

인터넷 서버에서 통계 그래픽의 처리 방법

정 남 철¹⁾

요 약

본 논문에서는 인터넷 상에서 통계 그래픽을 서버에서 처리하는 방법에 대하여 연구하고자 한다. 이 방법은 서버와 통계 패키지와의 인터페이스를 통하여 서버에서 Xlisp-Stat이 생성한 통계 그래픽을 브라우저가 표현할 수 있는 이미지로 변환하여 클라이언트에 전송하는 방법으로서 C언어를 사용하여 CGI 프로그램을 구현하였다.

1. 서 론

양방향 매체인 인터넷 상에서의 통계 분석은 정보 통신 분야의 핵심 기술의 발달로 새로운 사실이 아닌 것으로 인식되고 있다. 초고속 통신망의 발달과 더불어, 웹 서버의 발달, 다양한 네비게이터의 출현, 저작도구의 자동화, 웹 정보의 전달을 지원하는 프로토콜의 발달, 정보처리를 지원하는 프로그래밍 언어 등이 인터넷에서의 통계분석을 가능하게 한다.

인터넷 상에서 통계 패키지를 이용하여 통계분석을 하는 예로는 Garcia(1996), Shin(1996), Friendly(1995)가 있다. Garcia는 브라우저에서 사용자가 Xlisp-Stat 명령어를 전송하면 서버에 설치된 Xlisp-Stat 통계 패키지가 통계분석을 하고 그 결과를 브라우저로 전송하는 시스템이다. 그러나 통계분석 결과가 통계 그래픽인 경우에는 처리하지 못하는 단점이 있다. Shin은 Xlisp-Stat 통계 패키지를, Friendly는 SAS 통계 패키지를 이용하여 통계 그래픽을 처리하도록 고안하였다. 이들은 통계 패키지를 이용하여 통계 그래픽의 처리를 클라이언트 측의 브라우저(Netscape, Mosaic or Cello, etc)의 셋업을 통하여 처리하는 방법으로서 통계 패키지가 서버에서 실행되는 Garcia의 연구와는 처리 방법이 다르다.

한편 프로그래밍 언어를 이용하여 통계분석을 하는 예로는 Deleeuw(1995)와 Keith(1996)가 있다. Deleeuw는 그래픽 처리를 통계 패키지에 의하지 않고 'gd'라는 그래픽 라이브러리를 사용하여 C 언어로 구현하였으며, Keith는 Newcastle 대학의 하이퍼 교재인 'SurfStat australia'에서 자바 애플릿을 이용하여 클라이언트에서 동적 그래픽을 통한 통계분석이 가능하도록 개발하여 통계를 학습할 수 있도록 연구하였다.

인터넷 상에서 통계 그래픽을 처리하는 방법은 그래픽 생성 장소에 따라 서버에서 처리하는 방법과 클라이언트에서 처리하는 방법이 있다. 클라이언트에서 그래픽을 처리하기 위해서는 브라우저에서 plug-in이나 helper를 사용하여 처리하며 그와 관련된 소프트웨어들은 클라이언트 컴퓨터에 설치해야 한다.

1) (440-714) 경기도 수원시 장안구 정자동 동남보건대학 컴퓨터응용과 조교수

통계 그래픽을 서버에서 처리하는 방법은 구현 도구에 따라 두 가지 방법이 있다. 첫 번째 방법은 기존의 통계 패키지를 활용하는 방법과 두 번째로 프로그래밍 언어를 사용하는 방법이 있다. 첫 번째 방법을 사용할 때 장점으로서는 통계 패키지와 웹 서버간의 인터페이스를 통하여 통계 패키지가 그래픽을 생성하기 때문에 비교적 구현이 쉽고, 통계 교육을 위한 교육 시스템을 구축하는데 효율적이다. 그리고 이 시스템을 이용하는 사용자 측면에서 볼 때는 사용자는 단순히 데이터를 제공하면 되므로 이용하기 편리하다. 두 번째 방법은 서버가 프로그래밍 언어로 구현된 통계 그래픽의 생성 모듈을 직접 수행함으로써 빠르게 결과를 만들어 클라이언트에 전송할 수 있는 장점이 있지만 모든 통계 분석 모듈을 프로그래밍 해야 하므로 개발이 번거롭다. 본 논문에서는 서버에서 통계 패키지를 활용하여 통계 그래픽을 처리하는 방법에 대해서만 다루기로 한다.

인터넷 상에서 통계 패키지의 설치 위치는 통계분석시스템의 구축에 많은 영향을 주며 이에 따른 인터페이스 기술이 달라지게 되는데, 본 논문에서는 Garcia의 연구에서 통계 그래픽을 처리하지 못하는 점을 개선하여 서버에서 통계 그래픽을 처리하도록 연구하였으며, 특히 통계 소프트웨어 개발 환경을 지원하는 Xlisp-Stat(Tierney, 1990) 통계 패키지를 이용하여 브라우저에서 제시된 자료를 분석하고 그래프로 표현되는 분석 결과를 이미지 변환 프로그램으로 변환하여 이를 클라이언트에 전송하도록 구현하였다.

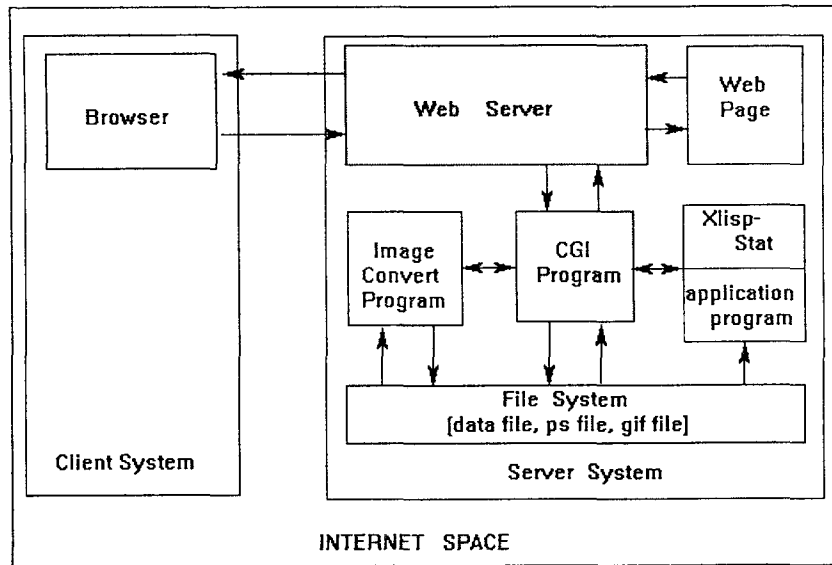
2. 인터넷 서버에서 통계 그래픽 처리 시스템의 구현

인터넷 서버에서 통계 그래픽의 처리 방법은 본 논문에서 구현한 시스템을 통하여 제시한다.

웹 서버 기반의 통계분석시스템(Web-server based on Statistical Analysis System)은 사용자가 클라이언트에서 분석하고자 하는 자료를 제시하면 서버에 설치된 통계 패키지가 통계 분석하고 그 결과를 클라이언트로 반환하는 통계분석시스템이다. 본 논문에서는 이와 같은 형태에 의하여 처리하고 통계 그래픽을 클라이언트로 전송하는 시스템을 CGI 프로그램으로 구현하였는데 이를 'W3SAS'라고 부르기로 한다.

2.1 W3SAS의 개요

본 논문에서 구현한 W3SAS(Statistical Analysis System on the WWW)는 분석하고자 하는 자료를 클라이언트에서 제시하고 서버에 설치된 Xlisp-Stat 통계 패키지를 이용하여 분석하고 그 결과인 그래프를 클라이언트로 반환하는 통계분석시스템이다. W3SAS는 NCSA httpd 웹 서버 하에서 UNIX 시스템을 기반으로 구축되었으며 공개 소프트웨어(freeware)인 Xlisp-Stat과 웹 서버간에 인터페이스를 할 수 있도록 ANSI-C 언어를 이용하여 CGI 프로그램을 개발하였다. 대부분의 웹 브라우저들은 포스트스크립트(postscript)로 표현되는 통계 그래픽을 취급할 수 없기 때문에 PS(PostScript) 파일을 GIF 파일이나 JPEG 등으로 변환하여 브라우저에 표현하여야 한다. 본 논문에서 구현한 CGI 프로그램은 실시간으로 Xlisp-Stat 통계 패키지가 생성한 PS 파일을 GIF 이미지 파일로 변환하도록 하며, 변환된 GIF 이미지를 브라우저에 전송하도록 하여 통계 그래프를 클라이언트에 표현한다. W3SAS의 예제로는 서버에 설치된 Xlisp-Stat을 이용하여 히스토그램을 생성하고 브라우저로 전송하도록 구현하였다. W3SAS의 구성은 그림 1과 같다.



<그림 1> W3SAS의 구조

2.2 W3SAS의 요소와 설계

2.2.1 서버의 구성

서버에는 Fill-In FORM을 설계한 HTML 웹 페이지, Xlisp-Stat 통계 패키지, Xlisp-Stat의 응용프로그램, 입력 데이터파일, 출력 이미지파일, 이미지 변환 유틸리티, CGIC 라이브러리, CGI 프로그램 등으로 구성되며, 특히 웹 서버 구축은 Httpd 데몬 프로그램이 사용되었다.

2.2.2 통계 분석 웹 페이지의 작성

모든 통계를 인터넷을 통하여 분석한다는 것은 방대한 작업일 뿐만 아니라 거의 불가능하므로 인터넷을 이용하여 통계를 분석할 경우에는 비교적 간단한 통계 분석이 바람직하다. 본 시스템에서는 히스토그램, Box-Plot, 기술통계를 모델로 설정하여 웹 페이지를 설계하였다.

2.2.3 Fill-In FORM의 설계

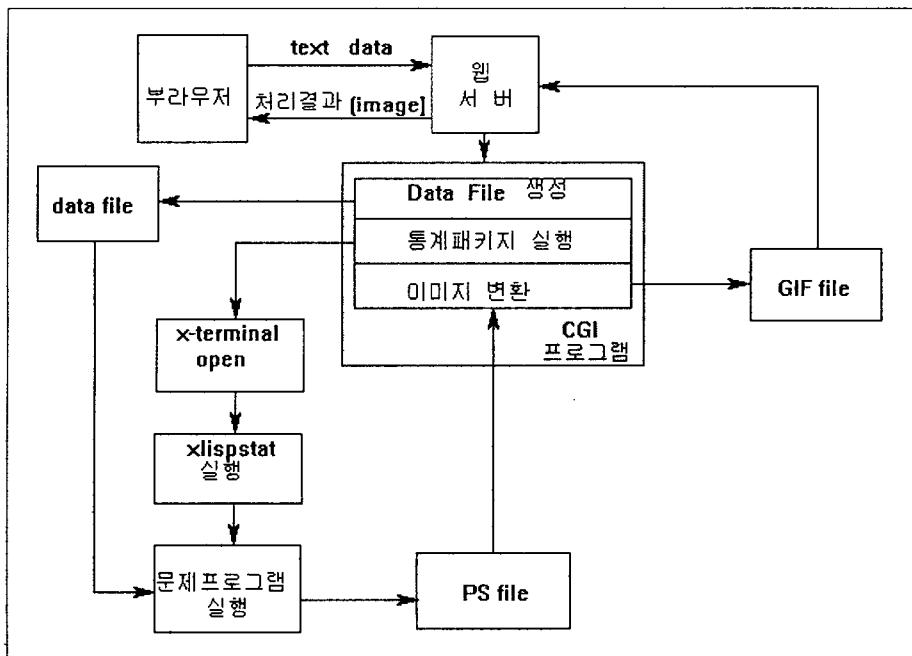
Fill-In Form은 분석하고자 하는 자료를 웹 서버로 전송할 수 있도록 설계된 클라이언트 측의 입력 화면에 대한 모듈이다. 이는 인터넷 저작도구인 HTML에 의하여 구현된 웹 페이지(그림 3)이며, 이는 웹 서버에 위치되어 있으며 사용자의 탐색에 의하여 기동(invoked)된다. 본 시스템에서는 1 차원 데이터를 입력할 수 있도록 설계하였으며 이 데이터는 히스토그램, Box-Plot, 기술통계량의 분석 자료로 이용된다.

2.2.4 통계 패키지의 선택

본 시스템은 통계 패키지인 Xlisp-Stat을 이용하여 설계를 하였다. Xlisp-Stat 통계 패키지는 첫째로, 공개 소프트웨어이므로 통계 패키지의 저작권 문제가 뒤따르지 않고, 둘째로, 통계 패키지의 분량이 크지 않으므로 서버의 자원을 낭비하지 않으며, 셋째로, 프로그래밍 언어(여기서는 C 언어임)와의 우수한 인터페이스 능력을 갖추고 있기 때문에 본 시스템인 W3SAS를 구현하는데 적합하다.

2.2.5 CGI 프로그램의 설계

CGI 프로그램은 웹 서버가 이해할 수 있는 데이터의 타입, 즉 MIME(Multipurpose Internet Mail Extensions)을 Content-type에 기록하면서 시작된다. 본 논문에서 구현한 CGI 프로그램은 웹 서버와 Xlisp-Stat 통계 패키지와의 인터페이스, Xlisp-Stat와 문제 해결 응용프로그램의 실행, 분석 결과에 대한 이미지 변환 등의 루틴이 포함되어 있다. 분석 결과에 대해서도 HTML 문서의 형태로 코드화하여 CGIC 라이브러리의 함수에 의해서 클라이언트에 전송하도록 설계하였다. 응용 프로그램(Xlisp-Stat의 문제 해결 프로그램)은 브라우저에서 입력한 데이터를 파일로 저장하는 역할과, 이 저장된 데이터를 이용하여 Xlisp-Stat에 의해 그래프를 생성하는 기능을 한다. 이미지 변환 패키지인 ImageMagick은 Xlisp-Stat 통계 패키지가 생성한 결과(PS 이미지 파일)를 GIF 이미지 파일로 변환하며, CGIC 라이브러리의 함수에 의하여 클라이언트로 전송하도록 설계하였다. 그림 2는 CGI 프로그램의 프로세스이다.



<그림 2> CGI 프로그램의 프로세스

2.2.6 이미지 변환 도구 및 선정

인터넷 상에서의 그래픽 변환하는 도구는 gd, Ghostscript, ps2gif, ImageMagick 등의 유틸리티 프로그램들과 저작도구가 있다. 그리고 C 언어를 이용한 CGI 프로그램을 통하여 이 유틸리티 프로그램들을 효율적으로 구현할 수 있도록 지원하는 CGIC 라이브러리가 있다.

gd는 GIF 화일을 만들기 위한 그래픽 라이브러리이다(Boutell(b),1995). 특히, gd는 웹 어플리케이션에서 GIF화일이 생성되는 즉시 처리되는 이미지일 경우에 유용하게 사용할 수 있다.

ImageMagick(Cristy,1995)은 X 윈도우 시스템에 대하여 이미지들의 상호 변환과 디스플레이를 위한 패키지이며, X 라이브러리와 인터페이스를 할 수 있다. 특히 이 도구는 내부의 명령('convert')을 직접 응용프로그램에서 코딩하여 이미지 변환, 합성 등의 기능이 있기 때문에 CGI 프로그램에 의한 웹 페이지를 구축할 때 유용하게 이용될 수 있다.

Ghostscript(Lang,1996)는 모든 플랫폼에 대하여 이용할 수 있는 PostScript와 PDF(Portable Document Format)을 위한 인터프리터이며, C 프로그램을 통해 이용할 수 있는 강력한 프로그램이다. 주로 Ghostscript는 PostScript 화일을 디스플레이하고 인쇄하기 위해 사용되며 'gs' 명령에 의하여 GIF 화일로 변환할 수 있다. 그밖에 PS 화일을 GIF 화일로 변환하기 위한 도구들은 ps2gif, www2dot, webgif 등이 있다.

CGIC는 CGI를 바탕으로 한 웹 응용프로그램을 만들기 위한 ANSI C 언어의 라이브러리이다(Boutell(a),1995). CGIC는 폼 데이터를 분석하고, GET과 POST 폼 데이터를 투명하게 받아들인다. 또한 일관성 있는 방법으로 폼 필드에 있는 line break들을 처리하며, 항상 null 값이 아닌 스트링에 CGI 환경 변수를 로드한다.

본 시스템에서는 C 언어로 작성된 CGI 프로그램 내에 ImageMagick 유틸리티와 CGIC 라이브러리를 이용하여 Xlisp-Stat이 생성한 PS 이미지 화일을 GIF 이미지 화일로 변환하여 브라우저에 전송하는 방법으로 설계하였다.

2.3 CGI 프로그램

그림 2는 통계 패키지를 실행하기 위한 CGI 프로그램의 프로세스이며, 이 프로세스는 C 언어에 의해 구현되었다. 웹 서버와 Xlisp-Stat과의 인터페이스는 이 CGI 프로그램의 processForm() 사용자 함수 내에서 이루어진다. 다음은 본 논문에서 통계 그래픽을 생성하고 이미지를 변환(GIF화일)하여 그 결과를 클라이언트에 전송할 수 있도록 구현한 processForm() 함수이다.

```

1: void processForm()
2: {
3:     char x[MAXCHAR], xcopy[MAXCHAR];
4:     char his[MAXYES];
5:     cgiFormString("x", x, MAXCHAR);
6:     cgiFormString("his", his, MAXYES);
7:     listify(xcopy, x);
8:     system("rm /usr/local/etc/httpd/htdocs/zzz*.gif");
9:     fprintf(cgiOut, "<H2>Results of one-dimensional data analysis</H2>\n");

```

```

10:    fprintf(cgiOut, "<PRE>\n");
11:    fprintf(cgiOut, "<IMG SRC=\"/blueline.gif\" HEIGHT=3 WIDTH=800>\n");
12:    if ((his[1] != '\0') && (x[1] != '\0'))
13:        {
14:            system("/usr/openwin/bin/xterm -display 203.252.32.233:0 -e
                /usr/local/bin/xlispstat /usr/local/etc/httpd/htdocs/edu/histo.lsp");
15:            strcpy(parmfile, "/usr/local/bin/convert /usr/local/etc/httpd/htdocs/edu/histo.ps");
16:            strcpy(filename, tempnam("/usr/local/etc/httpd/htdocs/", "zzz"));
17:            strcat(filename, ".gif");
18:            strcpy(filename1, filename);
19:            strcat(parmfile, filename);
20:            system(parmfile);
21:            filemake(filename1);
22:            fprintf(cgiOut, "<H2>your data :</H2>\n");
23:            fprintf(cgiOut, "<H3><I>X : %s</I></H3>\n",x);
24:            fprintf(cgiOut, "<H2>Selected Statistical Methods :</H2>\n");
25:            fprintf(cgiOut, "<P><H3><I>Histogram</I></H3></P>\n");
26:            fprintf(cgiOut, "<HR>\n");
27:            fprintf(cgiOut, "<CENTER>\n");
28:            fprintf(cgiOut, "<IMG SRC=\"/%s\">\n", file2);
29:            fprintf(cgiOut, "</CENTER>\n");
30:        }
31:    fprintf(cgiOut, "<IMG SRC=\"/blueline.gif\" HEIGHT=3 WIDTH=800\n");
32:    fprintf(cgiOut, "</PRE>\n");
33: }

```

위 processForm() 함수의 5:는 브라우저의 특정 입력 필드에 대하여 보내 온 스트링을 검출하는 CGIC 라이브러리 함수이다. 이 함수의 일반 형식은 다음과 같다.

```
cgiFormResultType cgiFormString(char *name, char *result, int max)
```

브라우저에서 전송되어 온 텍스트는 (max - 1) 바이트를 초과하지 않는 범위에서 "result"에 의해 정의된 버퍼로 복사된다. 브라우저에 의해 newline 형식으로 제시되었을지라도 cgiFormString은 항상 각각의 newline을 single line feed(ascii decimal 10)로 코드화 한다. 여기서는 브라우저에서 통계 분석을 위해 입력한 값을 "x"라는 이름의 변수로 버퍼 x에 복사한다.

6:은 브라우저의 선택 Fill-in 품인 checkbox를 "his"라는 이름의 변수로 버퍼 his에 저장한다.

7:은 cgiFormString() 함수에 의해 버퍼 x에 복사된 데이터를 데이터 화일인 data로 저장하기 위한 사용자 함수이다.

8:은 GIF 이미지 화일을 생성하기 전에 이미 생성한 GIF 이미지 화일을 삭제하는 시스템 함수이다. W3SAS는 브라우저에서 데이터를 실행시킬 때마다 16:의 tempnam() 함수와 17:에 의해 매

번 새로운 GIF 화일이름을 생성한다. 인터넷이 처음 탄생할 때, 주로 검색을 목적으로 이용되었기 때문에 브라우저에 캐쉬(cache)의 기능을 두어 이미 검색되어 브라우징한 화일은 캐쉬에 저장하여 새로이 같은 이름의 화일이 브라우징될 경우에 서버에서 로드하지 않고 캐쉬에 있는 화일을 브라우징한다. 검색을 목적으로 하는 웹의 이용에서는 시간 절약을 위해서 매우 유용하다. 그러나 본 시스템과 같은 경우에는 매번 브라우저로 로드되는 결과의 화일 이름이 달라야 한다. 만약 같은 이름으로 서버에서 브라우저로 처리 결과를 전송할 경우, 비록 결과의 내용이 다를지라도 처음에 로드한 결과만이 브라우징된다. 그러므로 항상 서버에서 전송되어 오는 GIF 화일이름은 브라우저로 전송될 때마다 바꾸어 주어야 한다.

14:는 X-terminal에 의해 Xlisp-Stat을 실행하여 통계 그래픽(PS 화일)을 생성하는 C 언어의 함수이다. 이 함수 내부에 포함되어 있는 histo.lisp는 Xlisp-Stat에서 수행되는 응용프로그램으로서, 7:에서 만든 데이터 화일을 이용하여 통계 그래픽을 PS 화일 형태로 생성하여 저장하도록 한다.

15:부터 19:까지는 20:의 system() 함수에서 PS 화일을 GIF 화일로 변환하기 위한 매개변수 parmfile을 만드는 과정이다.

20:은 14:에서 만든 PS 화일을 GIF 화일로 변환하기 위하여 ImageMagick 패키지의 convert 프로그램을 사용하며, 이를 C 언어 함수인 system()에 의해서 실행함으로써 GIF 화일을 16:에서 만든 화일 이름으로 저장하기 위한 함수이다.

21:은 GIF 화일 이름과 함께 연결(concatenation)된 디렉토리를 없애고 순수하게 GIF 화일이름만 남기기 위한 사용자 함수이다.

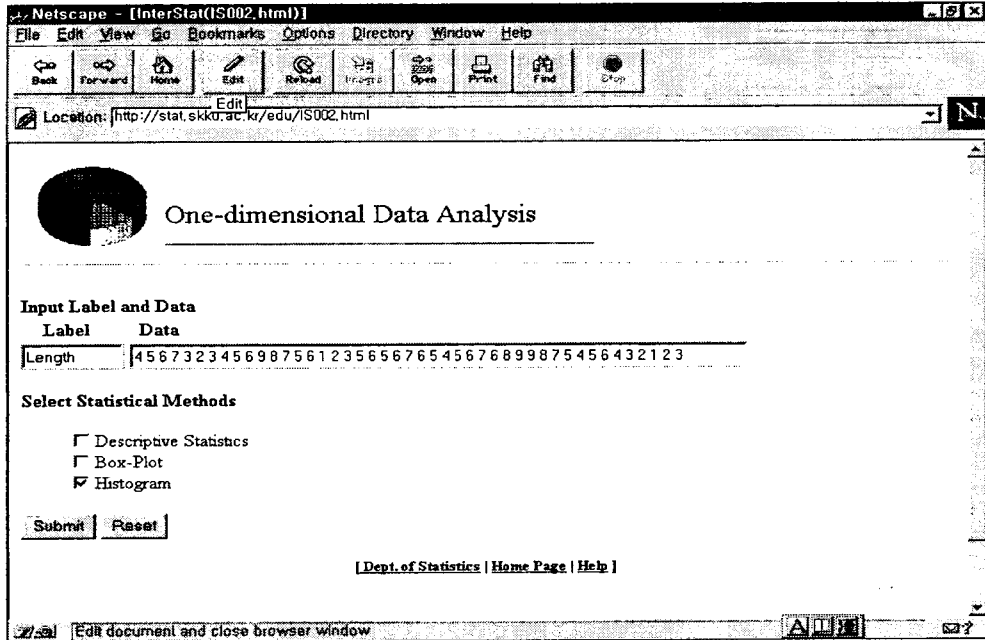
28:은 20:과 21:에서 변환한 GIF 화일을 브라우저에 출력한다.

한편 이 프로그램은 CGI 기반의 웹 응용프로그램을 만들기 위한 ANSI C 언어의 표준 라이브러리인 CGIC를 이용하였다.

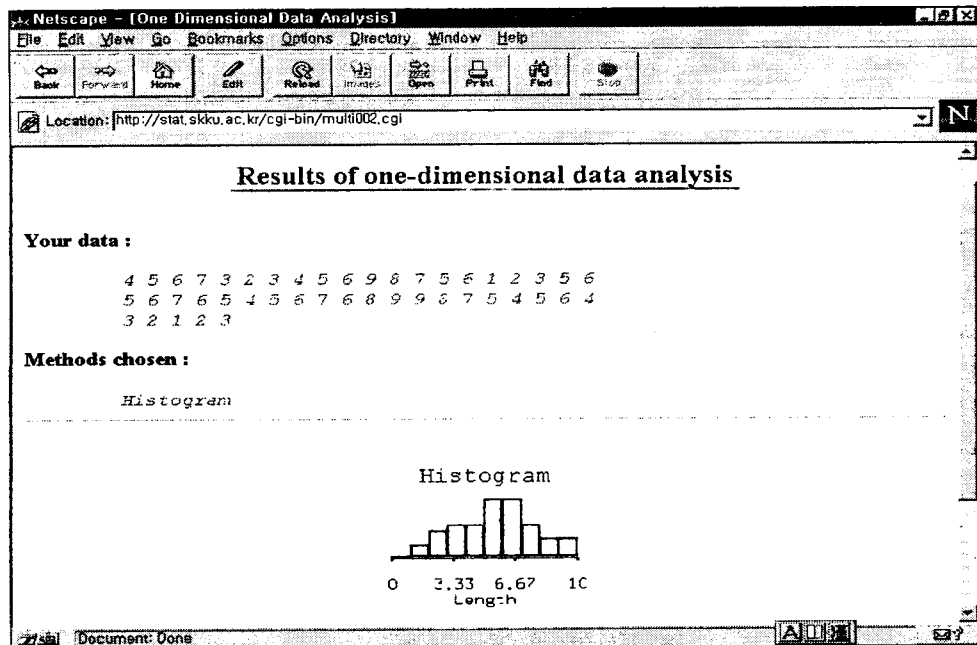
2.4 구현 예(例)와 결과 고찰

본 논문에서는 서버에 있는 통계 패키지가 생성한 히스토그램을 클라이언트에서 브라우징할 수 있도록 한 예를 사용하였다. 여기서 사용한 CGI 프로그램은 여러 프로그래밍 언어에 의해 작성할 수 있었으나 표준 언어인 C에 의해 구현한 결과, Xlisp-Stat과의 인터페이스에 문제가 없었다. 그림 3은 분석하고자 하는 데이터와 통계적 방법을 입력하는 화면으로 HTML에 의해 작성된 Fill-in 폼이다. 그림 4는 그림 3에서 데이터와 통계적 방법을 입력하고 실행 지시(submit)를 하면 CGI 프로그램에 의하여 서버에 있는 Xlisp-Stat을 수행하고 그 결과를 브라우저에 나타낸 것이다. 이와 같은 방식을 적용하여 통계분석시스템을 구축함으로써 사용자는 분석하고자 하는 데이터를 입력하여 그 결과를 그래픽으로 받아 볼 수 있다.

그러나 이 인터페이스를 구현함에 있어서 나타난 문제점은 원격의 클라이언트에서 통계 패키지를 액세스하여 결과로 나타난 PostScript 화일을 저장하고, PS 화일을 GIF 화일로 변환 및 저장하는 과정에서 화일의 쓰기(write 혹은 creation)를 허용함으로써 서버 시스템에 부담을 줄 수 있다. 또한 브라우저에 나타난 그래픽의 질(quality)이 만족한 만한 수준이 되지 못했다. 이는 PS 이미지 파일을 GIF 이미지 파일로 변환하는 유틸리티의 매개변수 조정이 관건이 되며, 또한 우수한 이미지 변환 유틸리티의 선정도 고려되어야 할 것이다. 그래픽의 질의 문제는 통계 패키지에 의존하지 않고 프로그램 언어를 이용한 모듈의 개발로 해결의 실마리를 찾을 수 있을 것이다.



<그림3> 서버의 Xlisp-Stat을 실행시키기 위한 입력 폼



<그림 4> 서버에 있는 Xlisp-Stat에 의한 실행 결과(실시간 그래픽 생성)

3. 결 론

최근 인터넷 기술이 도입되면서 통계분석에 있어서 그래픽을 처리하는 방법에 대한 많은 연구가 진행되고 있다. 본 논문에서는 서버에서 통계 그래픽을 처리하는 두 가지 방법 중에서 통계 패키지와 그래픽 변환 프로그램을 이용하여 실제로 통계 그래픽 처리를 구현하였으며, 순수하게 프로그래밍 언어를 활용하는 방법은 생략되었다. 본 논문에서 구현된 통계 그래픽을 서버에서 처리하는 방법에 있어서 다음과 같은 결과를 얻을 수 있었다.

장점으로는 개발자 입장에서 볼 때, 이해하기 쉽고 다루기 쉬운 개발 도구를 이용함으로써 시간적인 단축과 개발의 부담을 경감시켜 준다. 이는 웹 서버와 통계 패키지와의 인터페이스를 통하여 통계 패키지의 실행 및 클라이언트에 브라우저될 이미지 변환 등의 모듈이 포함된 CGI 프로그램을 작성하기만 하면 되기 때문이다.

단점으로는 웹 서버에 의존해서 통계 패키지를 이용할 때는 통계 패키지의 저작권 문제가 대두되므로 공개 소프트웨어를 이용해야 된다는 제한점이 있으며, 인터넷 이용자가 웹 서버에 데이터 화일을 생성하고 이미지를 변환하는 과정에서 서버에 부담을 줄 수 있다.

따라서 본 논문에서 단점으로 지적한 문제들의 해결 방법으로서, 웹 서버에서의 통계 패키지 사용에 따른 저작권의 문제는 본 논문에서와 같이 공개 프로그램을 이용하면 되지만, 그렇지 못할 경우에는 프로그래밍 언어(특히 JAVA)에 의하여 통계 분석 모듈이나 그래픽 생성 모듈을 개발하여 문제를 해결할 수 있다. 또한 웹 서버에서 통계 그래픽의 처리에 따른 웹 서버의 부담(overhead)은 API 등의 기법을 통하여 통계 그래픽을 처리하는 방안으로 해결할 수 있을 것으로 생각된다.

참고문헌

- [1] Boutell, T.(a) (1995), "CGIC : an ANSI C library for CGI Programming", <http://www.va.pubnix.com/man/cgic.html>
- [2] Boutell, T.(b) (1995), "gd1.2 : A graphics library for fast GIF creation", <http://www.boutell.com/gd/>
- [3] Cleveland, W. S. and McGill, R.(1987), *Dynamic Graphics for Statistics*, New York: Chapman & Hall.
- [4] Cristy, J.(1995), "ImageMagick", <http://www.wizards.dupont.com/cristy/>
- [5] December, J.(1995), *Presenting JAVA*, Sams Net
- [6] December, J. and Ginsburg, M.(1995), *HTML & CGI*, Sams Net
- [7] Deleeuw, J.(1995), "Regression", <http://www.stat.ucla.edu/cgi-bin/textbook/regres-comp.cgi>
- [8] Dwight, J. and Erwin, M. (1996), *Using CGI*, Que, 600-601
- [9] Friendly, M.(1995), "Running SAS on the Web", <http://pascal.math.yorku.ca/SCS/Online/sasweb/>
- [10] Garcia, J. L.(1996), "Submit Xlisp-Stat", <http://www.stat.ucla.edu/cgi-bin/Xlisp-Stat.cgi>
- [11] Gosling, J., Arnold, K.(1995), *The Java Programming Language*, Addison Wesley,1996.
- [12] Keith D.(1996), "SurfStat australia", <http://surfstat.newcastle.edu.au/surfstat>

- [13] Lang, R.(1996), "Ghostscript", <http://www.cs.wisc.edu/~ghost/>
- [14] Lemay, L. (1995), *Teach Yourself Web Publishing with HTML in a Week*, SAMS
- [15] Newman, A., et al.(1996), *Using JAVA*, QUE, 257-266
- [16] Phillips, R. L. (1991), "An Interpersonal Multimedia Visualization System", *IEEE Computer Graphics & Application*, May, 20-27
- [17] Rossini, T.(1995), "Xlisp-Stat and the WWW", <http://www.stat.psu.edu/~rossini/statlearn/#xlispstat>
- [18] Shin, T. S.(1996), "Teach Modules : Teaching Statistical Concepts to Undergraduate Students", <http://www.public.iastate.edu/~sts/lesson/head/head.html>
- [19] Tittel, E. & Gaither, M.(1995), *Internet Wolrd 60 Minute Guider to Java*, IDG books
- [20] Tierney, Luke (1990), *LISP-STAT : Object-Oriented Environment for Statistical Computing and Dynamic Graphics*, Wiley, NewYork
- [21] Tierney, Luke (1995), "Recent Developments and Future Directions in LISP-STAT", *Technical Reports*, No. 608., School of Statistics University of Minnesota.