1. Introduction

Nowadays many research organizations are working on developing intelligent agents on the World Wide Web that can learn a user's interest and find information in the World Wide Web based on the user's profile. (E.g., Pazzani et al., 1997, Goecks et al., 2001) Some researches found relationships between a user's behavior and user's interest level while others used a content analyzer to build predictive models to predict user's interest level on a web page when a user visits the web page. We think about building predictive models only using user's behavior with the user's explicit rating. There are some problems to predict a user's interest level to record and analyze the relationship between a user's interest level and the user's behavior on the World Wide Web. To detect a user's interest indicators such as detecting the number of the mouse clicks while a user uses a web browser, we need to have a detection software program to see what a user does on the web browser and how much he likes the web page. In addition to that, after we find the user's interest indicators general enough to build predictive models, we are able to build predictive models that can predict a user's interest level according to the user's behavior. We can think about building predictive models using the regression analysis and neural networks. The two main motivations follow:

- Can we record a user's interest indicators general enough to build predictive models?
- Can we build predictive models to predict a user's interest level only using the user's behavior after the user leaves the web page?

In this study, we show how we collect each user's interest indicators and build predictive models to predict a user's interest level by recording and analyzing only user's behavior with the user's explicit rating on the World Wide Web.

1.1 Problem statement

Research has shown that a user’s behavior is related to his interest level such as reading time spent on the web page. However, there were some technological problems and limitations in recording and analyzing a user’s behavior general enough to build predictive models to predict how much a user is interested in the web page.

Two main hypothesis of this thesis follows:

- There are users’ interest indicators that tell a user’s interest level on a web page
- I can build predictive models that predict a user’s interest level using the user’s behavior

In addition to that, we hypothesized that while a user is browsing a web page, the individual user’s behavior is different from other users; Every user has a different model to predict their interest level since every user will behave differently when he/she visits his/her interesting or non-interesting a web page depending on whether or not it is of interest to him/her.

1.2 Solution approach

To build a predictive model that can predict a user’s interest level, a system (web browser) should record a user’s explicit rating and the user’s behavior. A user can rate a web page explicitly by reporting his interest level to a web browser before he leaves the web page that he visits. We assumed that the user’s explicit rating matches his real interest. We tried two methods, regression analysis and neural network, to build predictive models.

The two main contributions of this research are:

- Software that records a user's behaviors in general.
- Personalized predictive models that predict a user's interest level on the World Wide Web.

In addition to the above contributions, we have 4 experimental user's 10 data sets and 1 experimental user's 7 data sets. In Figure 1.1, we show the overall architecture of the predictive system.

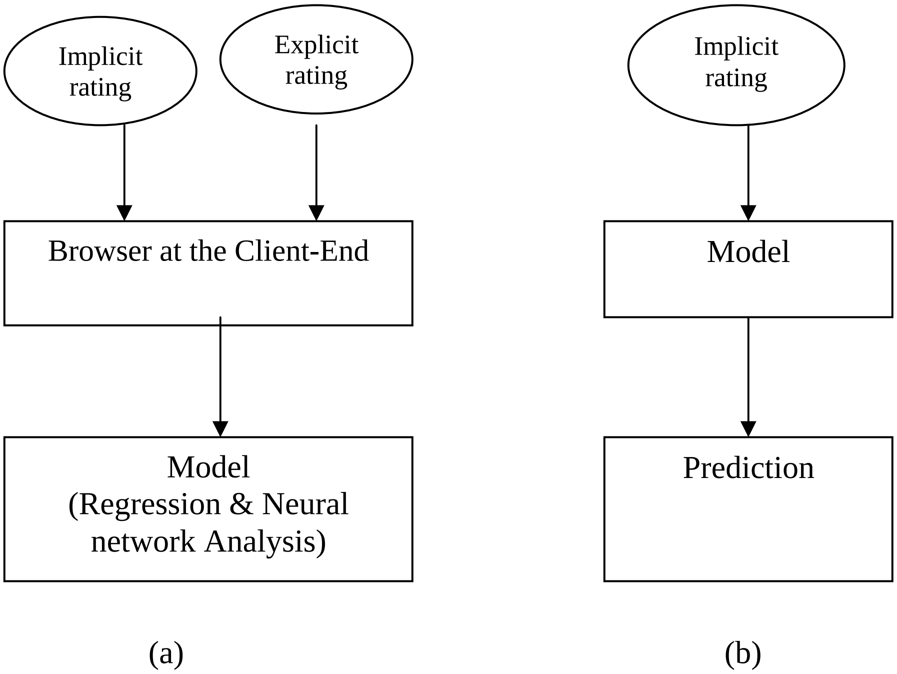

Implicit rating
Explicit rating
Browser at the Client-End
Model (Regression & Neural network Analysis)
(a)
Implicit rating
Model
Prediction
(b)

Figure 1.1 (a) Training (b) Operation

Figure 1.1 depicts the two phases, training and operation of our predictive system. In Figure 1.1(a), we have two inputs, the implicit rating and explicit rating, from a user to build the predictive models and in Figure 1.1(b), we have the predictive models to predict the user's interest level depending on the user's implicit interest rating.

1.3 Organization of the thesis

The organization of the rest of this paper is as follows: Chapter 2 describes related work which has been developed using implicit indicators; Chapter 3 describes software for recording a user's behavior and extracting a log file from a raw log file using an extractor after collecting the raw log file; Chapter 4 details the author's approach towards analyzing implicit interest predictors using proven indicators and a MSE table of the non-linear regression predictive model, the mouse-click-only regression predictive model, the linear regression predictive model, and a neural network to show how well each model fits; Chapter 5 presents conclusions and mentions possible future work and the applicable fields with an example of a personalized web search.

2. RELATED RESEARCH

2.1 Introduction

A web personalization system provides personalized suggestions about a user's web pages of interest. We can have an explicit rating and an implicit rating for each user. Followings are disadvantages of explicit rating and advantages of implicit rate on the web browse which support why we need to use more implicit rate with less explicit rating.

Disadvantages using explicit rating on the web:

- When a user browses on the World Wide Web, entering explicit ratings will somewhat alter normal patterns of the user's browsing behaviors (Claypool et al., 2000).
- Without an incentive to rate explicitly, a user may stop rating the pages (Avery et al., 1997; Grundin et al., 1994)
- Biased evaluators (Palme, 1997)
- GroupLens System (Sarwar et al., 1998) found that users were reading a lot more articles than they were rating
- Collaborative filtering requires many ratings to be entered for every item in the system in order to provide accurate predictions (i.e. the "sparsity" problem) (Sarwar et al., 1998)
- There might be a significant difference between a user's real interest level and the user's explicit rating since users sometimes have difficulty expressing their interest explicitly on a single numeric scale. (Morita et al., 1994)

Advantages of implicit rating:

- They reduce the cost of the user examining and rating items
- Potentially, every user's interaction with the system can contribute to an implicit rating

Although each implicit rating is likely to be less accurate than an explicit rating, they:

- Are almost free except additional processing on the client's side
- Are easy to add to other implicit ratings for a more accurate rating
- Can be combined with explicit ratings and a user profile at the beginning for an enhanced rating that compensates for explicit rating's 5th disadvantage
- However, explicit rating are fairly precise (Watson et al., 1998)

Those recommendation systems need a user interface to determine the level of a user's interest and use this feedback from each user to make suggestions for him/her. Not many systems have implicit rating system, nor is there an ability to detect various user's behaviors in those systems.

By comparing previous systems in Chapter 2.2, we can see what system has which feature for implicit rating. The goal of this paper is to collect, measure and evaluate the predictive power of some promising implicit interest indicators in the personalized web system.

2.2 Implicit systems

Malone et al. (1997) describe three forms of information filtering: cognitive (or content), economic, and social. Content-based filtering is dominant in IR (e.g. Foltz and Dumais, 1992), typified by profiles based on keywords, and economic filtering will become increasingly important as digital cash, micro-payments, and secure payment technologies emerge from research laboratories onto the Internet. Social filtering has moved on from the original description (of the importance of the identity of the sender of a message) to several research projects and a few actively used systems.

The main reason to use implicit ratings is that it removes the cost to the evaluator of examining and rating the indicators. Even though there remains a computational cost in storing and processing the implicit rating data this can be trivial and hidden from the user side. In a networked environment it is usually difficult for the user to separate network latency from extra application processing. Even though a user allows a limited storage/transport of implicit data in the client side, that storage/transport will not be an intensive task.

2.2.1 The Tapestry text filtering system

This system is developed by Nichols and others at the Xerox Palo Alto Research Center (PARC) and was the first to include social filtering (Nichols et al., 1997). Designed to filter personal e-mail, messages received from mailing lists, Internet News articles, and newswire stories, Tapestry (Douglas et al., 1993) allowed users to manually construct profiles based both on document content and on annotations made according to those documents by other users. Experience with several small-scale trials of social filtering suggests that a critical mass of users with overlapping interests is needed for social filtering to be effective.

2.2.2 GroupLens

GroupLens (Resnick et al., 1994) filters Internet news and has a freely redistributable text source. This system has content servers and Annotation servers, beginning in 1996 using a limited number of newsgroups and a single annotation server. GroupLens also addressed the potential benefit of implicit feedback (Konstan et al., 1997), although in this case it was used with social filtering. An experiment was done in 1996 using a limited number of Internet discussion groups (USENET news), to apply newsreader software for a user to enter explicit ratings and receive predictions. Over a seven-week trial, 250 registered users submitted a total of 47,569 ratings and received over 6,000,000 predictions for 22,862 different articles. Specially modified news browsers were provided that accepted explicit ratings and displayed predictions on a 1-5 scale where 1 was described as “this item is really bad” and 5 as “this article is great, we would like to see more like it.” Their study showed that predictions based on time spent reading are nearly as accurate as predictions based on explicit numerical ratings. They also suggested further actions, such as printing, saving, forwarding, replying to, and posting a follow up message to an article, as sources for implicit ratings.

2.2.3 InfoScope

Stevens developed a system called InfoScope (Stevens et al., 1992) that used automatic profile learning to minimize the complexity of exploiting information about the context in which words were used. InfoScope used three sources of implicit evidence about the user’s interest in each message: whether the message was read or ignored, whether it was saved or deleted, and whether or not a follow up message was posted.

2.2.4 URN

URN (Brewer et al., 1994) is an Internet News filtering system in which users can provide two types of information to support profile learning by making explicit binary judgments about the utility of the document. Those judgments were then used as a basis for a typical content-based ranked output system. Users can also collaboratively improve the system’s initial representation (or, for deletions, misrepresenting) of the content of the document. Each user maintains a separate content-based user model, while the annotation server effectively maintains a single collaboratively developed model of the document space.

2.2.5 Curious Browser

The contributions of Curious Browser (Claypool et al., 2001) are experimentally based statistical analysis of the correlation between the implicit interest indicators of mouse activity, keyboard activity, and time, with explicit interest.

A categorization of implicit interest indicators by Claypool et al. (2001) follows:

- Explicit Interest Indicators
- Marking Interest Indicators (bookmark, delete bookmark, save, e-mail the page, print)
- Manipulation Interest Indicators (Cut, paste, Open new windows)
- Navigation Interest Indicators (follow a link or not)
- External indicators (Physical response: heart-rate, perspiration, temperature, emotions and eye movements)
- Repetition Interest Indicators (Number of visits, scrolling, duration)

Their browser records a variety of implicit interest indicators:

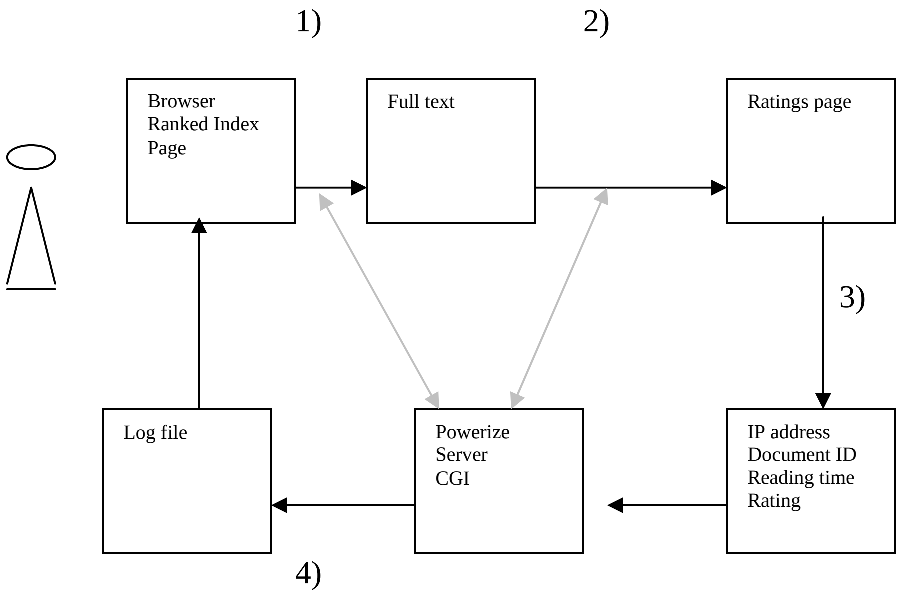
- Time moving mouse vs. Explicit Rating
- Number of mouse clicks vs. Explicit Rating
- The number of mouse events (clicks) on the horizontal and vertical scroll bars vs. Explicit Rating
- Time spent scrolling vs. Explicit Rating
- Amount of time holding arrow keys
- Number of arrow key presses
- Time spent on the web Page vs. Explicit Rating

They found that time is good implicit indicator of interest and mouse movement and mouse clicks by themselves are ineffective implicit interest indicators. However, in using mouse clicks and keyboard actions to infer the level of scrolling, they obtain an means of determining the “amount” of scrolling that also provides an effective indicator of interest.

2.2.6 Powerize server 1.0

Powerize Server™ (Jinmook et al., 2001, Oard et al., 1998) is Windows NT Web server-based text retrieval and filtering system that enables users to search distributed heterogeneous information sources. Profiles are used to periodically monitor specific sources for new information.

A custom version of Powerize Server 1.0 was created for these experiments by powerize.com. It was modified to measure reading time and printing behavior and to record user-entered ratings for individual documents. Table 2.1 illustrates the procedures for using the modified system, showing how reading time, printing behavior, and explicit ratings are recorded.



1) User clicks on a title.

2) User clicks on either the “Feedback” or the “Print(Save to File)” button.

3) User marks a rating and clicks on the “Submit” button.

4) CGI creates an entry in a log file.

Figure2.1 Procedures for using the modified Powerize Server 1.0 from Jinmook et al. (2001)

A modification on the Powerize Server was required to use the following category.

Category	Observable Behavior	Applicability to Powerize Server 1.0	Ease of Measurement**
Examination	Selection	Yes	1
	Scrolling behavior	Sometimes*	3
	Repetition	Yes	1
Retention	Save	Yes	2
	Print	Yes	2
	Delete		N/A
	Purchase		N/A
Reference	Forward	Yes	3
	Reply		N/A
	Post follow up		N/A
	Hypertext link		3
	Citation		N/A
	Cut & Paste	Yes	3
	Quotation		N/A

* Scrolling behavior is not applicable when the length of articles is not long enough to do scrolling.

"2" indicates that it is measurable by modifying either a Web browser or Powerize Server™

"3" indicates that it is not measurable without modifying a Web browser.

"N/A" indicates that it is not applicable to Powerize Server™, thus not measurable.

Table 2.1 Observable behavior using Powerize Server™

2.2.7 Jeremy Goecks' Agent

Jeremy Goecks' Agent (Goecks et al., 2000) mainly used surrogates to decide user's implicit interest's level. They took two following main steps to verify that.

Step1- When the user visits a web page, record (a) the HTML contents of the page; and (b) normal actions performed by the user on the page.When the user navigates to a new page, the agents records (a) the text of the HTML text; (b) the number of hyperlinks the user clicked on; (c) the amount of user scrolling activity; and (d) the amount of user mouse activity.

Setp2- Build labeled training instances from the information recorded in Step 1 and train on these instances. This agent showed these three surrogates could be useful in predicting a user's interest level.

2.3 Comparison between different systems based on features

On the following table 2.2, we mention only non-content-related aspects of those systems As we mentioned the goal of this paper on Chapter 1.

As we mentioned in Chapter 2.2, while Goeck's Agent detects a user's three surrogates in mouse activity and scroll activity, Curious browser detect the time spent on scrolling, clicking and keyboard activity. Keyboard activity is included in the scrolling activity with the mouse. These two Agents cannot detect the distance of mouse movement and scrolling distance of scroll bar due to technological limitations; without building a browser, it is almost impossible to detect them. The Better Bit Bureaus (BBBs) predict how much readers will like articles. While content filters would make predictions based on the presence or absence of words in the articles, the BBBs in GroupLens use the opinions of other people who have already rated the articles. If no one has read an article, the BBBs are unable to make predictions about it. Personalized user's behavior concept lacks on this system since they classified into some groups of people who behave similarly. Jeremy Goecks' browser hook mainly showed three surrogates can be used to predict user’s interest level on the web through content-filtering. However, the RMS Error on Test Sets was relatively big. Also, instead of using surrogates, using proven indicators will be more effective. Curious browser by Mark Claypool (implicit indicators vs. explicit rating on the web) tried and found more general and various types of implicit indicators such as scroll activities on the browser. But, they didn’t predict and make model to predict the level of user interest using the indicators they found, as they didn’t have enough indicators to model. Letiza (Liberman et al., 1995) works by detecting the user’s behavior under the automated browsing assistant program.

	Goecks’ browser hook	Curious browser	Group Lens	Amazon .com	Letizia	Powerize Server 1.0
Hyperlinks Clicked	+					
Scrolling Activity	+	+				
Mouse Activity	*1	*2			*4	
Keyboard Activity		*3				
Time on Page		+	+		+	+
Print						+
Following a link					+	
Passed over					+	
Prediction user’s interest for the page			+	+	+	
Content-Based	+		+		+	+
Explicit rating		+	+		+	+
Purchase-Based				+		
Pilot test			+	+		
Saving					+	
Number of visit					+	

Blank (” “) means system didn’t investigate it.
“+” means it is good indicator.
“*” means it is good indicator conditionally.
*1: Goeck’s Agent uses surrogates for mouse activity by assuming that the more mouse activity, the more status bar changes.
*2: Number of Mouse Clicks versus Explicit Rating in Curious browser showed the difference between interesting level1 and level5 but no difference between 1 and 2 and also no difference among 3,4,5. In addition, Time Moving Mouse vs. The explicit rating also doesn’t have difference among 2,3,4,5 while level1 and the other level have a significant difference.
*3: Curious browser used keyboard activity to get the time spent scrolling by the mouse and the keyboard.
*4: Letizia uses the number of mouse click on the hyperlink.

Table 2.2 Comparison between different systems based on features

All of the above three systems (Jeremy Goecks' browser, Curious browser, Letiza) mentioned that the time spent on the web-page/article is highly related to the user's interest. Powerize Server 1.0 verified printing behavior is a good indicator of user interest. Also, they showed the expected pattern of increasing reading time with increasing rating and mentioned that any obvious way of using reading time alone to make predictions would have missed some of those cases. The Tapestry text filtering system, InfoScope, GroupLens, and URN use social filtering. The limitation of the existing experimental work on social filtering is user motivation. In GroupLens, users annotate documents in order to improve the performance of their filter's ability to learn from other clients who have annotated the same documents. This creates a "chicken and egg" problem, though, since there is no incentive for the first user to annotate anything. If content-based and social filtering are integrated in the same system, however, then a synergy between the two techniques can develop. The URN system showed a more automatic method by which such synergy can be achieved by making explicit binary judgments about the utility of the document.

2.4 Discussion

Nichols (Nichols et al., 1997) presented a list of potential types of user's behaviors that could be used as sources for implicit feedback, such as purchases, assess, repeated use, print/save, delete, refer, reply, mark, examine/read, glimpse, associate, and query 'refer' behavior contains all those instances where one information item links to another item, including traditional academic citations as well as hyperlinks on Web pages or the threaded links between USENET news articles. Citation indexing has been well studied in the field of information retrieval, and this appears to be a promising source of implicit feedback in some applications.

Category	Observable Behavior
Examination	Selection Duration Edit wear Repetition Purchase (object or subscription)
Retention	Save a reference or save an object -With /without annotation -With or without organization Print Delete
Reference	Object-> Object (Forward, reply, post follow up) Portion->Object (hypertext link, citation) Object->Portion (cut & paste, quotation)

Table 2.3 Observable behaviors for implicit feedback.

All of systems evaluate a subject of this observable behavior and none of the system covered all categories. Since perfect cover of these categories is difficult with current technology, the system that covers examination, retention, and reference generally can produce a better score for user's interest level. Also, some of above system missed personalization aspect. Every human being is different; they behave differently over time. So, my research is focused on finding more user's interest indicators and personalizing those indicators to make the personalized model predict each individual's interest level differently.

3. Monitoring user's behavior

This Chapter describes a user's implicit interest data collected from the users and how we collected it. Chapter 3.1 describes the software's architecture and Chapter 3.2 describes the implicit interest indicators we am interested in. Chapter 3.3 describes mainly how an explicit interest rating is collected. Chapter 3.4 describes the overall methodology to generate a user's implicit interest data and Chapter 3.5 describes how a user's implicit interest data is extracted from a raw-log file to a table formatted log file.

3.1 Architecture

An implicit-explicit personalized predictive model is a model that can be affected by user's explicit response as well as user's implicit behavior as you see in Figure 1.1(a). We tried to find a correlation between candidate interest indicators and explicit ratings and a correlation between every possible and reasonably suspicious indicators and explicit rating since the more good predictors we find the more accurate predictive results we can get. The main function of this software is collecting explicit ratings and the user's implicit interest indicator data. Since every user has different interest predictive models, all indicator data for each user are supposed to be collected.

3.2 Implicit interest indicators from user's behavior

The Internet Client SDK and Microsoft's Common Object Model (COM®) provide 'hooks' that allow a program to observe, record, and measure a variety of user actions. However, there are still many limitations and difficulties in detecting a user's behavior such as detecting the distance of mouse movement, the distance of scrollbar movement and whether the text is highlighted or not and so on. Unless you build your own web-browser or have a web browser's source code and an almost perfect understanding of a browser, it will not be possible, or very difficult, to detect more general user's behavior. Here, we built a web browser called "kixbrowser" to detect most of the user's behavior and found good positive/negative interest indicators among the candidate indicators so that we could detect some of the user's implicit behavior. The behaviors monitored in this study are discussed below.

3.2.1 Duration

We assumed that the time spent on the web page might be a good predictor. We hypothesized that the longer a user stay on the web page, the more interesting level a user has.

Duration means the time interval between the time a user visits and leaves the web page and is measured in units of seconds. It includes all the actions and the actual reading time for the page as well as the time that the kixbrowser is not in focus for the purpose of editing such as pasting on the other editors. Thus, factors that influence its accuracy include loading time (which, in turn, depends upon speed of connection, CPU speed and the amount of Internet traffic) and how much of the active window time the user actually spends looking at the Web page (as opposed to going out for tea). In the raw log file, there is only the starting time. By using the extractor, you can subtract current web page's starting time from next web page's starting time. The formula follows:

Current web page's Duration (seconds) = Next web page's starting time – Current web page's starting time

3.2.2 Distance of Mouse Movement

Kixbrowser detects the distance the mouse moves by using its x and y coordinates on the active browser. The distance of mouse movement is measured as the total distance the mouse position is changing inside the active browser. Some users move the mouse while reading the window text or looking at interesting objects on the page, while others move the mouse only to click on interesting links. Either way, we hypothesized that the longer the distance of mouse movement is, the more a user is interested in the web page. Formula follows:

mouse_move_distance=|mousex-oldx|+|mousey-oldy|+ mouse_move_distance

“oldx” and “oldy” contains the previous x and y coordinates so that we can get the difference between the old and current location. To get the X and Y coordinates we used getX() and getY() functions from the mouse event library. As far as a user moves mouse on the same web page, mouse_move_distance will be accumulated.

3.2.3 Distance of Scrollbar Movement

We hypothesized that the longer the distance of scrollbar movement is, the more a user is interested in the web page. A user find interesting, most likely as they read the material occasionally as they search the page for interesting links to follow. A user might scroll by clicking on the scroll bar, clicking and dragging the scrollbar. Whenever a user scrolls vertical scrollbar and Horizontal scrollbar, the distance is summated on kixbrowser. The way we measure the distance of Scrollbar movement is the same way as the distance of mouse movement except that we used getHorizontalScrollBar() and getVerticalScrollBar() functions to get the location of scrollbar by using AdjustmentListener.

3.2.4 Number of Mouse Clicks

While a user reads a web page, a user clicks the mouse button as a habitual behavior or to hyperlink to another web page. We hypothesized that the greater the number of mouse clicks on a web page, the more a user is interested in it. Extractor will count how many times the mouse button has been pressed. If a mouse button is pressed, then the mouse button click counter will increase by 1 and if the mouse button is pressed at a location and released at the different location on the web browser, this is treated as highlighting text. To see if the x or y coordination is same or not, an extractor is used to compare mousePressedx and mousePressedy with mouseReleasedx and mouseReleasedy by using string matching. When the mouse is pressed, Is_pressed=true. To see whether two strings, string1 and string2, match or not, we used string1.equals(string2) function in Java.

3.2.5 Number of Highlighting text or sentence

Some users highlight text or sentences or keywords while they are reading web page. We hypothesized that if a user is interested in the web page, he will highlight a text or sentence more in the web page than if he is not interested in it. In Chapter 3.2.4, we explained how we detect highlighted text.

3.2.6 Frequency

Frequency means the number of visit on a web page. We hypothesized that frequency and explicit rating are positively related, which means that if a user likes a web page or is interested in reading the contents of a web page, a user will visit again and again. Whenever a user visits a web page, vis_num (The number of visit on the web page) will increase by 1. Comparing the current URL with the previously visited URL, the extractor can increase vis_num by 1 when two URL strings are matched or add this new record for the current URL by setting vis_num to 1 when two URL's strings are not matched. So, every vis_num column of the record is supposed to be more than 1.

3.2.7 Recency

Recency tells how recently the page has been visited. We hypothesized that the more recently a user has visited a web page, the more a user is interested in the web page. For example, the last previous visit on abc.com was on the 7th experiment and the current visit is during the 8th experiment. Then,

Recency = 10-(8-7)= 9

So, if a user visits a web site more recently, the recency's score will be higher.
Following is the formula:

Recency = 10-(Nth_experiment- the last vist)

3.2.8 Number of rolls over the hyperlinks

This predictor is very related with change of the status bar. Whenever the mouse is rolled over the hyper link, the status bar will show the different URL.Wehypothesized that the greater the number of roll over on the hyperlinks on a web page, the more a user is interested in the web page. On hyperlinklistener, when HyperlinkEvent's event type is enter, this shows mouse is rolling over a hyperlink. In java's expression, e.getEventType()==HyperlinkEvent.EventType.ENTERED.

E equals to hyperlink event and HyperlinkEvent has a EventType function. EventType has Enter and Exit.

3.2.9 Page after typing characters

After typing many characters, a user might be on an interesting web site. We hypothesized that the number of a user’s key input is related to a user’s interest level on the web page after a user inputs something on the web browser. This can be good indicator. For example, if a user visits a web-page and is interested in the web page, he will tried to answer any question or comment to get more articles or information.

3.2.10 Button/Ctrl key

Copy, paste, adding to bookmark, select all, back and forward actions will be performed by a user who can copy an interesting text or sentence using right click as well as Ctrl+c, Ctrl+v and Ctrl+x. We hypothesized these actions can indicate a user’s interest. Figure 3.1 is a popup menu, which comes out on the screen when a user clicks the right button.

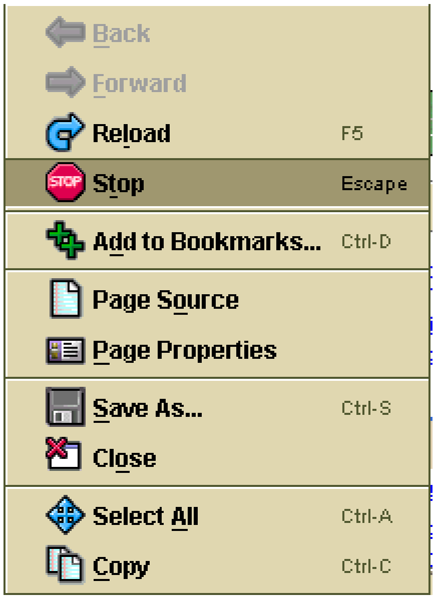
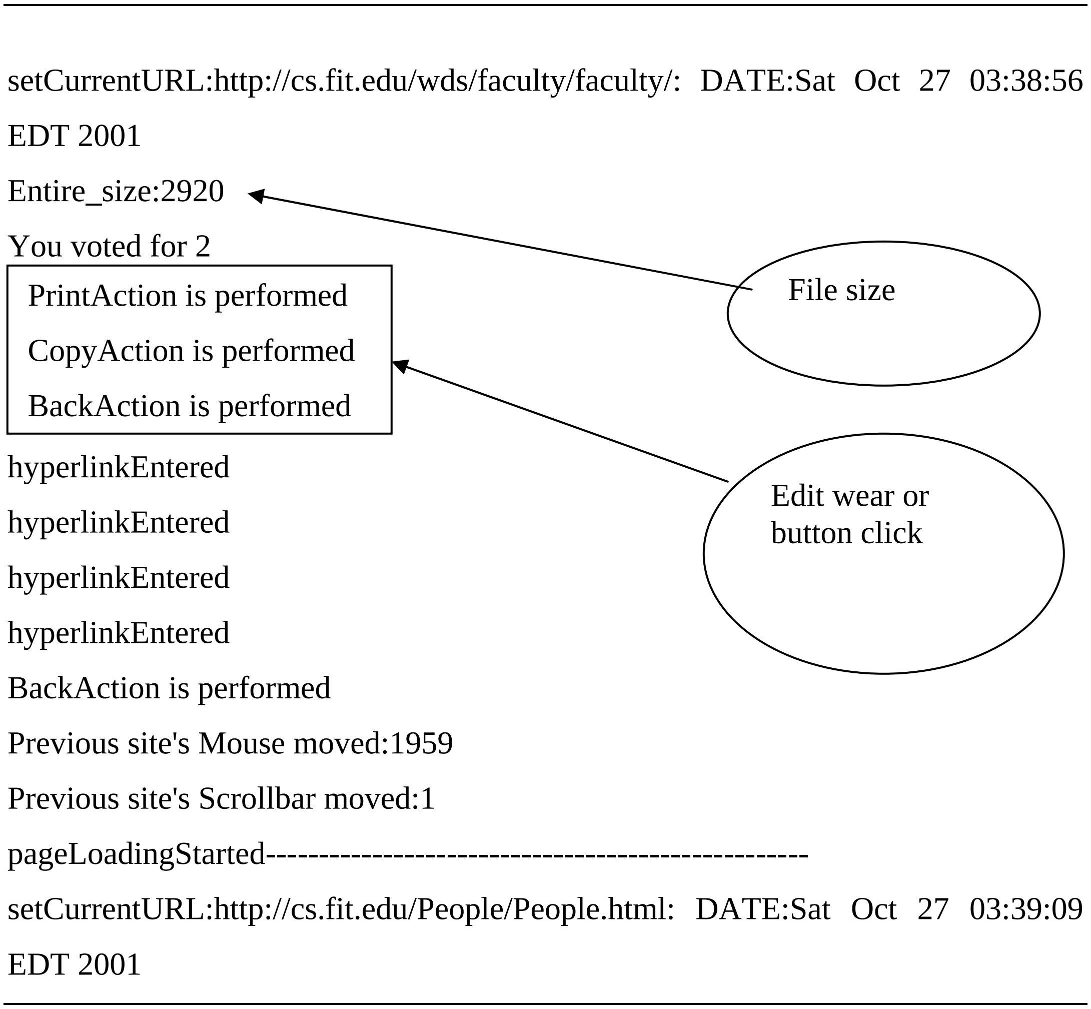


Figure 3.1 Edit (Copy, Paste, Cut) menu in popup menu

We recorded these actions whenever they were performed.

A sample of button click and Edit action in a raw log file format follows:



```
setCurrentURL:http://cs.fit.edu/wds/faculty/faculty/:  DATE:Sat  Oct  27  03:38:56
EDT 2001
Entire_size:2920
You voted for 2
PrintAction is performed
CopyAction is performed
BackAction is performed
hyperlinkEntered
hyperlinkEntered
hyperlinkEntered
hyperlinkEntered
BackAction is performed
Previous site's Mouse moved:1959
Previous site's Scrollbar moved:1
pageLoadingStarted--------------------------------------------------
setCurrentURL:http://cs.fit.edu/People/People.html:  DATE:Sat  Oct  27  03:39:09
EDT 2001
```

Figure 3.2 Edit action’s record

Figure 3.2 shows how PrintAction, CopyAction and BackAction is recorded in the raw log file. In the extractor program, the Print, Copy and Back columns will be set to 1.

3.2.11 File size

File size means that when a web page is loaded, a web page’s contents will be saved in the temporary file directory. Therefore,wecan use the GetFileize() function to obtain the file size in Figure 3.3. File size might be useful to find out the relationship between duration/size or size and user’s interest level. We hypothesized that file size or file size/duration of a web page increases if user is interested in the web page.

3.2.12 key input

We hypothesized that the number of key inputs on the previous page will affect the current web page, which means that the number of key inputs on the previous page increases, as a user is more interested in the current web page.

Those above candidate indicators can be used in modeling predictive user interest behavior for each web page after a user visits and leaves the web page. In this way, the user doesn’t have to rate explicitly whenever they visit a web page to tell the browser about the user interest level since the previous user’s explicit rating will model for the user and will predict according to the user’s behavior.

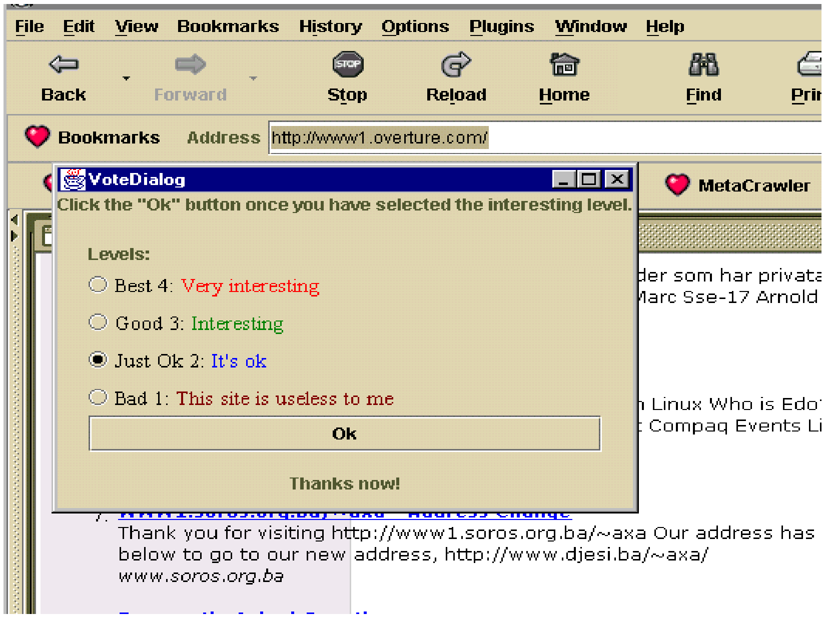
3.3 User’s Explicit Interest

Kixbrowser has a voting system that allows the user to rate the last page. Once he changes the web page by clicking on a hyperlink or typing in a URL text field or accessing it from a bookmark, the user is supposed to vote for the site. If a user visits and votes several times on the page, the recorded maximum value will be extracted by the log-file extractor. The web search site is excluded because of user’s different pattern of web surfing behavior and also because most users usually visits web-searching site first. We are not interested in which user likes which web-searching site because that the user will have an abnormal number of visits on that specific web-searching site.

By doing research on this system as described in Figure 1.1, we hypothesized that analyzing only the user’s behavior can be a good model to give a highly qualified information about a user.

A user can explicitly vote for the web page. There are 4 user interest levels: “very interesting”, “interesting”, “it’s ok” and “useless” (Bad). We had 5 experimental users. The reason that we use only 4 levels is because my hypothesis is that 4 levels are the levels that a user can differentiate each level. For example, if there are 10 levels, most users cannot see the difference between level 8 and level 9. Therefore, it is reasonable to have 4 levels, as we mentioned. Also, if the system has two levels on the negative side such as bad and worst, a user will be confused since both are same “bad” and a user doesn’t want to think about the difference. User 1, User 2, User 3 and User 5 did experiments for 10 hours by experimenting 1 hour each day, or on every available day and User 4 experimented total of 7 hours. This experiment continued for about 4 weeks since some users are busy and couldn’t 10 experiments in the short period of time. Also, we asked users to only experiment whenever they felt like doing so. Figure 3.3 shows what the voting window looks like.

For example, if a user rate it as “very interesting”, “You voted for 2” will be recorded in the raw log file, which means that a user think this web site is “just ok”. So, there is a minimum of 1 point (bad) and a maximum of 4 points on the explicit score. Since a user voted for 2 in the example, a user gives the web page 3 points on the explicit score.



http://www1.overture.com/

Click the "Ok" button once you have selected the interesting level.

Levels:

○ Best 4: Very interesting

○ Good 3: Interesting

● Just Ok 2: It's ok

○ Bad 1: This site is useless to me

Thanks now!

Thank you for visiting http://www1.soros.org.ba/~axa Our address has below to go to our new address, http://www.djesi.ba/~axa/ www.soros.org.ba

Figure 3.3 This picture is captured from kixbrowser

3.4 Overall Methodology to generate data

To generate the users' behavior log files, we built web browser and hooked recording functions and asked experimental user to use kixbrowser. After that,weextract each user's raw log file into a table formatted log file.

3.4.1 Building a web browser (kixbrowser)

We tried to imitate Internet explorer on GUI design as much as we could and used icons and picture taken from Dean S. Jones's work and took most of browser structure from Sun Microsystems web page (http://java.sun.com/). First of all, we built html parsers and rendering packages for the screen with basic GUI. We looked for a web browser parser for about a month. We get some information from the web site “http://java.sun.com/docs/books/tutorial/networking/index.html” and “http://www.mozilla.org”. Once we have an html parser, we add more GUI such as buttons and pull-down menus imitating the popular web browser such as Internet Explorer. Now, you have a good-looking web browser. We evaluate each function as much as we can. We find some functions’ partial evaluations from the World Wide Web and books. There are always time constraints. So, Kixbrowser can parse html file and text file but not flash, doc, ps, pdf and javascript partially. Also, there are no language support but English. Then, we can hook some recording function on the web browser in the next Chapter 3.4.2.

3.4.2 Hook recording functions to all implicit indicators related to user’s interest level and recorded into the raw log file

Recording functions will be performed in the following cases:

- When a user clicks any menu button on the screen
- When a user right mouse clicks, a popup menu shows some functions, such as Copy, Paste, Cut and Addtobookmark
- When a user opens a web page from bookmark

As you see Figure 3.4, a pop-up menu includes back, forward, reload, stop, add to bookmarks, page source, page properties, save as, close, select all and copy. Whenever each action is performed, it will be recorded. Figure 3.6 depicts a user’s raw log file. We will discuss the user’s raw log file in details in the later Chapter 3.5.1.

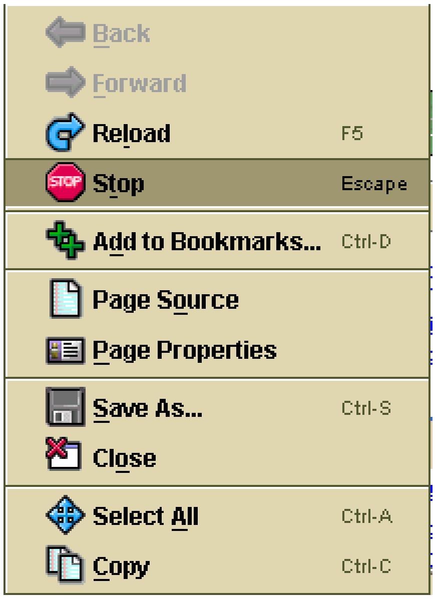


Figure 3.4 Right click Pop-up menus

3.4.3 Getting the data by doing experiments

There are five students who web search everyday more than 1 hour. Five students came to lab whenever they felt like using a web browser to get information of interest on the web. The first 9 data was for training set and the 10th one was for the prediction of the user interest level. But, in user 4's case, he performed the experiment only 7 times. So, we used only 6 experiments for training dataset and 1 for the test dataset.

3.4.4 Building a parser (extractor) for the raw log data to the log data

For example, kixbrowser records user's starting time and leaving time of the web page in the raw log data file. After all, the parser calculates this data to have duration. We excluded the web search web site since each user's behavior is different on a web-search site from a normal web site. For example, a user usually visits the web-search site whenever he/she starts to use the web browser. Of course the number of visit is very high even though the user's interest is not that high on the search web site. Also, we removed all the cases in the log file where a user doesn't rate for the web site. For example, when a user wants to go back to previous web site quick, the user sometimes forgets rating again for the web site. In this case, the web page's information will be removed. For the same web page, the user's interesting level might be different. We hypothesized that the maximum explicit rating on the same web page is the user's real interest level: this system uses only the maximum rated level.

3.5 Extracting Data

As you see in Figure 3.5, there are 3 steps to extract data before analyzing it. First of all, we have to have each user's raw-log file to make a calculated log file by an extractor. Then we changed the calculated log file's format to table log file format using SPSS or Excel.

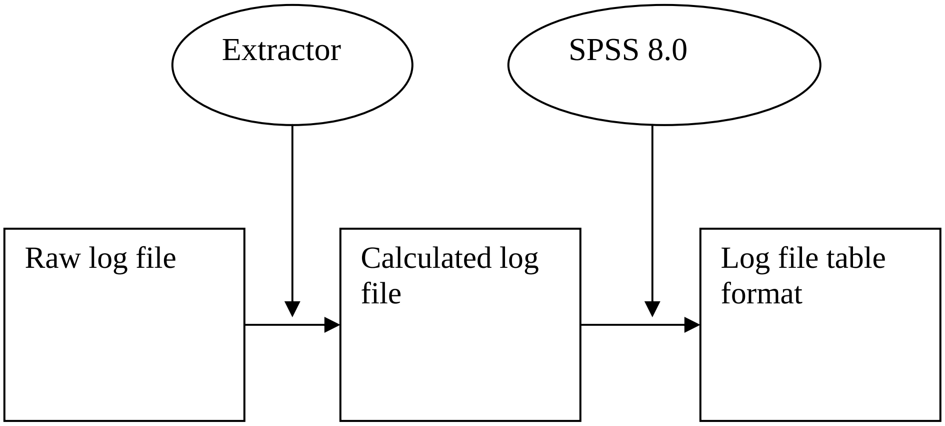
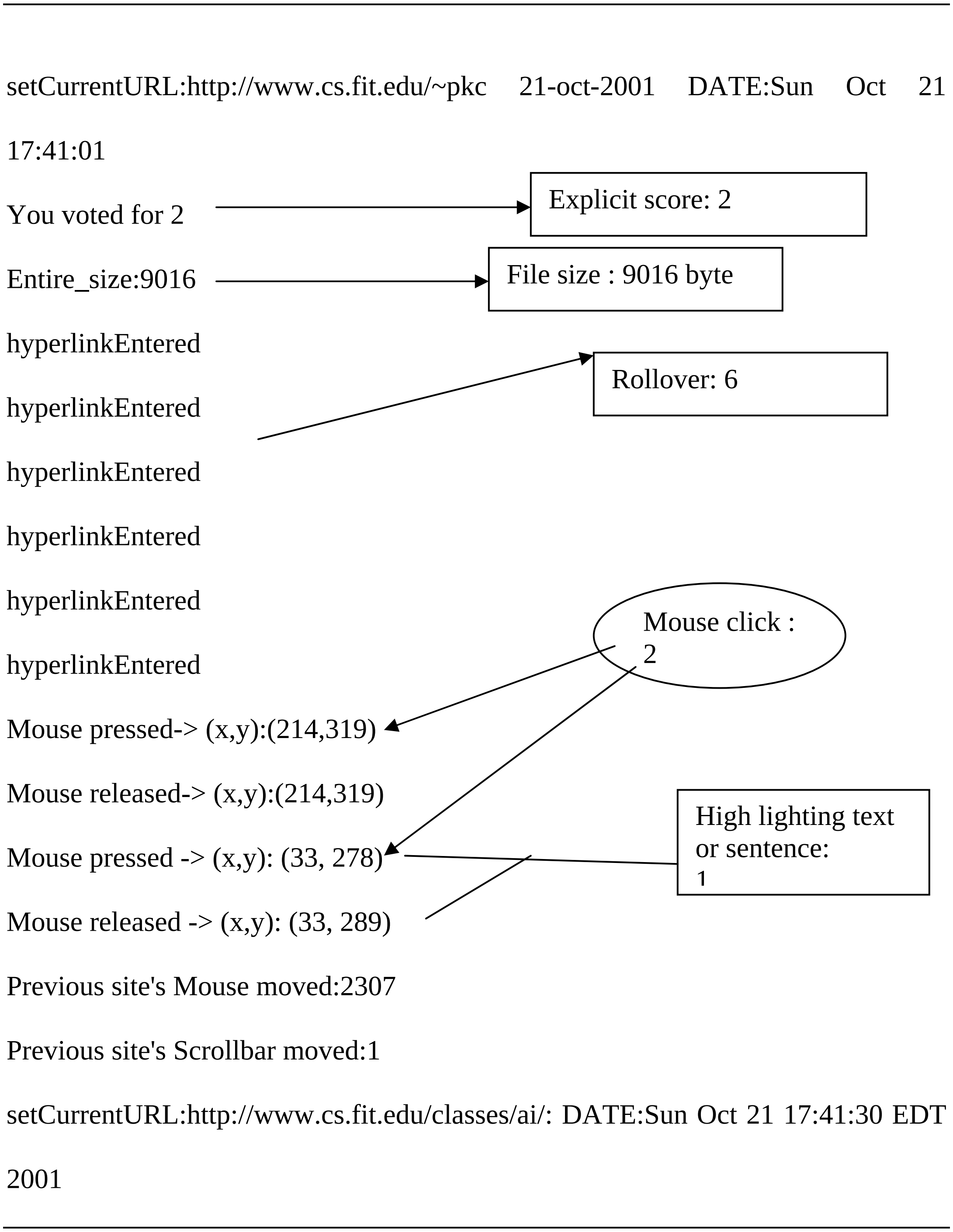


Figure 3.5 The process of extracting data

3.5.1 Getting each user’s raw-log file

The log file contains 26 candidate users’ indicators; it tells which actions a user performs. Figure 3.6 shows that it does not have to contain all indicators; it only contains all indicators that a user did. “You vote for 2” means a user is interested in the web site “http://www.cs.fit.edu/~pkc”. “Entire_size” is the web page’s file size. “HyperlinkEntered” means that a user rolled over a hyperlink on the web page. In this sample, a user rolled over a hyperlink 6 times. “Previous site's Mouse moved: 2307” means that a user moved the mouse 2307 (default mouse movement units in java) distance. “Previous site's Scrollbar moved:1” means that a user moved 1 (default scrollbar movement unit in java). These two will be calculated now for the previous visited web site before he visits “http://www.cs.fit.edu/~pkc”. Extractor will arrange all these indicators in the ascii table format in the next step.



```
setCurrentURL:http://www.cs.fit.edu/~pkc   21-oct-2001   DATE:Sun   Oct   21
17:41:01
You voted for 2
Entire_size:9016
hyperlinkEntered
hyperlinkEntered
hyperlinkEntered
hyperlinkEntered
hyperlinkEntered
hyperlinkEntered
Mouse pressed-> (x,y):(214,319)
Mouse released-> (x,y):(214,319)
Mouse pressed -> (x,y): (33, 278)
Mouse released -> (x,y): (33, 289)
Previous site's Mouse moved:2307
Previous site's Scrollbar moved:1
setCurrentURL:http://www.cs.fit.edu/classes/ai/: DATE:Sun Oct 21 17:41:30 EDT
2001
```

Figure 3.6 Raw log file

3.5.2 Extracting raw-log file to ASCII log file for analysis

As you see in Figure 3.7, the ASCII log file include URL, Start, Explicit, Size, rollover, mouse click, high light, mouse movement, scroll movement, back, add to book mark, number of key input, copy, open bookmark, arrow, select all, page source, print, find, copy hyperlink, forward, stop, frequency, duration, recency, n_th visit, with defaults of 0. All values can be calculated when the user leaves the current web page.

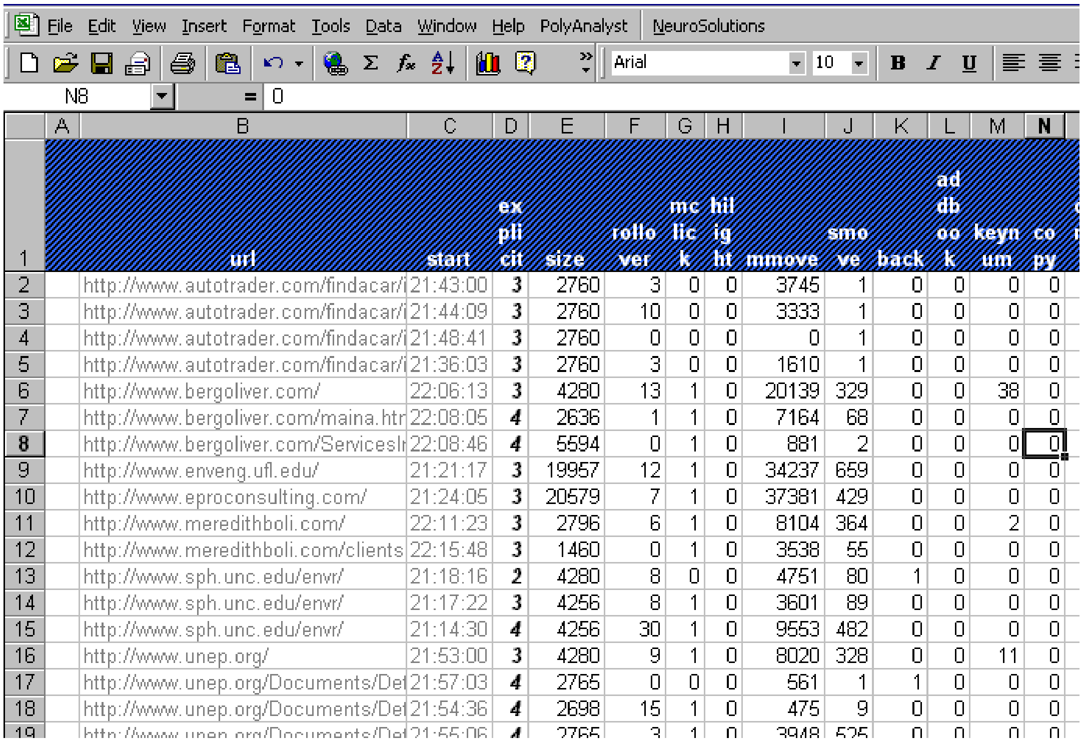
```
url     start   explicitsize    rollover        mclick hilight mmovesmove back
        addbook         keynum          copy    openbook        arrow   selectall
        pagesc print    find    copyhyper       forward         stop    numvisit
        duration        recency         nth

http://www.autotrader.com/findacar/index.jtmpl    1:23:50         3       2760    3
        0       0       3745    1       0       0       0       0       0       0       0
        0       0       0       0       0       0       2       9       10      1
```

Figure 3.7 ASCII log file

3.5.3 Importing this ASCII log file into a statistical package and Excel spread sheet

We chose SPSS 8.0 for regression analysis and Neuro solution in excel for neural network analysis. Two software packages allow me to import data in ASCII format to do analysis on each software. Figure 3.8 is the table formatted user’s implicit data.

Since we changed the ASCII table format to spread sheet table format, we are now ready to use data for the purpose of analysis.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1		url	start	explicit	size	rollover	mclick	hilight	mmove	smove	back	addbook	keynum	copy
2		http://www.autotrader.com/findacar/i	21:43:00	3	2760	3	0	0	3745	1	0	0	0	0
3		http://www.autotrader.com/findacar/i	21:44:09	3	2760	10	0	0	3333	1	0	0	0	0
4		http://www.autotrader.com/findacar/i	21:48:41	3	2760	0	0	0	0	1	0	0	0	0
5		http://www.autotrader.com/findacar/i	21:36:03	3	2760	3	0	0	1610	1	0	0	0	0
6		http://www.bergoliver.com/	22:06:13	3	4280	13	1	0	20139	329	0	0	38	0
7		http://www.bergoliver.com/maina.htr	22:08:05	4	2636	1	1	0	7164	68	0	0	0	0
8		http://www.bergoliver.com/ServicesIr	22:08:46	4	5594	0	1	0	881	2	0	0	0	0
9		http://www.enveng.ufl.edu/	21:21:17	3	19957	12	1	0	34237	659	0	0	0	0
10		http://www.eproconsulting.com/	21:24:05	3	20579	7	1	0	37381	429	0	0	0	0
11		http://www.meredithboli.com/	22:11:23	3	2796	6	1	0	8104	364	0	0	2	0
12		http://www.meredithboli.com/clients	22:15:48	3	1460	0	1	0	3538	55	0	0	0	0
13		http://www.sph.unc.edu/envr/	21:18:16	2	4280	8	0	0	4751	80	1	0	0	0
14		http://www.sph.unc.edu/envr/	21:17:22	3	4256	8	1	0	3601	89	0	0	0	0
15		http://www.sph.unc.edu/envr/	21:14:30	4	4256	30	1	0	9553	482	0	0	0	0
16		http://www.unep.org/	21:53:00	3	4280	9	1	0	8020	328	0	0	11	0
17		http://www.unep.org/Documents/Det	21:57:03	4	2765	0	0	0	561	1	1	0	0	0
18		http://www.unep.org/Documents/Det	21:54:36	4	2698	15	1	0	475	9	0	0	0	0
19		http://www.unep.org/Documents/Det	21:55:06	4	2765	3	1	0	3948	525	0	0	0	0

Fig 3.8 “Neuro Solution” embedded Excel spread sheet’s screen shot

4 Evaluation of implicit interest indicators

In this Chapter, we discuss briefly how we collect data from experimental users and how we analyzed using regression and neural network algorithm. Figure 1.1(b) is the overall architecture of Chapter 4. We used an average MSE (mean square error) to see how well a model predicts compared to the desired explicit rating, which is the real explicit rating on the 10th experiment.

4.1. Collecting data from users

We collected data from 5 experimental users. Four of experimental users were graduate students and the other was undergraduate student and all of them knew well about how to use a web browser. We used 9 training data sets (1st though 9th) and 1 test data set (10th). Each experiment (1st through 10th) is performed 10 times by user 1, user 2, user 3, user 5 and 7 times by user 4. In user 4's case, we used 6 training data sets and 1 test data set. Experimental users could use kixbrowser whenever they felt like to use it. We explained individually for 1 hour for each experiment and we was almost with experimental users while they were doing experiments. We had only one computer available in computer science Ph.D room. Since we had only key access, we opened the door for the experimental users every time they wanted to use the web browser. Since experimental users are good friends of the thesis' author, they joined this experiment without any payment, with only one incentive being "helping my friend by joining his experiment". We believe that this incentive will be very helpful to give us good data. Since we was with experimental users like a watchdog, we told them not to go the restroom or talk while they were using kixbrowser. There were 5 browser icons on the background that had each user's name on it such as xxbrowser where a user's name is "xx". Every individual's file name is changed every time experimental user finishes using kixbrowser for the purpose of the backup and analysis. Finally, we had 10 data log files from each of the 4 users and 7 data log files from user 4.

4.2. Correlations between each candidate interest indicator and explicit rating

In this Chapter, we describe how to find (a few) relevant indicators for each user. We used regression analysis for each candidate interest indicator related to the explicit rating after explaining basic statistical terminology.

To use regression analysis, we tried to find a correlation between every individual interest indicator for each user and their explicit rating so that we can build more accurate predictive models using the combination of those indicators. If we find the non-decreasing positive relationship between an indicator and the explicit rating, we will find the best curve fit for the interest indicator. If a regression analysis explains enough to use the interest indicator, we can add this regression curve form to non-linear or linear regression model. we used one user's case as an example to explain the process. For the other four users' analysis, the same methods and steps as the following example were used.

4.2.1 Statistical terminology

To explain the regression analysis of the output from the experiment, we want to talk about the definitions of statistical terminologies (reference.)

(a) Variance

The variance is a measure of how spread out a distribution is. It is computed as the average squared deviation of each number from its mean. For example, for the numbers 1, 2, and 3 the mean is 2 and the variance is:

$$\frac{(1-2)^2 + (2-2)^2 + (3-2)^2}{3} = .667$$

.

The formula (in summation notation) for the variance in a population is

$$\sigma^2 = \frac{\Sigma(X - \mu)^2}{N}$$

where m is the mean and N is the number of scores.

When the variance is computed in a sample, the statistic

$$S^2 = \frac{\Sigma(X - M)^2}{N}$$

(where M is the mean of the sample) can be used. S^2 is a biased estimate of s^2, however. By far the most common formula for computing variance in a sample is:

$$s^2 = \frac{\Sigma(X - M)^2}{N-1}$$

which gives an unbiased estimate of s^2. Since samples are usually used to estimate parameters, s^2 is the most commonly used measure of variance.

(b) Standard Deviation

The standard deviation is the square root of the variance. It is the most commonly used measure of spread.

An important attribute of the standard deviation as a measure of spread is that if the mean and standard deviation of a normal distribution are known, it is possible to compute the percentile rank associated with any given score. In a normal distribution, about 68% of the scores are within one standard deviation of the mean and about 95% of the scores are within two standards deviations of the mean. The standard deviation has proven to be an extremely useful measure of spread in part because it is mathematically tractable. Many formulas in inferential statistics use the standard deviation. Although less sensitive to extreme scores than the range, the standard deviation is more sensitive than the semi-interquartile range. Thus, the standard deviation should be supplemented by the semi-interquartile range when the possibility of extreme scores is present.

If variable Y is a linear transformation of X such that: Y = bX + A, then the variance of Y is: $b^2\ \sigma^2_x$where $\sigma^2_x$is the variance of X. The standard deviation of Y is b s_x where s_x is the standard deviation of X.

The standard deviation is by far the most widely used measure of spread. It takes every score into account, has extremely useful properties when used with a normal distribution, and is tractable mathematically and, therefore, it appears in many formulas in inferential statistics. The standard deviation is not a good measure of spread in highly-skewed distributions and should be supplemented in those cases by the semi-interquartile range.

The range is a useful statistic to know, but it cannot stand alone as a measure of spread since it takes into account only two scores.

The semi-interquartile range is rarely used as a measure of spread, in part because it is not very mathematically tractable. However, it is influenced less by extreme scores than the standard deviation, is less subject to sampling fluctuations in highly- skewed distributions, and has a good intuitive meaning. It should be used to supplement the standard deviation in most cases.

(c) The F distribution is the distribution of the ratio of two estimates of variance. It is used to compute probability values in the analysis of variance. The F distribution has two parameters: degrees of freedom numerator (dfn) and degrees of freedom denominator (dfd). The dfn is the number of degrees of freedom that the estimate of variance used in the numerator is based on. The dfd is the number of degrees of freedom that the estimate used in the denominator is based on. The dfd is often called the degrees of freedom error or dfe. In the simplest case of a one-factor between-subjects ANOVA,

dfn=a-1

dfd = N-a

where "a" is the number of groups and "N" is the total number of subjects in the experiment. The shape of the F distribution depends on dfn and dfd. The lower the degrees of freedom, the larger the value of F needed to be significant. For instance, if dfn = 4 and dfd = 12, then an F of 3.26 would be needed to be significant at the .05 level. If the dfn were 10 and the dfd were 100, then an F of 1.93

(d) The mean square error (MSE) is an estimate of the population variance in the analysis of variance. The mean square error is the denominator of the F ratio.

4.2.2 Finding the correlation between every individual's interest indicator for each user and explicit rating

In Figure 4.1, the error bar chart plots a confidence interval 95% which means that 95% of values are inside between the lowest bar and the highest bar for each distinct explicit rating. The box between the bars is the mean of each explicit rating. We tried to find whether or not the average of each indicator consistently increases or decreases as the explicit rating increases. If it increases or decreases consistently as the explicit rating increases, we want to use the user's interest indicator as a part of a predictive regression model. Using all the indicators that have a non-decreasing positive or negative relationship with the explicit rating, we can build predictive models. Here, we can reject or accept the hypotheses that we had on Chapter 3 according to Figure 4.1. Also, we can choose a candidate indicator to be used on the multiple indicator regression model. The explanation of indicators in Figure 4.1 follows:

(a)Number of Mouse Clicks versus Explicit Rating

Figure 4.1 (a) depicts an error bar chart of the number of mouse clicks versus explicit rating. The average of the number of mouse clicks increases consistently as the explicit ratings increase by 1. The 95% of confidence level also shows that the upper bound and lower bound of mouse clicks increases consistently as the explicit rating increases. Therefore, we might use this indicator to build the predictive models. In conclusion, the hypothesis that the number of mouse clicks versus explicit rating has a non-decreasing positive relationship. So, we have this indicator as a candidate as a part of the predictive model for this user.

(b)Number of Highlight versus Explicit Rating

Figure 4.1 (b) depicts an error bar chart of the number of highlighting versus the explicit rating. The average of the number of highlighting clicks increases consistently as the explicit ratings increase. Therefore, we might use this indicator to build predictive models. But, the lower bound of the number of highlight does not increase as the explicit rating increases. This 95% confidence level of the number of highlighting in the “2” explicit rating covers the whole 95% confidence level of the “1” explicit rating. It is not as good as the number of mouse clicks. We don’t know yet whether we can use it for the predictive models or not. It depends on how much a curve fit of this interest indicator can be explained by the regression. As we hypothesized, we found that the number of highlighting has a non-decreasing positive relation to this user’s explicit rating. This indicator can be a candidate to be used as a part of the predictive model for this user.

(c)Number of Key input versus Explicit Rating

Figure 4.1 (c) depicts an error bar chart of the number of key inputs versus the explicit rating. The average of the number of key inputs decreases consistently as the explicit rate increases by 1. Therefore, as we hypothesized, we found that the number of key inputs has a non-decreasing positive relation to this user’s explicit rating. This indicator can be a candidate to be used as a part of the predictive model for this user.

(d)Size versus Explicit Rating

Figure 4.1 (d) depicts an error bar chart of the file size versus the explicit rating. In the “1” explicit rating, the average of file size is the biggest. But, in the next explicit rating, “2” explicit rating, the file size is the smallest, which means that the file size does not increase or decrease consistently as the explicit rate increases by 1. We found the file size does not have a non-decreasing positive or negative relation to this user’s explicit rating. So, the hypothesis we mentioned is rejected. In conclusion, in this user’s case, there is no relationship between the file size and the user’s explicit rating. We don’t choose this implicit indicator to build the regression models in this user’s case.

(e)Number of Copy versus Explicit Rating

Figure 4.1 (e) depicts an error bar chart of the number of copy versus the explicit rating. The average of the number of copy in “1” explicit rating is greater than in the “2” explicit rating and the average of the number of copy in “3” explicit rating is greater than in the “2” explicit rating. It means that the average of the number of copy doesn’t increase consistently as the explicit rating increases. Therefore, it cannot be a candidate to be a part of the predictive model. However, you can see it increase as the explicit rating increases from the “2” explicit rating to the “4” explicit rating. We found the number of copy. We don’t choose this implicit indicator to build the regression models in this user’s case.

(f)Number of Rollover versus Explicit Rating

Figure 4.1 (f) depicts an error bar chart of the number of rollover versus the explicit rating. The average of the number of rollover in the “1” explicit rating and the average of the number of rollover in the “2” explicit rating is greater than in the “1” explicit rating. But, the average of the number of rollover in the “3” explicit rating is less than in the “2” explicit rating. It means that the average of the number of the rollover doesn’t increase consistently as the explicit rating increases. Therefore, it cannot be a candidate to be a part of the predictive model. However, except the “2” explicit rating’s case, it increases consistently as the explicit rating increases. But, in this user’s case, the hypothesis that the number of rollover increases monotonously as the explicit rating increases is rejected. We don’t choose this implicit indicator to build the regression models in this user’s case.

(g)Number of Mouse movement versus Explicit Rating

Figure 4.1 (g) depicts an error bar chart of the number of mouse movement versus the explicit rating. The average of the distance of the mouse movement does not increase consistently. It rejects the hypothesis that, for this user, the number of mouse movement and the explicit rating doesn’t have a non-decreasing positive or negative relationship. We don’t choose this indicator as a candidate implicit indicator to build the regression models in this user’s case.

(h)Number of Add to bookmark versus Explicit Rating

Figure 4.1 (h) depicts an error bar chart of the number of adding to bookmarks versus the explicit rating. We hypothesized that the number of adding a web page to bookmarks will have a non-decreasing positive relation to the explicit rating. The number of adding a web page to bookmarks doesn’t increase consistently as the explicit rating increases. It rejects the hypothesis. We don’t choose this indicator as a candidate implicit indicator to build the regression models in this user’s case.

(i)Number of Select All versus Explicit Rating

Figure 4.1 (i) depicts an error bar chart of the number of sellecting all function versus the explicit rating. The number of selecting all doesn’t increase consistently as the explicit rating increases. It rejects the hypothesis. We don’t choose this indicator as a candidate implicit indicator to build the regression models in this user’s case.

(j)Number of Page source versus Explicit Rating

Figure 4.1 (j) depicts an error bar chart of the number of viewing page sources versus the explicit rating. This figure shows that this user only view page sources when he is very interested in the page. Our hypothesis that the number of viewing the page source is related with this user's interest level is accepted because the number of viewing the page source and the explicit rating has a non-decreasing positive relationship. We found that he is very interested in the web page whenever he views the page source. The number of viewing page source doesn't increase consistently as the explicit rating increases. It rejects the hypothesis. We don't choose this indicator as a candidate implicit indicator to build the regression models in this user's case.

(k)Number of Print versus Explicit Rating

Figure 4.1 (k) depicts an error bar chart of the number of print versus the explicit rating. Figure 4.1 (k) accept the hypothesis that we have because, in this user's case, there is non-decreasing positive relationship between the number of prints and the explicit rating. We found this user uses print function on the web browser whenever he is very interested in this web page. But, this indicator cannot be used as a part of the regression model since the user's printing behavior does not often happen. The number of prints doesn't increase consistently as the explicit rating increases. It rejects the hypothesis. We don't choose this indicator as a candidate implicit indicator to build the regression models in this user's case.

(l)Number of Forward versus Explicit Rating

Figure 4.1 (l) depicts an error bar chart of the number of using the forward button versus the explicit rating. The number of forwarding a web page doesn’t increase consistently as the explicit rating increases. It rejects the hypothesis. We don’t choose this indicator as a candidate implicit indicator to build the regression models in this user’s case.

(m)Number of Stop versus Explicit Rating

Figure 4.1 (m) depicts an error bar chart of the number of using stop button versus the explicit rating. The number of stopping in a web page doesn’t increase consistently as the explicit rating increases. It rejects the hypothesis. We don’t choose this indicator as a candidate implicit indicator to build the regression models in this user’s case.

(n)Time on Page versus Explicit Rating

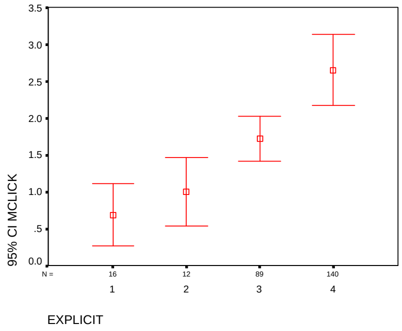
Figure 4.1 (n) depicts an error bar chart of the time spent on a page (duration) versus the explicit rating. We used an error bar chart to use this user’s indicator as a part of a predictive regression model for this user. The number of adding a web page to bookmarks doesn’t increase consistently as the explicit rating increases. It rejects the hypothesis. We don’t choose this indicator as a candidate implicit indicator to build the regression models in this user’s case. But, we notified the duration in the explicit rating “1” and “2”’s average is less than the duration in the explicit rating “3” and “4”.

(o)Number of Visits (Frequency) versus Explicit Rating

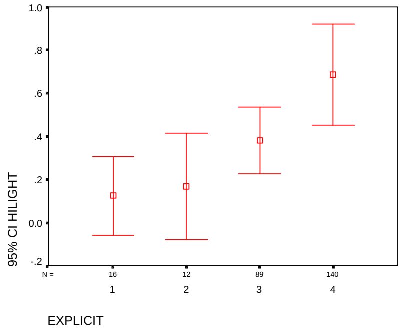
Figure 4.1 (o) shows an error bar chart of the number of visits (Frequency) versus the explicit rating. The number of the frequency doesn't increase consistently as the explicit rating increases. It rejects the hypothesis. We don't choose this indicator as a candidate implicit indicator to build the regression models in this user's case.

(p)Recency versus Explicit Rating

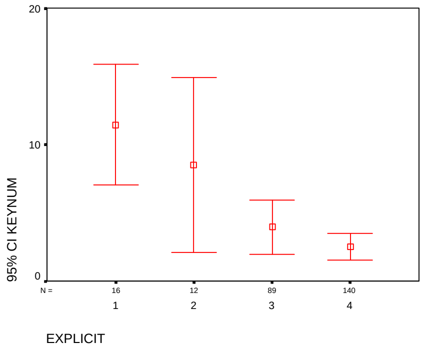
Figure 4.1 (o) shows an error bar chart of how recently the user visited in the last time (Recency) versus the explicit rating. The number of adding a web page to bookmarks doesn't increase consistently as the explicit rating increases. It rejects the hypothesis. We don't choose this indicator as a candidate implicit indicator to build the regression models in this user's case.



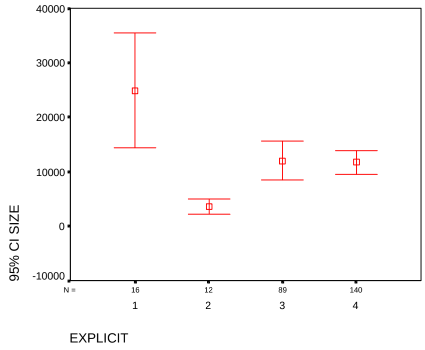
(a) Mouse clicks vs. Explicit



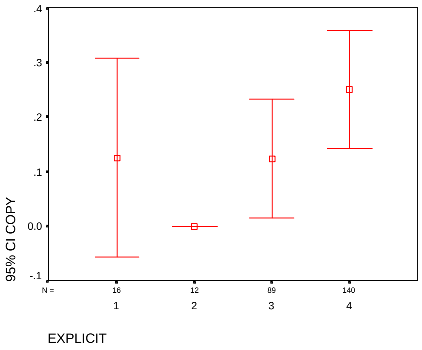
(b) Highlight vs. Explicit



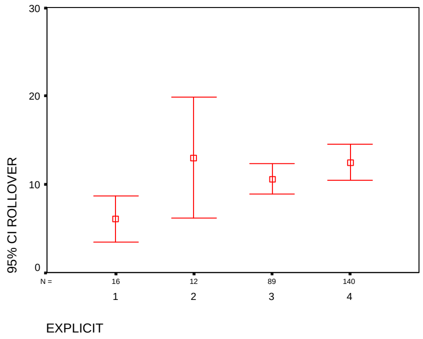
(d) Key input vs. Explicit



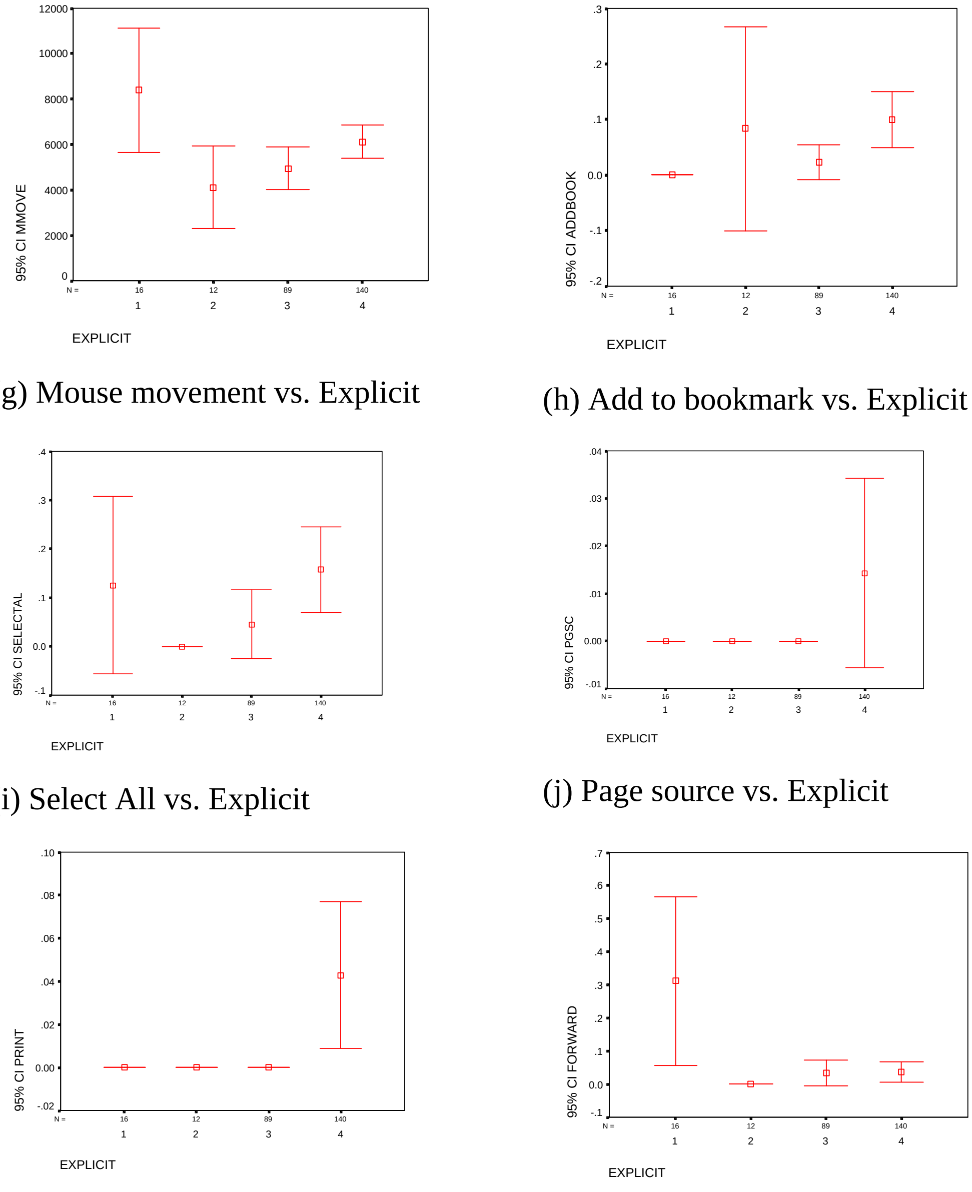
(d) Size vs. Explicit



(e) Copy vs. Explicit



(f) Rollover vs. Explicit



(g) Mouse movement vs. Explicit

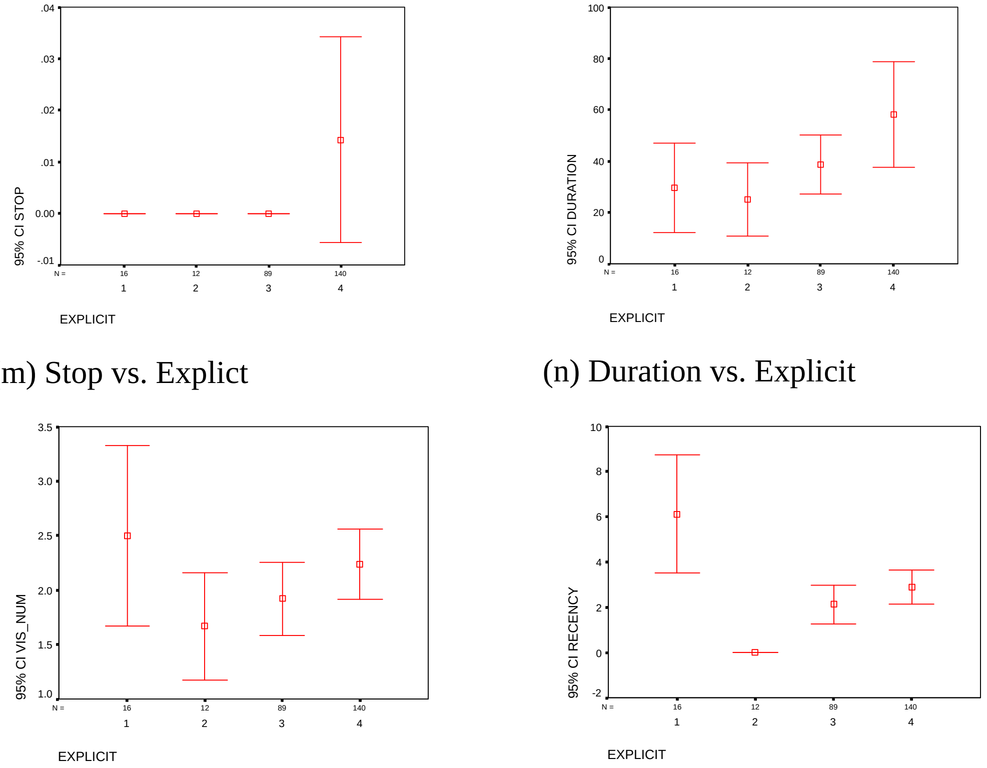
(h) Add to bookmark vs. Explicit

(i) Select All vs. Explicit

(j) Page source vs. Explicit

(k) Print vs. Explicit

(l) Forward vs. Explicit



(m) Stop vs. Explict

(n) Duration vs. Explicit

(o) The number of visits (Frequency) vs. Explicit

(p) Recency vs. Explicit

Figure 4.1 Error bar charts for 16 candidate predictors

Due to space reasons, for four of the users, the figures are in the appendix.

In user 1’s case, his interest indicators are the number of mouse clicks, the distance of scrollbar movements. But, in user 2’s case, his interest indicators are the number of mouse clicks and the number of forwarding a web page. User 3’s interest indicators are recency, the number of mouse clicks and the number of visits. User 4’s interest indicators are duration, the number of visits, the number of copy, the number of pressing back button and the recency. User 5’s interest indicators are the number of mouse clicks, the number of key inputs and the number of highlightings.

In conclusion, we found that the number of mouse clicks was the most common for these 5 users and every user interest indicator is different.

4.3 Regression Analysis on individual implicit interest indicators and mouse clicks only regression analysis

In this Chapter, we discuss how to find (a few) more fitted functions or the most fitted function for each of the relevant indicators found in 4.2, so that we can reduce the number of functions to be considered in 4.4.

If the candidate indicator shows any simple positive or negative relation with the explicit rating, it can be a candidate indicator to be used in linear/non-linear analysis. In the next Chapter 4.4, multiple combined implicit interest indicators will be used to build more accurate predictive models that will become a part of regression model.

We tried to have a best curve fit for each indicator to use that curve as a part of the predictive model. In case of highlighting, R squared value was 0.0277 for linear and 0.0275 for exponential curve, which means that this linear regression explains 2.77% of data and this exponential regression explains 2.75% of data. The difference of R squared values between two curve forms is 0.02% of data. If we have 10000 implicit interest data collected from a user, there is 2 data, which is closer to linear curve than exponential curve. We cannot choose linear form since the linear curve form explains the regression 0.02% more. So, we assume that each independent indicator has a normal distribution around the average on the each explicit rating 1 through 4 and the weight on each explicit rating 1 through 4 is uniformly distributed. We assume this because we estimate that we would need to have about more than 1 year's experimental user data to have a distribution for each explicit rating level and can’t find the other way. That is the reason that we used the average of indicator (independent variable) s data for each explicit rating level and we have a curve fit using each indicator’s average to see if it helps to predict the test set and showed those model had a good predictive ability.

In Figure 4.2, we tried to find the best-fit curve using the average of each indicator. There’s a brief statistical summary above each curve fit graph. “b0” and “b1” are constant for linear, log_e, inverse, power and exponential. So, these shapes are as follows:

Linear (LIN): b0+b1*highlight
Log   : b0+b1*log_e(highlight)
Inverse (INV):b0+b1/(highlight)
Power (POW): b0+Power (highlight,b1)
Exponential (EXP): b0+b1*exp (highlight)

This model is applicable for the following curve fit graphs. The dependent variable is explicit rating and the independent variable is each indicator.
“Mth” means mathematical form for the curve fit. “Rsq” means R squared: Goodness-of-fit measures a linear/non-linear model and is sometimes called the coefficient of determination. It is the proportion of variation in the dependent variable explained by the regression model. It ranges in value from 0 to 1. Small values indicate that the model does not fit the data well.

The F distribution is the distribution of the ratio of two estimates of variance. “F” means the f value of the F distribution. “d.f” stands for degrees of freedom used to calculate the F value. “Sigf” stands for Observed Significance Level often called the p value. It is the basis for deciding whether or not to reject the null hypothesis.

It is the probability that a statistical result as extreme as the one observed would occur if the null hypothesis were true. If the observed significance level is small enough, usually less than 0.05 or 0.01, the null hypothesis is rejected.

We underlined the best curve fit on the brief SPSS statistic summary. The formula for R squared is

$$R^2 = 1 - \frac{SS_{error}}{SS_{total}}$$

Another formula, which is mathematically equivalent is

$$R^2 = \frac{SS_{regression}}{SS_{total}}$$

where SSregression is the difference between SStotal and SSerror.

R^2 can be a lousy measure of goodness-of-fit, especially when it is misused.
By definition, R^2 is the fraction of the total squared error that is shown by the model. Thus values approaching one are desirable. But some data contain *irreducible* error, and no amount of modeling can improve on the limiting value of R^2. Sadly, many practitioners, including some who should know better, pursue very high order polynomial models in the mistaken but widely held belief that as the number of parameters approaches the number of observations, the model can be made to pass thorough every point. (It appears that the origin of this misconception is, as with many difficulties with applied statistics, not reading the fine print.)

Dependent	Mth	Rsq	d.f.	F	Sigf	b0	b1
EXPLICIT	LIN	.906	2	19.34	.048	.8577	4.7951
EXPLICIT	LOG	.969	2	61.85	.016	4.6463	1.6667
EXPLICIT	INV	.971	2	67.64	.014	4.4505	-.4419
EXPLICIT	POW	.876	2	14.12	.064	5.7253	.7380
EXPLICIT	EXP	.761	2	6.38	.127	1.0980	2.0468

Best Curve fit

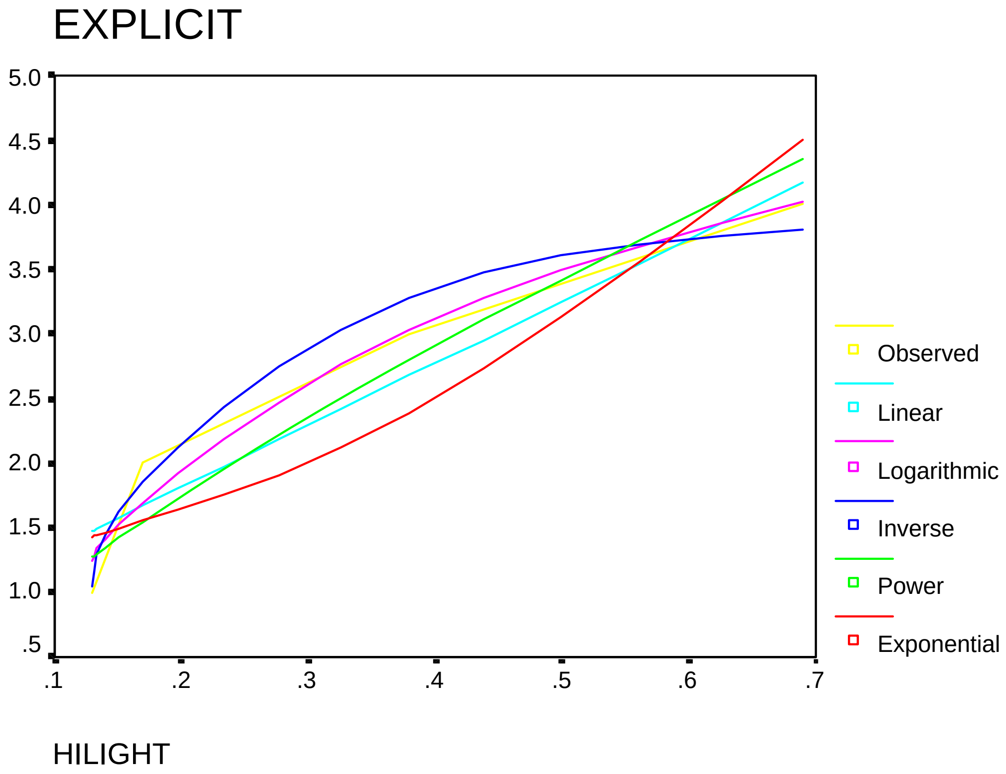


Figure 4.2 Best curve fit for highlighting text

```
Independent:  MCLICK
 Dependent   Mth     Rsq     d.f.  F            Sigf      b0         b1
  EXPLICIT LIN     .957     2     44.40       .022     .3035       1.4498
  EXPLICIT LOG  .995     2     428.52      .002     1.8773    2.1736
  EXPLICIT INV    .975     2     78.97       .012     4.7864   -2.6836
  EXPLICIT POW  .941     2    31.69       .030     1.6697   .9840
  EXPLICIT EXP   .842     2     10.68      .082     .8478     .6334
```

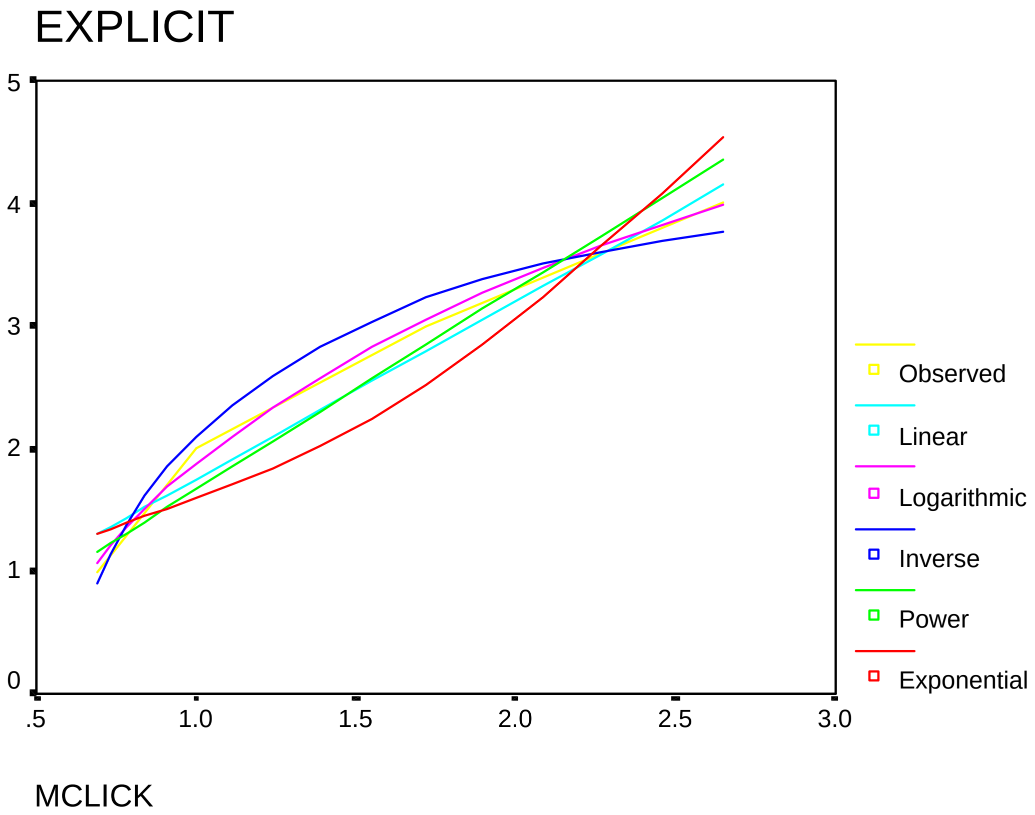


Figure 4.4 Best curve fit for the number of mouse clicks

4.4 Regression analysis on multiple implicit interest indicators

In this Chapter, we use the few more fitted functions found in 4.3 for the relevant indicators found in 4.2 and perform regression on multiple indicators. Once we have each correlation between every individual's interest indicator for each user and explicit rating, we can now start to do regression analysis on multiple implicit interest indicators. We hypothesized that regression analysis on multiple implicit interest indicators will give us the best predictive model since the more indicators a model has, the more accurate the predictive ability of the model is.

4.4.1 Linear predictor system

We used the SPSS statistic package to calculate the linear coefficients. Since we found the positive or negative non-decreasing relationship between each indicator and explicit rating, we used these indicators but it is obvious that the linear predictor system will not be as good as the non-linear predictor system. However, the linear predictor system's time spent on calculation will be shorter than the non-linear predictor system. In other words, it will take less time to compute this model. The linear predictor systems used the same indicators as the non-linear predictor's indicators but only the linear form of each indicator. The process to have the linear regression predictive model is the same as non-linear predictor system and consequently they will be explained together. The comparison between the non-linear predictor system and the linear predictor system is shown in Figure 4.4.

4.4.2 Non-linear predictor system

Since we have a best curve fit for each indicator, the indicators' forms are the same but the coefficient will have to be adjusted since we added more indicators in this predictor system. We used the SPSS statistic package and the same user as Chapter 4.2 to calculate the following non-linear predictive model:

Explicit rating= b1*mclick+b2* ln(keynum+1)+b3/(highlight+1)+b4

Also, this includes the linear form since we tried to find the best predictor model for the linear and non-linear combined regression predictor systems. So, this linear form will be like this:

Explicit rating=b1*mclick+b2*keynum+b3*hilight+b4

The higher R squared value the predictive model has, the more reliable the predictive model is as long as the standard error is less than or equal to 0.3, which means we allowed 30% error on the estimated coefficient. The reason that we add 1 to highlight indicator is that the standard error is too big and we found when we added 1 to highlight, standard error became smaller than what we allowed on this analysis. Results are as follows:

Explicit rating= .119909319 * mclick -.198639789 * LN(keynum+1) +.312951871 / (hilight+1) + 2.987368849

SPSS statistical summaries are following:

```
Nonlinear Regression Summary Statistics
Dependent Variable EXPLICIT
 Source               DF    Sum of Squares     Mean Square
 Regression            5    2952.35984         590.47197
 Residual             252   152.64016          .60571
 Uncorrected Total    257   3105.00000
 (Corrected Total)    256   180.14008
 R squared = 1 - Residual SS / Corrected SS =   .15266

                                  Asymptotic 95 %
                 Asymptotic     Confidence Interval
 Parameter   Estimate       Std. Error      Lower          Upper


 B1          .119909319   .032312065   .056273215   .183545423
 B2          -.198639789   .041035648   -.279456313  -.117823265
 B3          -.312951871   .280240928   -.864864623   .238960880
 B4          2.987368849   .297562868   2.401341881  3.573395817
```

Figure 4.4 The summary from the SPSS regression analysis

In Figure 4.4, the asymptotic 95% confidence Interval tells us about the lower bound and upper bound. A value from this model will appear with 95% confidence between the lower bound and the upper bound. The parameter is a coefficient that we would like to find to build a model.

We built this non-linear model within 30% error and with R square =0.15266.

The non-linear regression predictive regression model uses all possible predictors to predict user interest level. We hypothesized in the Chapter 1 that every user has a different predictive model. As you can see in Table 4.2, every user has a different model in the multiple indicator predictive model.

User	Explicit Rating Score
User 1	.760762387* $\text{mclick}^{0.5467}$ +0.068768222*ln(smove+1) +2.507089846
User 2	13.764728147+0.151431459*mclick+261.68100798/(forward-22.20515502)
User 3	.059450657* $\text{recency}^{0.5677}$+.259073980* LN(mclick + 1) + .011012400* $\text{vis\_num}^{2.1559}$ + 2.579012481
User 4	.160800300 *ln(duration+1)+.034649989*vis_num + .173556020 *copy -.043861436*exp(7.1997*back)+.118660875*Ln(recency+1)+ 2.158614140
User 5	.119909319*mclick-.198639789*LN(keynum+1)+.312951871/ (hilight+1) + 2.987368849

Table 4.2 Non-linear prediction model for 5 users.

Back: The number of back button clicks on a specific page.

Copy: The number of copy actions on a specific page.

Forward: The number of forward button clicks on a specific page.

Duration: The time spent on a specific page.

Hilight: The number of highlighting on a specific page.

Keynum: The number of key inputs on a specific page.

Recency: How recently he visited a specific page.

Smove: The distance of scrolling movement on a specific page.

Vis_num: The number of visits on a specific page.

4.4.3 Neural Network Analysis

We tried to use a neural network analysis since there seems to be a pattern on the predictive model. We doubt that a neural network analysis will give us the better result for prediction since we only used the user's behavior. If we had a content analyzer that could be combined with my data sets, it seems reasonable to be able to detect some patterns. Neural network doesn't give me really good result since the user's behavior alone is not enough to detect patterns and we believe that we would need to have at least 12 months of data of each user.

By training several inputs, we can have weights and outputs.

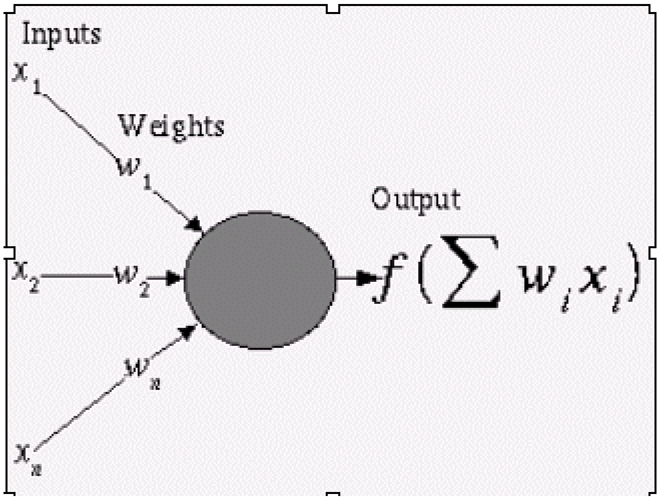


Figure 4.5 Simple structure of neural network

Figure 4.5 depicts single perceptrons that can only express linear decision surfaces. We can see input x_i and weight w_j for each node i and weights will be changed by training the neural net. We will get the expected output. In contrast, the kind of multiplayer networks learned by the BACKPROPAGATION algorithm are capable of expressing a rich variety of nonlinear decision surfaces. In this study, we used BACKPROPAGATION algorithm with 1 hidden layer and 5 nodes. One major difference in the case of multiplayer networks is that the error surface can have multiple local minima, in contrast to the single-minimum parabolic error surface. It means that gradient descent is guaranteed only to converge toward some local minimum, and not necessarily the global minimum error. This structure is shown in Figure 4.6.

Neural network terminology (Smith, 2001):

a) Training Set

A collection of input-output patterns that are used to train the network

b) Testing Set

A collection of input-output patterns that are used to assess network performance

c) Learning Rate-*η*

A scalar parameter, analogous to step size in numerical integration, used to set the rate of adjustments.

We can apply the value of each input parameter to each input node. Input nodes compute only the identity function. In Figure 4.6, this neural network has 1 hidden layer in the middle and the node is called "neuron".

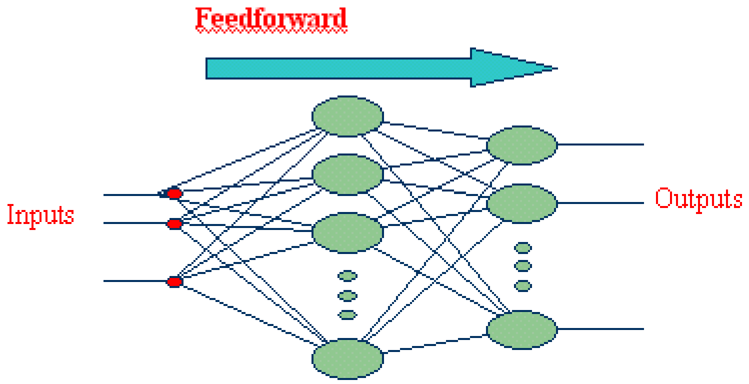


Figure 4.6 Multi perceptrons neural network structure

The hidden layer learns to *recode* (or to *provide a representation* for) the inputs. More than one hidden layer can be used. The architecture is more powerful than single-layer networks: it can be shown that any mapping can be learned, given two hidden layers (of units).

Formal algorithm of Stochastic Backpropagation (training examples, η, n_i, n_h, n_o) :

Each training example is of the form $\langle \vec{x}, \vec{t} \rangle$ where $\vec{x}$ is the input vector and $\vec{t}$ is the target vector. η is the learning rate (e.g., .01). n_i, n_h and n_o are the number of input, hidden and output nodes respectively. Input from unit *i* to unit *j* is denoted x_{ji} and its weight is denoted by w_{ji}.

Create a feed-forward network with n_i inputs, n_h hidden units, and n_o output units.
Initialize all the weights to small random values (e.g., between -.05 and .05)
Until termination condition is met, Do

For each training example $\langle \vec{x}, \vec{t} \rangle$, Do

1. Input the instance $\vec{x}$ and compute the output o_u of every unit
2. For each output unit *k*, calculate its error term (In our study, k=1 since there is only one explicit rating output)

$$\delta_k = o_k(1 - o_k)(t_k - o_k)$$

3. For each hidden unit *h*, calculate its error term

$$\delta_h = o_h(1 - o_h) \sum_{k \in Downstream(h)} w_{kh}\delta_k$$

4. Update each network weight w_{ji} as follows:

$$\begin{aligned} w_{ji} &\leftarrow w_{ji} + \Delta w_{ji} \\ \text{where} \quad \Delta w_{ji} &= \eta \delta_j x_j i \end{aligned}$$

To do neural network analysis, we used the software, “Neuro Solution”. We used 1 hidden layer and a 5 node neural network and an epoch of 1000. Also, we tried to have more layer and nodes but it did not have any significant difference. The learning rate (step size) is set to 0.001 and There are 9 training sets and 1 testing set for user 1, user 2, user 3, user 5 and 6 training sets and 1 testing set for user 4. Once we trained the neural nets enough (1000 epoch in this experiment), we used trained weights to predict the test data set in comparison to the real desired explicit score value. Using “Neuro Solution” software, we can have explicit rating and predicted output (explicit score).

4.4.4 Comparative analysis of regression techniques

In table 4.4, we can compare a predictor model to the others so that we can see how well a model fits. The table’s pixel values are the model’s MSE (Mean Square Error). Here follows a comparison of regression analysis and neural network analysis.

The user’s interest predictive model shows us that every user has a different model even though they have a common predictor such as the number of mouse clicks. Also, they might have the same interest indicators but a different shape of the model. For example, when user 1 and user 2 have a common indicator, the number of mouse clicks, user 1's mouse click model might look linear and user 2's mouse clicks shape might be a power or exponential. Every individual model has a different model containing the interest indicators.

	User 1	User 2	User 3	User 4	User 5
Multiple non-linear regression	1.3319	1.3071	1.1666	0.5932	2.2872
Mouse clicks only regression	1.3292	1.4362	1.612	0.6344	1.5736
Neural network	1.6423	2.3248	1.5555	2.9615	1.6861
Multiple linear regression	1.2551	1.4220	1.2009	0.5632	1.6861

Table 4.4 Comparison of models' MSE

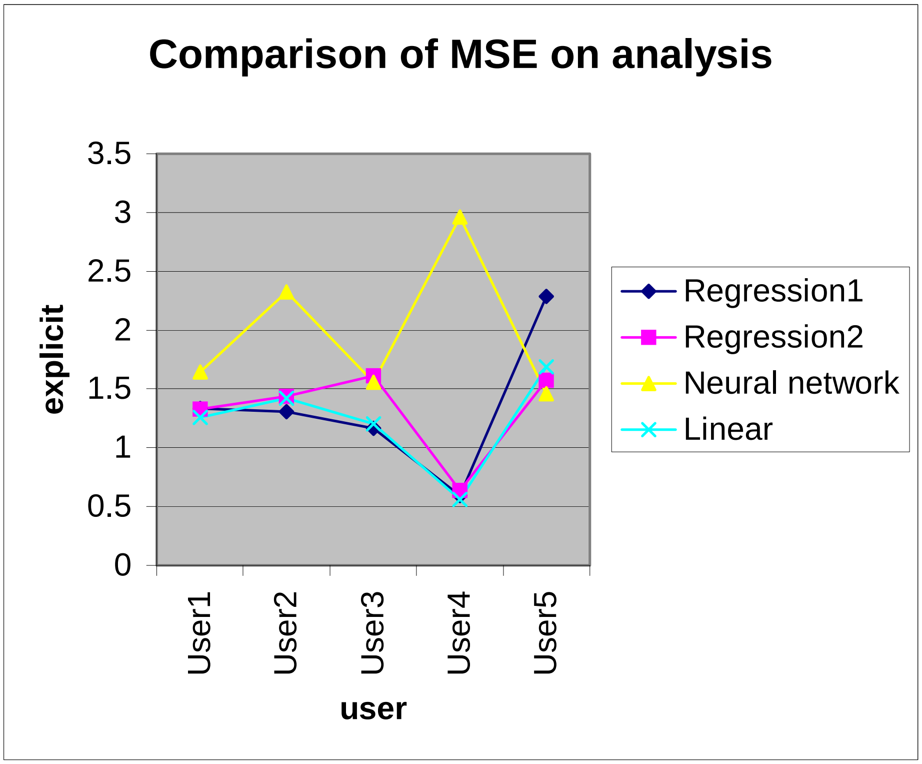


Fig 4.8 Comparison of MSE on analysis

Regression1: Non-linear regression model.

Regression2: Mouse clicks only regression model.

Table 4.4 and Figure 4.8 depict the comparison of the MSE. We used the MSE to tell how good the regression models' precisions are and how well data is distributed around the regression model. The smaller MSE of the model is, the better the model's precision is and the range of distribution of data became smaller. Since we have 1 through 4 explicit ratings, we can have the MSE value, 0 through 9.

The comparison of the MSE on each model shows how well each model predicts the testing data set using each user's model. According to Fig 4.8, the multiple indicator non-linear regression model predicts almost best and in case of user 1, user 2, user 3, user 4, the MSE of the predictive regression model is below 1.5, which means the MSE of the predictive model is low enough to tell that this model has a good predictive ability. But, in case of user 5, the MSE of the predictive regression model is somewhat high but still good. The multiple indicator linear regression model has very good prediction for the 5 users, too. It means that we can use linear regression model for 5 users without using the multiple indicator regression model. Mouse-clicks-only regression model has also very good predictive ability. So, in conclusion, we can use only one indicator, the number of mouse clicks, to predict these 5 users' interest level. However, the precision is not as good as the linear or non-linear regression analysis because the overall MSE of mouse click only regression model is greater than the MSE of the multiple indicator non-linear and linear regression model. At last, we used neural network to predict a user's interest level. Neural network also predict a user's interest level very well. But, it is not as good as the three regression predictive models.

For each user, there is the best model that has the least MSE value. For example, in user 1's case, we'd better use the multiple linear predictive regression model since the multiple linear predictive regression model has the least MSE value.

4.5 Mouse Clicks only regression model

There are several implicit indicators that can be used alone to build a predictive model. There are candidates following:
Duration, the number of mouse click, the distance of scrollbar movement, the distance of mouse movement, recency and file size.

The requirement to be a candidate is that a candidate should be used every time that a user visits a web page. If it is not used every time a user visits a web page, such as the "copy" function, the predictive system can not predict any reasonable result since "copy" action is not performed on the page. It would be very unwise to use "copy" as an indicator in this case.

From Figure 4.2 and Figure 4.3, only the number of mouse clicks can be chosen for the purpose of building one indicator regression model since the number of mouse click is used almost always at least one time whenever a user visits a web site. In case of the number of highlighting text, it does not happen often, nor does the number of key input. In this case, we can only use mouse clicks. Also, in the other users' cases, the number of mouse click alone was a good indicator according to candidate requirements. But, this model is not as good as the multiple indicators predictive models that have more indicators, which makes the prediction system more accurately.

We hypothesize that the only one indicator, the number of mouse clicks can sufficiently predict the user's behavior. Following results are when the model is applied to the test set.

User	Explicit Rating Score
User 1	$.775932172 * \mathrm{mclick}^{0.5467} + 2.808878149$
User 2	1.966219391 + .154819088* mclick
User 3	.244654146*ln(mclickc+1)+ 2.703313651
User 4	.001680012*mclick+2.807800756
User 5	.090576553*mclick+3.180757297

Table 4.1 Mouse-only prediction model for 5 users

We have a predictive model using only the number of mouse clicks. If this model has an acceptable level of predictive ability, we can use only this indicator without making any effort to calculate and detect the other user's implicit indicators. Results are shown in Figure 4.4. It was third best predictive model among the four candidate predictors and has a good predictive ability. Therefore, The above hypothesis is accepted. In conclusion, we can use only mouse clicks to predict the 5 experimental user's interest level.

4.6 Discussion

We studied the related work in Chapter 2. Since Curious browser's experiments were very similar to ours in a sense that they used a web browser on their experiments, we would like to discuss about their conclusions. Curious browser had found that the several user interest indicators had relationship between the explicit rating and users' behavior in the web page. We could detect more and general user's behavior and build the predictive models. Also, we found that every 5 experimental user has different interest indicators. A user's duration (the time spent in the web page) in a web page is related with his explicit rating. But, it does not tell us that the duration is also a good interest indicator to the other users. Also, Curious browser’s experiments concluded the number of mouse clicks is related with the user’s interest level by detecting the time that the mouse moves in the web page. They couldn’t use the distance of mouse movements but the time that the mouse moves that might give us wrong results while we can use the distance of mouse movements in the web page. Jeremy’s web agents could detect three surrogates since they had technical problems and limits to predict user’s interest level while we generally included surrogates as user interest indicators. They tried to find the amount of user scrolling activity by recording the time that the scroll moves while we could record the real distance of the scrollbar movements. In addition to that, to record the amount of user mouse activity, they recorded the number of change on the status bar while we could record the distance of mouse movements and the number of mouse clicks and the number of highlightings. In this way, we could cover most of indicators they mentioned and could build the predictive regression models. Nichols (Nichols et al., 1997) presented a list of potential types of user’s behaviors. We covered most of their lists to build the predictive model to predict a user interest level.

In conclusion, we found that every user has different interest indicators. Therefore, every user has different predictive models. So, any predictive system is supposed to be personalized to predict a user’s interesting level in the web page.

5 Summary of results & future work

5.1 Summary of contribution

There are two major contributions in this thesis paper. We…

- Built a software that can detect and record various user' behavior
- Built the three regression predictive models and neural network to look for a better predictive model

Since none of previous research could detect generally user's behavior for each user, they could not build a predictive system without any content analyzer. In this research, we have experimentally evaluated the effectiveness of several implicit interest indicators in determining the explicit interest in a web page and made different personalized models to predict each user's interest level. In addition to that, we built an extractor (a kind of parser) to have a well-formatted log file from a raw log file and had 5 experimental user's implicit interest rating data sets.

In conclusion, we found that there is a common user's indicator such as the number of mouse clicks. However, not all interest indicators are common for every user because every user has different interest indicators since every user behaves differently in the interesting or uninteresting web page. Therefore, every user has different predictive models. So, any predictive system is supposed to be personalized to predict a user's interesting level in the web page. Also, we showed it was very possible and accurate to predict a user's interest level only using user's implicit interest indicator.

5.2 Strength and weakness

Here, we discuss about our system’s advantages and disadvantages or limitations. While the explicit rating has a high cost, or need some incentives, detecting user’s behavior implicitly doesn’t cost almost anything except the increased load on the PC. By using only implicit interest indicators, we could build models for each user and predict each user’s interest level according to each user’s behavior on the web page. Every user’s behavior implies that a user’s interest level is different from the other users. The most common indicator, the number of mouse clicks could approximately predict the general user’s interest level. Fig 4.2 shows that the multiple indicator non-linear regression predictive models predict the overall best followed by the multiple indicator linear regression predictive models, the mouse clicks only non-linear regression models and the neural network predictive model in order. We assumed that the explicit rating was 100% accurate and that kixbrowser functioned the same as popular web browser IE. Also, we assume that the difference between the explicit ratings is the same. To build predictive model for each user, personalized statistical analysis was very effective.

5.2.1 Advantages of this system

This system:

- Found new user’s interest indicators and predictive models to predict a user’s interest level when he leaves a web page after he visits
- Is personalized so that every user has a different weight for each indicator
- Covers generally what Nichols et al. (1997) mentioned
- Content-based system can be combined with this system for the further evaluation

5.2.2 Disadvantages of this system or something not covered on this experiment

There are some disadvantages by using this web browser.
Kixbrowser…

- Doesn’t have content-based system. But, this system will be adapted to any system easily
- Is slower than popular commercial browser such as Netscape and Internet Explorer
- Support JavaScript partially
- Doesn’t support for Flash, doc, ps, pdf and multiligual text

But, it supports html and txt files perfectly. So, this browser is usually used for academic web site. With this reason, we strongly recommend that a user might use this browser for the purpose of browsing academic web sites. Also, in this predictive system there might be a better predictive model since we found a regression predictive model by adding all positive or negative indicators related to a user’s explicit rating. There might be a better predictive model by dividing and multiplying user’s interest indicators.

As we discussed earlier, a user’s explicit rating might be incorrect and inconsistent. Therefore, a predictive model will approximate according to a user’s implicit behavior. Finally, the longer a user use this system, the more accurate the prediction will become.

5.3 Future work

In the earlier Chapters, we mentioned that this system could be combined with a content-based system. We called that system “hybrid implicit user’s behavior and content-based system (HIUBC)”.

With a content-based system, this system can predict more accurately a user’s interest level. For example, the HIUBC system can be applied to every individual’s web search. Once a user profile is built and hot keywords are taken from web pages, a hot key extractor can be provided to produce good search results, the HIUBC can have personalized and better (more interesting) search results as well as filter not-interesting sites. Also, we can try to put statistical package into kixbrowser so that kixbrowser can show a PC owner’s favorites indicating behavior in graph (visualization). After automating the regression modeling for each user, we can connect kixbrowser with web-search engine at server side to verify that personalized search results produce better search results. In addition to that, we can change the log file system to an XML data base system that kixbrowser has already partially supports.

In the future, every user will find different search result dependent upon his/her profile or previous behavior and a personal model.

References

C. Avery and R. Zeckhauser. Recommendation Systems for Evaluating Computer Messages. *Communications of the ACM*, 40:88 – 89, March 1997.

Brewer, R. S., & Johnson, P. M. "*Toward Collaborative Knowledge Management within Large, Dynamically Structured Information Systems*". University of Hawaii, Dpt. Of Information and Computer Science. Honolulu, 1994

Mark Claypool, Phong Le, Makoto Waseda and David Brown. *Implicit Interest Indicators.* Computer Science Department Worcester Polytechnic Institute, 2001.

J. Goecks and J. W. Shavlik. *Learning Users' Interests by Unobtrusively Observing Their Normal Behavior*. In *Procedings of ACM Intelligent User Interfaces Conference (IUI)*, Jan. 2000.

J. Grundin. Groupware and Social Dyanmics: Eight Challenges for Developers. *Communications of the ACM*, 35:92- 105, 1994.

Jinmook Kim, Douglas W. Oard, and Kathleen Romanik. *Using Implicit Feedback for User Modeling in Internet and Intranet Searching*. College of Library and Information Services University of Maryland May, 2001.

Konstan, J.A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., and Riedl, J. (1997) GroupLens: Applying collaborative filtering to Usenet News. *Communication of the ACM* March 40(3), 77-87.

H. Liberman. Letizia: An Agent that Assists Web Browsing. *Proc. IJCAI-95, pp. 924-929. ,* 1995.

Malone, T.W., Grant, K.R., Turbak, F.A., Brobst, S.A. and Cohen, L.R. and Riedl, J. (1997), Applying collaborative filtering to usernet news, *Communications of the ACM* 30(5), 390-402.

Morita, M. and Shinoda, Y. Information filtering based on user behaviour analysis and best match text retrieval, *Proceedings of the 17th ACM Annual International Conference on Research and Development in Information Retrieval* (SIGTR'94), Dublin, Ireland, Springer-Verlag, 272-81, 1994.

Nichols, D.M., Twidale, M.B. and Paice, C.D. ,*Recommendation and Usage in the Digital Library*, Technical Report CSEG/2/97, Computing Department, Lancaster University, 1997.

Curt Stevens. *Knowledge-Based Assistance for Accessing Large, Poorly Structured Information Spaces*. PhD thesis, University of Colorado, Department of Computer Science, Boulder, 1992. http://www.holodeck.com/curt/my papers/CACM-12-92.ps.

D. Oard and J. Kim. Implicit Feedback for Recommendation Systems. In *Proceedings of the AAAI Workshop on Recommendation Systems*, July 1998.

Draper, Norman R. and Smith, Harry (1998). *Applied Regression Analysis, Third Edition*. New York NY: John Wiley and Sons, Inc. (ISBN: 0-471-17082-8).

Palme, J. (1997), Choices in the Implementation of Rating, in Alton-Scheidl, R., Schumutzer, R., Sint, P.P. and Tscherteu, G. (Eds.), *Voting, Rating, Annotation: Web4Groups and other projects; approaches and first experiences*, Vienna, Austria: Oldenbourg, 147-62

Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. GroupLens: *An open architecture for collaborative filtering of netnews*. In Richard K. Faruta and Christine M. Neuwirth, editors, Proceedings of the Conference on Computer Supported Cooperative Work, pages 175-186. ACM, October 1994. http://www.cs.umn.edu/Research/GroupLens/cscwpaper/paper.html.

Pazzani, M., Muramatsu J., and Billsus, D. (1996). Syskill & Webert: Identifying interesting web sites. *Proceedings of the National Conference on Artificial Intelligence* (pp.54-61). Portland, OR.

B. Sarwar, J.Konstan, A. Borchers, J. Herlocker, B. Miller, and J.Riedl. Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System. In *Proceeding of the ACM Conference on Computer Supported Cooperative Work* (*CSCW*), 1998

Leslie Smith Introduction to neural network Centre for Cognitive and Computational Neuroscience, Department of Computing and Mathematics University of Stirling Sept. 2001
http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html

Douglas B. Terry. A tour through tapestry. In *Proceedings of the ACM Conference on Organizational Computing Systems (COOCS)*, pages 21-30, November 1993.

A. Watson and M. A. Sasse. Measuring Perceived Quality of Speech and Video in Multimedia Conferencing Applications. In *Proceedings of ACM Multimedia Conference,* pages 55-60, Bristol, UK, Sept. 1998.

Appendix I

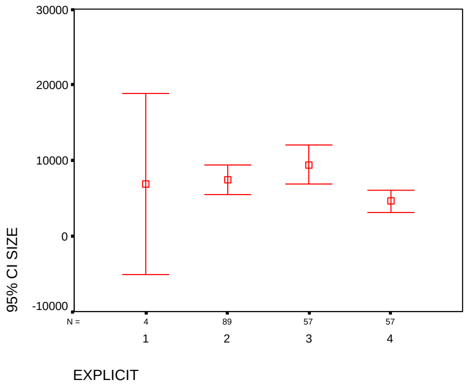
Sample of raw-log file:

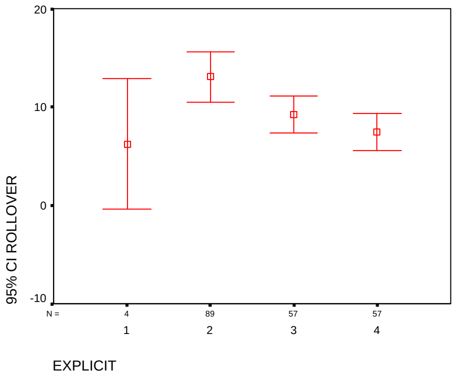
```
setCurrentURL:http://srd.yahoo.com/srst/157660/fifa/1/76/*http://www.fifa.com/:
DATE:Sat Oct 27 22:29:37 EDT 2001
Entire_size:2641
pageLoadingFinished-------------------------------------------------
Mouse pressed-> (x,y):(210,361)
Mouse released-> (x,y):(11783,361)
Mouse pressed-> (x,y):(11783,360)
Mouse released-> (x,y):(11783,360)
Open From bookmark
Previous site's Mouse moved:58354
Previous site's Scrollbar moved:616
pageLoadingStarted----------------------------------------------------
setCurrentURL:http://www.yahoo.com: DATE:Sat Oct 27 22:32:20 EDT 2001
```

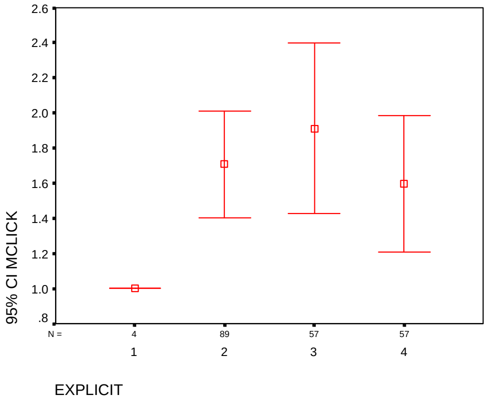
Appendix II

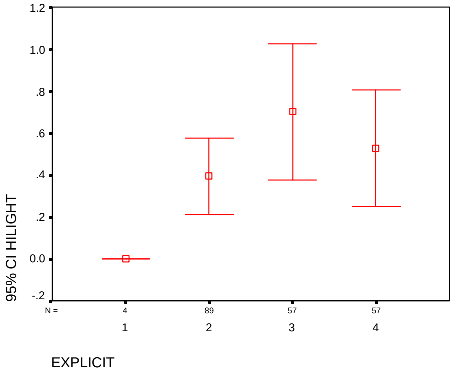
Explicit vs. each predictor

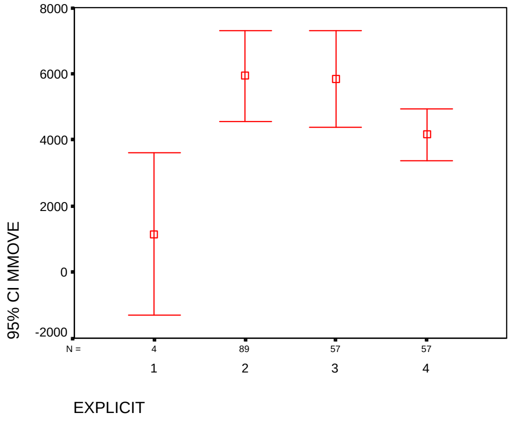
User J

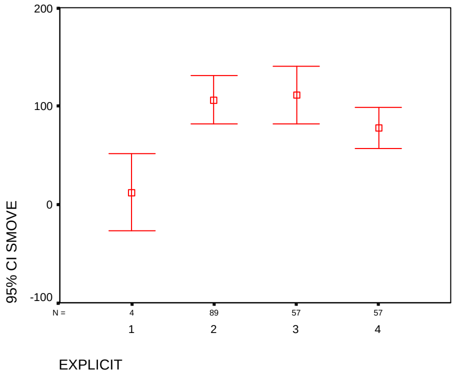


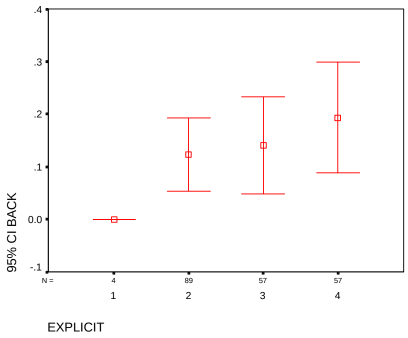


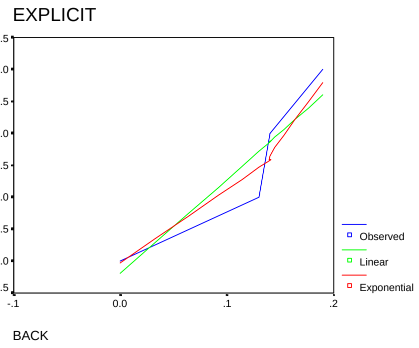


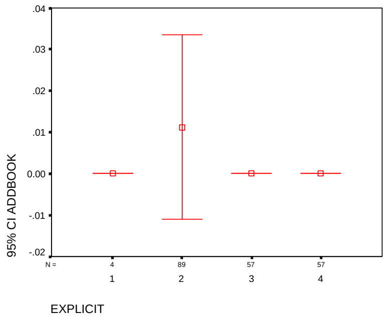


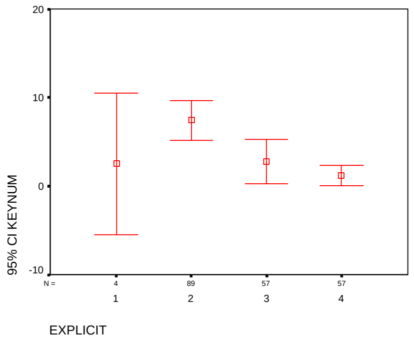


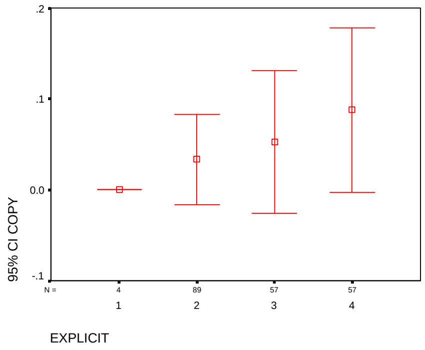


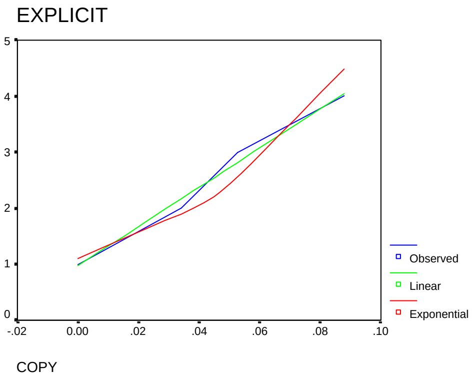

95% CI BACK
.4
.3
.2
.1
0.0
-.1
N =
4
89
57
57
1
2
3
4
EXPLICIT

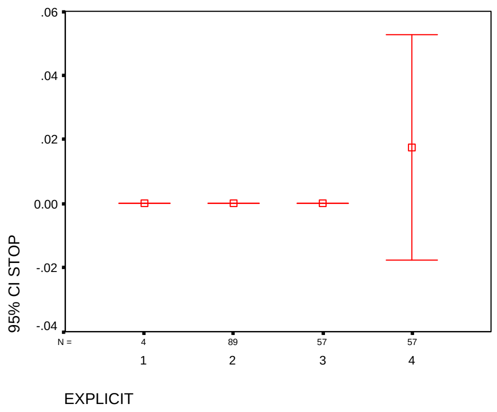

EXPLICIT
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
.5
-.1
0.0
.1
.2
BACK
Observed
Linear
Exponential

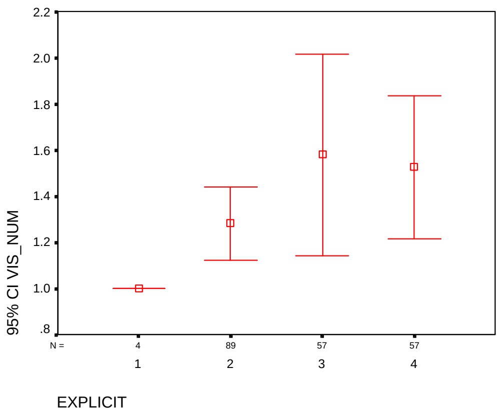

95% CI ADDBOOK
.04
.03
.02
.01
0.00
-.01
-.02
N =
4
89
57
57
1
2
3
4
EXPLICIT

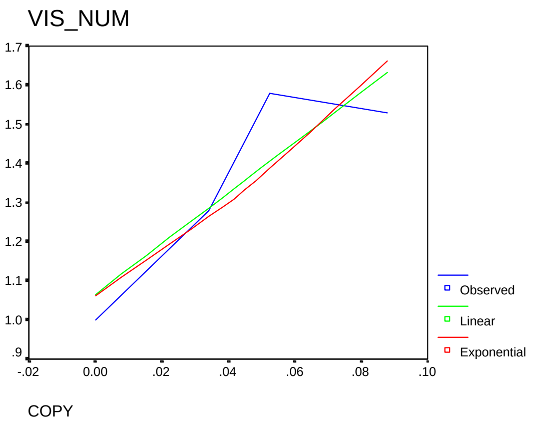

95% CI KEYNUM
20
10
0
-10
N =
4
89
57
57
1
2
3
4
EXPLICIT

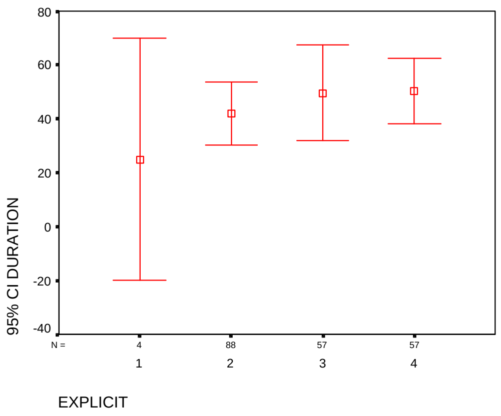

95% CI COPY
.2
.1
0.0
-.1
N =
4
89
57
57
1
2
3
4
EXPLICIT

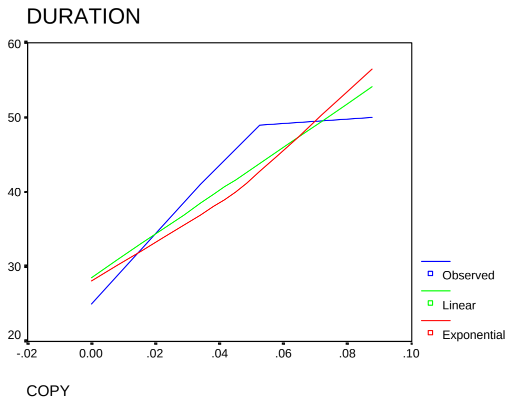

EXPLICIT
5
4
3
2
1
0
-.02
0.00
.02
.04
.06
.08
.10
COPY
Observed
Linear
Exponential

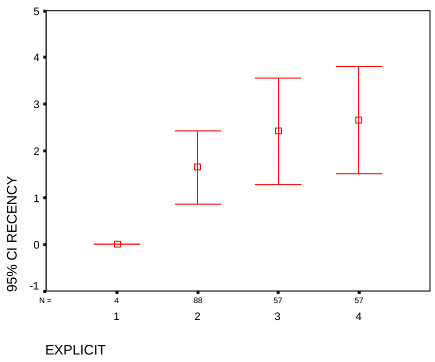

.06
.04
.02
0.00
-.02
-.04
95% CI STOP
N =
4
89
57
57
1
2
3
4
EXPLICIT

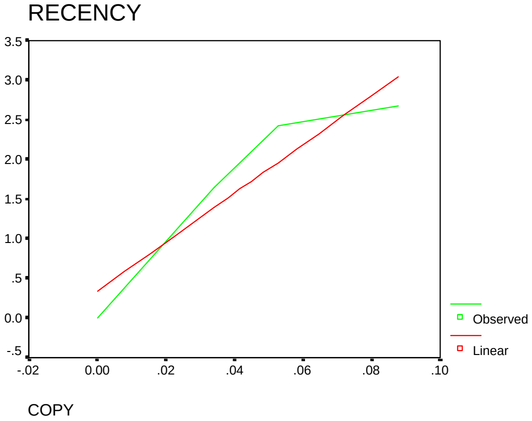

2.2
2.0
1.8
1.6
1.4
1.2
1.0
.8
95% CI VIS_NUM
N =
4
89
57
57
1
2
3
4
EXPLICIT


VIS_NUM
1.7
1.6
1.5
1.4
1.3
1.2
1.1
1.0
.9
-.02
0.00
.02
.04
.06
.08
.10
COPY
Observed
Linear
Exponential

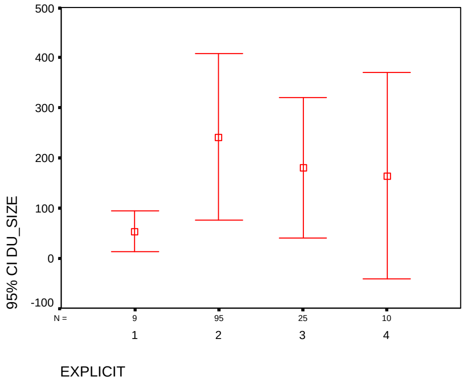

80
60
40
20
0
-20
-40
95% CI DURATION
N =
4
88
57
57
1
2
3
4
EXPLICIT

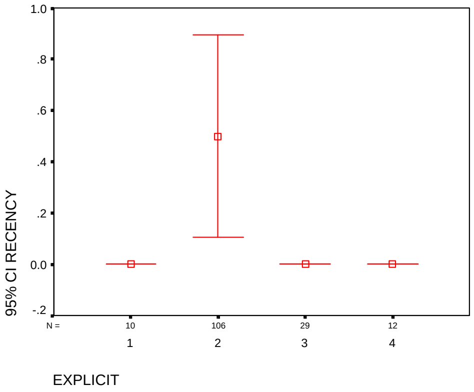

DURATION
60
50
40
30
20
-.02
0.00
.02
.04
.06
.08
.10
COPY
Observed
Linear
Exponential

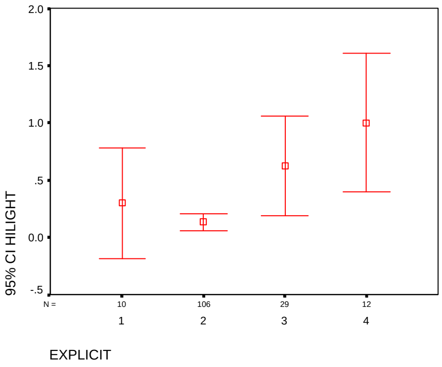

5
4
3
2
1
0
-1
95% CI RECENCY
N =
4
88
57
57
1
2
3
4
EXPLICIT

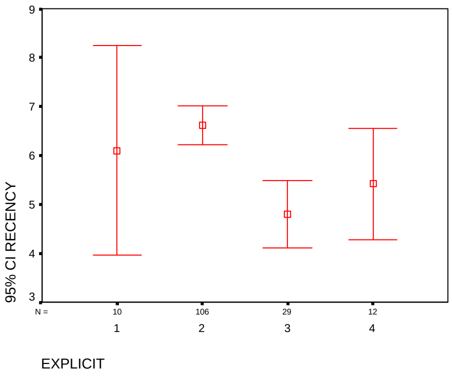

RECENCY
3.5
3.0
2.5
2.0
1.5
1.0
.5
0.0
-.5
-.02
0.00
.02
.04
.06
.08
.10
COPY
Observed
Linear

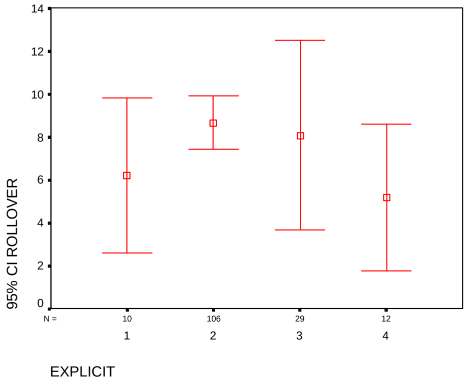
User r

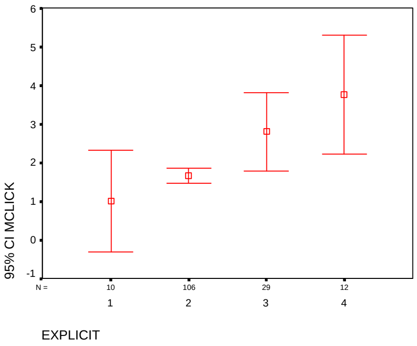

500
400
300
200
100
0
-100
95% CI DU_SIZE
N =
9
95
25
10
1
2
3
4
EXPLICIT

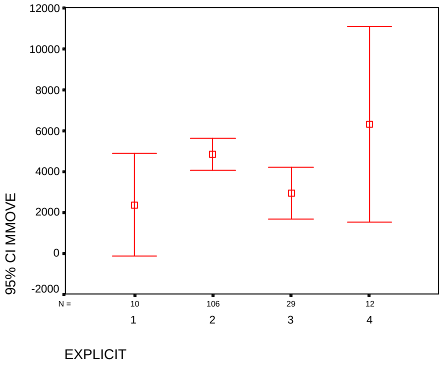

1.0
.8
.6
.4
.2
0.0
-.2
95% CI RECENCY
N =
10
106
29
12
1
2
3
4
EXPLICIT

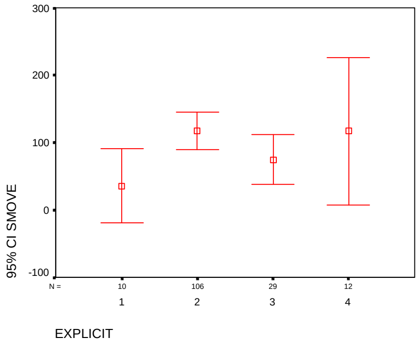

2.0
1.5
1.0
.5
0.0
-.5
95% CI HILIGHT
N =
10
106
29
12
1
2
3
4
EXPLICIT

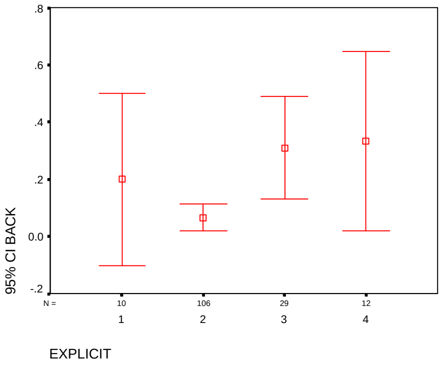

9
8
7
6
5
4
3
95% CI RECENCY
N =
10
106
29
12
1
2
3
4
EXPLICIT

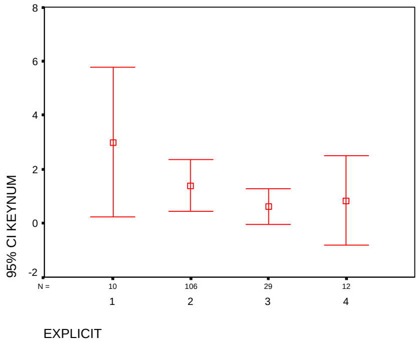

14
12
10
8
6
4
2
0
95% CI ROLLOVER
N =
10
106
29
12
1
2
3
4
EXPLICIT

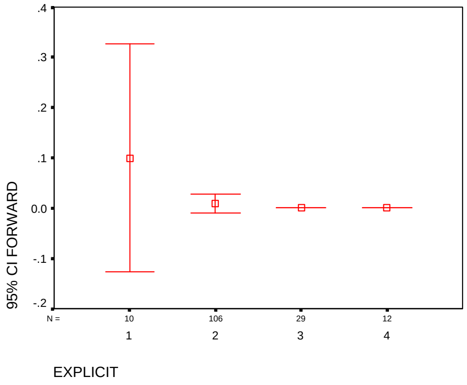

6
5
4
3
2
1
0
-1
95% CI MCLICK
N =
10
106
29
12
1
2
3
4
EXPLICIT

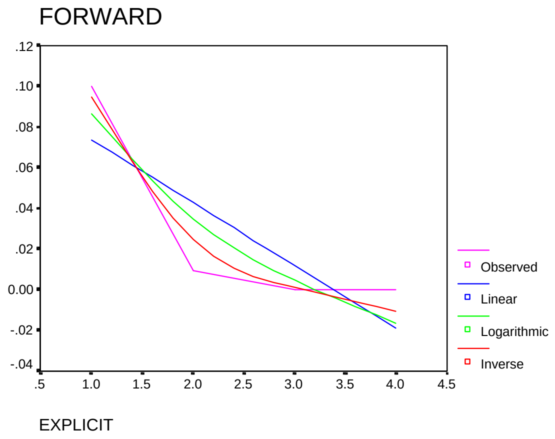

95% CI MMOVE
12000
10000
8000
6000
4000
2000
0
-2000
N =
10
106
29
12
1
2
3
4
EXPLICIT

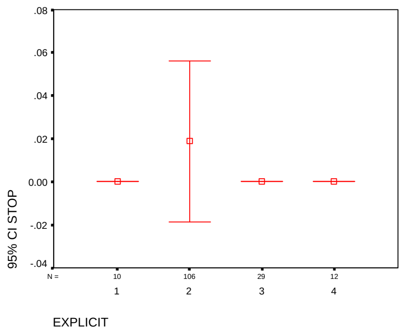

95% CI SMOVE
300
200
100
0
-100
N =
10
106
29
12
1
2
3
4
EXPLICIT

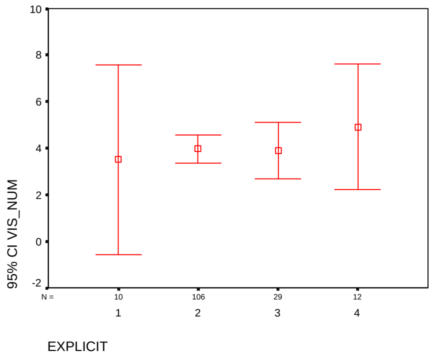

95% CI BACK
.8
.6
.4
.2
0.0
-.2
N =
10
106
29
12
1
2
3
4
EXPLICIT


95% CI KEYNUM
8
6
4
2
0
-2
N =
10
106
29
12
1
2
3
4
EXPLICIT

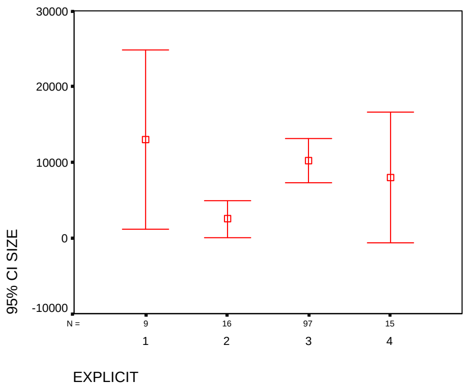

95% CI FORWARD
.4
.3
.2
.1
0.0
-.1
-.2
N =
10
106
29
12
1
2
3
4
EXPLICIT

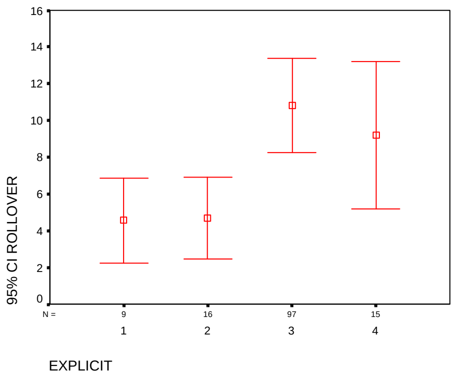

FORWARD
.12
.10
.08
.06
.04
.02
0.00
-.02
-.04
.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
EXPLICIT
Observed
Linear
Logarithmic
Inverse

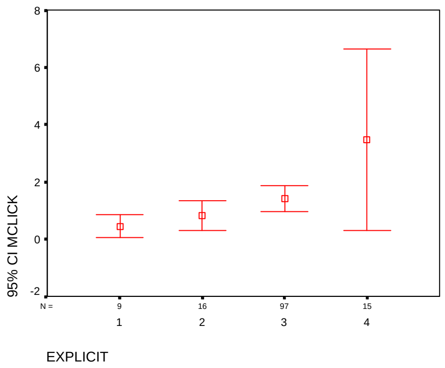

.08
.06
.04
.02
0.00
-.02
-.04
95% CI STOP
N =
10
106
29
12
1
2
3
4
EXPLICIT

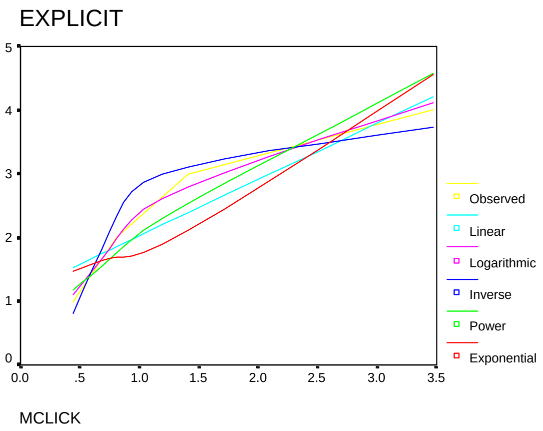

10
8
6
4
2
0
-2
95% CI VIS_NUM
N =
10
106
29
12
1
2
3
4
EXPLICIT

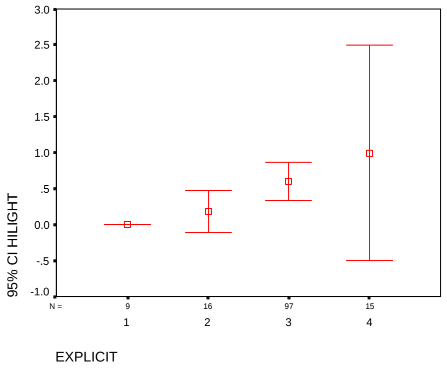
User s

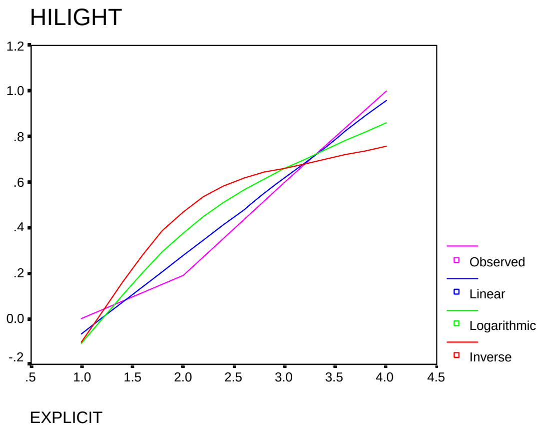

30000
20000
10000
0
-10000
95% CI SIZE
N =
9
16
97
15
1
2
3
4
EXPLICIT

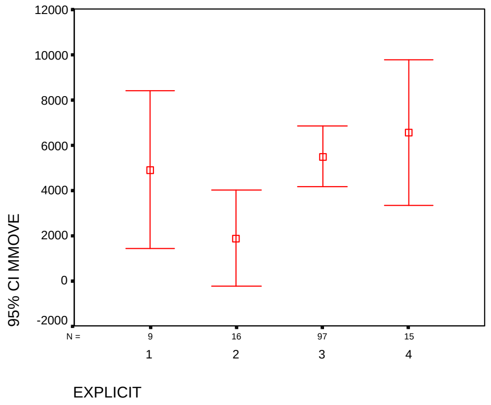

16
14
12
10
8
6
4
2
0
95% CI ROLLOVER
N =
9
16
97
15
1
2
3
4
EXPLICIT

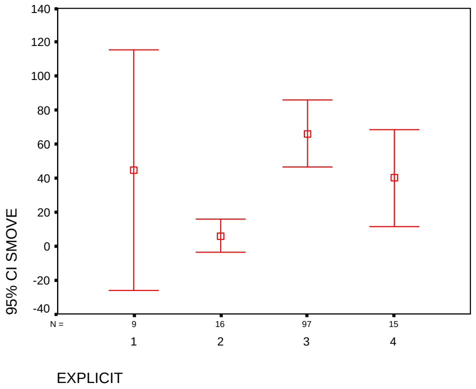

8
6
4
2
0
-2
95% CI MCLICK
N =
9
16
97
15
1
2
3
4
EXPLICIT

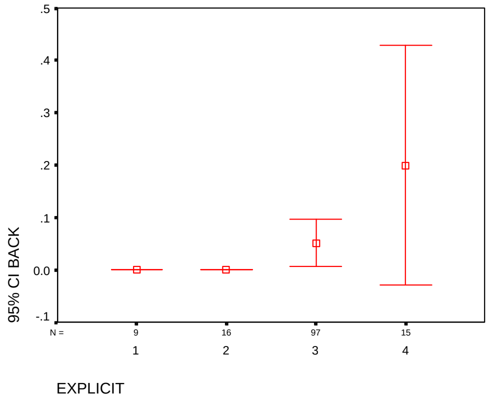

EXPLICIT
5
4
3
2
1
0
0.0
.5
1.0
1.5
2.0
2.5
3.0
3.5
MCLICK
Observed
Linear
Logarithmic
Inverse
Power
Exponential

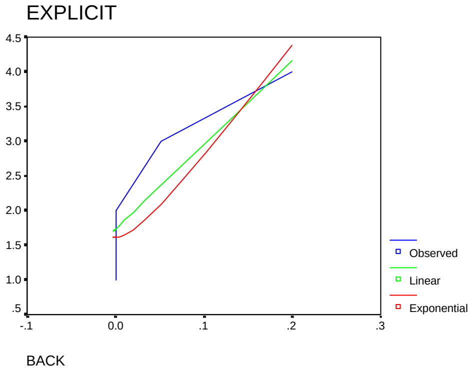

3.0
2.5
2.0
1.5
1.0
.5
0.0
-.5
-1.0
95% CI HILIGHT
N =
9
16
97
15
1
2
3
4
EXPLICIT

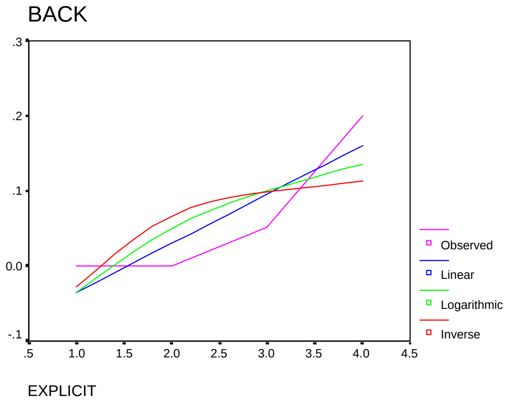

HILIGHT
1.2
1.0
.8
.6
.4
.2
0.0
-.2
.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
EXPLICIT
Observed
Linear
Logarithmic
Inverse

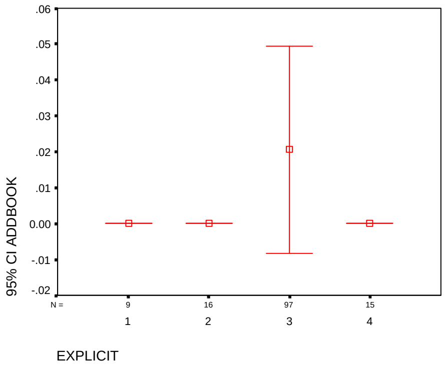

95% CI MMOVE
12000
10000
8000
6000
4000
2000
0
-2000
N =
9
16
97
15
1
2
3
4
EXPLICIT

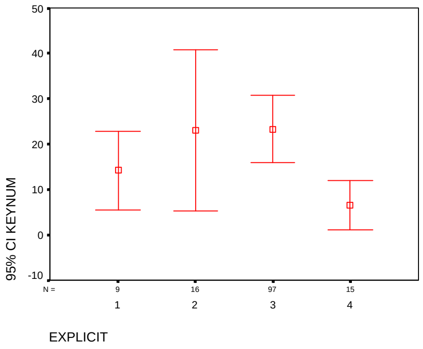

95% CI SMOVE
140
120
100
80
60
40
20
0
-20
-40
N =
9
16
97
15
1
2
3
4
EXPLICIT

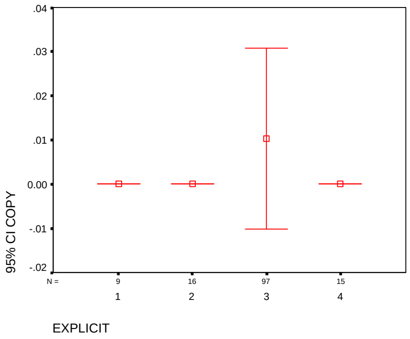

95% CI BACK
.5
.4
.3
.2
.1
0.0
-.1
N =
9
16
97
15
1
2
3
4
EXPLICIT

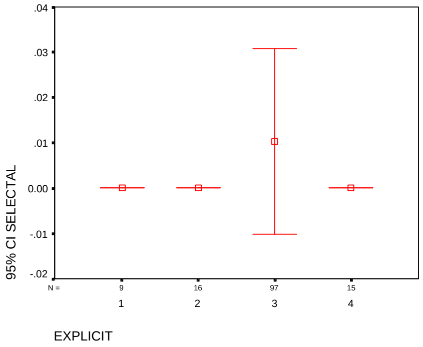

EXPLICIT
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
.5
-.1
0.0
.1
.2
.3
Observed
Linear
Exponential
BACK

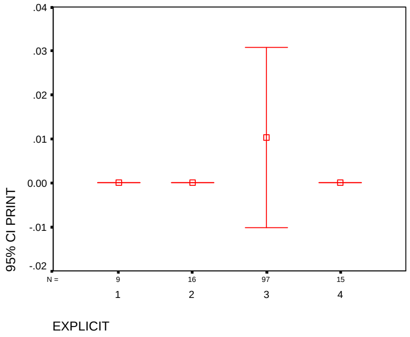

BACK
.3
.2
.1
0.0
-.1
.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
Observed
Linear
Logarithmic
Inverse
EXPLICIT

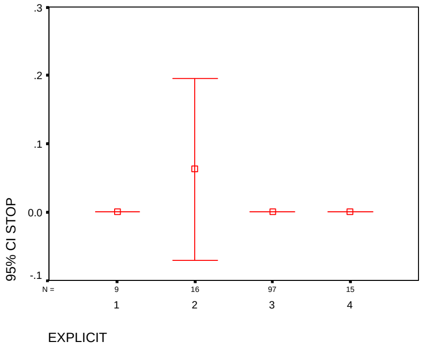

95% CI ADDBOOK
.06
.05
.04
.03
.02
.01
0.00
-.01
-.02
N =
9
16
97
15
1
2
3
4
EXPLICIT

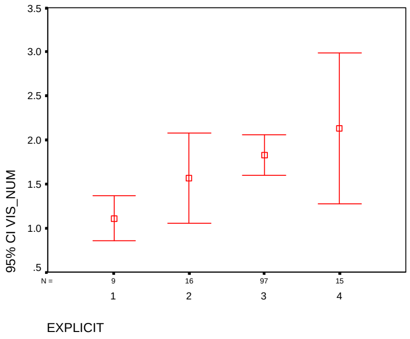

50
40
30
20
10
0
-10
95% CI KEYNUM
N =
9
16
97
15
1
2
3
4
EXPLICIT

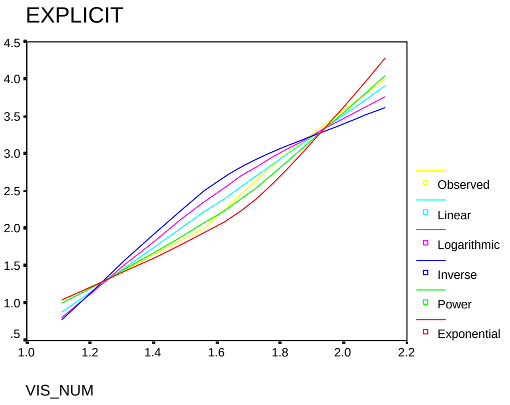

.04
.03
.02
.01
0.00
-.01
-.02
95% CI COPY
N =
9
16
97
15
1
2
3
4
EXPLICIT

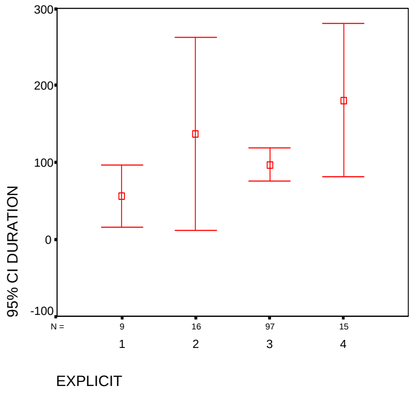

.04
.03
.02
.01
0.00
-.01
-.02
95% CI SELECTAL
N =
9
16
97
15
1
2
3
4
EXPLICIT

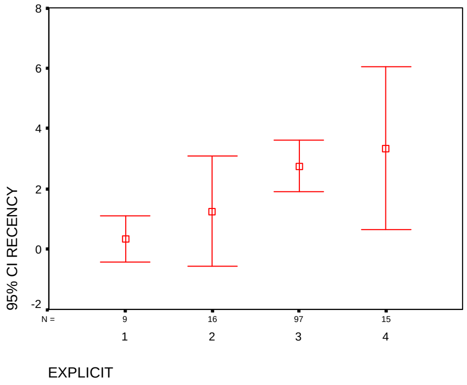

.04
.03
.02
.01
0.00
-.01
-.02
95% CI PRINT
N =
9
16
97
15
1
2
3
4
EXPLICIT

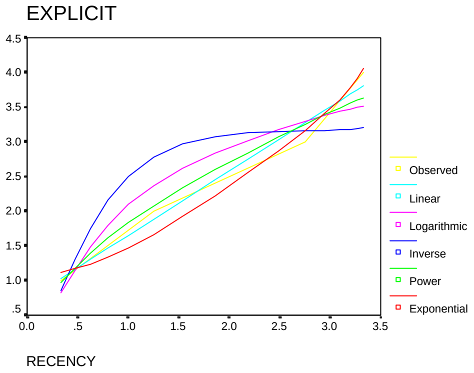

.3
.2
.1
0.0
-.1
95% CI STOP
N =
9
16
97
15
1
2
3
4
EXPLICIT


3.5
3.0
2.5
2.0
1.5
1.0
.5
95% CI VIS_NUM
N =
9
16
97
15
1
2
3
4
EXPLICIT

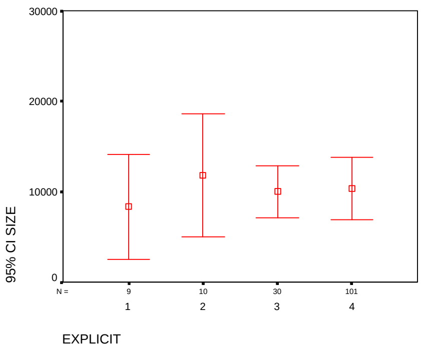

EXPLICIT
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
.5
1.0
1.2
1.4
1.6
1.8
2.0
2.2
VIS_NUM
Observed
Linear
Logarithmic
Inverse
Power
Exponential

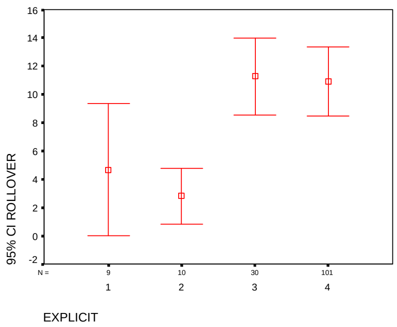

300
200
100
0
-100
95% CI DURATION
N =
9
16
97
15
1
2
3
4
EXPLICIT

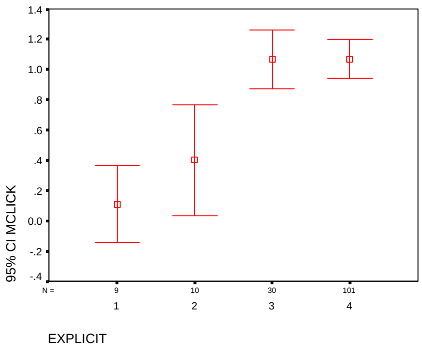

8
6
4
2
0
-2
95% CI RECENCY
N =
9
16
97
15
1
2
3
4
EXPLICIT

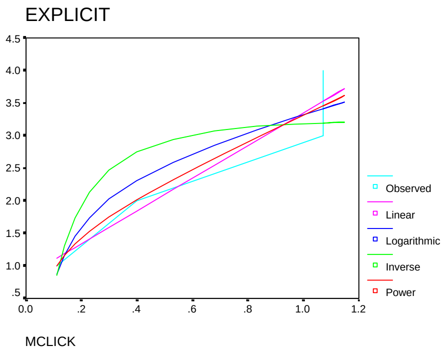

EXPLICIT
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
.5
0.0
.5
1.0
1.5
2.0
2.5
3.0
3.5
RECENCY
Observed
Linear
Logarithmic
Inverse
Power
Exponential

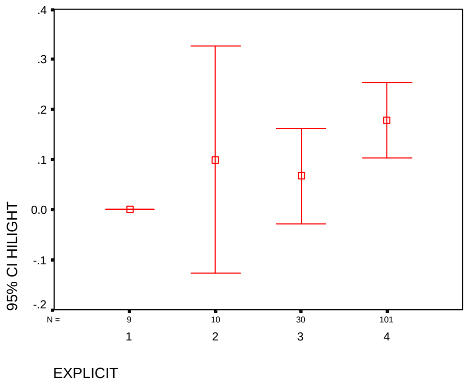
User p

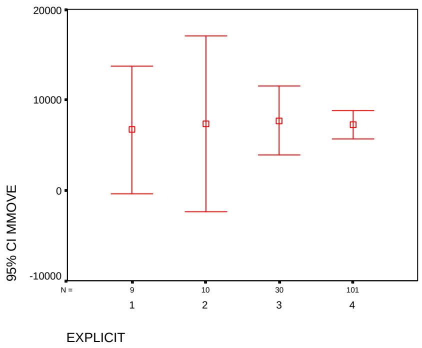

95% CI SIZE
30000
20000
10000
0
N =
9
10
30
101
1
2
3
4
EXPLICIT

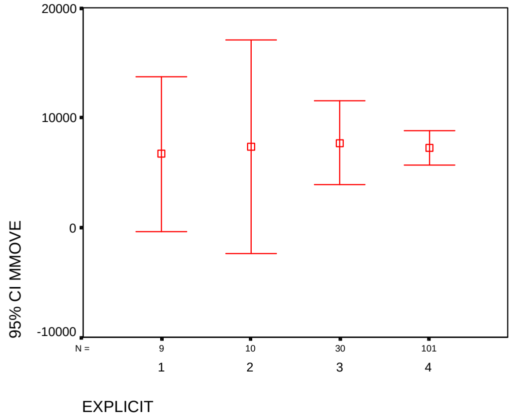

95% CI ROLLOVER
16
14
12
10
8
6
4
2
0
-2
N =
9
10
30
101
1
2
3
4
EXPLICIT

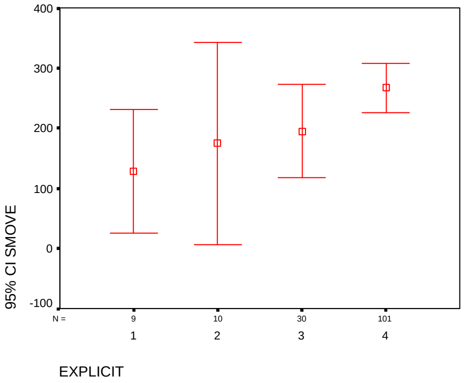

95% CI MCLICK
1.4
1.2
1.0
.8
.6
.4
.2
0.0
-.2
-.4
N =
9
10
30
101
1
2
3
4
EXPLICIT

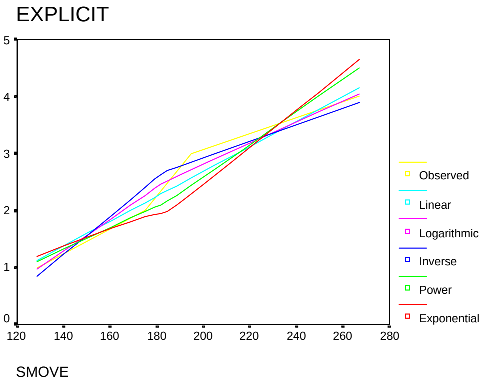

EXPLICIT
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
.5
0.0
.2
.4
.6
.8
1.0
1.2
Observed
Linear
Logarithmic
Inverse
Power
MCLICK

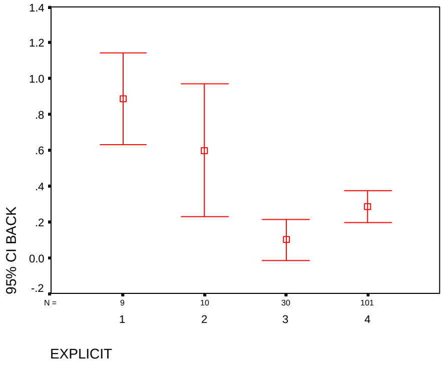

95% CI HILIGHT
.4
.3
.2
.1
0.0
-.1
-.2
N =
9
10
30
101
1
2
3
4
EXPLICIT

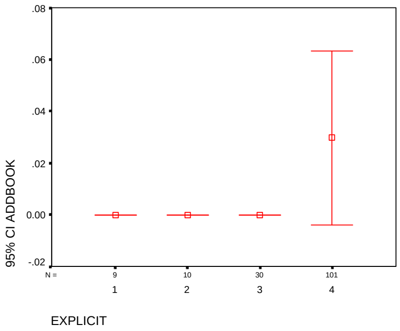

95% CI MMOVE
20000
10000
0
-10000
N =
9
10
30
101
1
2
3
4
EXPLICIT

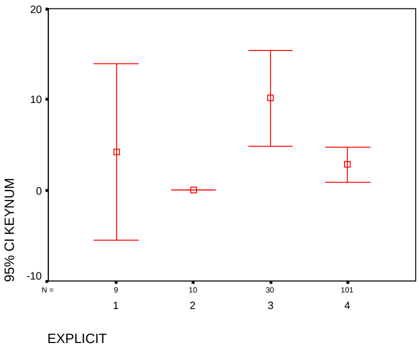

20000
10000
0
-10000
95% CI MMOVE
N =
9
10
30
101
1
2
3
4
EXPLICIT

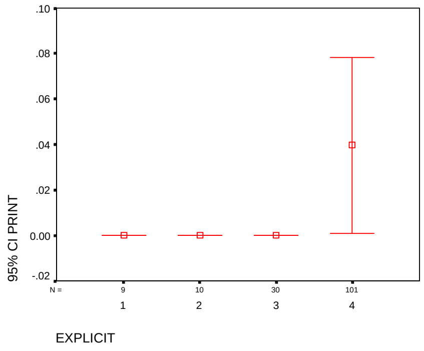

400
300
200
100
0
-100
95% CI SMOVE
N =
9
10
30
101
1
2
3
4
EXPLICIT

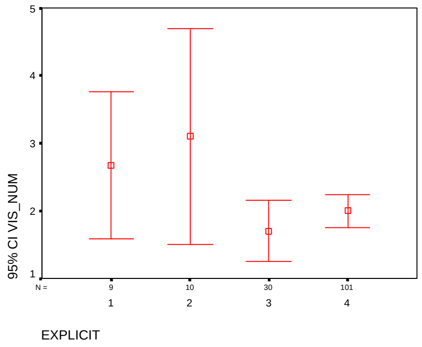

EXPLICIT
5
4
3
2
1
0
120
140
160
180
200
220
240
260
280
SMOVE
Observed
Linear
Logarithmic
Inverse
Power
Exponential

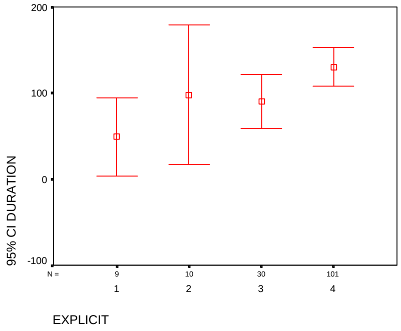

1.4
1.2
1.0
.8
.6
.4
.2
0.0
-.2
95% CI BACK
N =
9
10
30
101
1
2
3
4
EXPLICIT

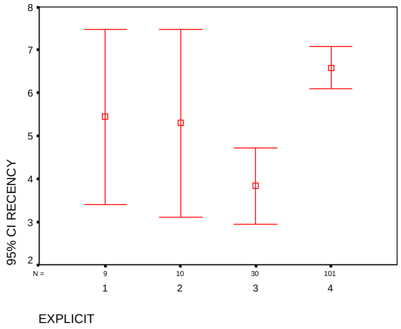

.08
.06
.04
.02
0.00
-.02
95% CI ADDBOOK
N =
9
10
30
101
1
2
3
4
EXPLICIT


20
10
0
-10
95% CI KEYNUM
N =
9
10
30
101
1
2
3
4
EXPLICIT


.10
.08
.06
.04
.02
0.00
-.02
95% CI PRINT
N =
9
10
30
101
1
2
3
4
EXPLICIT


5
4
3
2
1
95% CI VIS_NUM
N =
9
10
30
101
1
2
3
4
EXPLICIT


200
100
0
-100
95% CI DURATION
N =
9
10
30
101
1
2
3
4
EXPLICIT


8
7
6
5
4
3
2
95% CI RECENCY
N =
9
10
30
101
1
2
3
4
EXPLICIT