

Association Models for Web Mining

Modelli Associativi per il Web Mining

Paolo Giudici

Dipartimento di Economia Politica e Metodi Quantitativi
Università di Pavia
giudici@unipv.it

Riassunto: La presentazione riguarda i metodi statistici per l'analisi dei dati provenienti dalle sessioni di visita ad un sito web, tipicamente contenuti in un logfile. Nell'analisi dei dati sulle sequenze di visita, si è solitamente interessati a determinare le sequenze di pagine più ricorrenti. Nel lavoro, dopo avere descritto la struttura dei dati disponibili, nonché la preliminare analisi esplorativa, viene descritta la metodologia delle regole associative che si particolareggia, nel caso del web mining, all'analisi delle sequenze. Viene quindi proposta una classe di modelli statistici basati sulle catene di Markov, che permettono di irrobustire l'analisi dal punto di vista inferenziale, con particolare riferimento alla previsione delle sequenze più ricorrenti. Entrambe le metodologie verranno applicate e confrontate ad una base di dati transazionale reale, riguardante il flusso delle visite ad un sito di e-commerce.

Keywords: Association rules, Data Mining, Markov chain models

1. Introduction

The aim of this paper is to show how the information, concerning the order in which the pages of a web site are visited, can be profitably used to predict the visit behaviour at the site. Usually every click corresponds to the visualization of a web page. Thus, a web clickstream defines the sequence of the web pages requested by a user. Such a sequence identifies a user session.

Typically, a web mining analysis only concentrates on the part of each user session concerning the access at one specific site. The set of the pages seen in a user session, on a determinate site, is usually referred to with the term *server session* or, more simply, *visit*.

Our objective here is to show how web clickstream data can be used to understand the most likely paths of navigation in a web site, with the aim of predicting, possibly on-line, which pages will be seen, having seen a specific path of pages in the past. Such analysis can be very useful to understand, for instance, what is the probability of seeing a page of interest (such as the buying page in an e-commerce site) coming from a specified page. Or what is the probability of entering or (exiting) the web site from any particular page.

The data set that we consider to exemplify our methodology is the result of the elaboration of a log file concerning a site of e-commerce. The source of the data cannot be specified; however it is the website of a company that sells hardware and software products; it will be referred to as "a webshop". The accesses to the website

have been registered in a logfile for a period of about two years, since 30 september 1997 to 30 june 1999. The logfile has then been processed to produce a dataset, named "sequences". Such dataset contains the user id (*c_value*), a variable with the date and the instant the visitor has linked to a specific page (*c_time*) and the web page seen (*c_caller*). Table 1 reports a very small extract of the available dataset , corresponding to one visit.

Table 1: extract of the considered dataset

<i>c_value</i>	<i>c_time</i>	<i>c_caller</i>
70ee683a6df...	14OCT97:11:09:01	home
70ee683a6df...	14OCT97:11:09:08	catalog
70ee683a6df...	14OCT97:11:09:14	program
70ee683a6df...	14OCT97:11:09:23	product
70ee683a6df...	14OCT97:11:09:24	program

Table 1 describes that the visitor corresponding to the identifier (cookie) 70ee683a6df... has entered the site on the fourteenth of october, 1997, at 11:09:01, and has visited, in sequence, the pages “home”, “catalog”, “program”, “product”, “program”, leaving the website at 11:09:24.

The whole data set contains 250711 observations, each corresponding to a click, that describe the navigation paths of 22527 visitors among the 36 pages which compose the site of the webshop. The visitors are taken as unique, that is, no visitors appears in the dataset with more than one visit. On the other hand, we remark that a page can occur more than once in the same session.

This data set is a noticeable example of a transactional dataset. Another important examples of transactional dataset are those employed in market basket analysis. However, differently from what occurs with market basket data, here the order in which the pages are seen is important, and it is indeed the very objective of the analysis to understand the patterns that originate it.

So far we have insisted that the statistical units of the available transactional dataset are clicks rather than visit (here coinciding with visitors). We now describe the statistical variables that will be considered in the analysis. There are 36 binary variables, that describe whether each of the 36 webpages that constitute the web site is being clicked or not. For lack of space, we do not describe such pages, whose denomination we believe typically self-explanatory, and refer the reader interested in a detailed description to Giudici (2001, chapter 15).

2. Exploratory data analysis

Our main aim is to discover the most frequent association patterns of visit between the 36 webpages that form the website.

In order to obtain valid conclusions, the considered data have to be homogeneous. In order to assess whether the dataset is homogeneous we have deemed necessary to run first an exploratory analysis on the “visitors” dataset, which can be desumed from the transactional dataset in Table 1 by building a data matrix with the visitors in the rows and the web pages binary variables in the columns. To such a reduced

dataset we have appended additional variables, desumable from the logfile, such as the length of each visit (length), the time of the visit (time), the number of clicks (clicks) per visit and whether the visit has led to a purchase of an (e-commerce) product (purchase). Such additional variables have been used for exploratory (indeed pre-processing) purposes.

For instance, we have decided to proceed with outliers removal, and eliminate all observations above the 99-th percentile of the distribution of the variables clicks and length, which indeed behave similarly. On the other hand, we have decided not to remove observations possibly outlying with respect to the time variable. This because of the nature of the variable itself as well as of the observed distribution.

The resulting visitors dataset contains 22152 observations, in place of the initial 22527.

In consideration of the heterogeneous nature of the navigators of the site, confirmed by the exploratory stage, we have then decided to perform a cluster analysis, in order to find homogeneous clusters of behaviours. We remark that our primary goal here is not cluster analysis per se and, thus, cluster analysis can be seen as somehow exploratory or, better, preliminary, to the association models that we are seeking.

As clustering variables we have considered the three quantitative variables start, length, clicks, as well as the binary variable purchase, which can all be seen as instrumental to our objective, of understanding navigation patterns. For the cluster analysis, we have decided to first run a hierarchical method, to find the optimal number of groups, and then a non-hierarchical method, to allocate observations in the determined number of groups. As distance function we have considered the Euclidean distance; as hierarchical method the method of Ward (after some comparative experiments). Finally, to allocate observations we have chosen the K-means non-hierarchical method.

The results obtained from cluster analysis confirm heterogeneity of behaviours, and allocate the visitors in four distinct clusters. To the purpose of finding navigation patterns, we have decided to concentrate the analysis on only one cluster. This choice is obviously subjective but, nevertheless, has two important peculiarities. First, the visitors in the chosen cluster stay connected for long time, and visit many pages. Both these occurrences allow to better explore the navigation sequences between the web pages. Second, the chosen cluster has a high probability of purchase; it seems important to consider the typical navigation pattern of a group of high purchasers.

Therefore, in the following sections, we shall consider a reduced dataset, corresponding to examined cluster of visitors, with 1240 visits and 21889 clicks.

Before starting with the modelling results we remark that we have appended, in the cluster dataset, a start_session and an end_session page, respectively, before the first page and the last page visited in each session. Obviously, such pages are fictitious, and are not random, but certain. However, they will serve to find the most common entrance and exit pages.

Finally, we point out to the reader that an analysis of the entire dataset described here, and not of one cluster alone, has been performed by Blanc and Giudici (2001).

3. Sequence rules

In order to achieve our goal, that is, to find the most frequent patterns of visit, we shall compare two web mining models: sequence rules, which belong to the class of local data mining methods known as association rules; and markov chain models, which can be seen, on the other hand, as (global) predictive data mining methods. We first briefly describe what a sequence rule is. For more details the reader can consult a recent text on data mining, such as Han and Kamber (2001) or, from a more statistical viewpoint, Hand et al (2001), Hastie et al. (2001) and Giudici (2001).

An *association rule* is a statement between two sets of binary variables (itemsets) A and B, that can be written in the form $A \rightarrow B$, to be interpreted as a logical statement: *if A, then B*. If the rule is ordered in time we have a sequence rule and, in this case A preceeds B.

In web clickstream analysis, a sequence rule is typically *indirect*: namely, between the visit of page A and the visit of page B other pages can be seen. On the other hand, in a *direct* sequence rule A and B are seen consecutively.

A sequence rule model is, essentially, an algorithm that searches for the most interesting rules in a database. The most common of such algorithms is the Apriori model, introduced by Agrawal et al. (1995). In order to find a set of rules, statistical measures of “interestingness” have to be specified. The measures more commonly used in web mining to evaluate the importance of a sequence rule are the indexes of support and confidence.

Consider a sequence $A \rightarrow B$ and indicate as $N_{A \rightarrow B}$ the number of visits which appear in such sequence, at least once. Let N be the total number of the server sessions. The *support* for the rule $A \rightarrow B$ is obtained dividing the number of server sessions which satisfy the rule by the total number of server sessions:

$$support\{A \rightarrow B\} = \frac{N_{A \rightarrow B}}{N}$$

Therefore, it is a relative frequency that indicates the percentage of the users that have visited in succession the two pages. In presence of a high number of visits, as it is usually the case, it is possible to state that the support for the rule approximates the probability an user session contains the two pages in sequence:

$$support\{A \rightarrow B\} = Pr\{A \rightarrow B\}$$

The confidence for the rule $A \rightarrow B$ instead is obtained dividing the number of server sessions which satisfy the rule by the number of sessions containing the page A:

$$confidence\{A \rightarrow B\} = \frac{N_{A \rightarrow B}}{N_A} = \frac{\frac{N_{A \rightarrow B}}{N}}{\frac{N_A}{N}} = \frac{support\{A \rightarrow B\}}{support\{A\}}$$

Therefore, the confidence approximates the conditional probability that in a server session in which has been seen the page A is subsequently required page B.

What just said has been referred to itemsets A and B containing one page each; however, each itemset can contain more than one page, and the previous definition carries through. The order of a sequence is the total number of pages involved in the rule. For instance, the rules discussed so far are sequences of order two.

Let us apply the previous concepts to our web clickstream data. Sequence rules are to be calculated on the transactional dataset corresponding to the chosen cluster, in which visits to the pages are ordered by the time of the visit, recorded in the logfile. In order to find a sequence model, we have applied the Apriori algorithm implemented in SAS Enterprise Miner, setting a support threshold equal to $0.05 \times \text{support}(\text{mode})$, where $\text{support}(\text{mode})$ indicates the support of the most seen page, among the 38 considered. This means that the search of the most interesting rules will be based on the support index, and all itemsets with support lower than the specified threshold will not be considered.

The output of the algorithm can be visualised in terms of the sequence rules with the highest interestingness, as measured, for instance, by the support and confidence of the rules that are selected.

Table 2 describes the most likely sequences of order two, for the considered cluster dataset. In other words, we search a pattern of sequences, characterised by maximum support among all sequences of order two.

Table 2: The most frequent sequences of order two

	Support[%]	Confidence[%]	Transaction Count	Rule
1	100.00	100.00	1240	start_session ==> end_session
2	88.79	88.79	1101	start_session ==> product
3	88.79	100.00	1101	product ==> end_session
4	77.66	77.66	963	start_session ==> program
5	77.66	100.00	963	program ==> end_session
6	73.39	94.50	910	program ==> product
7	64.68	72.84	802	product ==> product
8	62.10	62.10	770	start_session ==> home
9	62.10	100.00	770	home ==> end_session
10	56.13	56.13	696	start_session ==> catalog
11	56.13	100.00	696	catalog ==> end_session
12	53.23	53.23	660	start_session ==> login
13	53.23	100.00	660	login ==> end_session
14	52.26	93.10	648	catalog ==> program
15	51.37	91.52	637	catalog ==> product
16	50.32	81.04	624	home ==> product
17	49.92	56.22	619	product ==> program
18	49.76	100.00	617	addcart ==> end_session
19	49.76	49.76	617	start_session ==> addcart
20	48.71	54.86	604	product ==> addcart
21	48.39	48.39	600	start_session ==> logpost
22	48.39	100.00	600	logpost ==> end_session
23	48.15	61.99	597	program ==> program
24	45.89	73.90	569	home ==> program
25	44.76	44.76	555	start_session ==> p_info
26	44.76	100.00	555	p_info ==> end_session

From Table 2 notice that most of the sequences involve start_session and /or end_session, and this is obvious, as each session will contain both of them, and we are considering indirect rules. Among the other sequences, program → product is the sequence with the highest frequency, followed by product → product. Notice that the latter cannot be detected by association rules. Table 2 establishes that the estimated probability that the product page is seen twice at least once in a session is equal to 64.68%.

We have then searched the most supported indirect sequences, among those of any order . It turns out that the most frequent sequences are, besides those already examined previously, those that have to do with visiting the most seen pages in the cluster, namely start_session → product → end_session and start _session → program → end_session.

We now turn to search the most likely direct sequences. While direct sequences of order two can be easily obtained from the software, this is not the case for higher order ones. In any case, Table 3 reports the most likely order two direct sequences.

Table 3: Most frequent direct sequences

	Support(%)	Confidence(%)	Transaction Count	Rule
1	60.65	78.09	752	program ==> product
2	46.21	82.33	573	catalog ==> program
3	45.81	45.81	568	start_session ==> home
4	35.73	67.12	443	login ==> logpost
5	31.77	63.86	394	addcart ==> freeze
6	25.56	28.79	317	product ==> addcart
7	25.24	28.43	313	product ==> p_info
8	24.68	67.85	306	pay_req ==> pay_res
9	24.19	54.55	300	freeze ==> pay_req
10	19.68	43.96	244	p_info ==> product
11	19.52	61.58	242	register ==> regpost
12	17.02	17.02	211	start_session ==> program
13	15.48	17.44	192	product ==> product
14	14.52	14.52	180	start_session ==> product
15	14.35	45.41	178	shelf ==> download
16	12.66	20.39	157	home ==> program
17	12.18	42.66	151	pay_res ==> shelf
18	11.77	24.33	146	logpost ==> catalog
19	11.69	26.13	145	p_info ==> p_info
20	11.53	38.03	143	cart ==> freeze
21	10.97	17.66	136	home ==> login
22	10.97	17.66	136	home ==> catalog
23	9.84	20.33	122	logpost ==> cart
24	9.76	15.71	121	home ==> product
25	9.68	18.18	120	login ==> register
26	9.19	9.19	114	start_session ==> logpost

From Table 3 notice that program → product is the most likely direct sequence. Notice however that its support is now about 0.60, against 0.73 when indirect sequences are considered. This means that, in 13% of the times, between program

and product are seen other pages. On the basis of Table 3, we can also make sensible conclusions on which pages are the entrance pages: home, program, product, logpost, which are listed among those with highest support of occurring together with start_session. Notice, on the other hand, that there are no sequences containing end_session as head of a rule: this means that, while the site is typically entered from a few pages (the ones listed before), it is exited from a wide variety of pages.

4. Markov chain models

We now consider a different model for the analysis of the transactional dataset. Association rules (of which sequence rules are a special instance) are an instance of a *local model*: they take into account only a portion of the data, i.e. that which satisfies the rule being examined. We now consider a *global model*, for which association patterns are discovered on the basis of the whole dataset. A global model suited to analyse the web clickstream data is the Markov chain model. Precisely, here we consider discrete Markov chains.

The idea is to introduce dependence between time-specific variables. In each session, to each time point i , here corresponding to the i -th click, it corresponds a discrete random variable, with as many modalities as the number of pages (these are named states of the chain). The observed i -th page in the session is the observed realisation of the Markov chain, at time i , for that session. Time can go from $i=1$ to $i=T$, and T can be any finite number. Note that a session can stop well before T : in this case the last page seen is said an absorbing state (end_session for our data).

A Markov chain model establishes a probabilistic dependence between what seen before time i and what will be seen at time i . In particular, a first-order Markov chain, which is the model we consider here, establishes that what seen at time i depends only on what seen at time $i-1$.

This short-memory dependence can be assessed by a transition matrix that establishes what is the probability of going from anyone page to any other one in one step only. For 36 pages there are 36×36 probabilities of this kind.

The conditional probabilities in the transition matrix can be estimated on the basis of the available conditional frequencies. If we add the assumption that the transition matrix is constant in time, as we shall do, we can use the frequencies of any two adjacent pairs of time-ordered clicks to estimate the conditional probabilities.

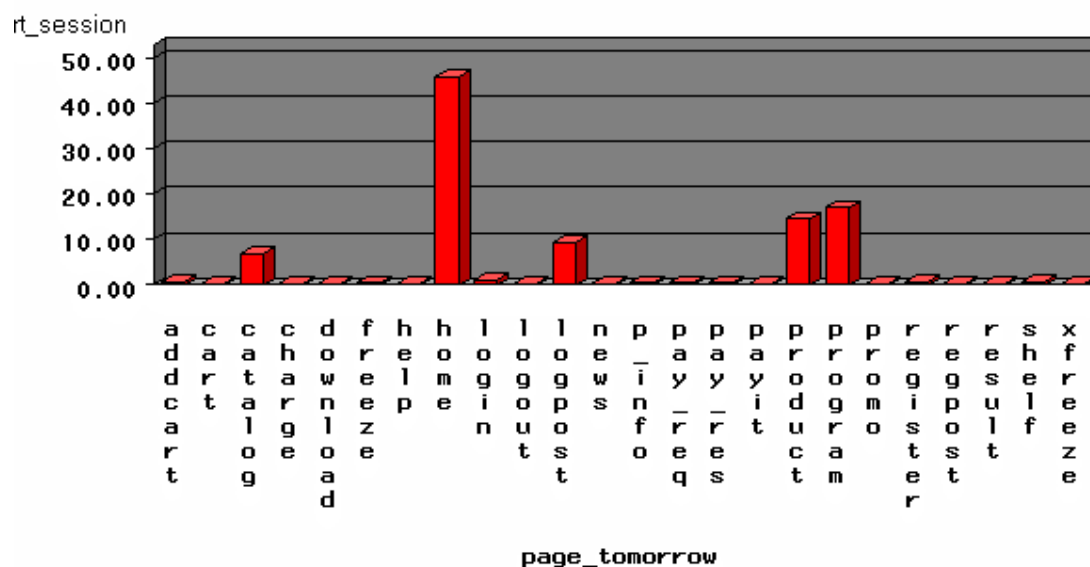
Note the analogy of Markov chains with direct sequences. It can be shown that a first order Markov chain is a model for direct sequences of order two; a second-order Markov model is a model for direct sequences of order three, and so on. The difference is that the Markov chain model is a global and not a local model. This is mainly reflected in the fact that Markov chains consider all pages and not only those with a high support. Furthermore, the Markov model is a probabilistic model and, as such, allows inferential results to be obtained.

For space purposes, we now briefly consider some of the results that can be obtained from the application of Markov Chain models. For more details, see Rognoni et al. (2002).

For instance, we can evaluate where it is most likely to enter the site. To obtain this we have to consider the transition probabilities of the start_session row. Figure 1

reports graphically such probabilities (excluding those estimated to have a null probability).

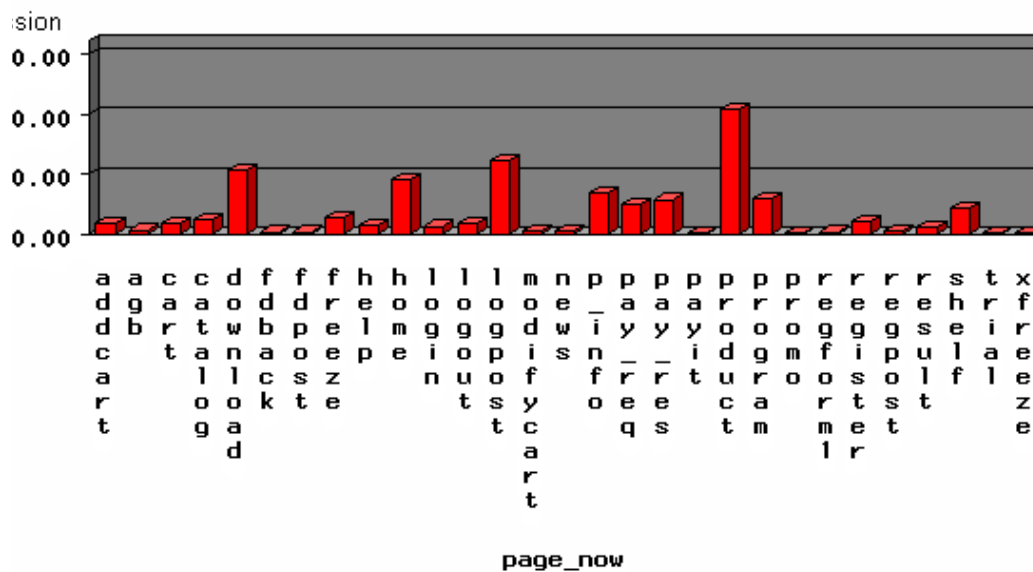
Figure 1: The most frequent entrance pages



From Figure 1 notice that the most frequent entrance page is home (48,81%), followed by program (17,02%), product (14,52%), logpost (9,19%) and catalog (6,77%). This is consistent with the nature of the visitors belonging to this cluster.

We can then consider the most likely exit pages. To obtain this we have to consider the transition probabilities of the `end_session` column. Figure 2 reports graphically such probabilities (excluding those estimated to have a null probability).

Figure 2: Most frequent exit pages



From Figure 2 it results that the most likely exit page is product (20,81%), followed by logpost (12,10%), download (10,56%), home (9,19%), (p_info, 6,94%). Finally, for comparison purpose we remark that, from the transition matrix we can establish a path that connects nodes through the most likely transitions. One possibility is to proceed forward from start_session. From start_session, we can connect it to the page with which start_session has the highest probability of transition: this is the home page. Next follows program, then product and then P_info. From P_info the most likely transition is back to product and, therefore, the path ends. We can compare what the previous path would be, using the confidence index in the sequence rule algorithm. Initially, from start_session, the highest confidence is reached for home (45,81%), as it occurs for Markov chains, followed by *program* (20.39, more than before), *product* (78,09% , against 70,18%). The next most probable link is now with addcart (28,79%) rather than link. The differences are due to the fact that the direct sequences consider only pages which pass a certain support.

5. Conclusions

We have considered two classes of statistical models to model web mining data. It is quite difficult to choose between them. Here the situation is complicated by the fact that we have to compare local models (such as sequence rules) with global models (such as Markov chains).

For global models, such as Markov chains, statistical evaluation can proceed in terms of classical scoring methods, such as likelihood ratio scoring, AIC or BIC. Or, alternatively, by means of computationally intensive predictive evaluation, based on cross-validation and/or bootstrapping. But the real problem is how to compare them with sequence rules.

A simple and natural scoring function of a sequence rule is its support, that gives the proportion of the population to which the rule applies. Another measure of interestingness of a rule, with respect to a situation of irrelevance, is the lift of the rule itself. The lift is the ratio between the support of the confidence of the rule $A \rightarrow B$ and the support of B. Recalling the definition of the confidence index, the lift compares the observed absolute frequency of the rule with that corresponding to independence between A and B.

Ultimately, though, the assessment of an association pattern has to be judged by their utility for the objectives of the analysis at hand. In the present case-study, for instance, the informative value of the `start_session` $\rightarrow$ `end_session` rule, which in table 1 has the largest support and confidence (100%) is, for instance, null. On the other hand, the informative value of the rules that go from `start_session` to other pages, and from other pages to `end_session` can be extremely important for the design of the website.

We remark that other statistical methods can be applied to web clickstream data, in order to detect association rules. For instance, Blanc and Giudici (2001) consider, besides sequence rules, odds ratios and graphical loglinear models, while Blanc and Tarantola (2002) consider bayesian networks and dependency networks. Finally, in a recent paper, Di Scala and La Rocca (2001) also consider the application of Markov chain models to web data, with main emphasis on assessing the homeogeneity of the considered Markov chain.

References

- Agrawal, R., Manilla, H., Srikant, R., Toivonen, H. and Verkamo, A.I. (1995) Fast discovery of association rules, in: *Advances in knowledge discovery and data mining*, AAAI/MIT Press, Cambridge.
- Blanc, E. and Giudici P. (2001) *Association models for web clickstream analysis*. Submitted.
- Blanc, E. and Tarantola, C. (2002) Dependency networks and bayesian networks for web mining. Submitted.
- Di Scala, L. , and La Rocca, L. (2001) A Markov model for web data. Submitted.
- Giudici, P. (2001) *Data mining, metodi statistici per le applicazioni aziendali*, McGraw-Hill, Milano.
- Hastie, T., Tibshirani, R. and Friedman, J. (2001) *The elements of statistical learning: data mining, inference and prediction*. Springer-Verlag, Berlin.
- Han, J. and Kamber, M. (2001) *Data mining: concepts and techniques*. Morgan Kaufman, New York.
- Hand, D.J., Mannilla, H. and Smyth, P (2001) *Principles of Data Mining*, MIT Press, New York.
- Rognoni, M., Giudici, P. and Polpettini, P. (2002) Markov chain clustering for web clickstream analysis. Submitted.