Cohort Install Attribution Model

Author: Greg Murray, Senior Data Scientist EA Mobile

INTRODUCTION

The goal for this work is to attribute the installs and revenue from the SKAdNetwork (SKAN) postbacks to a cohort date(s). This is needed in order to train the elasticity model which will be used to inform User Acquisition spend decisions. It is vital that the estimate of the installs and revenue for each day and cohort, respectively, be as accurate as possible for the elasticity model to provide useful spend signals. **Additionally, the output of this model can be used to derive priors for the parameters of the SKAN media mix model for campaign spend optimization and revenue attribution.**

This attribution will be accomplished in two parts. First, the installs must be attributed to a date. Second, once the installs are attributed, daily cohorts' DX revenue can be calculated by combining the revenue estimates associated with the corresponding conversion value.

This model will leverage the knowledge of SKAN along with the postback conversion values and batch send times to infer the Bernoulli probabilities of the install distributions for each campaign-conversion_value (Part I). Once calculated, we can use those Bernoulli probabilities as inputs to a second model that uses maximum likelihood estimation techniques to find the most probable install ratio (Part II). More specifically, we will utilize the knowledge that:

1.) the postback chain "breaks" exactly 24 hours after the last conversion event if no other conversion value increase occurs to reset the 24-hour timer
2.) the conversion value indicates the days since install date in the first two bits (up to 3 days after install date)
3.) the batches are sent every 24 hours and include all postbacks that occurred since the last batch send and none of the postbacks that occurred before the previous batch send*
4.) the postbacks contain the total number of installs for every campaign-conversion_value pair
5.) the conversion values indicate a range of revenue (spender buckets)

It is preferrable that the model in Part II be used in production, but in case any of the model's assumptions are egregiously violated, or time/computational constraints render that model impractical in the short term, the model in Part I can be used on its own. It should be noted, however, that the model in Part I will likely not provide powerful signals because it is only able to leverage data with low information content (SKAN send times).

Both model derivations deal only with installs as cohorts' DX revenue can be inferred once installs for each conversion value are attributed to a date.

**This is not entirely certain but is the consensus among most familiar with SKAN at EA, Singular and Facebook*

PART I

Bernoulli Cohort-Install Distributions

In this section we will derive the Bernoulli probability distribution over possible install dates from the batch send times based on a number of elements and axioms defined below.

Definitions

- $D_{install}$ is a discrete random variable representing day of install
- D_{last} is a discrete random variable representing day of last event
- D_{chain} is a discrete random variable representing day of chain break
- d_{skan} is the day of the SKAN postback receipt (*deterministic*)
- $T_{install}$ is a continuous random variable representing time (seconds) of install [0, 86400)
- T_{last} is a continuous random variable representing time (seconds) of last event [0, 86400)
- T_{chain} is a continuous random variable representing time (seconds) of chain break [0, 86400)
- t_{skan} is the time (seconds) of SKAN postback batch receipt [0, 86400) (*deterministic*)
- r is the number of days from retention bits in the conversion value [0-3] (*deterministic*)
- h is the threshold in seconds that marks the start of a new date [0, 86400); assumed to be 86400 (24:00) (*deterministic*)
- n is the conversion value [0-63] (*deterministic*)

Axioms

i. $D_{install} = D_{last} - r$
ii. $D_{last} = D_{chain} - 1$
iii. $d_{skan} - 1 \leq D_{chain} \leq d_{skan}$
iv. $t_x = t_x \, mod \, 86400$
v. $t_{skan} = T_{chain} + X, X \sim U[0, 86400)$*
vi. $p(D_{install} = d_{skan} - r - 2) + p(D_{install} = d_{skan} - r - 1) = 1$
vii. $T_{chain}, t_{skan} <$ h
viii. h=86400

***Assumption**

$$D_j := D_{install} + r + 1$$

(i-viii.)

$$D_j = d_{skan} - 1 = D_{chain} \Leftrightarrow T_{chain} > t_{skan}$$

$$\Rightarrow p\big(D_j = d_{skan} - 1\big) = p(T_{chain} > t_{skan})$$

(v., vii., viii.)

$$= f_{unif}(t_{skan} < T_{chain} < h)$$

$$= \frac{|h - t_{skan}|}{86400}$$

where $f(x)$ is the probability density function for a uniform distribution. So,

$$p(D_{install} = d_{skan} - r - 2 \mid t_{skan}) = \frac{|h - t_{skan}|}{86400}.$$

PART II

Maximum Posterior Estimation

Having inferred a Bernoulli probability distribution over possible install dates from the batch send times, we can go one step further and combine our estimated distribution with the data in the EA server by simulating the conversion values for our users there. To combine them in this way, we have to be cognizant of the co-dependencies of all the parameters and random variables involved. The distribution of installs for any given campaign is conditionally dependent on the probability distribution of installs from all other campaigns (for that game and conversion value) during that date range, and directly dependent on:

1.) the installs in the EA server on that day with the corresponding simulated conversion value (constraint)
2.) the total count of installs in that campaign's postback (constraint)

Because of the enormous number of co-dependencies, estimating the probability directly is intractable.

Fortunately, there is a way of estimating the ratio of installs between *day_a* and *day_b* without directly estimating the probability distribution. To do so, we use a discrete R.V. (ie: Irwin-Hall or Poisson) to model the conditional probability that: *install_count on day_a = x, given the installs in the EA server.*

Because we are mainly interested in the **most likely value of the parameters (campaigns' installs on day *a* & *b* campaign) given the install counts in the EA server on both days***, and they are iterable quanta (computationally feasible), we can simply maximize the product of the likelihood and prior (maximum posterior estimation) to turn the problem into one of constrained optimization. The only stipulation is that in order to compute the likelihood term, we must calculate the probability distribution of organic install counts ahead of time. Luckily, we have estimates of those distributions as we are currently able to distinguish paid and organic installs and can supplement those estimates with k-factor analysis currently underway.

posterior

Definitions

- $d_a := d_{skan} - r - 2$
- $d_b := d_{skan} - r - 1$
- ${i^{EA}_{x}}_{t_{skan}}n$ is the count of installs in EA server on day x, where $x \in \{a, b\}$, and $time_{EA} > t_{skan}$ if day=*a* and $time_{EA} \leq t_{skan}$ if day=b, for conversion value n.

- ${I^{c}_{x}}_{t_{skan}}n$ is a Irwin-Hall random variable of count of installs on day x where $time_{EA} > t_{skan}$ if day $= a$, and $time_{EA} \leq t_{skan}$ if day $= b$, for campaign c and conversion value n

- $ORGANIC_{xn}$ is a discrete random variable of count of organic installs on day x for conversion value *n*
- K^c_n is the total count of installs in the SKAN postback for campaign c and conversion value *n*
- C_n is the total count of campaigns currently active for the game being modelled and conversion value n

**Subscript notation will omit t_{skan} and n as the time operator is implied by day=a or day=b (>t_{skan} or <=t_{skan}), and the conversion value n is constant for each model but is important to note in the definitions for context.*

Constraints

i. $organic_a + \sum_{c=1}^{C} i_a^c = i_a^{EA}$
ii. $organic_b + \sum_{c=1}^{C} i_b^c = i_b^{EA}$
iii. $i_a^c + i_b^c = k^c \ , \forall c$

Axioms

i. $\boldsymbol{p(I_x^1 = i_x^1) \perp p(I_x^2 = i_x^2) \perp \cdots \perp p(I_x^c = i_x^c), \forall c \ , \forall x, c \in C, \ x \in (a, b)^*}$

ii. $p(I_x^c = i_x^c \mid i_a^{EA}, i_b^{EA}) \neq p(I_x^c = i_x^c), \forall c, \forall x, \mathbf{c} \in C, x \in (a, b)$

iii. $I_x^{EA} = ORGANIC_x + \sum_{c=1}^{C} I_x^c, \forall c, \forall x, \mathbf{c} \in C, x \in (a, b)$

iv. $i_x^{EA} = organic_x + \sum_{c=1}^{C} i_x^c, \forall c, \forall x, \mathbf{c} \in C, x \in (a, b)$

v. $p(organic_a) \perp p(organic_b)$

vi. $\sum_{i=0}^{K^c} p(I_a^c = i_a^c) + p(I_b^c = K^c - i_a^c) = 1, \forall c, \mathbf{c} \in C$

vii. $p(organic_x) \perp p(I_x^c = i_x^c), \forall c, \forall x, c \in C, x \in (a, b)$*

viii. $p(I_x^c = i_x^c, I_x^{c'} = i_x^{c'} \mid i_x^{EA}) \neq p(I_x^c = i_x^c \mid i_x^{EA}) * p(I_x^{c'} = i_x^{c'} \mid i_x^{EA}), \forall c, \forall x, \mathbf{c} \in C, x \in (a, b)$

*Assumption

We first define our posterior as seen below in purple. This arises naturally from the business problem and formulation of the model. Because of the conditional dependence of all campaigns' install counts (for each day) on the install counts in the EA server, estimating this probability distribution directly is practically impossible as it requires estimating the conditional probabilities of all possible combinations of campaigns! Consequently, even if we had access to a way of directly estimating those conditional probabilities (we don't), the computational complexity alone renders a direct estimation of the posterior infeasible.

Instead, we can leverage the clever simplicity of Bayes theorem so need only estimate the right side of the proportionality below,

$$Posterior \propto \mathcal{L} * \pi.$$

$$Posterior := p(I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \dots, I_a^c = i_a^c, I_b^c = i_b^c \mid I_a^{EA} = i_a^{EA}, I_b^{EA} = i_b^{EA})$$

Prior Derivation

$$\pi := p(I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \dots, I_a^c = i_a^c, I_b^c = i_b^c)$$

(i., vi, $p(D_{install} = d_{skan} - r - 2 \mid t_{skan})$ ~ Bernoulli RV, **Poisson Binomial PMF)**

$$p(I_a^c = i_a^c) \equiv p(I_a^c = \overline{f(\boldsymbol{p_c}; k_c, K_c)}),$$

where $f(p; k)$ is the Poisson Binomial PMF (the sum of n Bernoulli RV's with different p(x)'s), $\boldsymbol{p_c}$ is the vector of Bernoulli probabilities for the observations in campaign c, and k_c is the number of installs on day A for campaign c and K_c is the total number of installs in the postback for campaign c.

Because the marginal probabilities of each campaign's installs are independent, their joint distribution is simply the product of all the Poisson Binomial RV's. Because the formal PMF of the Poisson Binomial distribution is computationally infeasible for n>~20, the discrete Fourier transform is used in its place.

(Formal Poisson Binomial PMF)

$$\boldsymbol{\pi}(\boldsymbol{p}; \boldsymbol{k}, \boldsymbol{K}) = \prod_{c=1}^{C} \left[\sum_{A \in F_k} \left(\prod_{i \in A} p_i \prod_{j \in A^C} (1 - p_j) \right) \right]$$

(PMF used for practical implementation; discrete Fourier transform)

$$\boldsymbol{\pi}(\boldsymbol{p}; \boldsymbol{k}, \boldsymbol{K}) = \prod_{c=1}^{C} \left[\frac{1}{n+1} \sum_{l=0}^{n} F^{-lk} \prod_{m=1}^{n} (1 + (F^l - 1) * p_m) \right],$$

where $F := \exp(\frac{2i\pi}{n+1})$, $i = \sqrt{-1}$, and

$p := p(\, D_{install} = d_{skan} - r - 2 |\, t_{skan}).$

Likelihood Derivation

The likelihood derivation is slightly more involved,

$$\boldsymbol{\mathcal{L}} := p\big(I_a^{EA} = i_a^{EA}, I_b^{EA} = i_b^{EA} \mid I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \ldots, I_a^c = i_a^c, I_b^c = i_b^c\big)$$

(ii., iii.)

$$= p(ORGANIC_a + \sum_{c=1}^{C} I_a^c = organic_a + \sum_{c=1}^{C} i_a^c \ , \ ORGANIC_b + \sum_{c=1}^{C} I_b^c = organic_b + \sum_{c=1}^{C} i_b^c \ | \ I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \dots, I_a^c = i_a^c, I_b^c = i_b^c)$$

$$\equiv p\Bigg(ORGANIC_a = organic_a, \sum_{c=1}^{C} I_a^c = \sum_{c=1}^{C} i_a^c \ , \ ORGANIC_b = organic_b, \sum_{c=1}^{C} I_b^c = \sum_{c=1}^{C} i_b^c \ \Bigg| \ I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \dots, I_a^c = i_a^c, I_b^c = i_b^c)$$

(vii.)

$$= p(ORGANIC_a = organic_a, ORGANIC_b = organic_b \mid I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \dots, I_a^p = i_a^p, I_b^p = i_b^p) * p(\sum_{c=1}^{C} I_a^c = \sum_{c=1}^{C} i_a^c, \sum_{c=1}^{C} I_b^c = \sum_{c=1}^{C} i_b^c \mid I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \dots, I_a^c = i_a^c, I_b^c = i_b^c)$$

(p(X=x|X=x)=1)

$$= p(ORGANIC_a = organic_a, ORGANIC_b = organic_b \mid I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \dots, I_a^c = i_a^c, I_b^c = i_b^c)$$

(iii.)

$$= p(ORGANIC_a = i_a^{EA} - \sum_{c=1}^{C} i_a^c, ORGANIC_b = i_b^{EA} - \sum_{c=1}^{C} i_b^c \mid I_a^1 = i_a^1, I_b^1 = i_b^1, I_a^2 = i_a^2, I_b^2 = i_b^2, \dots, I_a^c = i_a^c, I_b^c = i_b^c)$$

(v.)

$$\mathcal{L} = p(\boldsymbol{ORGANIC_a = i_a^{EA} - \sum_{c=1}^{C} i_a^c} \mid I_a^1 = i_a^1, I_a^2 = i_a^2, \dots, I_a^c = i_a^c) * p(\boldsymbol{ORGANIC_b = i_b^{EA} - \sum_{c=1}^{C} i_b^c} \mid I_b^1 = i_b^1, I_b^2 = i_b^2, \dots, I_b^c = i_b^c).$$

Optimality Equation

So to get the parameters that maximize the posterior probability for our Irwin-Hall distribution of installs we take,

$$\underset{i_a^1, i_b^1, i_a^2, i_b^2 \dots, i_a^c, i_b^c}{\operatorname{argmax}} \ \mathcal{L}_{log} + \pi_{log}$$

$$
\begin{aligned}
&subject\ to \\
&\qquad organic_a + \sum_{c=1}^{C} i_a^c = i_a^{EA}, \\
&organic_b + \sum_{c=1}^{C} i_b^c = i_b^{EA}, \\
&\qquad i_a^c + i_b^c = k^c \ , \forall c \ .
\end{aligned}
$$

PART III

Quantum (Rounding) Error

While we transformed an intractable problem into a manageable one by discretizing our posterior in a form we can calculate more easily, it did not come without cost. Ideally, we would want the parameters of the marginal probability distribution itself because that is the maximum likelihood estimator and ensures our distribution of installs will account for the uncertainty in our estimations more accurately than the discrete count solution. This is especially true when the SKAN install counts are low for a given conversion value but luckily the difference between the two distributions' means converges to zero as sample size goes to infinity. For example, if there are 1000 installs in a given conversion value and the true probability is 25.7% an install occurred on *day_a* from that conversion value, then the maximum likelihood estimate (MLE) of our model will be 257 installs on *day_a* and there will be zero "quantizing" error. However, if there are only 10 installs for that conversion value in the SKAN postback, then the model's MLE will be 3 installs on *day_a* which is off by 16.7% of the true MLE.

PART IV

Limitations and Future Iterations

The potentially significant limitation of the maximum posterior solution derived in Part II is that axioms ***v.*** and ***vii.*** do not hold in most situations. However, since problem complexity increases when incorporating the conditional dependence of organics on campaign installs (which is really endogeneity, both caused by UA impressions to some degree), and the codependence between organic and paid installs is believed to be marginal, this assumption allows for a tractable solution that is believed to not overly bias the results. Fortunately, there

has been work done that measures the relationship between organics and UA spend which would serve as a useful starting point. Future iterations of this work should try to account for those co-dependencies between organic and paid installs in the likelihood term.

The biggest limitation of this approach is undoubtedly the large number of corner solutions that result from a large number of parameters and weak signal. In other words, there are many ways you can distribute the installs across the various campaigns that result in the same posterior density since the only distinguishing factor between them is the receipt times (priors). The more granular the send times the more effective this model will be in identifying unique optimal parameter values.

Lastly, another assumption that can be improved is that of the uniform random variable for time between chain break and postback send time. One possible improvement is to model the rate of install with a Poisson or time series trained on different panels.