Web Browser Based Statistical Software - The Next Generation of Statistical Computing

Ratul Chakraborty

Abstract*— There are essentially two ways to deliver an application on PC/Laptop/tablet/smartphone: as a client-side/native application (developed using the appropriate platform-dependent development kit and installed on user devices) or as a web application (developed using web standards and accessed through a web browser-there's nothing to install on user devices). Traditionally, we are familiar with the native applications. But the recent trend shows that, in near future web applications will become more competitive with native applications due to the ubiquity of web browsers and platform independent programming features. HTML5, Java Scrip and WebGL will bring a new level of computing to the web. At present we have a bunch of native Statistical Computing applications (for PCs and Laptops only) but there is a scarcity of good web application of such type which can run on any computing device (from PC to smartphone) without any hazard.

Index Terms*—Native applications, Web applications, Graphical User Interfaces, Programming Language, Statistical Software.

I. INTRODUCTION

The web may have been created to share information, online forms, interface for remote instructions and emails but today web browsers are capable of supporting sites that are getting close to the look and feel of applications we run directly on our computers, tablets and smartphones.

The native applications we had used to on computers/smartphones/tablets are tied to a specific operating system (Windows, Linux, Mac, Android etc.). Under normal circumstances, a windows application can only run on a window PC or an Android application can only run on a smartphone running with an Android OS.

In contrast, Applications built around standard technologies created to build modern websites (HTML5, CSS and JavaScript) offer the promise of being able to build one application that can reaches nearly every device in the world that runs a browser. So, in near future, the web could be the platform of choice for building an application, rather than Windows, Linux, Mac, Android, etc.

Till date, various types of web applications have been developed for different purposes. For example, Office tools(word processors, spreadsheets, presentation tools, etc.), Business & Accounting tools, Project management tools, Computer-aided design and video editing tools, Games, News & Weather tools, Social & Communication tools are broad classification of web applications. But there is a scarcity of good web application for statistical data analysis. This paper will investigate past and recent developments in the area of web statistical data analysis applications and throw light on developing a good one in respect of efficiency and free of cost.

Manuscript Received on February 16, 2015.

Ratul Chakraborty, Department of Statistics, M. B. B. College, Tripura University, Agartala, India.

II. PAST AND RECENT DEVELOPMENTS OF STATISTICAL DATA ANALYSIS SOFTWARES USING WEB APPLICATIONS

In this section we are going to introduce some web applications which are designed to perform statistical computations in online or offline mode. A "comprehensive" list of such web applications are: StatCrunch, Rweb, SOCR, JSTAT, OpenEpi, SISA, SciStatCalc, VSUstat, ProtoGenie and Statlets. A brief discussion of these applications is given below.

A. StatCrunch (http://www.statcrunch.com/)

StatCrunch developed by Webster West is powerful web-based statistical software that allows users to collect data, perform complex analyses, and generate compelling results. It is developed on the basis of client-server architecture. With StatCrunch we can analyze data by a large number of Statistical Procedures. Details of Statistical Procedures Available in StatCrunch may be found in http://www.statcrunch.com/learn-about/features.php#proced ures . Inspite of different complex analyses supported by StatCrunch it have the following demerits:

1) It is commercial software.
2) We can't use it in offline mode.
3) We can't process any user-defined scripts by StatCrunch Engine.

B. Rweb (http://www.math.montana.edu/Rweb/)

Rweb developed by Jeff Banfield is a freely accessible statistical analysis environment that is delivered through the web browser. Like StatCrunch it is based on client-server architecture. In Rweb, the commands and scripts inserted in client (browser) are processed in the Rweb server through the popular free statistical software R. Although any R script can be analyzed through Rweb, it has the following drawbacks:

1) Without internet it is useless.
2) There is no menu driven user interface for Analysis.
3) Spreadsheet like data editor is not present in it for editing and inserting data.

C. SOCR (http://www.socr.ucla.edu/)

The Statistics Online Computational Resource (SOCR) is a suite of online/offline tools and interactive aids for hands-on learning and teaching concepts in statistical analysis and probability theory developed at the University of California, Los Angeles. The SOCR resource is HTML, XML and

Java based and all materials and tools are freely available over the Internet. Although SOCR supports a large and comprehensive library of educational and computational tools, it has the following drawbacks:

1) It's analyses part covers only Test of Statistical Hypothesis, Linear Regression, Logistic Regression, PCA, Survival Analysis, ANOVA 1-Way, ANOVA 2-Way and Hierarchical Clustering.
2) As the SOCR Analyses runs through Java Applets, appropriate version of JRE should be installed in the PC.
3) As Java applet can't access local file system, local data files can't be loaded in SOCR Data editor for analysis. Accessing of PC clipboard is also restricted in Java applets. So typing of data into the SOCR Data editor is the only way of analyze the data.
4) In online mode SOCR Applets takes a long time to load, especially in a low speed internet zone like India.

D. JSTAT (http://jstat.github.io/)

jStat is a statistical library written in JavaScript that allows us to perform advanced statistical operations in a R like environment (claimed by the developers) from a browser. Being a relatively young project, it has the following drawbacks:

1) The library is partially complete.
2) Only a subset of R environment is implemented in this library.
3) It is not a standalone library. It depends on jQuery and the flot plug-in.
4) It's not a full-fledged stats package like R, SPSS etc.

E. Open Epi (http://www.openepi.com/)

OpenEpi is free and open source software for epidemiologic statistics. The programs are written in JavaScript and HTML. We can use it from any JavaScript enabled browsers in online or offline mode. It has the following drawbacks:

1) Only some simple analyses (T-test, ANOVA 1-Way, Chi Square test, etc.) are included in this software.
2) Data editor is not available for inserting raw data.
3) Local data files can't load for analysis.
4) This software is useful for epidemiologic statistics only.

F. SISA (http://www.quantitativeskills.com/sisa/)

Simple Interactive Statistical Analysis (SISA) based on HTML & JavaScript allows us to do statistical analysis directly on the Internet through a browser like OpenEpi discussed above. Compared to OpenEpi it has some additional analyses and raw data can be inserted in a textbox for analysis. Other drawbacks of this software are same as OpenEpi.

Rests of the applications (SciStatCalc, VSUstat, ProtoGenie and Statlets) have some or all the drawbacks discussed above.

III. NECESSARY REQUIREMENTS OF A GOOD WEB APPLICATION FOR STATISTICAL COMPUTATION

1) The application must have a simple but professional look user interface.
2) The application must have a spreadsheet like data editing interface supporting large data set and this data editing interface must be capable of handling clipboard content (Copy, Paste & Cut).
3) It should be capable of handling local file system.
4) Its analyses should cover almost all branches of statistics.
5) The computation time of analyses must be nearest or less compared to the well-known applications SPSS, R, etc.
6) It should be scriptable, i.e. it must be design in a manner such that the user can define his/her own script for analysis like R scripts.
7) The software must be menu driven so that non-programmers can use it easily through menu and variable/option selection dialog boxes like SPSS, Minitab, etc.
8) We should design the application in such a way that most well-known web browsers can run this application efficiently in online/offline mode for optimum coverage.
9) It must have a full supporting documentation and Help content.
10) It must be a free software for rapid development by the world-wide community.

IV. NECESSARY TECHNOLOGY SUPPORT FOR BUILDING A GOOD STATISTICAL SOFTWARE FOR WEB BROWSER

For building a statistical software discussed in the previous section we need a web browser supporting the following technologies:

1) Java Script: An efficient Java Script engine is necessary for making (from GUI to Analyses) a good browser based statistical software.
2) HTML5 Canvas: This technology is necessary for preparing Statistical Graphs.
3) HTML5 content-editable attribute: It is required for editing outputs.
4) FileReader API: For reading local data files, FileReader API is required.
5) Clipboard API: For accessing clipboard data (through Cut, Copy & Paste) this API is required.
6) Typed Arrays: For fast calculation and low memory use, typed arrays (char, int, float, double, etc.) is the only way rather than using Java Script normal array.

V. SELECTION OF WEB BROWSER

The web browser is a critical part of PC. All web browsers are not same in respect to supporting technologies. Considering the needs of heavy numerical computation, statistical Graphs and maximum coverage of worldwide users it's important to select the optimal browser out of many, many browsers available for different operating system. To narrow the selection down, let's consider the most popular: Internet Explorer, Chrome, Firefox, Safari and Opera. From the following discussion we will be able to select the optimum browser for our work.

A. *Market Share of Desktop Web Browsers & Operating System Support*

Table I: Market Share of Desktop Web Browsers - December 2014

	Browser					
Source	**Chrome**	**Internet Explorer**	**Firefox**	**Safari**	**Opera**	**Other**
Stat Counter	50%	25%	18%	5%	2%	2%
W3Counter	43%	18%	16%	15%	3%	7%
Wikimedia	48%	18%	17%	5%	2%	11%
Average	**47%**	**20%**	**17%**	**8%**	**2%**	**7%**

Source: Official websites of StatCounter, W3Counter & Wikimedia

.

From the above table it is clear that we should focus on five browsers (Chrome, Internet Explorer, Firefox, Safari and Opera) only for maximum coverage (93%).

Table II: Operating System support

	Browser				
OS	**Chrome**	**Internet Explorer**	**Firefox**	**Safari**	**Opera**
Windows	✓	✓	✓	*X*	✓
OS X / Mac	✓	*X*	✓	✓	✓
Linux	✓	*X*	✓	*X*	✓
BSD	*X*	*X*	✓	*X*	✓
Unix	*X*	*X*	✓	*X*	✓
Android	✓	*X*	✓	*X*	✓

Source: Official websites of different Browsers.

From the above table, it is seen that, Internet Explorer supports only Windows OS and Safari is designed for Mac OS only. But the other browsers support most of the Oss. So it would be natural to zoom our focus on Chrome, Firefox and Opera.

B. *Necessary Web Technology Support for Statistical Data Analysis*

Table III: Web Technology Support

		Browser				
Technology	**Version**	**Chrome**	**Internet Explorer**	**Firefox**	**Safari**	**Opera**
Java Script	Current	✓	✓	✓	✓	✓
	Future	✓	✓	✓	✓	✓
HTML5 Canvas (for Statistical Graphs)	Current	✓	✓	✓	✓	✓
	Future	✓	*!*	✓	*!*	✓
HTML5 content-editable attribute (for editing outputs)	Current	✓	✓	✓	✓	✓
	Future	✓	*!*	✓	*!*	✓
FileReader API (for reading data files)	Current	✓	✓	✓	✓	✓
	Future	✓	*!*	✓	*!*	✓
Clipboard API (for cut, copy & paste)	Current	✓	✓	✓	✓	✓
	Future	✓	*!*	✓	*!*	✓
Typed Arrays (for fast calculation & low memory use)	Current	✓	✓	✓	✓	✓
	Future	✓	*!*	✓	*!*	✓

Source: Official websites of different Browsers

.

At present all of our selected web browsers support the necessary web technologies for building a good statistical software, but there is no certainty of these kind of supports in future form Internet Explorer and Safari. So it will be better to consider Chrome, Firefox and Opera rather than Internet Explorer and Safari for this task.

C. Computation time (in milliseconds) for various types of analysis

We know that for numerical computing, FORTRAN, C and C++ are the languages of programmers/scientists. These three languages are the backbone of all mathematical/statistical libraries and softwares (eg: R, SPSS, SAS, Matlab, numpy/scipy, etc.). But for web browser based applications, Java Script is the one and only choice of programmers. Most of the Java Script engines of popular web browsers are written in C/C++. So in the table given below we compare the efficiency of different web browsers Java Script engines with respect to C. For this comparative study we select a list of statistical analyses/tools which are used frequently for developing various statistical applications. Run/Computation time (in milliseconds) for each of these analyses (for different programming languages with same algorithm) are recorded using Windows-7 (for C, Chrome, IE, Firefox and Opera) & MAC (for Safari) PCs (Intel(R) Core(TM) 2 Duo CPU E7500 @ 2.93GHz 2.93 GHz, 4 GB RAM).

Table IV: Run/Computation Time (in milliseconds) for Some Selected Statistical Analyses

		Programming language					
		C	**Java Script**				
		Compiler	**Browser**				
Analysis		GCC with O2 optimization	Chrome 31	Internet Explorer 11	Firefox 25	Safari 7	Opera 18
Random Sampling. Size 10000000	Uniform [0, 1)	372	374	461	380	373	374
	Exponential (2)	788	790	1080	1142	787	827
	N(0, 1)	901	905	1562	1395	957	905
	Gamma (2, 3)	1900	1872	3246	3431	2392	1903
	Beta-I (2, 3)	3911	3931	8133	7669	6321	4025
R-matrix based on Pearson's correlation coefficient for 10 variables and 1000000 records		225	229	1092	283	487	249
Computation of median for a random sample of size 10000000		200	201	245	235	279	218
Sorting of 10000000 random numbers by Quick Sort algorithm		1875	1897	2760	2445	2515	1909
Ranking of 10000000 random numbers		4410	4523	6546	4834	6013	4680
Matrix multiplication of 1000x1000 and 1000x1000 matrices of random numbers		6630	6969	17093	7871	54194	6927
Inversion of a matrix (1000x1000) of random numbers		6998	6958	17206	7603	55308	7005
Eigen analysis for a random matrix of order 700x700		9126	5358	13347	6619	42981	5445
Singular value decomposition for a random matrix of order 400x500		2098	2104	5133	5030	5239	2200
Total		**39434**	**36111**	**77904**	**48937**	**177846**	**36667**

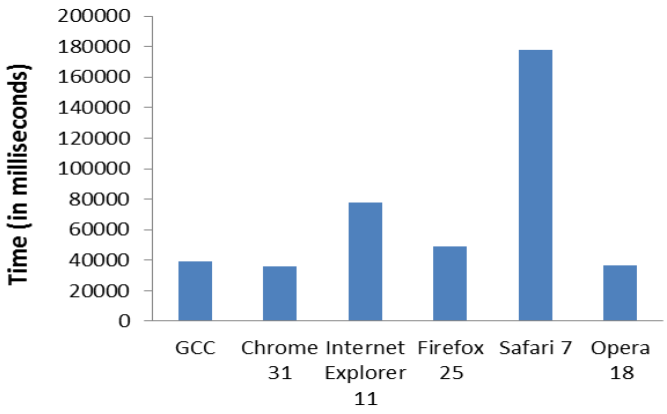

Fig.1. Total Computation Time for Some Selected Statistical Analyses

From the above table and bar diagram it is observed that the total run time of the analyses for C, Chrome and Opera are more or less same but in case of Internet Explorer and Safari it is higher. Moreover in case of Safari the variation in computation times is very high compared to others. These higher total and variations in computation times are due to Matrix multiplication, Inversion of a matrix and Eigen analysis. But these analyses are essential tools for most of the widely used statistical analyses like Regression, Linear models, PCA, Factor Analysis, etc. So for better performance (in respect of less computation time), Chrome and Opera should be our first choice.

Form the above discussion we can give ranks to web browsers in order to fulfil the necessity of developing a good web application for statistical data analysis.

Table V: Ranks of Web Browsers

Rank	1	2	3	4	5
Browser	Chrome	Opera	Firefox	Internet Explorer	Safari

VI. CONCLUSION

We cannot make a forecast of how statistical computing environments will look like in the future from the present time. But we can visualize about what statistical computing can look like in the future if we make web browsers as base of Statistical software. For this purpose, web browsers play a significant role and these can be noted as follows:

1) Google predicts that, in near future computer applications will be based on the so called "web-apps" accessed through web browsers that run on terminal-like computers/mobile devices. As part of its strategy, Google engaged into developing web applications that would mimic the functionalities of classic applications, but running entirely on the web. Google went even further and also developed a web browser (Google Chrome) that would make those applications to look even more like the classic ones (better layout, capacity to run off-line and some in-device storage). Other web browser developers (Firefox, Opera, Safari, etc.) are also following the footprints of Google.
2) The particular strength of web browsers is certainly the possibility to write libraries/packages, which are no longer bound to a certain platform.
3) Java Script can be used as a universal and efficient tool to develop and exchange statistical methods.
4) With HTML5 graphical user interfaces are much easier to implement.

REFERENCES

1. http://caniuse.com/
2. http://statcounter.com/
3. http://clicky.com/
4. http://w3counter.com/
5. http://wikimedia.org/
6. http://statpages.org/
7. http://statcrunch.com/
8. http://www.math.montana.edu/Rweb/
9. http://www.socr.ucla.edu/
10. http://jstat.github.io/
11. http://www.openepi.com/
12. http://www.quantitativeskills.com/sisa/

AUTHORS PROFILE

Ratul Chakraborty, graduated in Statistics from M. B. B. College, Tripura University, Agartala, India and received the Master degree in Statistics from The University of Burdwan, West Bengal, India in 2000. Since July 2005, he is with M. B. B. College, Agartala, India, occupying teaching position (Assistant Professor) in the Department of Statistics.