Identifying Subgroups with Enhanced Peer Influence Using High-dimensional Data

Shan Huang[1], Tong Wang[2], and Haojun Wu[3]

[1]MIT

[2]Tippie College of Business, University of Iowa

[3]Tencent Corporation

Abstract

We develop a machine learning model that identifies user-friend pairs where social cues (i.e., friends likes) have enhanced effects on users' ad engagement, using high-dimensional data. The impact of social cues in social ads is jointly determined by the characteristics of influencers (influence), those influenced (susceptibility) and their relationships. Our model searches for the user-friend pairs exhibiting enhanced effects of social cues. Such pairs are characterized by the interpretable rules constructed from the demographic, behavioral and network characteristics of users, their friends shown in ads and the tie strength and social embeddedness between the users and the friends. We present a Bayesian framework for learning a rule set. The Bayesian model consists of a prior to control the size and shape of a rule set and a Bayesian logistic regression to characterize interactions of features, treatment and subgroup membership. We also develop an efficient inference method for learning a MAP model. Our results on ad engagements for different product categories demonstrate that the enhanced effects of social cues associated with the identified pairs can be much larger than that on the entire population. As a result, targeting the contagious user-friend pairs is an effective strategy to improve social ads effectiveness.

Contact. shanh@mit.edu, tong-wang@uiowa.edu, hanswu@tencent.com.

1. INTRODUCTION

Individuals are influencing and influenced by their peers, when they adopt products, express political views, share posts in social platforms, etc (Tucker 2012, Bakshy et al. 2012, Bond et al. 2012). Identification of peer influence is also critical to understanding information diffusion and designing effective network marketing strategies (Aral 2011). Nowadays, peer influence becomes even larger and more pervasive spreading in large online social networks, such as Facebook, WeChat and Twitter. Previous research suggested that both influence and susceptibility would affect peer influence (Aral and Walker 2011, Bakshy et al. 2012, Muchnik et al. 2013, Bond et al. 2012, Jones et al. 2013). Recent literature also identified the moderating effects of dyadic relationships on influence (Aral and Walker 2014). However, no research has identified specific subgroups considering influence, susceptibility and relationship together, in which peer influence has stronger effects.

In this paper, we develop a model to identify such subgroups using data from a large-scale randomized field experiment conducted on WeChat Moments ads, a type of social advertisements displayed in WeChat users' newsfeed. WeChat is a world leading mobile social network platform, with more than 800 million active users. Our goal is to identify subgroups where peer influence exerts enhanced effects on users behaviors in social advertisements - where a social cue (i.e., a friend's like) on an ad has enhanced effect on a user's response (i.e., clicking) to the ad. Such a subgroup is characterized by interpretable rules constructed from the demographic, behavioral and network characteristics of a user and his/her affiliated friend (An *affiliated friend* refers to the individual who generated the social cue displayed on the ad) and the attributes of dyadic relationships between users and the friend. (See Figure 1 for illustration of the experiment and related features.) To the best of our knowledge, this is the first paper that identified the enhanced subgroup effects of peer influence in social

advertisements across different product categories, while previous work only estimated the average effects of influence(Aral 2011, Huang et al. 2016). This paper also provides important implications for increasing the value of peer influence in social advertisements of different product categories.

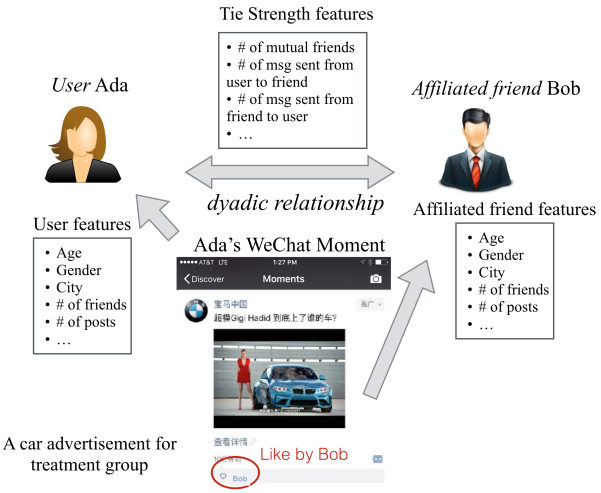

Figure 1: An example of a car advertisement shown on a user's WeChat Moment

Problems related to ours are referred to as subgroup analysis or subgroup-treatment effect interactions (Rothwell 2005). Two classes of methods for identifying subgroups with enhanced treatment response have been developed recently. One class includes methods that fit statistical models incorporating features and treatment-by-feature interactions by employing various methods of regularized regression, often referred to as moderated regression analysis Cohen et al. (2013). However, the methods rely on clear a prior hypothesis about which subgroups are involved in the interactions and the result is often a complex "black box" model for capturing potentially complicated interactions of features, which is not directly actionable in a real application. The other class of widely adopted approach is recursive partitioning that partitions data into smaller subsets till a stopping criteria is met. Several models fall into this category, including Subgroup Identification based on Differential Effect

Search, Virtual Twins (Foster et al.), Interaction Trees (Su et al. 2009), etc. Recursive partitioning methods are a natural way to analyze a large number of features and assess potentially complicated interactions of the features. It also has an advantage over other methods for the interpretability of the model since a model since a subgroup can be clearly identified by rules. However, in recursive partitioning, data is partitioned greedily and splits are not based on the treatment effect of the endpoint of interest, therefore the identified subgroups.

To address the shortcomings of previous methods, we propose a Causal Rule Set (CRS) model to capture a subgroup. We present a Bayesian framework for learning a CRS, that consists of a prior for rule set and a Bayesian logistic regression for modeling the conditional likelihood of data. The prior is constructed to control the size and shape of an output rule set. The Bayesian logistic regression characterizes the interactions of features, treatment and subgroup. Our method does not depend on any a priori knowledge of subgroup or features and is learned by optimizing the posterior. The output is a small set of short rules which is easily interpretable and actionable.

We propose an efficient inference method for learning a maximum a priori (MAP) solution. The algorithm consists of 3 steps to effectively reduce computation. First, a rule space is identified via learning from labels generated by two separate random forests trained on control and treatment groups, respectively. Then, a metric is developed to replace the more computationally heavy objective during the search. Lastly, the algorithm applies an exploration-with-exploitation strategy globally at the simulated annealing chain level, and also locally when choosing neighbors.

Another main contribution of our paper is the carefully designed large-scale randomized controlled experiments on a real social network platform. Peer influence is often

endogenous(Manski 1993, Shalizi and Thomas 2011). The gold standard for the influence identification is randomized field experiment, which largely eliminates the bias created by homophily and other confounding factors and enables the detection of peer influence in many different subpopulations. Our experiment was conducted on 28,668,980 users across 99 ads, in which we manipulated the number of social cues displayed on WeChat Moments ads at ad-user level. After cleaning the data, our analysis focuses on 2,578,664 user-friend-ad observations. Peer influence in our experiment refers to the effects of seeing the first friend's like under an ad on the adoption (i.e., clicking) of the ad.

The paper is organized as follows: we first describe prior work related to ours, then we present the Bayesian method for learning a CRS model. Next, we describe the inference method and strategies we use to efficiently search for a MAP solution. Finally, we present the experiment we conducted and apply CRS and other baseline models to the data set and compare the results.

2. PRELIMINARIES

We work with data $S = \{(\mathbf{X}_i, Y_i, T_i)\}_{i=1}^n$ comprised of n instances, each of which corresponds to a user described by a covariate vector $\mathbf{X}_i \in \mathbb{R}^{J+1}$ that consists of a constant and J features, $\mathbf{X}_i = \{1, X_{i1}, X_{i2}, ..., X_{iJ}\}$. Let $T_i \in \{0, 1\}$ denote the treatment assignment for user i and 1 indicates the user receives the treatment. Let $Y_i \in \{0, 1\}$ denote the binary outcome and 1 indicates the better outcome. We use Rubin Causal Model framework Rubin (1974) with potential outcomes $Y_i(1), Y_i(0) \in \{0, 1\}$ under treatment and control, respectively, and define the treatment effect at $\mathbf{x}$ as

$$\tau(\mathbf{X}) = \mathbb{E}\big[Y_i(1) - Y_i(0) | \mathbf{X}_i = \mathbf{x}\big]. \tag{2.1}$$

Our goal is to identify a subgroup of users on which the treatment demonstrates an enhanced effect. The main difficulty in finding such a subgroup and most problems in causal analysis is one can only observe one of the two potential outcomes for a given example, therefore one cannot directly estimate the differences in the form $Y_i(1) - Y_i(0)$. A standard way is to assume unconfoundedness Rosenbaum and Rubin (1983), i.e., that the treatment assignment is independent of the potential outcomes for Y_i conditional on $\mathbf{X}_i$:

$$\big\{Y_i(0), Y_i(1)\big\} \perp T_i|\mathbf{X}_i. \tag{2.2}$$

This assumption allows us to treat nearby observations as having come from a randomized experiment, therefore any observation's potential outcomes will be in general consistent with those' that are within close proximity. For example, in causal forests Wager and Athey (2015), observations in the same leaf in a tree are considered to have come from a randomized experiment.

In our model, we generalize the notion of "proximity" from subgroups defined by one rule to a set of rules. A rule is a conjunction of conditions and the number of conjunctions is referred to as the length of the rule. For example, "female AND age > 50 AND married" is a rule and has length 3. Let $a(\mathbf{X}_i) \in \{0, 1\}$ represent if example i satisfies rule a or as we also call it, "*covered*" by rule a. Let A denote a rule set. An observation satisfies the rule set if it satisfies at least one rules in the set, represented as below.

$$A(\mathbf{X}) = \begin{cases} 1 & \exists a \in A, a(\mathbf{X}) = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

We use S_A to represent the subgroup defined by A and an observation in S_A needs to satisfy

at least one rule in A

$$S_A = \{(\mathbf{X}_i, Y_i, T_i) \in S | A(\mathbf{X}_i) = 1\}. \tag{2.4}$$

The average treatment for S_A is

$$\tau(S_A) = \frac{\sum_{\{i:T_i=1, A(\mathbf{X}_i)=1\}} Y_i}{|\{i : T_i = 1, A(\mathbf{X}_i) = 1\}|} - \frac{\sum_{\{i:T_i=0, A(\mathbf{X}_i)=1\}} Y_i}{|\{i : T_i = 0, A(\mathbf{X}_i) = 1\}|}. \tag{2.5}$$

3. CAUSAL RULE SETS

Our goal is to find a set of rules to capture a subgroup of users that demonstrate an enhanced treatment effect. We propose a Bayesian framework for learning a rule set A from data. The Bayesian approach turns this problem into finding a MAP solution $P(A|S, H)$, given training data S and a set of hyperparameters denoted by H. The model consists of a prior for forming a rule set A, Prior(A), and a Bayesian logistic regression $\mathcal{B}(S; A, \mathbf{w})$ for modeling the conditional likelihood of data. We now describe them in detail.

3.1. Prior for a Rule Set

We construct a prior for A that favors a small model, i.e., a small set of short rules. A small model is easy to interpret since it contains fewer conditions to comprehend. Besides, the shorter a rule, the larger the support, which naturally avoids overfitting. We use a generative process described in Wang et al. (2016) to form a rule set. Let $\mathcal{A}$ denote a rule space from where A is drawn. This rule space can be further partitioned into pools of rules with equal lengths, indexed by rule length $l, l = \{1, ..., L\}$, L being the maximum rule length a user allows. Therefore, $\mathcal{A} = \mathcal{A}_1 \cup \cdots \mathcal{A}_L$.

Assume the interpretability of a rule is only associated with its length. Then rules in the same pool l are drawn uniformly randomly with probability p_l. We place a beta prior on p_l

to control the size and shape of an output rule set, yielding

$$\text{Select a rule from } \mathcal{A}_l \sim \text{Bernoulli}(p_l), \tag{3.1}$$

$$p_l \sim \text{Beta}(\alpha_l, \beta_l) \text{ for } l \in \{1, ..., L\}. \tag{3.2}$$

Let M_l notate the number of rules drawn from $\mathcal{A}_l$. Then the probability of selecting a rule set A is

$$\begin{aligned} \text{Prior}(A) &= \prod_{l=1}^{L} p(M_l; \alpha_l, \beta_l) \\ &= \prod_{l=1}^{L} \frac{B(M_l + \alpha_l, |\mathcal{A}_l| - M_l + \beta_l)}{B(\alpha_l, \beta_l)}, \end{aligned} \tag{3.3}$$

where $B(\cdot)$ represents the Beta function. The parameters $\{\alpha_l, \beta_l > 0\}_{l=1,...,L}$ on the priors control the expected number of rules of each length in the rule set. We usually choose $\alpha_l \ll \beta_l$ so that the model tends to choose a smaller set from each pool.

3.2. Bayesian Logistic Regression for Conditional Likelihood

A Bayesian Logistic Regression model controls for confounding covariates, and models the impact of receiving the treatment and being in the subgroup, giving the conditional likelihood:

$$\begin{aligned} p(Y_i = 1 | \mathbf{X}_i, T_i) = \sigma\Big(&\mathbf{v}\mathbf{X}_i + \gamma^{(0)} T_i + \gamma^{(1)} T_i A(\mathbf{X}_i) - \\ &\gamma^{(2)}(1 - T_i) A(\mathbf{X}_i)\Big), \end{aligned} \tag{3.4}$$

where $\sigma(\cdot)$ is a sigmoid function. The exponent in formula (3.4) models the interaction between the attributes, treatment assignment T_i and the rule set A. $\mathbf{v}$ represents the vector

of weights for attribute including an intercept. $\mathbf{vX}_i$ captures the contribution from the attributes towards getting a good outcome $Y_i - 1$, regardless of whether receiving a treatment. $\gamma^{(1)}$ can be regarded as a measurement of additional treatment effect for being in subgroup A. $\gamma^{(1)} T_i A(\mathbf{X}_i)$ implies that the additive treatment effect exists only when user i receives the treatment, i.e., $T_i = 1$, and it belongs to the subgroup, i.e., $A(\mathbf{X}_i) = 1$. $\gamma^{(0)} T_i$ captures the base treatment effect on the entire population. A significant interaction for γ_1 implies differential treatment effects between subgroup defined by A and the rest of the data. The term $\gamma^{(2)}(1 - T_i)A(\mathbf{X}_i)$ discourages $A(\mathbf{X}_i)$ from covering observations with a positive outcome without treatment.

Let $\mathbf{w} = \{\mathbf{v}, \gamma^{(0)}, \gamma^{(1)}, \gamma^{(2)}\}$. Assuming data is i.i.d, the conditional likelihood of data S is

$$\begin{aligned}
&\text{CondLikelihood}(S; A, \mathbf{w}) = \prod_{i=1}^{n} p(Y_i | \mathbf{X}_i, T_i) = \\
&\prod_{i=1}^{n} \sigma\Big[\mathbf{vX}_i + \gamma^{(0)} T_i + \gamma^{(1)} T_i A(\mathbf{X}_i) - \gamma^{(2)}(1 - T_i) A(\mathbf{X}_i)\Big]^{Y_i} \cdot \\
&\Big[1 - \sigma\big(\mathbf{vX}_i + \gamma^{(0)} T_i + \gamma^{(1)} T_i A(\mathbf{X}_i) - \gamma^{(2)}(1 - T_i) A(\mathbf{X}_i)\big)\Big]^{(1 - Y_i)}
\end{aligned} \tag{3.5}$$

parameterized by:

$$A \sim \text{Prior}(A), \tag{3.6}$$

$$\mathbf{w} \sim \mathcal{N}(\mu, \Sigma). \tag{3.7}$$

μ is a vector of means and Σ is a variance matrix. We define

$$\mathcal{B}(S; A, \mathbf{w}) = \log \text{CondLikelihood}(S; A, \mathbf{w}) + \log \text{Prior}(\mathbf{w}) \tag{3.8}$$

4. INFERENCE METHOD

We describe in this section how to find an optimal rule set A^* that maximize the posterior $p(A|H,S)$, where H denotes a set of hyperparameters. This is equivalent to maximizing:

$$F(A,\mathbf{w};S,H)=\mathcal{B}(S;A,\mathbf{w})+\log \text{Prior}(A). \tag{4.1}$$

We write the objective as $F(A,\mathbf{w})$, Bayesian logistic regression as $\mathcal{B}(A,\mathbf{w})$, omitting dependence on hyperparameters and data when appropriate.

Given a rule set A, the corresponding parameters are obtain by maximizing $F(A,\mathbf{w})$, which is equivalent to maximizing $\mathcal{B}(A;\mathbf{w})$ since only $\mathcal{B}(A;\mathbf{w})$ in $F(A,\mathbf{w})$ depends on $\mathbf{w}$. Let $\mathbf{w}_A$ denote the optimal parameters for a given rule set A. $\mathbf{w}_A$ is estimated via

$$\mathbf{w}_A=\max_{\mathbf{w}}\mathcal{B}(A,\mathbf{w}) \tag{4.2}$$

Our goal is to find an optimal pair $(A^*,\mathbf{w}_{A^*})$, such that

$$A^*,\mathbf{w}_{A^*}=\max_{A,\mathbf{w}}F(A,\mathbf{w}). \tag{4.3}$$

The main procedure follows the steps of simulated annealing which generates a sampling chain that starts from a random state, proposes the next state by selecting from the neighbors, and converges to the optimal solution as the temperature cools down. In the context of our model, where each state is defined as $s^{[t]}=(A^{[t]},\mathbf{w}_{A^{[t]}})$, where t is the time stamp. Given a rule set $A^{[t]}$, a neighboring solution is a rule set whose edit distance is 1 to the current rule set (one of the rules is different). Therefore, $A^{[t+1]}$ is generated by adding, removing or replacing a rule from $A^{[t]}$. A candidate for next state $s^{[t+1]}$ is proposed by first generating

$A^{[t+1]}$ and then obtaining $\mathbf{w}_{A^{[t+1]}}$ that maximizes $F(A^{[t+1]}, \mathbf{w})$. Given a temperature schedule function over time steps, $T(t) = T_0^{1-\frac{t}{N_{\text{iter}}}}$, the proposed state $(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}})$ is accepted and assigned to $s^{[t+1]}$ with probability $\min(1, \exp\{\frac{F(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}}) - F(s^{[t]})}{T(t)}\})$.

Inference for rule set models is challenging since the number of rules grows exponentially with the number of conditions and classic simulated annealing that generates a random neighbor at each iteration converges very slowly, making the problem even more difficult to solve in practice.

We describe a three-step inference algorithm for efficiently searching for a MAP Model. First, we generate a set of candidate rules $\mathcal{A}$ that contains promising rules with non-negligible support and restrict our search within only this meaningful set. Then we develop a simple metric to replace to computationally heavy objective. Lastly, we run a modified simulated annealing that applies exploration-with-exploitation strategies globally and locally to improve the search efficiency.

4.1. Candidate Rule Generation

Our rule generation approach borrows concepts from Virtual Twins methods Foster et al. where a model is built to generate an estimate of $P(Y_i = 1)$ in treatment and control group, respectively. The difference in the two values is an estimate of treatment effect at the individual level. We then binary code this estimated treatment effect and apply a rule mining algorithm to generate rules that cover users with a high estimated treatment effect. We describe the two-step procedure in detail.

First, two random forest Breiman (2001) are built to predict $P(Y_i = 1)$ in control and

treatment groups, respectively.

$$\hat{P}_i(1) = P(Y_i = 1|T_i = 1, \mathbf{X}_i), \tag{4.4}$$

$$\hat{P}_i(0) = P(Y_i = 1|T_i = 0, \mathbf{X}_i). \tag{4.5}$$

The difference in $\hat{P}_i(1)$ and $\hat{P}_i(0)$ is can be regarded as an estimate of the treatment effect for user i. In Foster et al., $\hat{P}_i(1) - \hat{P}_i(0)$ is then directly used as a classification or regression result to build a tree and get subgroups. However, this approach produces suboptimal solutions in our model since our goal is to find an optimal set of rules that *collectively* cover a subgroup with enhanced treatment effect. A single rule with high estimated treatment effect is not necessarily selected since it might work poorly with other selected rules. Nonetheless, this estimation is useful in narrowing down the search space, since it is unlikely that a collectively good set of rules contains a rule that is particularly bad. Therefore, we need only find rules with a high estimated treatment effect.

In order to find these rules, we define a binary variable to indicate if $\hat{P}_i(1) - \hat{P}_i(0)$ is above certain threshold:

$$Z_i = \mathbb{1}(\hat{P}_i(1) - \hat{P}_i(0) \geq \tau(S) + \delta). \tag{4.6}$$

The value of $\tau(S)$ is an estimate of the treatment effect over the entire population. δ represents the level of additional affect that is desired. δ affects the size of $\mathcal{A}$. A large δ results in a smaller rule space which makes the inference algorithm more efficient but may at the cost of leaving out potentially good rules.

Z_i is then used as the classification variable in discovering association rules from the original data. For our model, we are only interested in rules leading to a positive outcome, so we only mine frequent itemsets in the positive class ($Z_i = 1$). There are many off-the-shelf

techniques in literature that can mine rules sufficiently fast. In this particular work, we used FP-Growth Borgelt (2005). FP-Growth takes binary coded data where each column represents if the attribute in a data point satisfies a condition, either a continuous attribute is within a range (for example, age is between 10 to 20), or a categorical attribute equals a specific category (for example, gender is male). It then generates all rules that satisfy a maximum length and a minimum support that are pre-determined. Since the rules are only mined from the positive class, they cover users with larger estimated treatment effect than the general population.

These rules become the rule space $\mathcal{A}$ from where we will find an optimal set A^*.

4.2. A Theoretically Grounded Heuristic

Simulated annealing proposes a state by randomly choosing from neighbors. It takes tens of thousands of iterations to converge according to Letham et al. (2015), Wang and Rudin (2015) for a classification problem which is not practical for a large real system. Wang et al. (2016) proposed a more efficient algorithm that converges much faster but needs an evaluation on every neighbor. Their algorithm works well on supervised problems since every data point is labeled and it is easy to evaluate which neighbor is the best without computing a posterior. In our problem, however, Y_i and T_i are not the targets. The target is if an observation belongs to a subgroup which remains unknown during the search. Therefore, the objective value need to be computed to evaluate every neighboring model at every iteration, which is computationally impractical since evaluating $F(A, \mathbf{w})$ includes fitting a Bayesian logistic regression and computing priors. We would like to label some of the data and develop simpler heuristic to avoid computing $F(A, \mathbf{w})$.

A key observation in our model is that the selected rule set will try to "cover" users whose

$Y_i(1) - Y_i(0) = 1$. While we are unable to observe both potential outcomes to find out exactly which users to include, we are able to determine, by observing one of the potential outcomes and the treatment assignment, which users to exclude. We divide the outcome space into four regions based on Y_i and T_i: $S = \mathcal{E}_0 \cup \mathcal{E}_1 \cup \mathcal{U}_0 \cup \mathcal{U}_1$, where for group index $c \in \{0, 1\}$,

$$\mathcal{E}_c = \{(\mathbf{X}_i, Y_i, T_i) | T_i = c, Y_i = 1 - c\}, \tag{4.7}$$

$$\mathcal{U}_c = \{(\mathbf{X}_i, Y_i, T_i) | T_i = c, Y_i = c\}. \tag{4.8}$$

We observe that when a user i is in $\mathcal{U}_c$, it is *unknown* whether the treatment is effective without knowing the other potential outcome. It needs to be inferred by the model and cannot directly be labeled. However, if a user is in $\mathcal{E}_c$, the treatment effect is non-positive. $\mathcal{E}_0$ represents a group of users that already have good outcomes ($Y_i = 1$) in the control group so treatment is not necessary. $\mathcal{E}_1$ represents a group that shows bad outcomes ($Y_i = 0$) under treatment so the treatment is not helpful. Therefore a good subgroup should contain less or none of the $\mathcal{E}_c$ area. From this intuition we define an *Ideal Rule Set*.

Definition 1. *Given a data set* $S = \{(\mathbf{X}_i, Y_i, T_i)\}_{i=1}^n$, *an* Ideal Rule Set $\bar{A}$ *is defined as:*

$$\bar{A}(\mathbf{X}_i) = \mathbb{1}(\{\mathbf{X}_i, Y_i, T_i\} \in \mathcal{U}_0 \cup \mathcal{U}_1).$$

An ideal rule set "covers" only $\mathcal{U}_c$. We call such a rule set ideal because it achieves maximum conditional likelihood with prior on $\mathbf{w}$. We show in Theorem 1 that $\mathcal{B}(S; \bar{A}, \mathbf{w}_{\bar{A}})$ is the upper bound on the bayesian logistic regression on dataset S given any rule set A and its corresponding parameters $\mathbf{w}_A$, i.e.,

Theorem 1. $\forall A$. *Let* $\mathbf{w}_A = \max_{\mathbf{w}} \mathcal{B}(A, \mathbf{w})$ *and notate the elements in* $\mathbf{w}_A$ *as* $\mathbf{w}_A =$

$\{\mathbf{v}_A, \gamma_A^{(0)}, \gamma_1^{(0)}, \gamma_A^{(2)}\}$. *If* $\gamma_A^{(1)}, \gamma_A^{(2)} \geq 0$,

$$\mathcal{B}(A, \mathbf{w}_A) \leq \mathcal{B}(\bar{A}, \mathbf{w}_{\bar{A}}).$$

Proof. We notate elements in $\mathbf{w}_{\bar{A}}$ as $\mathbf{w}_{\bar{A}} = \{\mathbf{v}_{\bar{A}}, \gamma_{\bar{A}}^{(0)}, \gamma_{\bar{A}}^{(1)}, \gamma_{\bar{A}}^{(2)}\}$. Since $\bar{A}(\mathbf{X}_i) = 1$ when $\mathbf{X}_i \in \mathcal{U}_0 \cup \mathcal{U}_1$, we rewrite $\bar{A}(\mathbf{X}_i)$ as

$$\bar{A}(\mathbf{X}_i) = T_i Y_i + (1 - T_i)(1 - Y_i) \tag{4.9}$$

Expanding $\mathcal{B}(\bar{A}, \mathbf{w}_{\bar{A}})$ using formula (3.5) and plugging in (4.9) yields

$$\begin{aligned}\mathcal{B}(\bar{A}, \mathbf{w}_{\bar{A}}) = \log \text{Prior}(\mathbf{w}_{\bar{A}}) + \sum_{Y_i=1} \log \sigma\Big(\mathbf{v}_{\bar{A}}\mathbf{X}_i + \gamma_{\bar{A}}^{(0)} T_i + \\ \gamma_{\bar{A}}^{(1)} T_i\Big) + \sum_{Y_i=0} \log \left[1 - \sigma\Big(\mathbf{v}_{\bar{A}}\mathbf{X}_i + \gamma_{\bar{A}}^{(0)} T_i - \gamma_{\bar{A}}^{(2)}(1 - T_i)\Big)\right]\end{aligned} \tag{4.10}$$

We then upper bound the conditional likelihood of data S and the prior of parameters given any rule set A and $\mathbf{w}_A$.

$$\begin{aligned}\mathcal{B}(A, \mathbf{w}_A) &= \sum_{i=1}^{n} \log P(Y_i | \mathbf{X}_i, T_i; A, \mathbf{v}_A) + \log \text{Prior}(\mathbf{w}_A) \\ &= \log \text{Prior}(\mathbf{w}_A) + \sum_{Y_i=1} \log \sigma\Big(\mathbf{v}_A \mathbf{X}_i + \gamma_A^{(0)} T_i + \gamma_A^{(1)} T_i A(\mathbf{X}_i)\end{aligned}$$

$$
\begin{aligned}
&- \gamma_A^{(2)}(1-T_i)A(\mathbf{X}_i)\Big) + \sum_{Y_i=0} \log\Bigg[1-\sigma\Big(\mathbf{v}_A\mathbf{X}_i + \gamma_A^{(0)}T_i + \\
&\gamma_A^{(1)}T_iA(\mathbf{X}_i) - \gamma_A^{(2)}(1-T_i)A(\mathbf{X}_i)\Big)\Bigg] \\
\leq &\log \text{Prior}(\mathbf{w}_A) + \sum_{Y_i=1} \log\sigma\Big(\mathbf{v}_A\mathbf{X}_i + \gamma_A^{(0)}T_i + \gamma_A^{(1)}T_i\Big) + \\
&\sum_{Y_i=0} \log\Bigg[1-\sigma\Big(\mathbf{v}_A\mathbf{X}_i + \gamma_A^{(0)}T_i - \gamma_A^{(2)}(1-T_i)\Big)\Bigg] \qquad (4.11) \\
= &\mathcal{B}(\bar{A}, \mathbf{w}_A) \leq \mathcal{B}(\bar{A}, \mathbf{w}_{\bar{A}}) \qquad (4.12)
\end{aligned}
$$

(4.11) follows because $\sigma(x)$ increases monotinoically with x, and $\mathbf{v}_A\mathbf{X}_i+\gamma_A^{(0)}T_i+\gamma_A^{(1)}T_iA(\mathbf{X}_i)-\gamma_A^{(2)}(1-T_i)A(\mathbf{X}_i) \leq \mathbf{v}_A\mathbf{X}_i+\gamma_A^{(0)}T_i+\gamma_A^{(1)}T_iA(\mathbf{X}_i)$ and $\mathbf{v}_A\mathbf{X}_i+\gamma_A^{(0)}T_i+\gamma_A^{(1)}T_iA(\mathbf{X}_i)-\gamma_A^{(2)}(1-T_i)A(\mathbf{X}_i) \geq \mathbf{v}_A\mathbf{X}_i+\gamma_A^{(0)}T_i+\gamma_A^{(1)}T_iA(\mathbf{X}_i)-\gamma_A^{(2)}(1-T_i)$. (4.12) follows since $\mathbf{w}_{\bar{A}} = \max_{\mathbf{w}} \mathcal{B}(\bar{A}, \mathbf{w})$.

□

This theorem states that an Ideal Rule Set is an optimal solution to $F(A, \mathbf{w})$, if ignoring the prior probability Prior(A). However, this Theorem only provides a mathematically achievable upper bound on $\mathcal{B}(\cdot)$. It does not guarantee that an Ideal Rule Set is a MAP model since there might not exist a rule set that only covers examples in $\mathcal{U}_c$ at all. However, Theorem 1 provides illuminates on criteria for evaluating a rule set, that a good rule set needs to cover much of $\mathcal{U}_0 \cup \mathcal{U}_1$ and little of $\mathcal{E}_0 \cup \mathcal{E}_1$. We define *precision* of a rule set.

Definition 2. *Given a data set* $S = \{(\mathbf{X}_i, Y_i, T_i)\}_{i=1}^n$*, the precision of a rule set* A *is*

$$Q(A) = \frac{|S_A \cap (\mathcal{U}_0 \cup \mathcal{U}_1)|}{|\mathcal{U}_0 \cup \mathcal{U}_1|}. \qquad (4.13)$$

4.3. Exploration-with-Exploitation

Given metric (4.13), we are able to evaluate neighbors obtained by performing one of the actions, adding, removing or replacing. Define

$$\epsilon^{[t]} = \{(\mathbf{X}_i, Y_i, T_i) | \mathbf{X}_i \in \mathcal{E}_0 \cup \mathcal{E}_1, A^{[t]}(\mathbf{X}_i) = 1\}, \tag{4.14}$$

$$u^{[t]} = \{(\mathbf{X}_i, Y_i, T_i) | \mathbf{X}_i \in \mathcal{U}_0 \cup \mathcal{U}_1, A^{[t]}(\mathbf{X}_i) = 0\}. \tag{4.15}$$

To choose one action to perform from adding, removing and replacing, at iteration t, an example k is drawn uniformly from $\epsilon^{[t]} \cup u^{[t]}$. Let $\mathcal{R}_1(\mathbf{X}_k)$ represent a set of rules that $\mathbf{X}_k$ satisfies and $\mathcal{R}_0(\mathbf{X}_k)$ represent a set of rules that $\mathbf{X}_k$ does not satisfy. If $\mathbf{X}_k \in \epsilon^{[t]}$, it means $A^{[t]}$ covers wrong data and we then find a neighboring rule set that covers less, by removing or replacing a rule from $A^{[t]} \cap \mathcal{R}_0(X_k)$. If $\mathbf{X}_k \in u^{[t]}$, then as explained previously, it is not sure if X_k should or should not be covered. Therefore, the new rule set is proposed by randomly choosing from adding a rule from $\mathcal{R}_1(X_k)$, or removing or replacing a rule from $A^{[t]} \cap \mathcal{R}_0(\mathbf{X}_k)$.

After determining the best action, we choose a rule $z^{[t]}$ to perform the action on. We evaluate $Q(\cdot)$ on all neighbors produced by performing the selected action. Then a choice is made between exploration, choosing a random rule, and exploitation, choosing the best rule. We denote the probability of exploration as q. This randomness helps avoid local minima and helps the Markov Chain to converge to the global optima. We detail the three actions below.

- ADD: 1) With probability q, draw $z^{[t]}$ randomly from $\mathcal{R}_1(X_k)$; with probability $1 - q$,

$$z^{[t]} = \underset{a \in \mathcal{R}_1(X_k)}{\arg\max}\, Q(A^{[t]} \cup a).$$

  2) Then $A^{[t+1]} \leftarrow A^{[t]} \cup z$.

- REMOVE: 1) With probability q, draw $z^{[t]}$ randomly from $A^{[t]} \cap \mathcal{R}_0(X_k)$; with probability $1-q$,

$$z^{[t]} = \underset{a \in A^{[t]} \cap \mathcal{R}_0(X_k)}{\arg\max} \; Q(A^{[t]} \backslash a).$$

  2). Then $A^{[t+1]} \leftarrow A^{[t]} \backslash z$.

- REPLACE: 1) REMOVE, 2) ADD.

The proposal strategy assesses the current model and evaluates all neighbors in order to make sure that the selected action improves the current model, and the selected rule makes maximizes the improvement (coordinate descent). This is significantly more practically efficient than proposing moves uniformly at random.

Now we present the full algorithm in Algorithm 1.

5. EXPERIMENTS

We conducted a large-scale randomized mobile-based field experiment on WeChat Moments to identify peer influence across different products. Our goal is to identify subgroups defined by users, affiliated friends, and their relationship, where a social cue has enhanced treatment effect. We compared with state-of-the-art baseline methods. We analyzed the results and obtained useful insights in peer influence and social marketing context.

Figure 2: Experimental Control and Treatment

Algorithm 1 Search Algorithm

procedure CRS(S,H,T)
  $\mathcal{A} \leftarrow$ FP-Growth(S)
  $A^{[0]} \leftarrow \emptyset$
  $v^{[0]} = F(\emptyset, \mathbf{w}_\emptyset)$
  for $t = 0, ..., T$ **do**
    $\mathbf{X}_i \leftarrow$ an example randomly drawn from $\epsilon^{[t]} \cup u^{[t]}$
    if $\mathbf{X}_i \in \epsilon^{[t]}$ **then**
      $A^{[t+1]} = \begin{cases} \text{REMOVE}(A^{[t]}) \text{ with probability } \frac{1}{2} \\ \text{REPLACE}(A^{[t]}) \text{ with probability } \frac{1}{2} \end{cases}$
    else
      $A^{[t+1]} = \begin{cases} \text{REMOVE}(A^{[t]}) \text{ with probability } \frac{1}{3} \\ \text{ADD}(A^{[t]}) \text{ with probability } \frac{1}{3} \\ \text{REPLACE}(A^{[t]}) \text{ with probability } \frac{1}{3} \end{cases}$
    end if
    $\mathbf{w}_{A^{[t+1]}} = \arg\max_{\mathbf{w}} F(A^{[t+1]}, \mathbf{w})$
    $v^{[t+1]} = \max\{v^{[t]}, F(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}})\}$
    if $F(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}}) \geq v^{[t]}$ **then**
      $A^*, \mathbf{w}_{A^*} = A^{[t+1]}, \mathbf{w}_{A^{[t+1]}}$
    end if
    $(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}}) = (A_t, \mathbf{w}_{A^{[t]}})$ with probability $\exp\left(\frac{F(A^{[t+1]}, \mathbf{w}_{A^{[t+1]}}) - F(A^{[t]}, \mathbf{w}_{A^{[t]}})}{T(t)}\right)$
  end for
  return $A^*, \mathbf{w}_{A^*}$
end procedure

5.1. Experimental Design

The experiment was conducted over a 21-day period, during which 28,668,980 users and 99 ads participated in the experiment. The peer influence in our experiments refers to the effects of a social cue, (i.e.,like), representing *first-degree friends'* endorsements of an ad on users' response (i.e.,clicking) to the ad.

During the experiment, as users received a new ad, they were randomly assigned into two groups: without any social cue (control) and with maximum one like (treatment) (See Figure 2). Every ad stayed in users' newsfeed for maximum 48 hours. After 48 hours, the old ad would disappear and a new ad was received. In this way, users saw only one

experiment-related ad at one time. Randomization would happen again whenever users received a new ad. Users could be in a different treatment group or outside the experiment for different ads. The randomization was at ad-user level.

Our experimental design carefully avoided many known sources of bias in influence identification and network experiments. First, it eliminated bias created by homophily through randomly assigning the social cues such that observed and unobserved attributes of users are equally distributed across treatment groups through. Second, the randomization controls for external confounding factors, because users are equally likely to be exposed to external stimuli that affect adoptions across different treatment groups. Third, all the ads involved in the experiments were new and distinctive, so there were no external sources for users to get access to the ads before or outside the experiment. Fourth, "likes" from different users are shown in identical forms in Moments (See Figure 2), eliminating the heterogeneity of unmeasurable characteristics of social cues. Fifth, this method guarantees the stable unit treatment value assumption (SUTVA)(Rubin 1990). Users would not receive different treatments from different ads at the same time, because of one-ad limit within 48 hours. Sixth, since randomization (re)occurred for every 48 hours, it was very unlikely that users noticed they were being treated during the experiments. The treatment effects, therefore were not confounded by unmeasurable psychological factors of users, who suspected or realized they were in an experiment.

Assignment to treatment and control groups was random, with no significant mean or distributional difference between subjects of different groups, economically or statistically, in terms of their age, gender, network degree (number of WeChat friends), and level of WeChat Moments activity (log-in days) in November of 2015, the month right before the experiment. (t-test, mean difference/mean $< 0.5\%$, $p > 0.1$.)[1] These evidences taken together confirm

[1]Hypothesis testing : μ_0=μ_1 would always be rejected in a very large dataset with extremely high statistical

the integrity of the randomization procedure.

5.2. Data Description

Since the maximum number of displayed social cues is limited by the number of peers who had liked the ads (i.e. affiliated peers). We can manipulate only the maximum number of social cues rather than the exact number of them. In order to get the treatment of exact number of social cue, we filtered the data on the condition that ONLY one friend had liked the ads before the users saw the ad at the first time and got 2,578,664 observations in total coming from two groups: control group with no displayed social cue and treatment group with one displayed social cue.

The dataset for our analysis consists of 2,578,664 observations, each corresponding to a user-friend-ad triples constructed from 2,459,428 users and 93 advertisements. These ads were categorized into 22 categories by WeChat ads department. The control group contains 1,282,867 observations and the treatment group contains contains 1,295,797 observations. An observation consists of a binary target variable Y_i, indicating whether the user clicked the ad, treatment assignment T_i, indicating whether the user saw the like by the affiliated friend, and a covariate vector $\mathbf{X}_i$. $\mathbf{X}_i$ includes features regarding user i, his/her affiliated friend who liked the advertisement and their relationship information. The features are categorized into demographics, behavior, network, tie strength and social embeddedness (See Table 1 for a complete list of features.) We split the data by product category and obtained 22 datasets in total.

power, since there is almost no two sample with exactly the same mean.We thus test the hypothesis: μ_1-$\mu_0 < 0.5\% * \mu_0$ instead.

Table 1: Datasets and feature list

Types of products and sizes of corresponding datasets	Features of user i, his/her friend and their dyadic relationships
1. Baby foods (20,101) 2. Beverage (297,260) 3. Cars (660,944) 4. Clothes (263,043) 5. Cosmetics (169,005) 6. Daily products (26,263) 7. Elec appliance (119,759) 8. Financial service (13,870) 9. Foods (133,290) 10. Home furnishing (4,908) 11. Insurance (30,830) 12. Jewelry (240,442) 13. Mobile games (105,957) 14. Real estate (48,032) 15. Smart phones (130,421) 16. Tires (123,512) 17. Tourist Spots (48,456) 18. TV shows (37,701) 19. Web portals (32,162) 20. Web of service (9,414) 21. Web of tourism (57,351) 22. Wine (3,412)	**User and the affiliated friend** Demographic Characteristics: Age Gender City Behavioral Characteristics: # of messages sent # of messages received # of posts in moments # of endorsements sent # of endorsements received Network Characteristics: # of friends **Relationship:** Tie Strength: # of msg sent to the friend # of msg received from the friend # of endrs sent to the friend # of endrs received from the friend Social embeddedness: # of mutual friends

5.3. Experimental Results

5.3.1. Baselines

We compared our model to two state-of-the-art subgroup identification methods: Differential Effect Search and Virtual Twins (VT). Both of which generate interpretable results in the form of rules, like ours. SIDES is a greedy algorithm that recursively partitions a database into two subgroups at each parent group based on certain criteria. We implemented the algorithm with three different splitting criteria, one was based on maximizing the differential effect between the two child subgroups, denoted as SIDES1 in our paper, one was based on maximizing the treatment effect in at least one of the two child subgroups, denoted as SIDES2, and the third one was a hybrid of the two, denoted as SIDES3. Virtual twins first builds different models to predict responses in control and treatment groups, respectively, then uses the response as the outcome in a classification or regression tree. We implemented a VT model with classification tree, denoted as VT(C), and a VT model with regression tree, denoted as VT(R).

5.3.2. Evaluation metric

Given a subgroup S_A, we define the *lift* of this subgroup as the ratio between the treatment effect on subgroup S_A and the average treatment effect on the population. [2]

$$\text{lift}(S_A) = \frac{\tau(S_A)}{\tau(S)}. \tag{5.1}$$

$\tau(S_A)$ is computed using formula (2.5). All $\tau(S)$ were positive in our dataset.

[2]The magnitude of the treatment effect is considered confidential information by the company and cannot be disclosed in the paper. So we normalize it with the treatment effect on the entire population and only report this ratio.

5.3.3. Performance comparison

We applied CRS and baseline methods on the 22 datasets. In all experiments, we set the rule length to be 3 for all methods, i.e., rules in CRS and SIDES will have at most 3 conditions in a rule and trees in VT will have a maximum depth of 3. Every dataset was split into 60% training and 40% testing. For CRS models on all datasets, We set the minimum support of a rule to be 5% and the variance matrix to be an identity matrix. The expected means were set to 0 for all coefficients except the $\gamma^{(1)}$ and $\gamma^{(2)}$ whose expected means were set to 1 since only positive coefficients are effective in finding the right subgroups.

We report the lifts for all competing models in Table 2.

Each row in Table 2 corresponds to one type of product ads. The highest lift of each dataset is written in bold text and about half of the time our model achieved the highest lift. Negative lifts are underscored. Since $\tau(S)$ is positive for all datasets, a negative lift means that the model failed to identify a subgroup with enhanced treatment effect and ended up with a subgroup with a negative effect.

Negative lifts happened to most baseline methods at least once but none happened to CRS. This was because the baseline methods are greedy algorithms that depend on local information to partition the data. But a local optimal split might not benefit the overall subgroup characterization. It might even, like the few examples in our data, hurt the final result. Our model, on the other hand, is obtained by maximizing a global objective. Besides, baseline methods do not regularize model and have the risk of overfitting whereas our model uses a prior to favor simpler models.

To visualize the overall performance on all datasets, we plot box plots of the ranking of all methods in Figure 3. Our model achieved best overall rank compared to other models.

Table 2: Lifts for CRS and baseline methods on datasets of different categories of product advertisements. All models have maximum rule length 3.

	CRS	SIDES1	SIDES2	SIDES3	VT(R)	VT(C)
Baby foods	1.28	0.86	0.00	0.00	-0.60	**2.39**
Beverage	**2.04**	1.28	1.64	1.32	1.90	1.10
Cars	**2.59**	1.02	1.09	0.80	2.88	1.20
Clothes	**3.03**	1.21	0.00	0.66	2.79	1.56
Cosmetics	1.67	1.09	0.00	1.93	0.00	**2.83**
Daily products	2.75	1.66	1.66	0.68	**3.73**	0.00
Elec appliance	1.76	0.79	0.79	1.76	**2.16**	1.63
Financial service	1.47	0.92	1.57	1.25	-2.94	**2.19**
Foods	**2.53**	1.28	1.28	1.12	0.90	0.80
Home furnishing	**1.42**	1.24	1.24	0.00	0.00	1.05
Insurance	**2.23**	1.29	0.41	-0.89	3.67	1.00
Jewelry	1.88	1.00	0.90	1.76	1.12	**2.24**
Mobile games	**1.48**	1.38	1.38	0.20	1.38	1.04
Real estate	**2.69**	1.18	-0.89	1.81	1.48	1.09
Smart phones	2.48	1.25	1.49	0.48	**2.79**	1.38
Tires	2.52	1.35	1.35	**4.08**	1.79	2.29
Tourist Spots	1.43	1.12	1.30	0.72	0.00	**1.53**
TV shows	5.1	0.16	5.58	**7.44**	3.53	2.36
Web portals	3.34	1.43	2.33	**3.97**	3.16	1.00
Web of service	4.51	**13.89**	10.07	0.00	-30.22	-0.15
Web of tourism	**1.26**	1.18	0.00	0.27	0.97	0.62
Wine	**1.98**	0.00	0.00	1.37	1.80	0.00

5.3.4. Case study

It is interesting to look at some of the specific results: in what subgroups peer influence has stronger effects in ads adoptions of a particular product category? Let us first take clothes as an example. The lift for clothes is 3.03, indicating that the subgroup treatment effects are 3.03 times as large as the average treatment effects on clothes test data. The output rule set contains one rule, which is:

[User and the affiliated friend messaged each other in the past month] AND [# of user's friends $<$ 25% quantile] AND [User's gender is male]

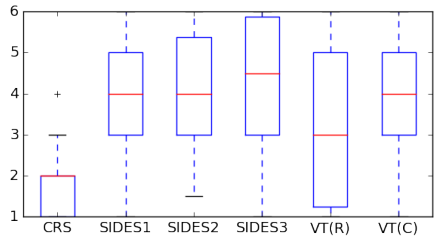

Figure 3: Rankings of lifts on datasets of 22 categories of product ads reported in Table 2.

This refers to a subgroup of males whose number of friends is in the lowest 25% of the sample and had sent messages to the affiliated friend in the month right before he saw the ad. This subgroup of males is most susceptible to this affiliated friend's influence mediated by the social cue displayed on the ad. This is easy to explain. First of all, individuals are more likely to be influenced when they do not know much about something (most men are not expert in clothes) are more likely to be influenced for things they do not know. Second of all, fewer friends means the person will have higher chance to see this ad. An ad is more likely to be neglected if the user has too many friends and sees too many posts every day.

The CRS obtained for tourist spot is

[# of mutual friends $\leq$ 50% quantile] AND [# of user's friends $<$ 25% quantile]

This refers to subgroups in which users whose friend number is in the lowest 25% of the sample and the number of mutual friends between users and their affiliated friend is in the lowest 50% of the sample, are associated with the greatest effects of peer influence. This result is counterintuitive in that social embeddedness represented by the number of mutual friends was shown to positively affect peer influence on averageAral and Walker (2014). This implies that the same factors, such as social embeddedness, may have different effects in identifying the subgroups with the greatest treatment effects from estimating the moderating effects.

The estimated lift for the above model is 1.43, which is much smaller than that for clothes. This suggests that identifying the most effective subgroup for tourist spot is not as useful as that for clothes, in terms of increasing the treatment effects.

5.3.5. Subgroup Effects by Product Category

Figure 4 displays treatment effects of the subgroups, in which a social cue has enhanced effects on users' clicking to the ads and compares them with the average treatment effects for each product category. Three main observations arise from consideration of the results in Figure 4. First, consistent with the results in Table 2, the enhanced subgroup treatment effects are always larger than average treatment effects in all the product categories. Second, treatment effects are very heterogeneous across 22 product categories for both of average treatment effects and enhanced subgroup treatment effects. This strongly indicates that peer influence has heterogeneous effects on the ads adoptions of different product categories. Web portals are associated with the largest and website of life service are associated with the smallest enhanced subgroup treatment effect. Wine has the largest and website of life service has the smallest average treatment effect. Third, we observed that enhanced subgroup effect can be very different from average treatment effect. For example, TV show is associated with a large enhanced subgroup treatment effect but has very small average treatment effect.

6. CONCLUSION

We proposed a method for identifying subgroups with enhanced treatment effect. Our model captures a subgroup with a set of rules. We constructed a Bayesian framework for learning a rule set that simultaneously considers the simplicity and performance of a model, and developed an inference algorithm that applies various strategies to improve search efficiency.

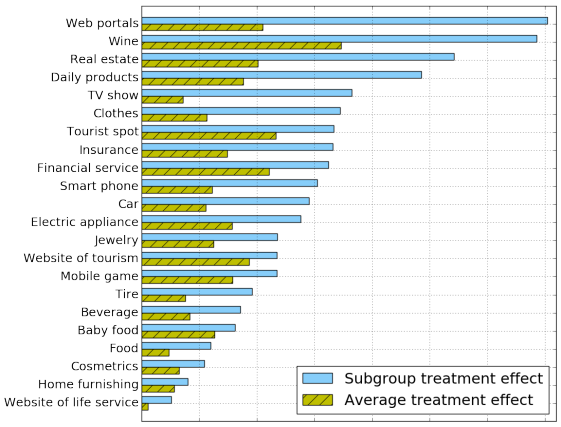

Figure 4: Treatment Effects by Product Category

We conducted a large-scale real world experiment on a social network platform: WeChat. Our analysis involves roughly 2.5 million observations. We applied our model and two state-of-the-art methods to the data set. Results show that our model outperformed the baselines and provided insightful discoveries regarding influence, susceptibility, and dyadic relationships.

This paper also contributes to the literature of peer influence and has important implications for social advertising. First, this is the first paper that identified the subgroups with enhanced peer influence in social advertisements, considering influence, susceptibility and effects of dyadic relationships. Second, this is among the first papers that show the heterogeneous effects of peer influence in product decisions of different categories, in terms of both average treatment effects and enhanced subgroup treatment effects (Huang et al. 2016). Third, our results provided the first evidence that enhanced subgroup treatment effects can be very different from average treatment effects in peer influence across different product categories. Prior research measured peer influence only using average treatment effects. Our study provided a different angle of understanding the degree of peer influence by subgroup treatment effects. Finally, our method provides useful insights for advertisers to increase

the effectiveness of peer influence in propagating new products in social networks, through targeting specific subgroups with enhanced peer influence. We also presented and compared the effectiveness of our method among the ads of different product categories.

References

S Aral. Commentary-identifying social influence: A comment on opinion leadership and social contagion in new product diffusion. *Marketing Science*, 30(2):217–223, 2011.

S. Aral and D. Walker. Creating Social Contagion Through Viral Product Design: A Randomized Trial of Peer Influence in Networks, 2011.

Sinan Aral and Dylan Walker. Tie Strength, Embeddedness, and Social Influence: A Large-Scale Networked Experiment. *Management Science*, 60(6):1352–1370, 2014. ISSN 0025-1909. doi: 10.1287/mnsc.2014.1936.

E Bakshy, D Eckles, R Yan, and I Rosenn. Social influence in social advertising: evidence from field experiments. *Proceedings of the 13th ACM Conference on Electronic Commerce*, 1(212): 146–161, 2012.

Robert M. Bond, Christopher J. Fariss, Jason J. Jones, Adam D. I. Kramer, Cameron Marlow, Jaime E. Settle, and James H. Fowler. A 61-million-person experiment in social influence and political mobilization. *Nature*, 489(7415):295–298, 2012. ISSN 0028-0836. doi: 10.1038/nature11421.

Christian Borgelt. An implementation of the fp-growth algorithm. In *Proc of the 1st interl workshop on open source data mining: frequent pattern mining implementations*, pages 1–5. ACM, 2005.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Jacob Cohen, Patricia Cohen, Stephen G West, and Leona S Aiken. *Applied multiple regression/correlation analysis for the behavioral sciences*. Routledge, 2013.

Jared C. Foster, Jeremy M. G. Taylor, and Stephen J. Ruberg. Subgroup identification from randomized clinical trial data. 30(24):2867–2880. ISSN 1097-0258. doi: 10.1002/sim.4322.

Shan Huang, Sinan Aral, Y. Yu (Jeffrey) Hu, and Erik Brynjolfsson. Social Influence in Search and Experience Goods. In *Conference on Digital Experimentation (CODE)*, pages 1–24, Boston, 2016.

Jason J Jones, Robert M Bond, Eytan Bakshy, Dean Eckles, and James H Fowler. Social Influence and Political Mobilization : Further Evidence from a Randomized Experiment in the 2012 U . S . Presidential Election. 2013.

Benjamin Letham, Cynthia Rudin, Tyler H. McCormick, and David Madigan. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *Annals of Applied Statistics*, 2015. accepted with minor revision.

Charles F. Manski. Identification of Endogenous Social Effects: The Reflection Problem. *The Review of Economic Studies*, 60(3):531, 1993.

Lev Muchnik, Sinan Aral, and Sean J Taylor. Social influence bias: a randomized experiment. *Science (New York, N.Y.)*, 341(6146):647–51, 2013.

Paul R Rosenbaum and Donald B Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

Peter M Rothwell. Subgroup analysis in randomised controlled trials: importance, indications, and interpretation. *The Lancet*, 365(9454):176–186, 2005.

Donald B Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688, 1974.

Donald B. Rubin. Comment: Neyman (1923) and Causal Inference in Experiments and Observational Studies. *Statistical Science*, 5(4):472–480, 1990.

Cosma Rohilla Shalizi and Andrew Thomas. Homophily and Contagion are generically confounded in observational social network studies. *Rociological Methods & Research*, 40(2):211–239, 2011.

Xiaogang Su, Chih-Ling Tsai, Hansheng Wang, David M Nickerson, and Bogong Li. Subgroup analysis via recursive partitioning. *Journal of Machine Learning Research*, 10(Feb):141–158, 2009.

C E Tucker. Social Advertising. 2012.

Stefan Wager and Susan Athey. Estimation and inference of heterogeneous treatment effects using random forests. *arXiv preprint arXiv:1510.04342*, 2015.

Fulton Wang and Cynthia Rudin. Falling rule lists. In *Proceedings of Artificial Intelligence and Statistics (AISTATS)*, 2015.

Tong Wang, Cynthia Rudin, Velez-Doshi Finale, YImin Liu, Erica Klampfl, and Perry MacNeille. Bayesian rule sets for interpretable classification. In *The IEEE International Conference on Data Mining*, 2016.