# Music Lyrics Pilot Data

*Gurdit Chahal, Adam Letcher, Carlos Castro, Noah Randolph*

*8/4/2018*

### Results from Pilot Testing of Experiment Site

```
dt_raw <- fread('pilot_results.csv')
```

### Cleanup Columns

```
dt <- dt_raw[, .(hear_song = as.integer(as.character(factor(Check1, levels = c('A', 'B'),
                                                            labels = c(1, 0)))),
                 piano_playing = as.integer(as.character(factor(Check2,
                                                                levels = c('A', 'B'),
                                                                labels = c(1, 0)))),
                 q1 = as.factor(Q1),
                 q2 = as.factor(Q2),
                 q3 = as.factor(Q3),
                 q4 = as.factor(Q4),
                 q5 = as.factor(Q5),
                 age = factor(Q6, levels = c('A', 'B', 'C', 'D', 'E',
                                             'F', 'G', 'H', 'I', 'J'),
                              labels = c('<12', '12-17', '18-24', '25-34', '35-44',
                                         '45-54', '55-64', '65-74', '>75', 'decline')),
                 gender = factor(Q7, levels = c('A', 'B', 'C', 'D'),
                                 labels = c('male', 'female', 'other', 'decline')),
                 own_dog = as.integer(as.character(factor(Q8, levels = c('A', 'B'),
                                                          labels = c(1, 0)))),
                 education = factor(Q9, levels = c('A', 'B', 'C', 'D', 'E',
                                                   'F', 'G', 'H', 'I', 'J'),
                                    labels = c('none', '8th grade', 'some high school',
                                               'high school completed', 'some college',
                                               'vocational', 'associates', 'bachelors',
                                               'masters', 'doctorate')),
                 occupation = Q10,
                 native_english = as.integer(as.character(factor(Q11,
                                                                 levels = c('A', 'B'),
                                                                 labels = c(1, 0)))),
                 heard_lyrics = as.integer(as.character(factor(Q12, levels = c('A', 'B'),
                                                               labels = c(1, 0)))),
                 is_turk = as.integer(as.character(factor(isTurk,
                                                          levels = c('true', 'null'),
                                                          labels = c(1, 0)))),
                 time,
                 correct_count = correctCount,
                 assigned_lyrics = as.integer(as.character(factor(lyrics, levels = c('true', 'false'),
                                                                  labels = c(1, 0)))))]
```

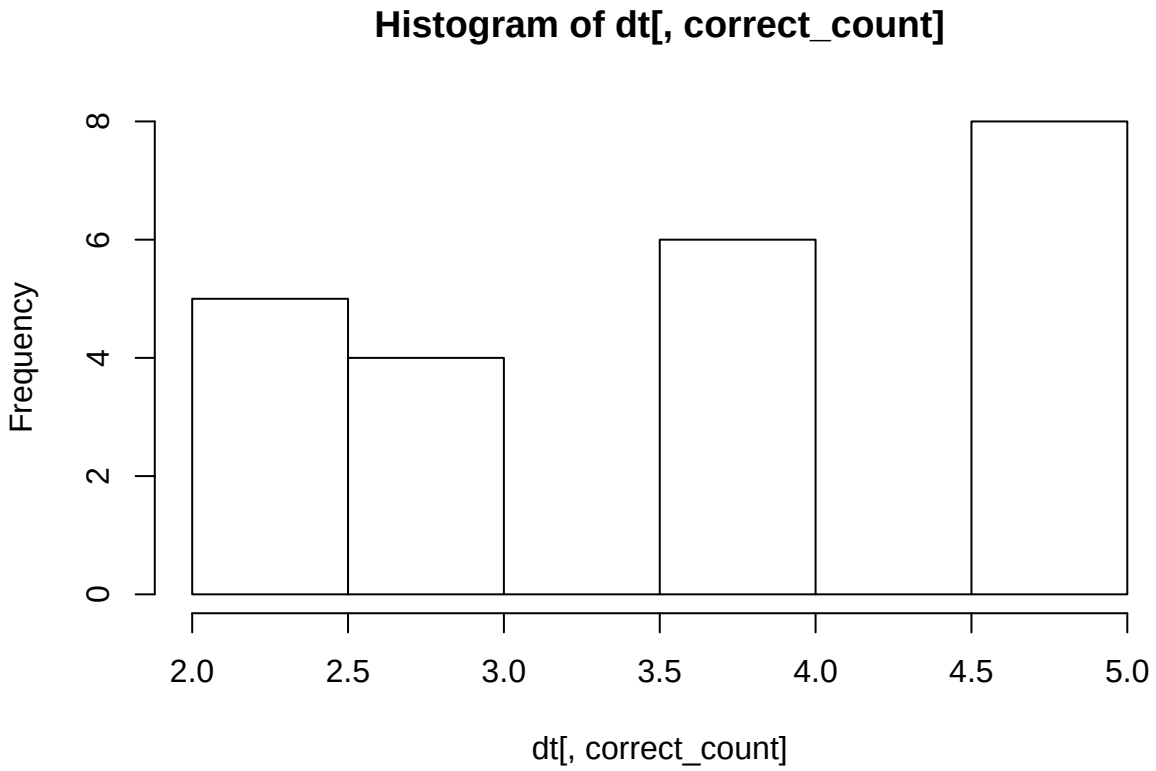
## EDA

```
stargazer(dt, header=FALSE, type='latex')
```

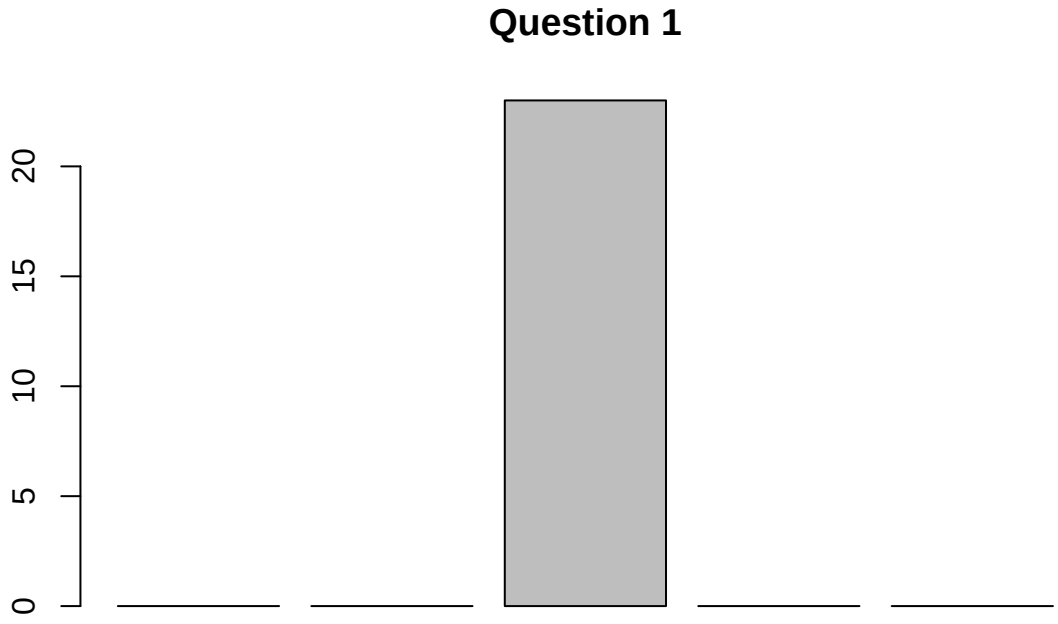
Table 1:

| Statistic | N | Mean | St. Dev. | Min | Max |
|---|---|---|---|---|---|
| hear_song | 23 | 1.000 | 0.000 | 1 | 1 |
| piano_playing | 23 | 1.000 | 0.000 | 1 | 1 |
| own_dog | 23 | 0.304 | 0.470 | 0 | 1 |
| native_english | 22 | 0.727 | 0.456 | 0 | 1 |
| heard_lyrics | 22 | 0.545 | 0.510 | 0 | 1 |
| is_turk | 23 | 0.217 | 0.422 | 0 | 1 |
| time | 23 | −590.696 | 310.507 | −1,478 | −182 |
| correct_count | 23 | 3.739 | 1.176 | 2 | 5 |
| assigned_lyrics | 23 | 0.435 | 0.507 | 0 | 1 |

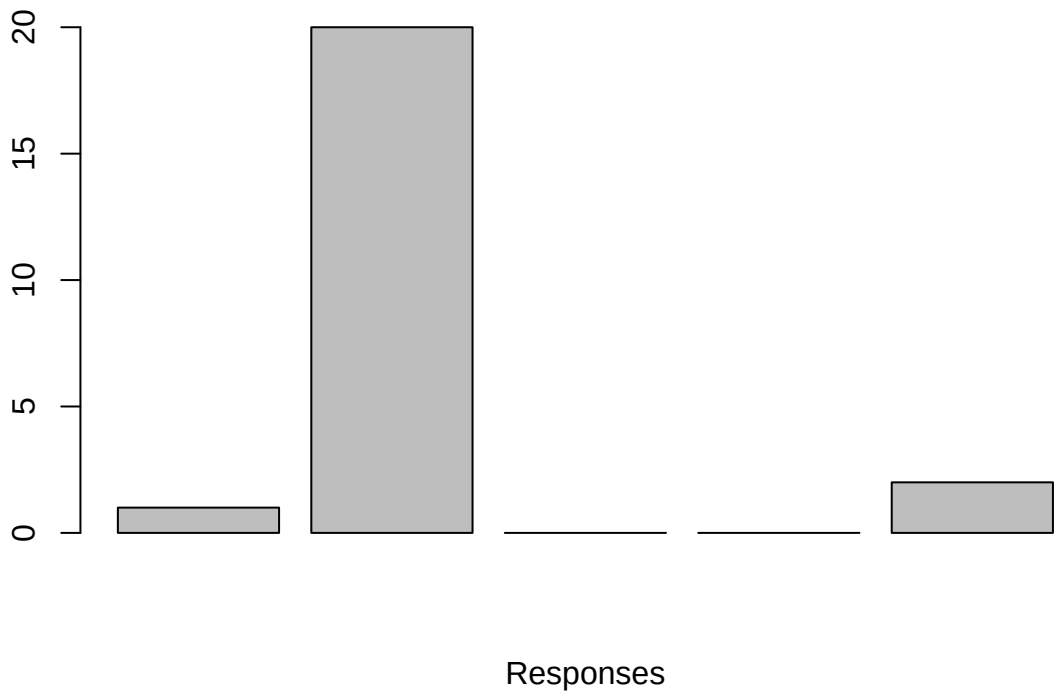
```
hist(dt[, correct_count])
```



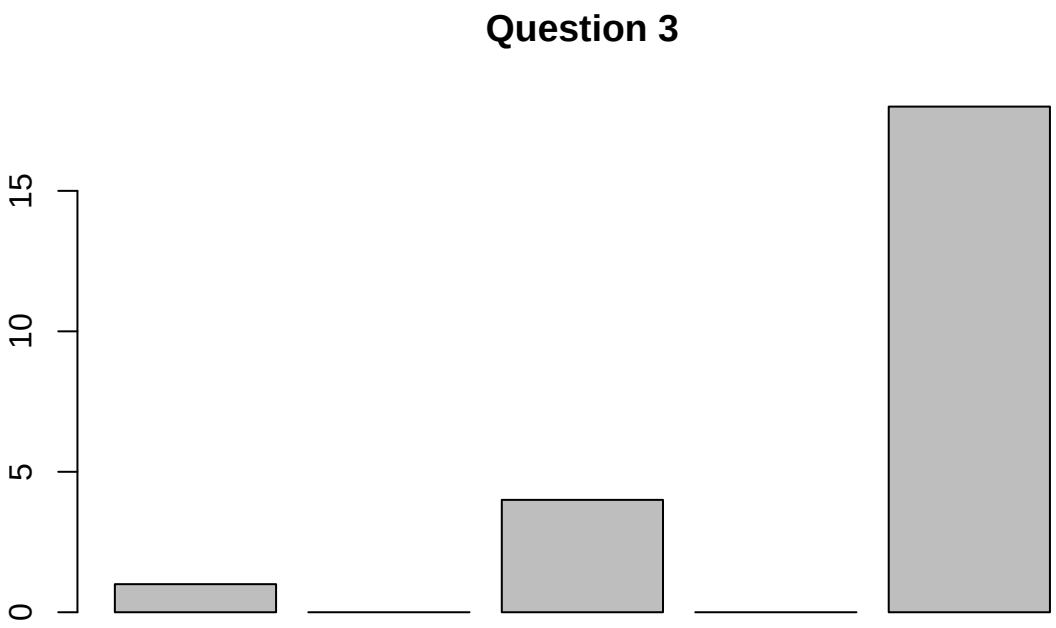

```
q1_counts <- c(sum(dt[,q1=='A']), sum(dt[,q1=='B']), sum(dt[,q1=='C']),
               sum(dt[,q1=='D']), sum(dt[,q1=='E']))
barplot(q1_counts, main="Question 1",
    xlab="Responses")
```



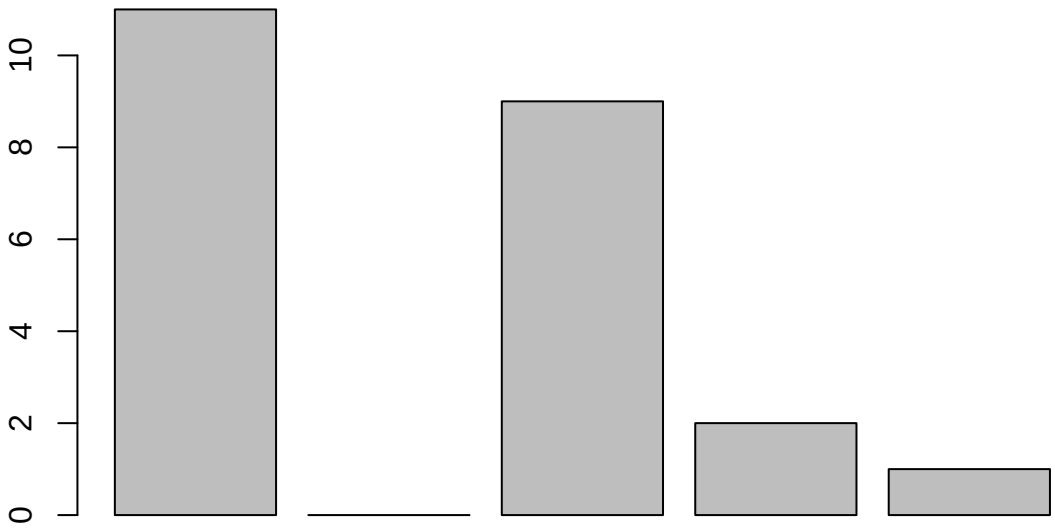

```
q2_counts <- c(sum(dt[,q2=='A']), sum(dt[,q2=='B']), sum(dt[,q2=='C']),
               sum(dt[,q2=='D']), sum(dt[,q2=='E']))
barplot(q2_counts, main="Question 2",
    xlab="Responses")
```



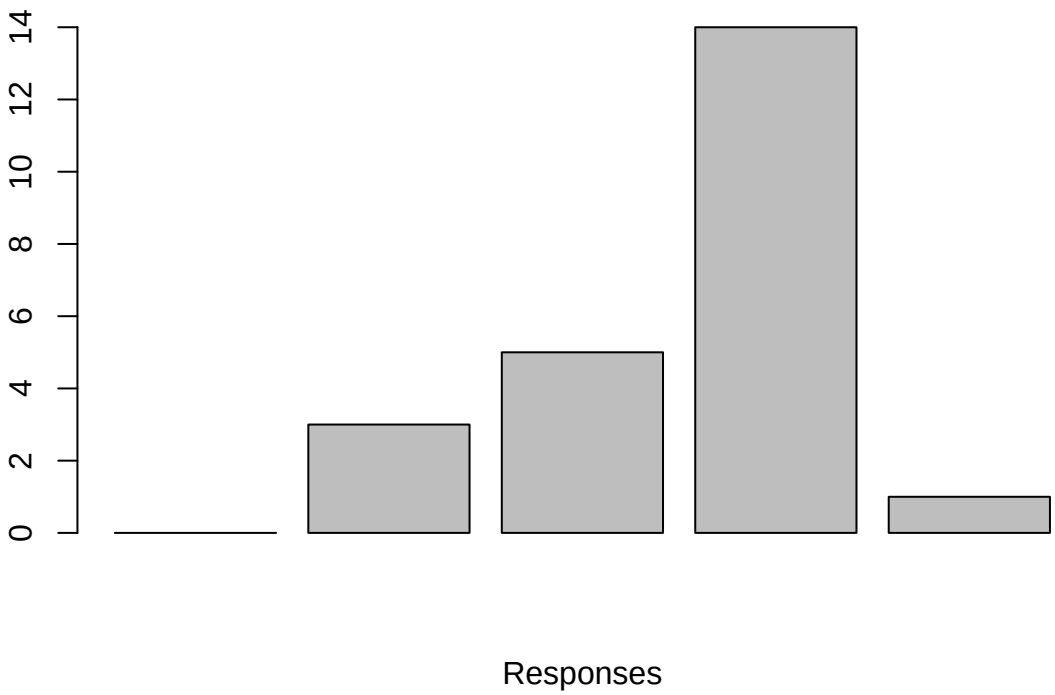
```
q3_counts <- c(sum(dt[,q3=='A']), sum(dt[,q3=='B']), sum(dt[,q3=='C']),
               sum(dt[,q3=='D']), sum(dt[,q3=='E']))
barplot(q3_counts, main="Question 3",
    xlab="Responses")
```




```
q4_counts <- c(sum(dt[,q4=='A']), sum(dt[,q4=='B']), sum(dt[,q4=='C']),
               sum(dt[,q4=='D']), sum(dt[,q4=='E']))
barplot(q4_counts, main="Question 4",
    xlab="Responses")
```




```
q5_counts <- c(sum(dt[,q5=='A']), sum(dt[,q5=='B']), sum(dt[,q5=='C']),
               sum(dt[,q5=='D']), sum(dt[,q5=='E']))
barplot(q5_counts, main="Question 5",
    xlab="Responses")
```



```r
summary(dt)
```

```
##    hear_song piano_playing q1     q2     q3     q4     q5          age
##  Min.   :1   Min.   :1     C:23   A: 1   A: 1   A:11   B: 3   25-34  :11
##  1st Qu.:1   1st Qu.:1            B:20   C: 4   C: 9   C: 5   18-24  : 6
##  Median :1   Median :1            E: 2   E:18   D: 2   D:14   35-44  : 2
##  Mean   :1   Mean   :1                           E: 1   E: 1   65-74  : 2
##  3rd Qu.:1   3rd Qu.:1                                         45-54  : 1
##  Max.   :1   Max.   :1                                         55-64  : 1
##                                                                (Other): 0
##      gender      own_dog                        education
##  male   :12   Min.   :0.0000   bachelors             :11
##  female :11   1st Qu.:0.0000   masters               : 7
##  other  : 0   Median :0.0000   associates            : 3
##  decline: 0   Mean   :0.3043   high school completed: 1
##               3rd Qu.:1.0000   doctorate             : 1
##               Max.   :1.0000   none                  : 0
##                                (Other)               : 0
##   occupation        native_english    heard_lyrics       is_turk
##  Length:23          Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  Class :character   1st Qu.:0.2500   1st Qu.:0.0000   1st Qu.:0.0000
##  Mode  :character   Median :1.0000   Median :1.0000   Median :0.0000
##                     Mean   :0.7273   Mean   :0.5455   Mean   :0.2174
##                     3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.0000
##                     Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##                     NA's   :1        NA's   :1
##       time         correct_count   assigned_lyrics
##  Min.   :-1478.0   Min.   :2.000   Min.   :0.0000
##  1st Qu.: -794.0   1st Qu.:3.000   1st Qu.:0.0000
##  Median : -506.0   Median :4.000   Median :0.0000
##  Mean   : -590.7   Mean   :3.739   Mean   :0.4348
##  3rd Qu.: -369.5   3rd Qu.:5.000   3rd Qu.:1.0000
##  Max.   : -182.0   Max.   :5.000   Max.   :1.0000
##
```

## Regression

```r
#Covariate Balance Check
# dt[ , t.test(as.numeric(age) ~ assigned_lyrics)]
# dt[ , chisq.test(as.numeric(education), assigned_lyrics)]
# dt[ , t.test(native_english ~ assigned_lyrics)]
# dt[ , t.test(is_turk~assigned_lyrics)]
```

```r
fit_pilot <- lm(correct_count ~ assigned_lyrics, dt)
summary(fit_pilot)
```

```
##
## Call:
## lm(formula = correct_count ~ assigned_lyrics, data = dt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9000 -0.9000  0.3846  1.1000  1.3846
```

```
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.6154     0.3314  10.910 4.12e-10 ***
## assigned_lyrics   0.2846     0.5025   0.566    0.577
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.195 on 21 degrees of freedom
## Multiple R-squared:  0.01504,    Adjusted R-squared:  -0.03186
## F-statistic: 0.3207 on 1 and 21 DF,  p-value: 0.5772
```

```
#fit_pilot_with_covariates <- lm(correct_count ~ assigned_lyrics + age + education + is_turk + time, dt
fit_pilot_with_covariates <- lm(correct_count ~ assigned_lyrics + age + education + is_turk, dt)
summary(fit_pilot_with_covariates)
```

```
##
## Call:
## lm(formula = correct_count ~ assigned_lyrics + age + education +
##     is_turk, data = dt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.17354 -0.70024  0.03762  0.63167  1.23786
##
## Coefficients: (1 not defined because of singularities)
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          2.31917    1.61897   1.433    0.178
## assigned_lyrics     -0.07524    0.64309  -0.117    0.909
## age25-34            -0.33617    0.74380  -0.452    0.659
## age35-44            -0.93689    1.13003  -0.829    0.423
## age45-54            -2.73058    1.88360  -1.450    0.173
## age55-64             0.68083    2.02872   0.336    0.743
## age65-74            -1.17354    1.11793  -1.050    0.315
## educationassociates  1.87136    1.62065   1.155    0.271
## educationbachelors   1.85437    1.57063   1.181    0.261
## educationmasters     2.65534    1.69505   1.567    0.143
## educationdoctorate       NA         NA      NA       NA
## is_turk             -0.24393    1.10158  -0.221    0.828
##
## Residual standard error: 1.223 on 12 degrees of freedom
## Multiple R-squared:  0.4107, Adjusted R-squared:  -0.08041
## F-statistic: 0.8363 on 10 and 12 DF,  p-value: 0.6061
```

## Power Calculation

Determine what the effect size should be OR use the effect size from the pilot experiment:

```
cohen.ES(test = 't', size = 'small')
```

```
##
##      Conventional effect size from Cohen (1982)
##
##            test = t
##            size = small
```

```
##     effect.size = 0.2
```

```
pwr.t.test(power = 0.8, d = 0.28, sig.level = 0.05, type = "two.sample", alternative = "greater")
```

```
##
##      Two-sample t test power calculation
##
##               n = 158.399
##               d = 0.28
##       sig.level = 0.05
##           power = 0.8
##     alternative = greater
##
## NOTE: n is number in *each* group
```