Social Network Classification of Ad Responses

Guduguntla Vamshi
North Carolina State University,Raleigh,NC

Abstract

A social network classification model to predict the ad response behaviour is presented.The model demonstrates the idea of k-nearest-neighbour classification of a given user using Depth First Search and Floyd-Warshall algorithms.We observe how the social network structure would influence the ad click-through.

1 Introduction

The goal of this paper is to develop a statistical model to predict which website users will click on an ad. The simulated data include information on 1,000 users, and 100 ads, with the network structure of the users.

2 Constructing a Social Stucture

The data set had the ad impression, click-through information for each individual user. Since different ads are shown to different users, an individual social structure is constructed for a given user. Starting, I have used the Depth First Search algorithm to retrieve the vertex coverage i.e., one starts at the root and exploring as far as possible along each branch before backtracking.

Input: A graph G and a vertex v of G Output: All vertices reachable from v labeled as discovered

The psudo-code for the Depth First search algorithm looks like the following :

```
procedure DFS(G,v):
     label v as discovered
     for all edges from v to w
                    in G.adjacentEdges(v) do
            if vertex w is not labeled
                        as discovered then
                recursively call DFS(G,w)
```

3 Mapping the similarity between users

3.1 Scaling the attributes between users

The data set has user-attributes such as age,gender,income and education.The personal attributes and the degree of connection are scaled to a standardized scale using the population maximum and minimum of the attributes.

```
for each attribute in population:
    scaled_val = {val-min(val)}/{max(val)-min(val)}
```

3.2 Algorithm for degree of connection

The degree of connection between the users is computed using Floyd-Warshall algorithm. The psudo-code to find the degree of connection is given :

```
let dist be a |V|  |V| distances set to
for each vertex v
   dist[v][v]  0
for each edge (u,v)
   dist[u][v]  w(u,v) #the weight = 1
for k from 1 to |V|
   for i from 1 to |V|
      for j from 1 to |V|
         if dist[i][j] > dist[i][k]+dist[k][j]
             dist[i][j]  dist[i][k]+dist[k][j]
        end if
```

These five attributes are used to calculate the similarity between the users using Euclidean Distance.

4 Pruning the social structure

The initial social structure constructed for each individual users using DFS algorithm is a maximal cover of the vertices. We need to prune the each individual structure

so as to find the optimal structure to predict the ad click through.

For pruning purposes, I have made use of the ad impression information. For any given individual social structure, the elements(vertex) with no ad impressions are pruned. This ensures that the elements with known ad impression for a particular ad served best predicts the click-through for the user.

5 Probability of an ad-click

To find the probability of ad-click for a particular user, the social structure is inferred.

The core intuition behind the modeling of probability relies on the overall click through rate of the social structure. For instance, you may belong to a social group who are say less interested in click the ads. Others may enjoy exploring the content in the advertisements.

5.1 Group overall clicks rate

So, the probability is modeled based on your friends' click-through rate. Out of the k friends you have, the overall-click rate for the group is given by:

```
Overall click rate of group (OCRG) =
(i=1 to k) clicks-rate_i for an individual/k
```

5.2 Classification

Given an advertisement id for a particular user, the rule to predict whether click happens is given by:

```
Pr = #clicks by friends who saw the ad/
                    #total number of views

User clicks on ad) =
1 - if Pr > OCRG
0 - if Pr < OCRG
```

References

Notes

[1]Remember to use endnotes, not footnotes!