Data Challenge 2

Introduction

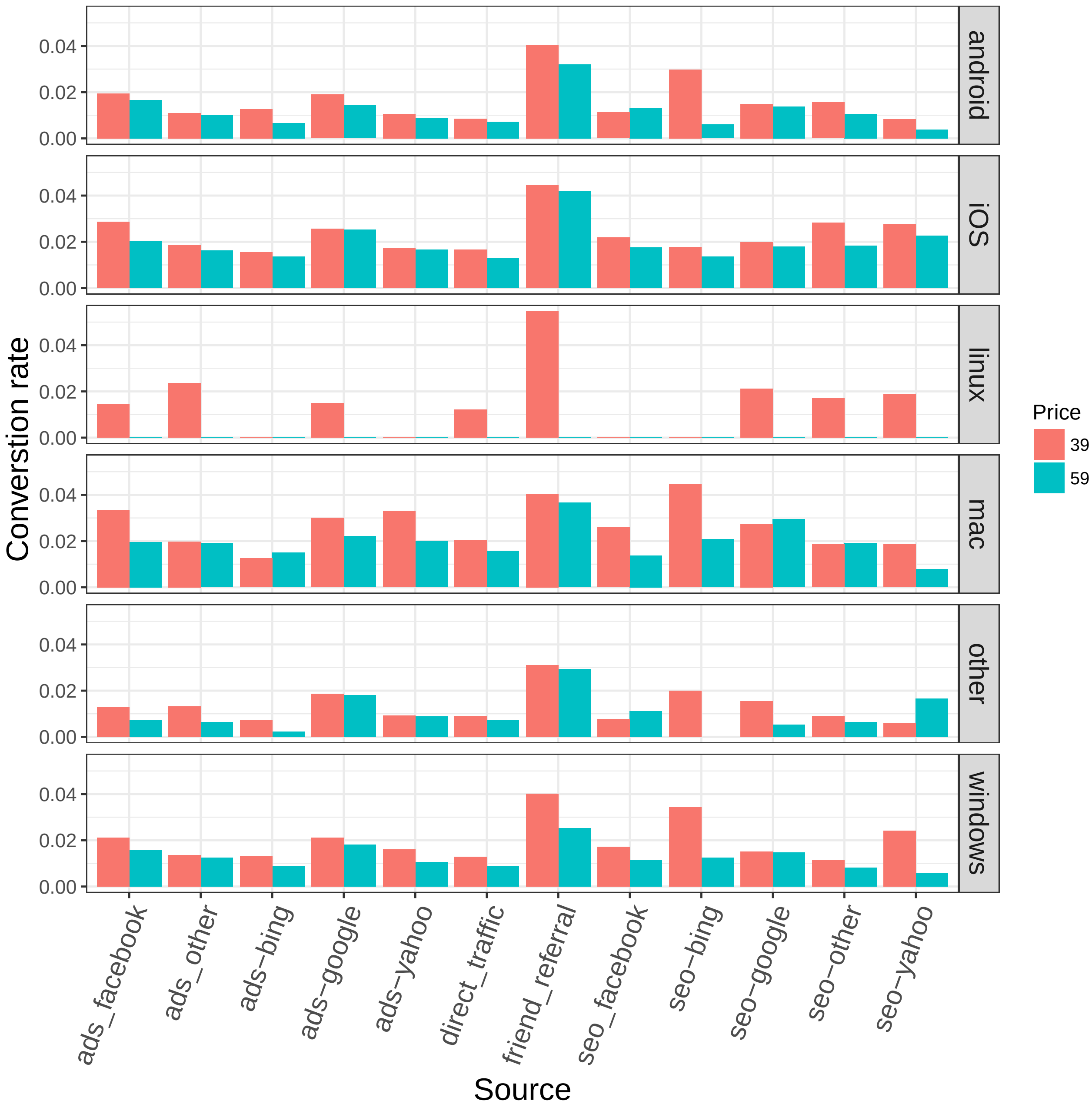
In this data challenge we were give results from an online sales price A/B test. The normal price for the product it $39 and a random sample of 33% of users were exposed to a new higher price of $59. We are given background information on users and information of their exposures to the product, including whether they made a purchase or not.

Let's start out by taking a look at the data

```
set.seed(101)
setwd('/Users/jalealsanjak/Documents/Research/Insight/interview_prep/data_challenge/Insight_data_challe
library(tidyverse)
library(lme4)
library(knitr)
conversion_data <- read_csv("Pricing_Test/test_results.csv")
user_data <- read_csv("Pricing_Test/user_table.csv")
all_data <- inner_join(conversion_data,user_data,by="user_id")
kable(head(all_data))
```

user_id	timestamp	source	device	operative_system	test	price	converted	city	coun
604839	2015-05-08 03:38:34	ads_facebook	mobile	iOS	0	39	0	Buffalo	USA
624057	2015-05-10 21:08:46	seo-google	mobile	android	0	39	0	Lakeville	USA
317970	2015-04-04 15:01:23	ads-bing	mobile	android	0	39	0	Parma	USA
685636	2015-05-07 07:26:01	direct_traffic	mobile	iOS	1	59	0	Fayetteville	USA
820854	2015-05-24 11:04:40	ads_facebook	web	mac	0	39	0	Fishers	USA
169971	2015-04-13 12:07:08	ads-google	mobile	iOS	0	39	0	New York	USA

```
conversion_rates <- all_data %>% group_by(price,source,operative_system) %>%
  summarise(conversion_rate = mean(converted),total = n(),converted=sum(converted))
p <- ggplot(conversion_rates, aes(x=source,y=conversion_rate,fill=as.factor(price))) + geom_bar(positio
  theme_bw() + facet_grid(operative_system ~ .) + xlab("Source") + ylab("Converstion rate")+ guides(fil
p <- p + theme(axis.text.x = element_text(angle = 70,hjust=1,size=14),
         axis.text.y = element_text(size=10),
         axis.title =element_text(size=16),
         strip.text.x= element_text(size=14),
         strip.text.y= element_text(size=14))
```



Modeling approach

It appears that there is a decent bit of variation amongst the different sources of referral to the site and the operating system. It appears that friend referrals have the highest converstion rate and that linux users are extremely prive sensitive (althought the sample size is low for linux).

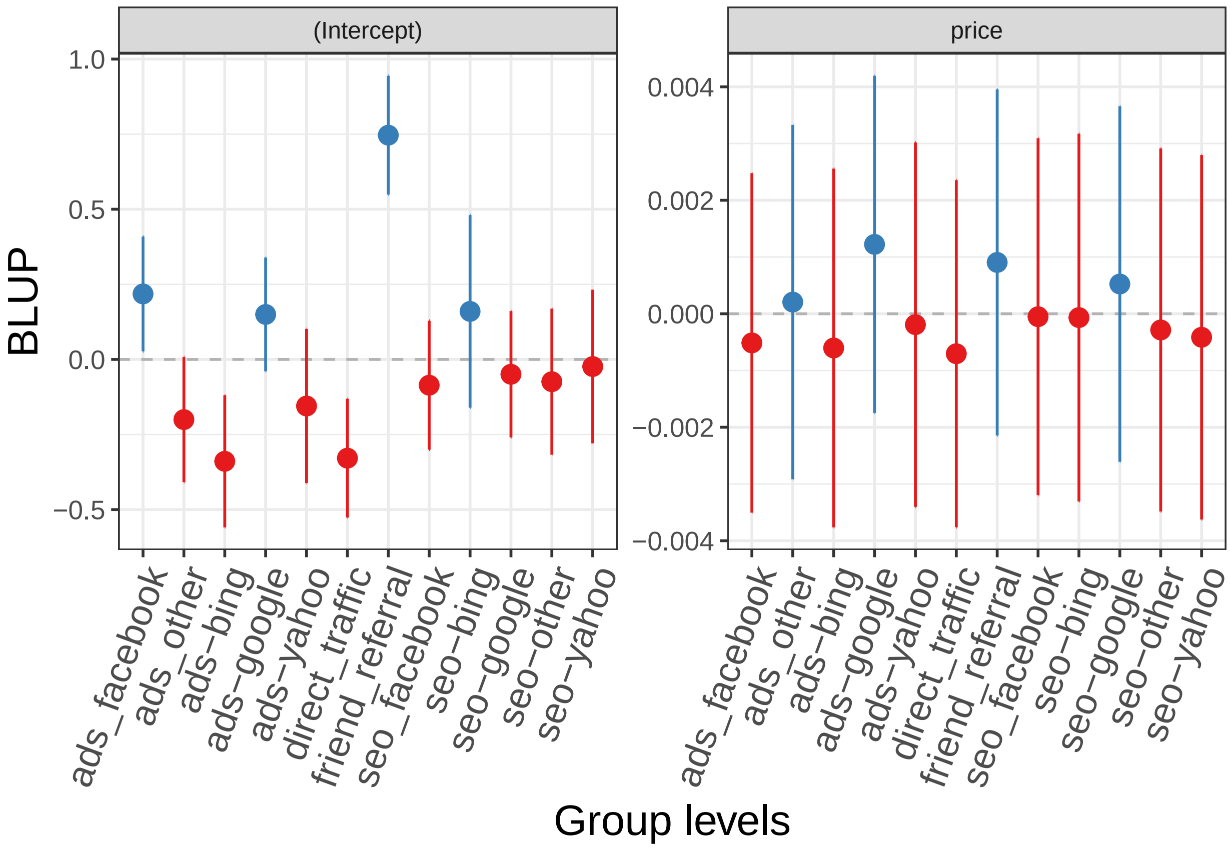
Given the structure in these data, my first thought was to use a linear mixed effects model. More specifically, because our response data is binomial (0/1 for converted and not converted) we use a generalized linear mixed effects model (glmm) with a binomial error model and logit link, i.e. a logistic mixed effects model.

(G)LMMs are a broad family of models that can account for multiple levels of structure in the data. There are fixed effects and random effects.

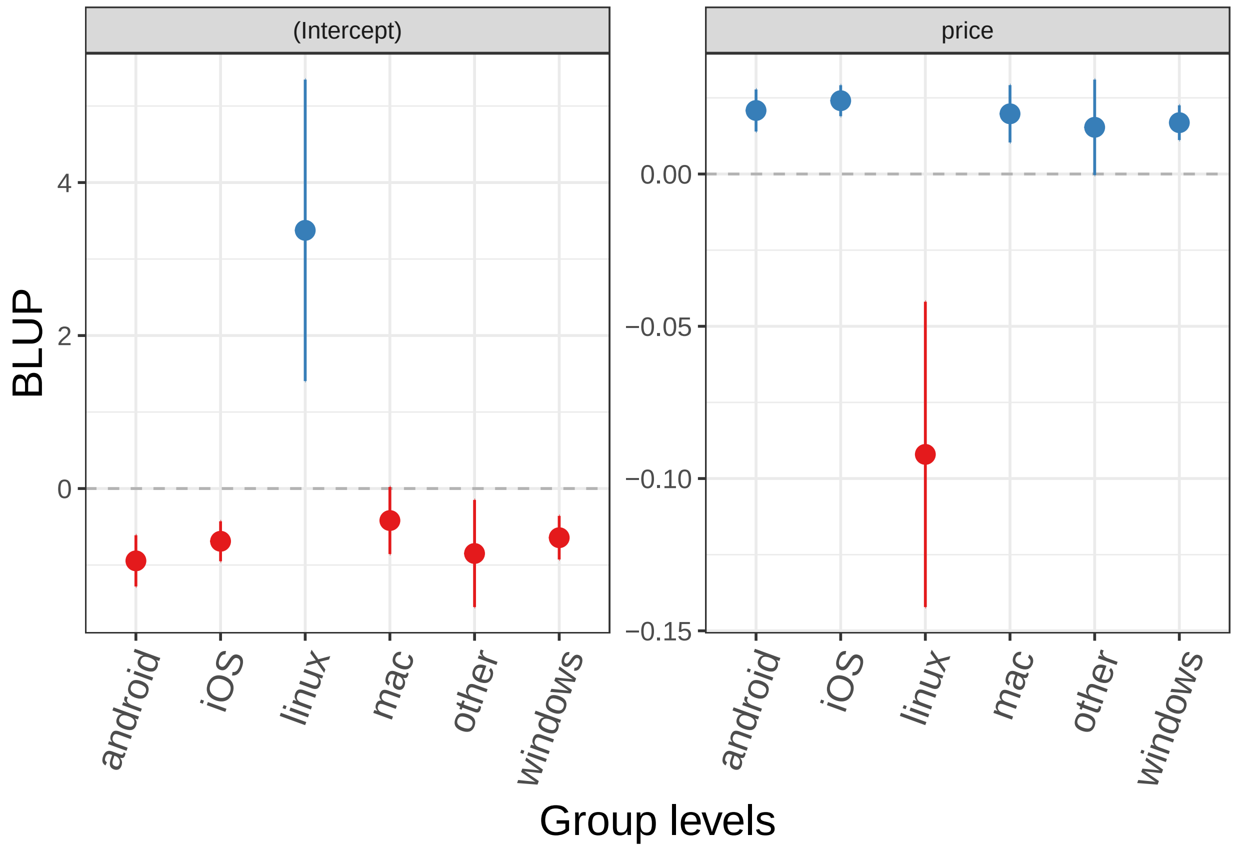
```r
####
#conversion_model_slope <- glmer(converted ~ price + (1 + price | source) + (1 + price | operative_syst
conversion_model_slope <- readRDS("conversion_slope_model.rds")
summary(conversion_model_slope)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## converted ~ price + (1 + price | source) + (1 + price | operative_system)
##    Data: all_data
##
##      AIC      BIC   logLik deviance df.resid
##  49398.5  49482.8 -24691.3  49382.5   275608
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -0.2358 -0.1487 -0.1287 -0.1127 12.1453
##
## Random effects:
##  Groups           Name        Variance  Std.Dev. Corr
##  source           (Intercept) 9.331e-02 0.305469
##                   price       2.882e-06 0.001698 -0.01
##  operative_system (Intercept) 2.717e+00 1.648299
##                   price       1.985e-03 0.044551 -0.99
## Number of obs: 275616, groups:  source, 12; operative_system, 6
##
## Fixed effects:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.75742    0.37768  -7.301 2.86e-13 ***
## price       -0.03251    0.01050  -3.097  0.00196 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##       (Intr)
## price -0.939
## convergence code: 0
## Model failed to converge with max|grad| = 0.00947359 (tol = 0.001, component 1)
## Model is nearly unidentifiable: very large eigenvalue
##  - Rescale variables?
```

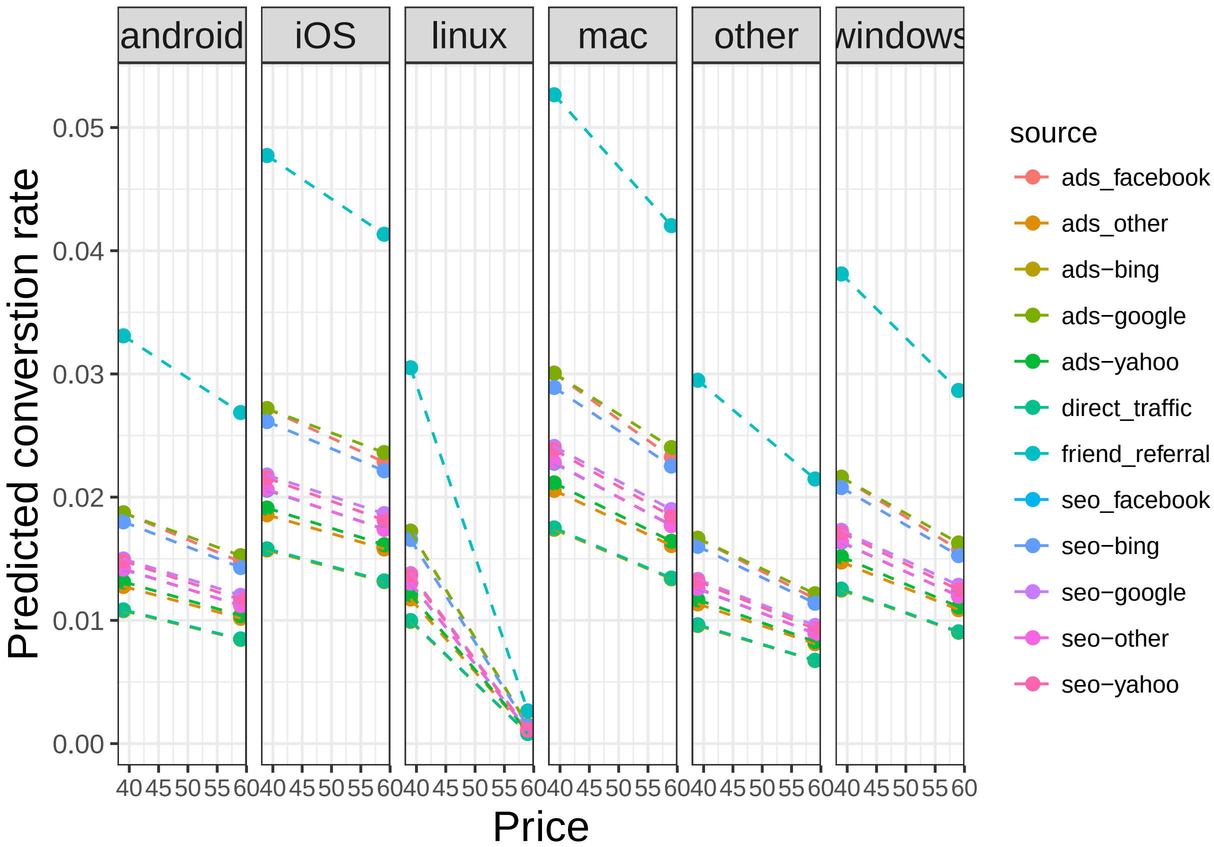
```r
library(lme4)
library(sjPlot)

p <- sjp.lmer(conversion_model_slope,free.scale=TRUE,
              prnt.plot = FALSE,show.values=FALSE)$plot.list

plot(p[[1]] + theme_bw() + theme(axis.text.x = element_text(angle = 70,hjust=1,size=14),
                 axis.text.y = element_text(size=10),
                 axis.title =element_text(size=16) ))
```




```
plot(p[[2]] + theme_bw() + theme(axis.text.x = element_text(angle = 70,hjust=1,size=14),
                  axis.text.y = element_text(size=10),
                  axis.title =element_text(size=16) ))
```




```
model_mat <- all_data %>% group_by(price,source,operative_system) %>%
  summarise(N=n()) %>% select(price,source,operative_system)

pred_or <- predict(conversion_model_slope,newdata=model_mat,type="response")
model_mat <- ungroup(model_mat) %>% mutate(predicted = pred_or)

p <- ggplot(model_mat,aes(price,predicted,color = source)) +
  geom_point(size=2) + geom_line(linetype="dashed") + facet_grid(.~operative_system) + xlab("Price") + y

p + theme_bw() + theme(axis.text.y = element_text(size=10),
                      axis.title =element_text(size=16),
                      strip.text.x= element_text(size=14),
                      strip.text.y= element_text(size=14))
```



Discussion

There are a few things that are apparrent from the mixed effect model.

- In general price has a negative effect on conversion rate, as expected
- Friend referral is more effective than any other other source of advertising.
- Facebook is also slightly more effective
- Bing and direct traffic seem to perform poorly
- Linux users appear to be extremely price sensitive

Questions

Should the company sell at the higher price?

Based on my analyses, yes the company should sell at the higher price. If we assume that raising the price does not effect the composition or the total of their usership, then we can simply look at the average conversion rate between the two price groups. Taking the product of the average rate and the price gives us the expected revenue per user.

```
conversion_mean <- all_data %>% group_by(price) %>% summarise(total= n(),conversion_rate= mean(converte
kable(conversion_mean)
```

price	total	conversion_rate	revenue
39	176376	0.0197533	0.7703769

price	total	conversion_rate	revenue
59	99240	0.0154676	0.9125857

How long did this test need to be?

This is a question about power. We want to know what our sample size needs to be in order to detect a certain effect size of interest. To do this, we need to determine what a relevant effect size is. In this case, perhaps it is the break even point. At the break even point:

$$conversion_1 * price_1 = conversion_2 * price_2$$

$$conversion_1 * \frac{price_1}{price_2} = conversion_2$$

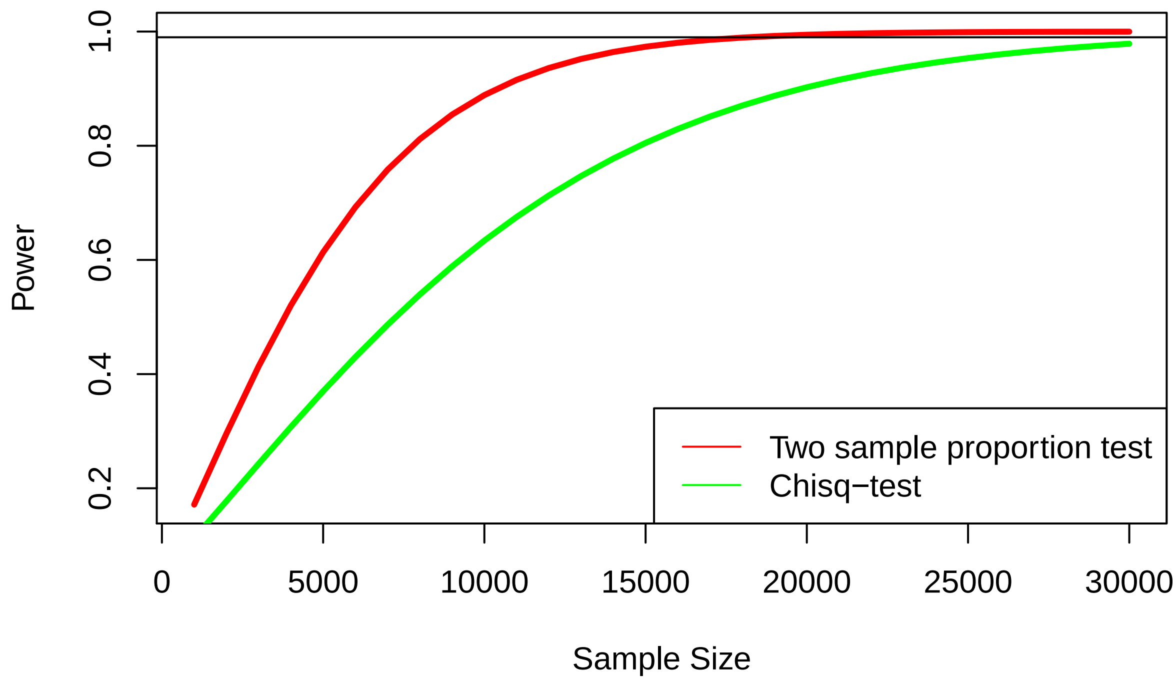
```
break_even_rate = conversion_mean$conversion_rate[1]*39/59
break_even_rate
```

```
## [1] 0.01305724
```

In general, doing power analysis on GLMM's is pretty challenging and typically requires simulation. I did not have time to implement this. Therefore I will use a more simple approach where I am testing the power of a two-sample proportion test. This tests the null hypothesis that the proportions of conversions (conversion rate) is the same in the two samples. This would represent our ability to see a difference in total conversion rate between the two price groups either a) overall or b) within a specific group. I also checked the power assuming we did a chi-squared test on the 2*2 price by converted converted contingency table

Overall two sample power at $\alpha = 0.05$

```
library(pwr)
N = seq(1000,30000,1000)
N2_ratio = conversion_mean$total[2]/conversion_mean$total[1]
power_2p = sapply(N,function(x) pwr.2p2n.test(h = ES.h(p1 =conversion_mean$conversion_rate[1],
                                                       p2 = break_even_rate),
                                             n1=x,n2=N2_ratio*x, sig.level = 0.05)$power)

all_data_fake <- all_data %>% group_by(price) %>% mutate(conversion=ifelse(price == 39, conversion_mean$
all_data_fake <- ungroup(all_data_fake) %>% mutate(simulated = rbinom(nrow(all_data_fake),1,prob =all_da

chisq_test_real <- chisq.test(table(all_data$price,all_data$converted))
chisq_test_fake <- chisq.test(table(all_data_fake$price,all_data_fake$simulated))
eff_size <- sqrt(chisq_test_fake$statistic/sum(chisq_test_fake$observed)) #chi squared effect size

power_chisq = sapply(N,function(x) pwr.chisq.test(w=eff_size,N=x,df=1,sig.level=0.05)$power)



plot(N,power_2p,type='l',col='red',ylab = "Power",xlab="Sample Size",lwd=3)
lines(N,power_chisq,col='green',lwd=3)
abline(h=0.99)
legend("bottomright",lty=c(1,1),col=c('red','green'),legend = c("Two sample proportion test","Chisq-tes
```



This suggests that we only need about 15,000 samples to have really high power at $\alpha = 0.05$. According to the timestamps, we could have had that amount of data in about a week. But, if we cared about the within group power, then we need to calculate the sample sizes in each group to get an estimate of the power. In general, we will need larger overall sampel sizes to estimate within group effects, but the mixed model is more powerful than subgroup analysis with small subgroup sample sizes.