratio <- MS.Batch/ MS.Error
L <- qf(crit/2, dfB, dfE)
U <- qf(1-crit/2, dfB, dfE)
Est.va <- (MS.Batch - MS.Error)/r    # MME of among group variance
Est.ve <- MS.Error                   # MME of with group variance
CI.ve <- sqrt((A-B)/qchisq(c(.975, .025),dfE))
Est.ICC <- Est.va/(Est.va+Est.ve)    # MME of IIC
CI.ICC <-c((F.ratio-U)/(F.ratio+(r-1)*U),(F.ratio-L)/(F.ratio+(r-1)*L))
```

```
# ANOVA Procedure
anova(lm(x~Bat))

# MME
sqrt(Est.va)
sqrt(Est.ve)
CI.ve
Est.ICC
CI.ICC
```

☆ Output

ANOVA table for Dataset 2: F not significant.
& MME

```
> anova(lm(x~Bat))
Analysis of Variance Table

Response: x
           Df Sum Sq Mean Sq F value Pr(>F)
Bat        29  14931     515  0.9453  0.552
Residuals 120  65362     545
>
> # MME
> sqrt(Est.va)
[1] NaN
Warning message:
NaNs produced in: sqrt(Est.va)
> sqrt(Est.ve)
[1] 23.33838
> CI.ve
[1] 20.72229 26.71643
> Est.ICC
[1] -0.01106697
> CI.ICC
[1] -0.09743496  0.13537798
>
```

Classroom Simulation: Understanding One-Way Random-Effect ANOVA
Eric A. Suess, Bruce E. Trumbo, and Yun Jiang, CSU, East Bay
Poster: Section on Statistics Education, JSM 2005, Minneapolis, MN

 # Fig.6. Maximum likelihood estimates for Dataset 1.

To get the MME for Dataset2, just change the value of batch sd. and error sd.

```
#simulate dataset 1
set.seed(1237) # R
g <- 30     # number of batches
r <- 5      # number of replications per batch
mu.grand <- 1000   # the grand mean
sd.Bat <- 25       # the batch stand deviation
sd.Err <- 15       # the error stand deviation
Y <- matrix(0, nrow=r, ncol=g)
mu <- numeric(g)
for (i in 1:g)
 {mu[i] <- rnorm(1, mu.grand, sd.Bat)
   for (j in 1:r)
      {Y[j,i] <- rnorm(1, mu[i], sd.Err)
      }
  }
x <- round(as.vector(Y))
Bat <- as.factor(rep(1:g, each = r))
FRAM <- data.frame(x, Bat)
X <- round(t(Y))

# MLE Estimate
""# R Montana
require(nlme)
f.fit <- lme(x ~ 1, random = ~1|Bat, method="ML")
f.fit
intervals(f.fit)
```

☆ Output

Fig.6. Maximum likelihood estimates for Dataset 1.

```
> ""# R Montana
[1] ""
> require(nlme)
Loading required package: nlme
[1] TRUE
> f.fit <- lme(x ~ 1, random = ~1|Bat, method="ML")
> f.fit
Linear mixed-effects model fit by maximum likelihood
  Data: NULL
  Log-likelihood: -654.0247
  Fixed: x ~ 1
(Intercept)
   1003.253

Random effects:
 Formula: ~1 | Bat
        (Intercept) Residual
StdDev:    23.81335 14.49575

Number of Observations: 150
Number of Groups: 30
> intervals(f.fit)
Approximate 95% confidence intervals

 Fixed effects:
               lower     est.    upper
(Intercept) 994.332 1003.253 1012.175
attr(,"label")
[1] "Fixed effects:"

 Random Effects:
  Level: Bat
                    lower     est.    upper
sd((Intercept)) 18.14331 23.81335 31.25536

 Within-group standard error:
   lower     est.    upper
12.77308 14.49575 16.45074
>
```

Classroom Simulation: Understanding One-Way Random-Effect ANOVA
Eric A. Suess, Bruce E. Trumbo, and Yun Jiang, CSU, East Bay
Poster: Section on Statistics Education, JSM 2005, Minneapolis, MN

 # Maximum likelihood estimates for Dataset 2.

not shown in the paper

```
#simulate dataset 2
set.seed(12) # R
g <- 30     # number of batches
r <- 5      # number of replications per batch
mu.grand <- 1000    # the grand mean
sd.Bat <- 1        # the batch stand deviation
sd.Err <- 25        # the error stand deviation
Y <- matrix(0, nrow=r, ncol=g)
mu <- numeric(g)
for (i in 1:g)
 {mu[i] <- rnorm(1, mu.grand, sd.Bat)
   for (j in 1:r)
     {Y[j,i] <- rnorm(1, mu[i], sd.Err)
     }
  }
x <- round(as.vector(Y))
Bat <- as.factor(rep(1:g, each = r))
FRAM <- data.frame(x, Bat)
X <- round(t(Y))

# MLE Estimate
""# R Montana
require(nlme)
f.fit <- lme(x ~ 1, random = ~1|Bat, method="ML")
f.fit
intervals(f.fit)
```

☆ **Output**

Maximum likelihood estimates for Dataset 2.

```
> ""# R Montana
[1] ""
> require(nlme)
[1] TRUE
> f.fit <- lme(x ~ 1, random = ~1|Bat, method="ML")
> f.fit
Linear mixed-effects model fit by maximum likelihood
  Data: NULL
  Log-likelihood: -684.054
  Fixed: x ~ 1
(Intercept)
   1000.033

Random effects:
 Formula: ~1 | Bat
        (Intercept) Residual
StdDev:   0.5599993 23.12995

Number of Observations: 150
Number of Groups: 30
> intervals(f.fit)
Error in intervals.lme(f.fit) : Cannot get confidence intervals on
var-cov components: Non-positive definite approximate
variance-covariance
>
```