Predicting organic installs multiplier

Samuel Chan

24 April 2017

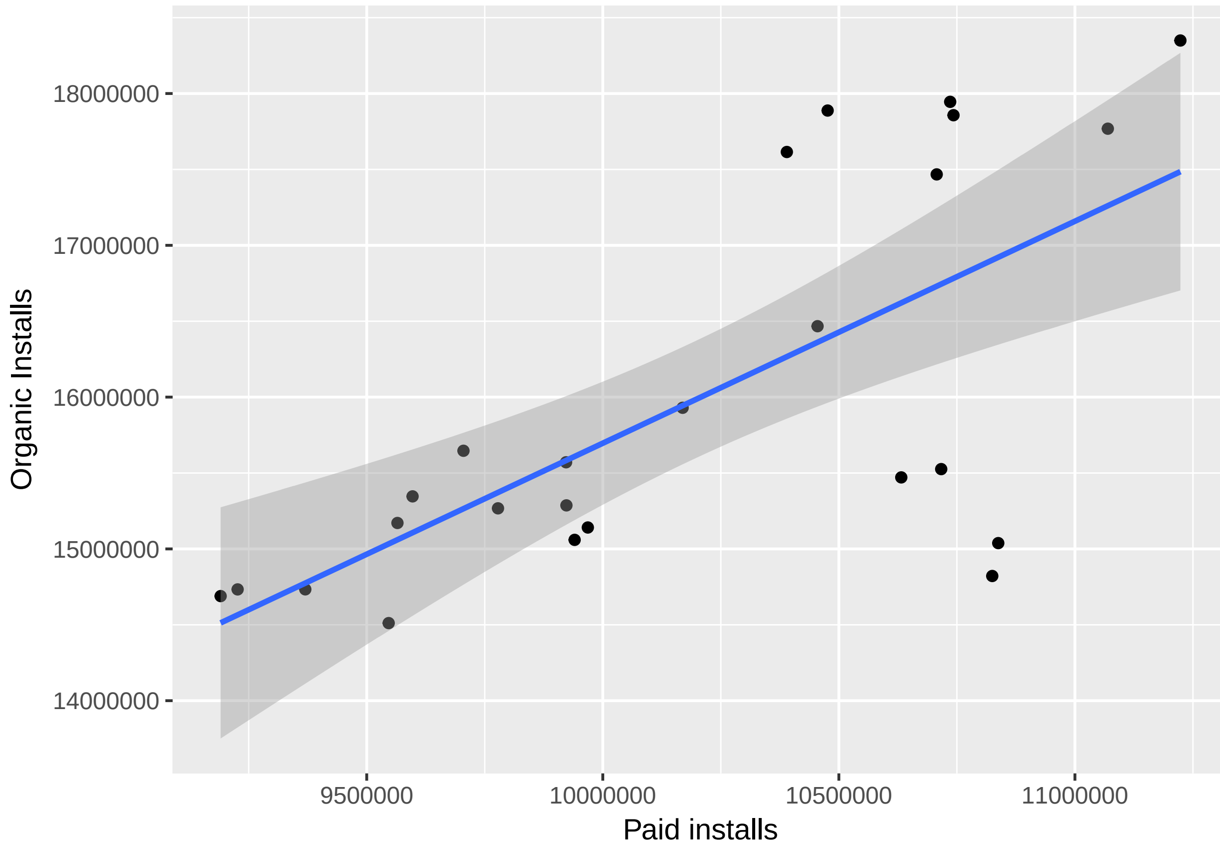
This exercise is motivated by a request from an industry friend, who is interested to learn the **organic multiplier** rate for mobile app campaigns in the Shopping category in an unspecified geo / country.

Some initialization and configuration

```
# clear our global environment
rm(list=ls())

# effectively prevent scientific notation e.g e+04
options(scipen = 999)
```

Read in the data and simple exploratory

```
# Read in the data
data <- read.csv("data2.csv", header=TRUE)

# Feature construction
data$total <- data$paid + data$organic

# Examine if there is a fairly linear relationship between paid and organic
library(ggplot2)
ggplot(data = data, aes(x=paid, y=organic))+geom_point()+labs(x="Paid installs", y="Organic Installs", 
```



Notice our use of `method=lm` automatically assume a 95% confidence region

Fit our first linear regression line

```
fit <- lm(organic ~ paid, data = data)
summary(fit)
```

```
##
## Call:
## lm(formula = organic ~ paid, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2081239  -508920   105710   507742  1495192
##
## Coefficients:
##                Estimate  Std. Error t value Pr(>|t|)
## (Intercept) 1072240.956 3245114.935   0.330 0.744076
## paid              1.462       0.318   4.599 0.000126 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 933400 on 23 degrees of freedom
## Multiple R-squared:  0.4791, Adjusted R-squared:  0.4564
## F-statistic: 21.15 on 1 and 23 DF,  p-value: 0.0001264
```

Interpreting the outlut: Values of coefficients are 1072240 and 1.462, hence prediction equation for model using the provided dataset is as below: **Organic installs = 1072240 + 1.462 * paid installs**

We also observed that our *cost* is 933400 (residual standard error)

Compare organic installs predicted by the linear regression model vs actual values

```
actual.o <- data$organic
pred.o <- fitted(fit)
comparison <- as.data.frame(cbind("Actual Organic"=actual.o, "Predicted Organic" = pred.o, "Difference
comparison[1:25,]
```

```
##    Actual Organic Predicted Organic Difference (%)
## 1        15346209          15107137           1.56
## 2        15170811          15060273           0.73
## 3        15570750          15582630           0.08
## 4        17614907          16266382           7.66
## 5        17467287          16730566           4.22
## 6        14511285          15033147           3.60
## 7        14733265          14565222           1.14
## 8        14689045          14512609           1.20
## 9        14733520          14774845           0.28
## 10       15646495          15264760           2.44
## 11       17887966          16392774           8.36
## 12       17945417          16772376           6.54
## 13       15059549          15609091           3.65
## 14       15471055          16620840           7.43
## 15       15286845          15583683           1.94
## 16       15267369          15371687           0.68
## 17       15929826          15943584           0.09
## 18       17857219          16782568           6.02
## 19       17768430          17260688           2.86
## 20       15141070          15649990           3.36
## 21       14821458          16902697          14.04
## 22       15038147          16921271          12.52
## 23       15525773          16744555           7.85
## 24       16467219          16361509           0.64
## 25       18349397          17485431           4.71
```

Our formula here: **Organic installs = 1072240 + 1.462 * paid installs** can quite reliably predict the organic installs of many days with roughly 2% to 8% of error (difference). Depending on your use-case, this may or may not be a sufficient predictive model. There were 2 days of outlier (21st March and 22nd March respectively), but those are likely the result of extra boost campaign unaccounted in the data – or possibly an Apple AppStore feature or Google Play feature.

If we have removed the outlying 21st and 22nd March's data

```
data2 <- data[-c(21,22),]
fit2 <- lm(organic ~ paid, data = data2)
summary(fit2)
```

```
##
## Call:
## lm(formula = organic ~ paid, data = data2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1600686  -296598    59908   350119  1199575
##
## Coefficients:
##                  Estimate    Std. Error t value    Pr(>|t|)
## (Intercept) -2390008.4073  2581170.1819  -0.926       0.365
## paid               1.8211        0.2543   7.161 0.000000464 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 706500 on 21 degrees of freedom
## Multiple R-squared:  0.7094, Adjusted R-squared:  0.6956
## F-statistic: 51.27 on 1 and 21 DF,  p-value: 0.0000004642
```

We would achieve a multiple R-squared value of ~0.7.

Caveat

It is important to note here that the dependent variable (organic installs / multiplier) is correlated with the independent variable (paid installs) but this correlation do not imply causation. That is, as the number of paid installs changes, we observe a change in the organic installs. This should not be inferred or interpreted to mean that the paid installs has *caused* the number of organic installs to change.

However, we should also note that the model has a multiple R-squared value of *0.6956*. An easy way to intepret this is that the linear model as a whole explains nearly 70% of the variation in our dependenable variable. We have after all, fit a simple linear model and this exercise should in no way be considered robust or scientifically rigorous. To improve our model's performance, we could:

- Obtain larger datasets

- Construct multiple features on each observation (app's placement on top free download rankings, top free shopping rankings)
- Variability (datasets spanning across multiple months)

We should also acknowledge that a game or mobile app's inherent design, including its virality features, social features, in-game social gifting features etc might also a role in the organic multiplier. Obtaining bigger sample sizes or more data could help in our hypothesis design.

Despite its simplicity, we also observe that a linear model is powerful enough as a predictor for organic multiplier on the provided data, and has the added advantage of being effective despite the relatively small sample size.