Mixed Models with Cupcakes

Pseudoreplication

Consider the simplification of a cupcake dataset that contains measurements of cupcake heights from a designed experiment and assesses differences in height for 375 and 325 degrees.

```
cupcakes_q1 <- read_csv('CupcakeHeights.csv') %>%
  filter(Temp.F %in% c(325, 375)) %>%
  dplyr::select(Temp.F, Height.cm)
```

- **Temp.F:** temperature of oven
- **Height.cm** measured height of cupcake

```
summary(lm(Height.cm ~ factor(Temp.F), data = cupcakes_q1))
```

```
##
## Call:
## lm(formula = Height.cm ~ factor(Temp.F), data = cupcakes_q1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31667 -0.12083 -0.01667  0.08333  0.28333
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        4.33333    0.03716 116.619  < 2e-16 ***
## factor(Temp.F)375  0.18333    0.05255   3.489  0.00136 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1576 on 34 degrees of freedom
## Multiple R-squared:  0.2636, Adjusted R-squared:  0.242
## F-statistic: 12.17 on 1 and 34 DF,  p-value: 0.001362
```

This type of analysis results in pseudoreplication. Note the standard error on the contrast term for comparison.

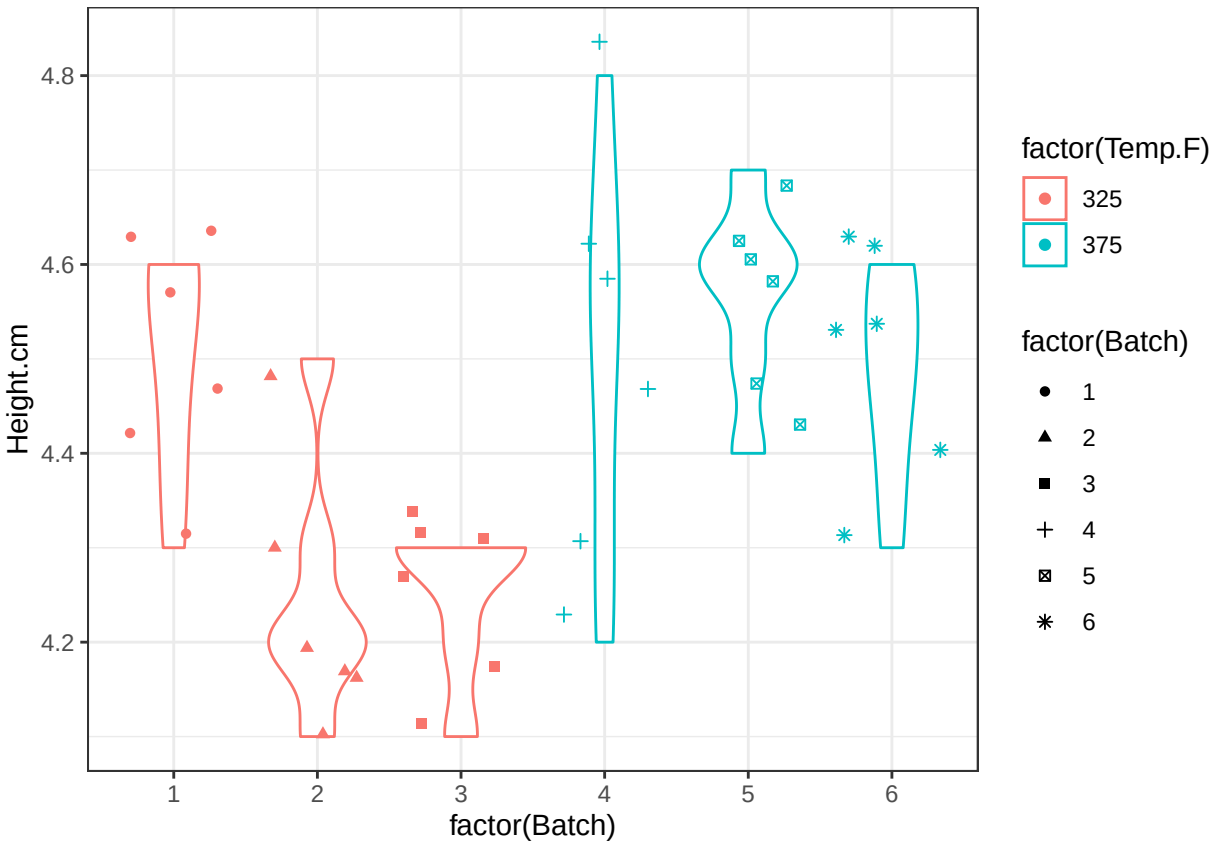
Averaging Models

This complete cupcake dataset includes information about the batch number of cupcakes that corresponds to pan the cupcakes were baked in. Cupcakes from the same batch are cooked in the oven together and would likely be correlated. Failing to account for this results in pseudo-replication.

```
cupcakes_q2 <- read_csv('CupcakeHeights.csv') %>%
  filter(Temp.F %in% c(325, 375)) %>%
  mutate(Batch = case_when(Batch.Number == 1 & Temp.F == 325 ~ 1,
                           Batch.Number == 2 & Temp.F == 325 ~ 2,
                           Batch.Number == 3 & Temp.F == 325 ~ 3,
                           Batch.Number == 1 & Temp.F == 375 ~ 4,
                           Batch.Number == 2 & Temp.F == 375 ~ 5,
                           Batch.Number == 3 & Temp.F == 375 ~ 6)) %>%
  dplyr::select(Temp.F, Batch, Height.cm)
```

- **Temp.F:** temperature of oven
- **Batch** identifier for batch of cupcakes (cooked in same pan)
- **Height.cm** measured height of cupcake

```
cupcakes_q2 %>%
  ggplot(aes(x = factor(Batch), y = Height.cm, color = factor(Temp.F), shape = factor(Batch))) +
  geom_violin() + theme_bw() + geom_jitter()
```



Now consider taking the average height of all cupcakes in a batch and analyzing that data.

```
cupcakes_q2 %>% group_by(Batch, Temp.F) %>%
  summarize(Height.cm = mean(Height.cm), .groups = 'drop') %>%
   lm(Height.cm ~  factor(Temp.F), data = .) %>% summary()
```

```
##
## Call:
## lm(formula = Height.cm ~ factor(Temp.F), data = .)
##
## Residuals:
##        1        2        3        4        5        6
##  0.16667 -0.08333 -0.08333 -0.01667  0.05000 -0.03333
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        4.33333    0.06161  70.330 2.45e-07 ***
## factor(Temp.F)375  0.18333    0.08714   2.104    0.103
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1067 on 4 degrees of freedom
## Multiple R-squared:  0.5253, Adjusted R-squared:  0.4067
## F-statistic: 4.427 on 1 and 4 DF,  p-value: 0.1032
```

Note the difference in the standard error

Hierarchical Models

Finally we can formally account for this by fitting a hierarchical model using `lmer`. Compare the standard error of the hierarchical model with the

```
cupcakes_q2 %>%
  lmer(Height.cm ~  1 + factor(Temp.F)+ (1 |Batch), data = .) %>%
  summary()
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Height.cm ~ 1 + factor(Temp.F) + (1 | Batch)
##    Data: .
##
## REML criterion at convergence: -27.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.2069 -0.5305  0.1939  0.6907  2.1396
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Batch    (Intercept) 0.008213 0.09063
##  Residual             0.019056 0.13804
## Number of obs: 36, groups:  Batch, 6
##
## Fixed effects:
##                   Estimate Std. Error t value
## (Intercept)        4.33333    0.06161  70.330
## factor(Temp.F)375  0.18333    0.08714   2.104
##
## Correlation of Fixed Effects:
##             (Intr)
## fct(T.F)375 -0.707
```

```
cupcakes_q2 %>% lm(Height.cm ~factor(Temp.F) , data = .) %>% summary()
```

```
##
## Call:
## lm(formula = Height.cm ~ factor(Temp.F), data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31667 -0.12083 -0.01667  0.08333  0.28333
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        4.33333    0.03716 116.619  < 2e-16 ***
## factor(Temp.F)375  0.18333    0.05255   3.489  0.00136 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1576 on 34 degrees of freedom
## Multiple R-squared:  0.2636, Adjusted R-squared:  0.242
## F-statistic: 12.17 on 1 and 34 DF,  p-value: 0.001362
```

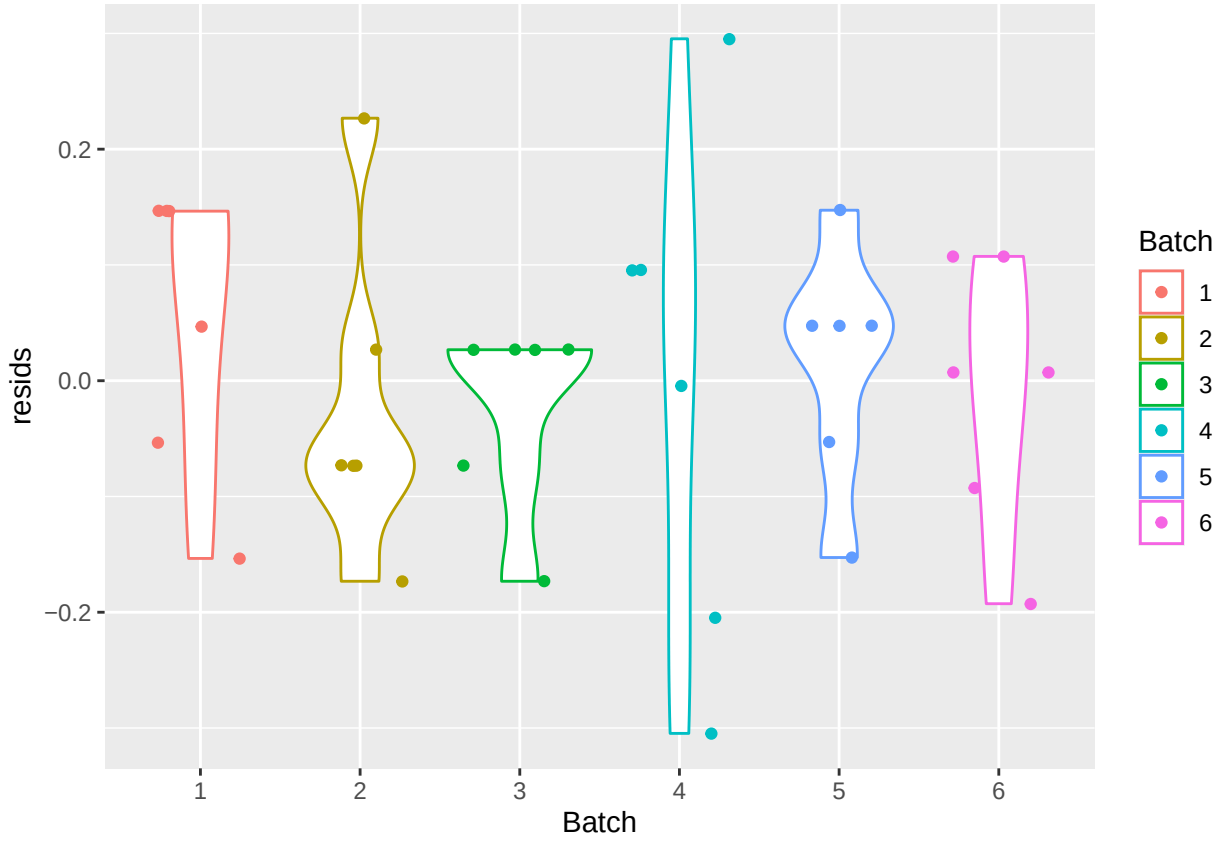
```r
cupcakes_q2 %>% group_by(Batch, Temp.F) %>%
  summarize(Height.cm = mean(Height.cm), .groups = 'drop') %>%
   lm(Height.cm ~  factor(Temp.F), data = .) %>% summary()
```

```
##
## Call:
## lm(formula = Height.cm ~ factor(Temp.F), data = .)
##
## Residuals:
##        1        2        3        4        5        6
##  0.16667 -0.08333 -0.08333 -0.01667  0.05000 -0.03333
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        4.33333    0.06161  70.330 2.45e-07 ***
## factor(Temp.F)375  0.18333    0.08714   2.104    0.103
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1067 on 4 degrees of freedom
## Multiple R-squared:  0.5253, Adjusted R-squared:  0.4067
## F-statistic: 4.427 on 1 and 4 DF,  p-value: 0.1032
```

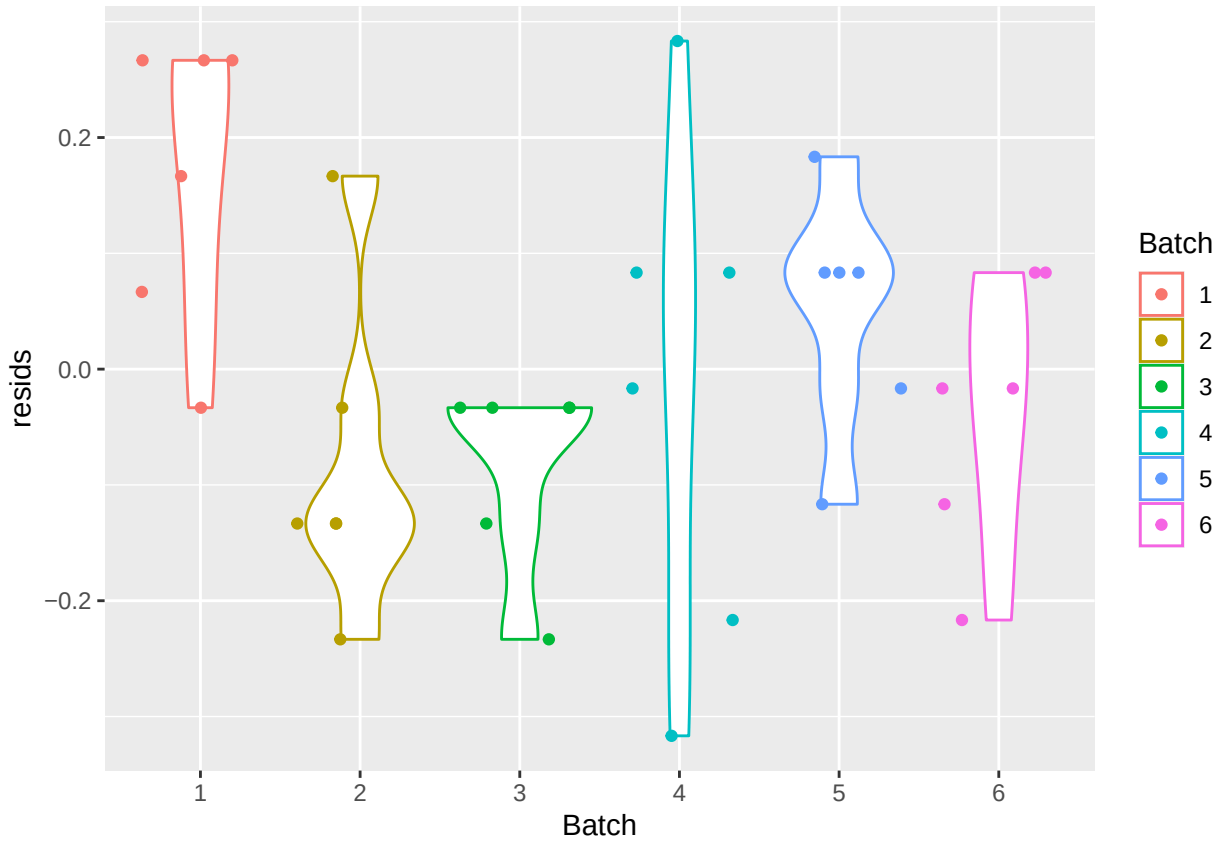
Model Assumptions

Now we will re-examine the residuals from the models fit with the hierarchical model for Batch) and the pseudoreplicated linear model.

```
tibble(resids = cupcakes_q2 %>%
  lmer(Height.cm ~  1 + factor(Temp.F)+ (1  |Batch), data = .) %>%
  residuals(),
  Batch = factor(cupcakes_q2$Batch)) %>% ggplot(aes(y = resids, x = Batch, color = Batch)) +
  geom_violin() + geom_jitter()
```




```
tibble(resids = cupcakes_q2 %>%
  lm(Height.cm ~  factor(Temp.F), data = .) %>%
  residuals(),
  Batch = factor(cupcakes_q2$Batch)) %>% ggplot(aes(y = resids, x = Batch, color = Batch))  +
  geom_violin() + geom_jitter()
```


resids
0.2
0.0
−0.2
1
2
3
4
5
6
Batch
Batch
1
2
3
4
5
6