

HW3_solutions

Dustin Pluta

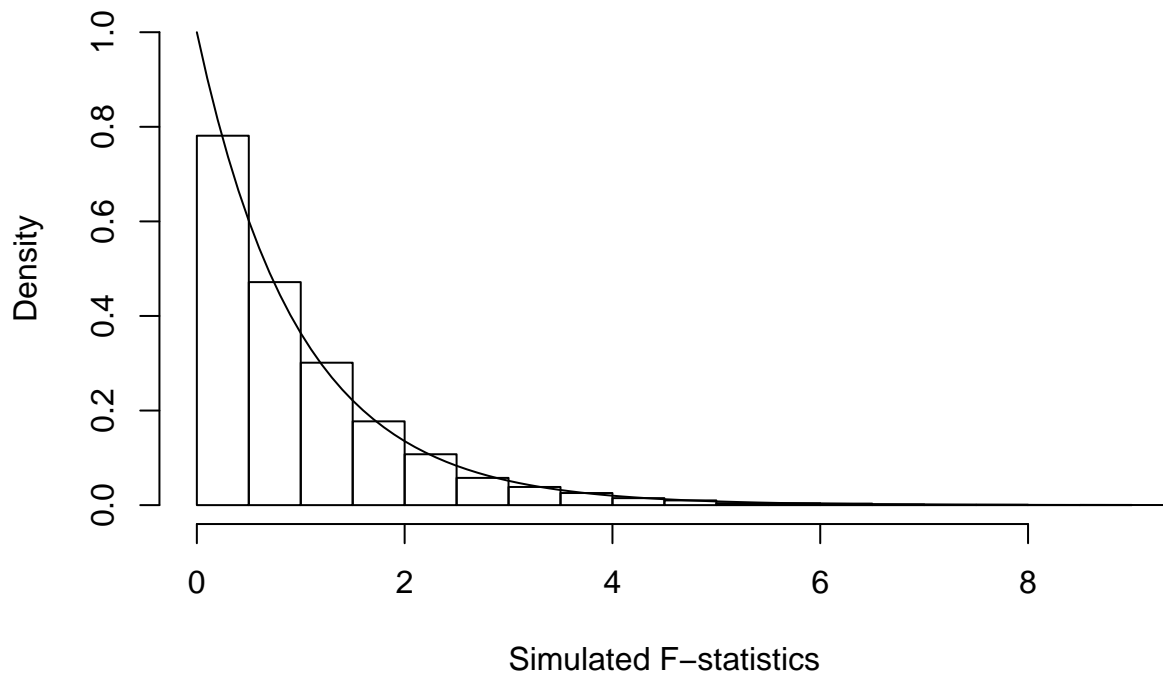
April 25, 2019

Problem 1

b (i)

```
#####  
##### Calculate the observed type I error rate  
#####  
  
n_reject <- sum(result[, 2] < .05 )  
sim_type1_error <- n_reject / n_sims  
sim_type1_error  
  
## [1] 0.0456
```

Histogram of result[, 1]



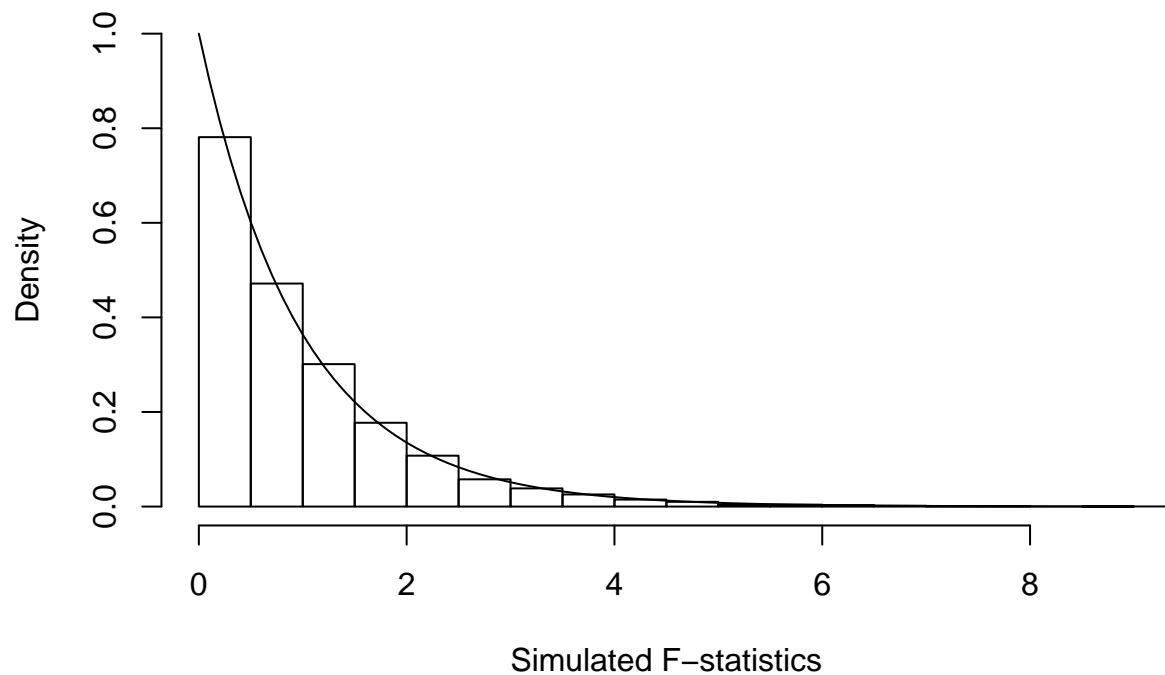
b (ii)

```
#####  
##### Calculate the observed type I error rate  
#####
```

```
n_reject <- sum(result[, 2] < .05 )
sim_type1_error <- n_reject / n_sims
sim_type1_error
```

```
## [1] 0.0456
```

Histogram of result[, 1]



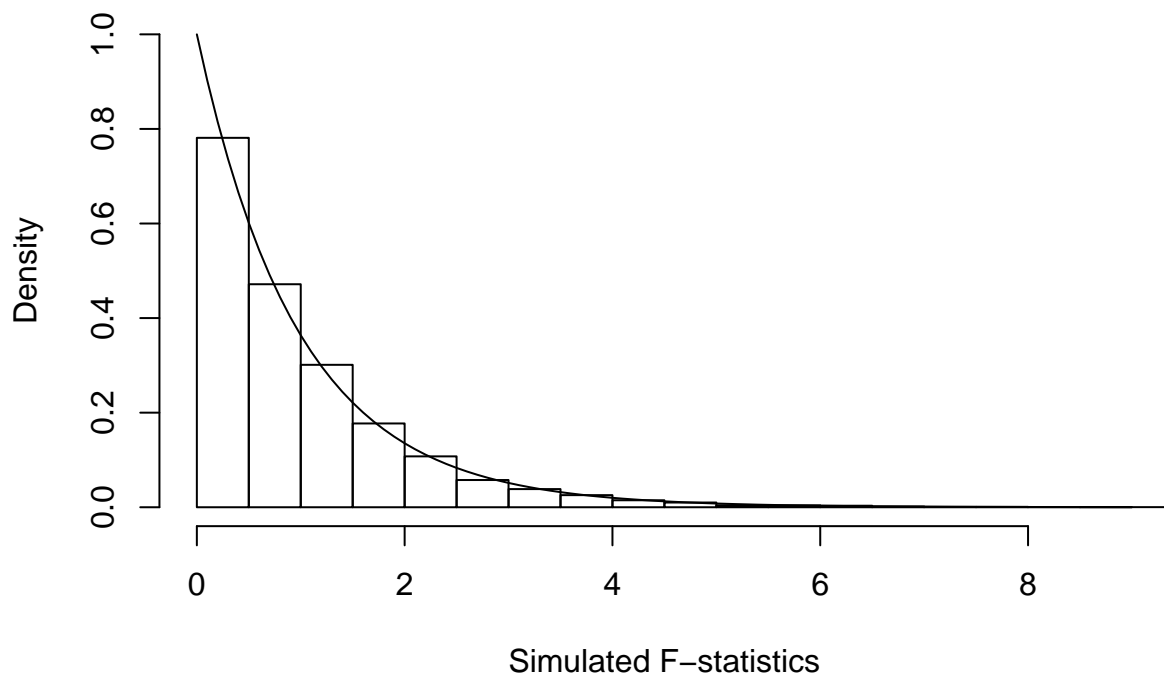
b (iii)

```
#####
##### Calculate the observed type I error rate
#####
```

```
n_reject <- sum(result[, 2] < .05 )
sim_type1_error <- n_reject / n_sims
sim_type1_error
```

```
## [1] 0.0456
```

Histogram of result[, 1]



Problem 2

(a)

Source	SS	df	MS	F
Treatment	4.66	2	2.33	1.674
Error	197.64	142	1.392	
Total	202.3	144		

(b)

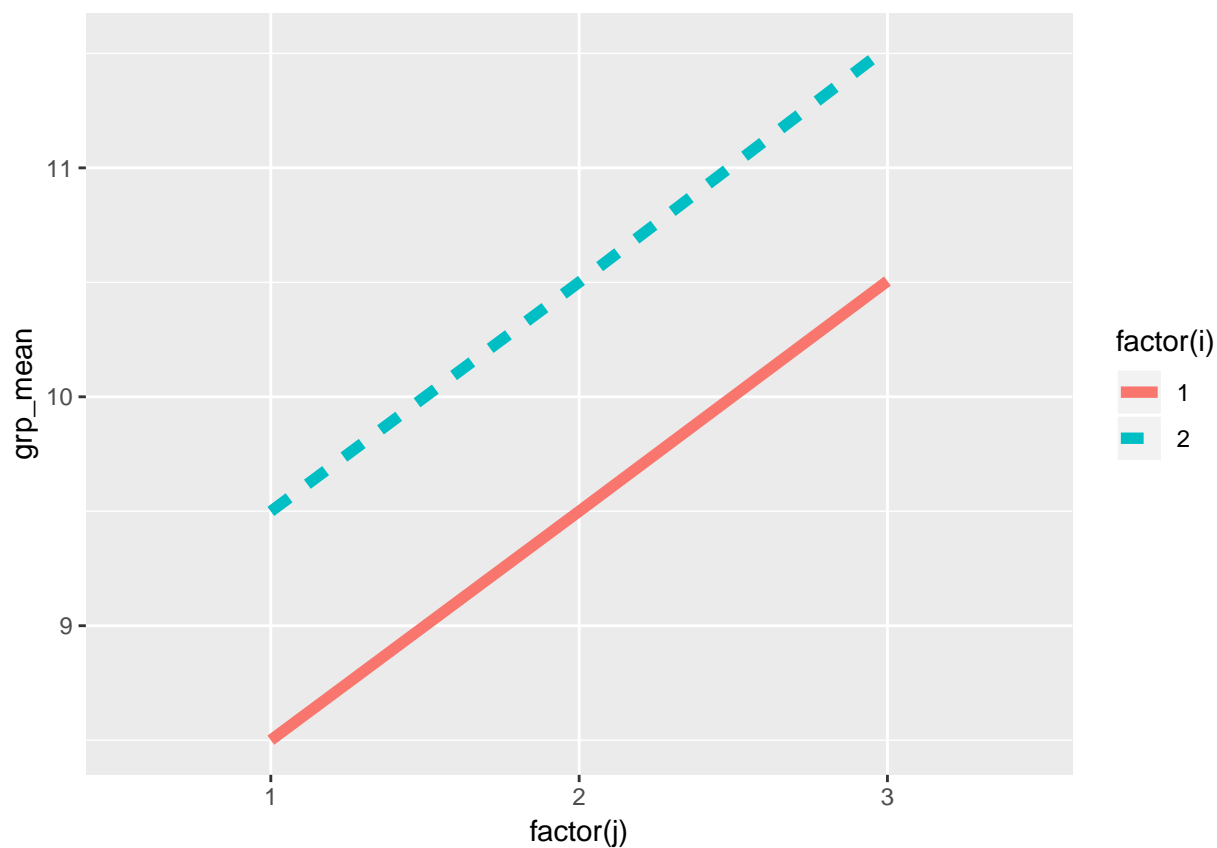
```
## [1] 0.1911742
```

Problem 3

(a) & (b)

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

The true interaction plot consists of two parallel lines.



(c)

The true deltas are

$$\delta = \begin{pmatrix} -1 & 2 & -1 \\ 1 & -2 & 1 \end{pmatrix}.$$

```
####
K <- 10000
mu <- 1
alpha <- c(-0.5, 0.5)
beta <- c(-1, 0, 1)
```

```

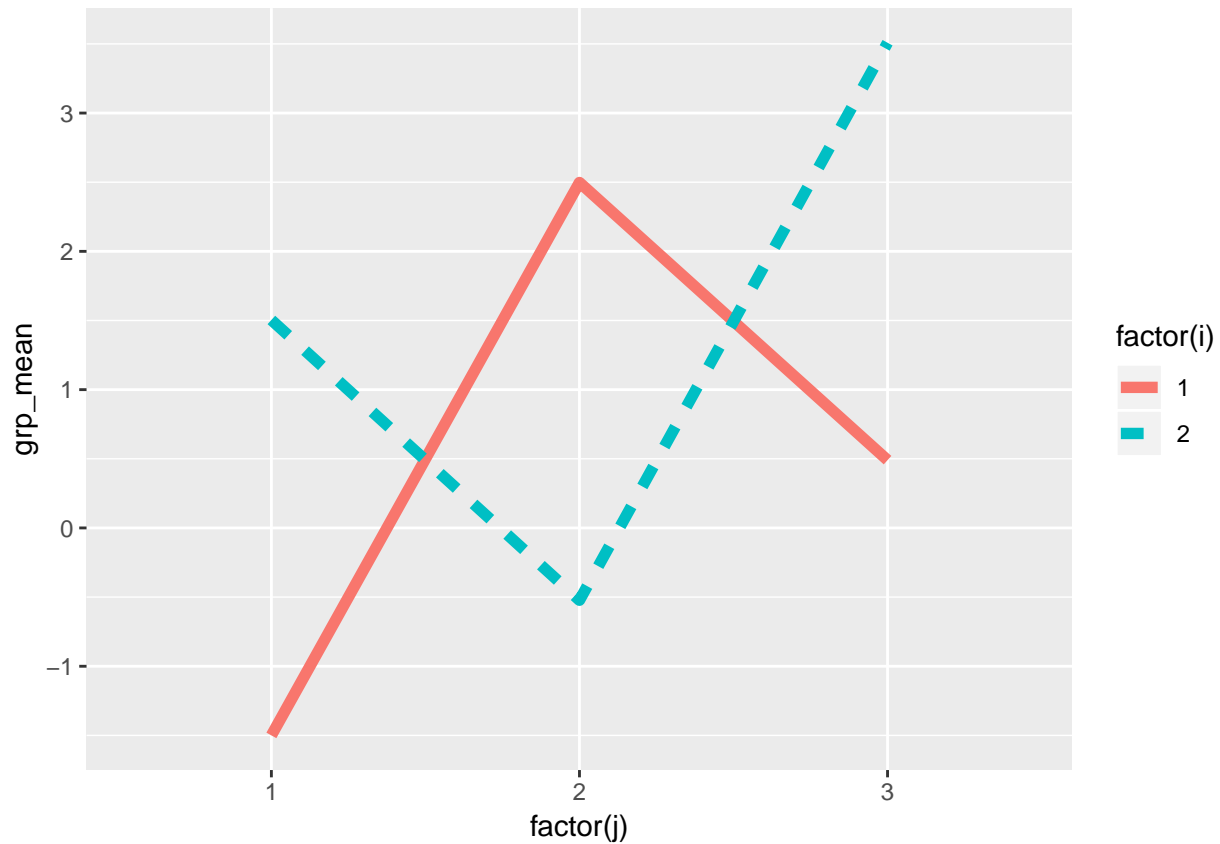
delta <- matrix(c(-1, 1, 2, -2, -1, 1), nrow = length(alpha), ncol = length(beta))
sigma <- 1

simulate_data <- function(mu, alpha, beta, delta, K, sigma) {
  I <- length(alpha)
  J <- length(beta)
  dat <- as.data.frame(expand.grid(i = 1:I, j = 1:J, k = 1:K, Y = NA))
  for (obs in 1:nrow(dat)) {
    dat$Y[obs] <- with(dat, mu + alpha[i[obs]] + beta[j[obs]] + delta[i[obs], j[obs]] + rnorm(1, 0, sigma))
  }
  return(dat)
}

dat <- simulate_data(mu, alpha, beta, delta, K, sigma)
dat_grp_means <- dat %>% group_by(i, j) %>% summarize(grp_mean = mean(Y))

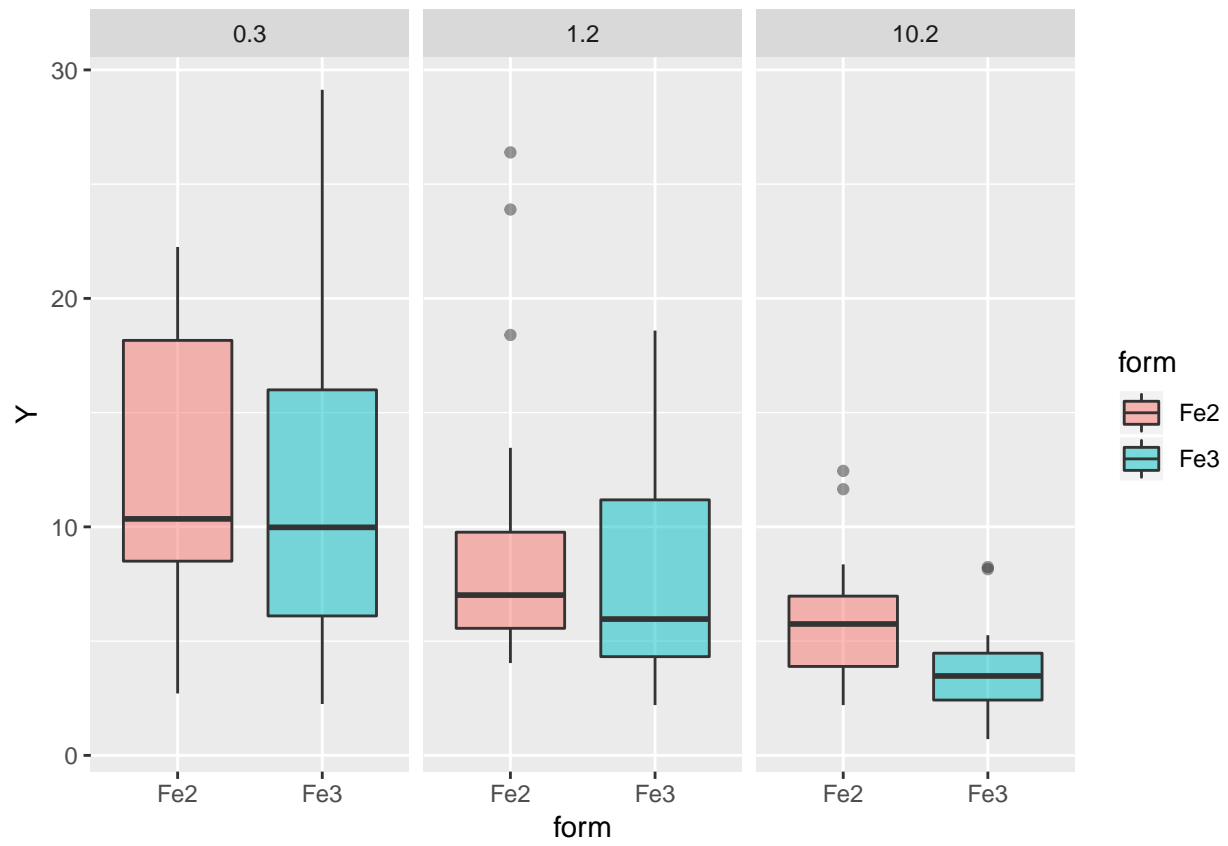
ggplot(dat_grp_means) +
  geom_line(aes(x = factor(j), y = grp_mean, group = factor(i), linetype = factor(i), color = factor(i)))

```



Problem 4

(a)



(b)

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
form	1	62.25926	62.25926	1.67921	0.1978441
Residuals	106	3930.11200	37.07653	NA	NA

(c)

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
factor(dose)	2	983.6176	491.8088	17.16323	4e-07
Residuals	105	3008.7537	28.6548	NA	NA

(d)

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
form	1	62.259259	62.259259	2.1613357	0.1445996
factor(dose)	2	983.617580	491.808790	17.0731859	0.0000004
form:factor(dose)	2	8.290813	4.145406	0.1439082	0.8661428
Residuals	102	2938.203611	28.805918	NA	NA

Problem 5

(a) See class notes.

(b)

Yes, it is possible to have interactions with no main effects, although this requires a specific form of symmetry in the interaction plot due to the constraints $\sum_i \delta_{ij} = \sum_j \delta_{ij} = 0$. For instance,

```
####
K <- 10000
mu <- 1
alpha <- c(0, 0)
beta <- c(0, 0, 0)
delta <- matrix(c(-1, 1, 2, -2, -1, 1), nrow = length(alpha), ncol = length(beta))
sigma <- 1

simulate_data <- function(mu, alpha, beta, delta, K, sigma) {
  I <- length(alpha)
  J <- length(beta)
  dat <- as.data.frame(expand.grid(i = 1:I, j = 1:J, k = 1:K, Y = NA))
  for (obs in 1:nrow(dat)) {
    dat$Y[obs] <- with(dat, mu + alpha[i[obs]] + beta[j[obs]] + delta[i[obs], j[obs]] + rnorm(1, 0, sigma))
  }
  return(dat)
}

dat <- simulate_data(mu, alpha, beta, delta, K, sigma)
dat_grp_means <- dat %>% group_by(i, j) %>% summarize(grp_mean = mean(Y))

ggplot(dat_grp_means) +
  geom_line(aes(x = factor(j), y = grp_mean, group = factor(i), linetype = factor(i), color = factor(i)))
```

