

Graphical analyses of clinical cancer data using SAS macros and SAS / Graph

Karen L. Knorr, M.S., David G. Mascorro, B.S., Greg S. Shaw,
Susan G. Hilsenbeck, Ph.D., John J. Hough, B.S., Gary M. Clark, Ph.D.

University of Texas Health Science Center at San Antonio
Division of Medical Oncology, Department of Medicine
7703 Floyd Curl Drive
San Antonio, Texas 78284-7884

Address reprint requests to: Gary M. Clark, Ph.D.
University of Texas Health Science Center
Department of Medicine/Medical Oncology
7703 Floyd Curl Drive
San Antonio, TX 78284-7884, U.S.A.

This research was supported in part by NIH grants CM07305 and CA54174.

Summary

The objectives of phase I studies of new anticancer agents are to evaluate the safety of new agents and to determine the best doses to use in future studies. In order to achieve these goals, a large number of variables are repeatedly measured on each patient during the course of these studies, and these data are entered and maintained in relational databases. Data managers need methods to perform quality control checks of these data and to identify data entry and data recording errors. The clinicians involved in the studies need to follow patients' reactions to the agents over time, both individually and in groups, in order to spot possible toxicities.

Graphical presentations display data in a compact, yet highly informative manner. By using the SAS macro environment, we produced an easy to use, interactive system that gives investigators the ability to transform their choices of variables into two different types of graphs to help address different types of questions. SAS windows permits non-SAS trained users to readily choose among studies, types of graphs, and the huge array of variables collected on each patient. SAS macros also allows automation of certain choices, thus reducing the time and effort required of the user. Once chosen, the data are manipulated by different SAS procedures and data steps in order to produce the needed values and formats. We produced the two different types of graphs by using GPLOT in the SAS graphics package. The "annotate" option places special features such as patient labels or arrows for treatment times. The SAS graph template system then arranges the graphs one, four, six or eight to a page so they can be more quickly printed and read. As a unit, the system reads and subsets the data, manipulates it into the proper form, produces graphs, places the graphs in a template and prints them, all within a user-friendly system aimed at giving the user control of the raw data.

Key words: clinical trials, SAS, graphs, data presentation, data management

Introduction

Important information can often be overwhelmingly out of reach simply because of the huge bulk of the raw data. Graphs can help overcome this barrier by summarizing information in an intuitive and compact manner. This paper presents an application designed to help put researchers and data managers of phase 1 cancer drug studies in better contact with their data.

Background:

Phase 1 Anticancer Trials

Phase 1 anticancer trials investigate drugs that have never been tested on humans before with the main purpose of determining toxicities and side effects and how these factors limit the dosage. Using patients who have either failed all standard therapies or have no standard therapy as an option, the doctors gradually increase the dosage of the drug to ascertain the maximum effectiveness without serious side effects or toxicities. Since anticancer agents are generally harsh drugs, patients on these studies need to be monitored closely on a large number of variables. In addition, these variables must be collected at least once a week, sometimes more often. The result is a daunting pile of raw data for a single patient, and even more for a whole study.

Relational Databases

To handle this mass of data, our drug development team uses Informix™ relational database running on a SUN Microsystems™ network. Informix uses tables of related variables as the building blocks of the database. Typically, we need thirty or more tables to capture all of the information over the length of the study for each patient. Each study consists of approximately 30 patients and these make up the rows of the tables. These tables are related to each other by patient number, date, and other factors determined by the needs of the study, such as tumor location. Although there is potentially a great number of toxicities these patients could experience, many can be identified through routine laboratory tests. The most common problems with anticancer agents are hematological, such as destruction of white blood cells or platelets, or biochemical, such as abnormal levels of sodium, potassium, bilirubin, or a host of others. Anticancer drugs can affect any of these parameters, and most

patients can only tolerate small departures from the normal ranges. Thus, investigators constantly need to monitor all of these potential problems.

Users

We have different users accessing the phase 1 database, all with different interests. Besides the varying software needs, there are three different types of hardware that interface with the SUN network: simple CRT terminals, SUN SPARCstation, and MacIntoshes across a TCP/IP network.

The data for the phase 1 trials are collected by nurse data managers who obtain laboratory reports and transfer the results to hand written case report forms. Since they are responsible for the accuracy of the data, their main concern is checking for extreme and unlikely points. They access the database through all three interfaces. They are next entered by data entry people using CRT terminals. Finally, the clinicians, who monitor the progress of patients and write up the results of the study, need to see patient data summaries for all the variables so they can detect any adverse events. They access the database primarily through MacIntoshes.

Problem

The investigators need a summary of each patient's progress over time on any or all of forty-four laboratory or physical exam values, while the data managers require a simple method of scanning for extreme and improbable data points. Graphs provided the ideal answer to both of these questions, since they summarize a great deal of information in an easily interpretable form that can be scanned quickly. We developed two types of graphs to serve these needs: one is a graph showing a patient's laboratory values over time, the other is a scatterplot of medians versus ranges.

From an interfacing viewpoint, the users need the ability to choose which study, type of graph, and laboratory values they want to graph. During the development of this program, we also had to consider that these users were not trained in SAS, so we needed to produce an extremely user-friendly interface between the users and the database, as well as build ways to automate the whole system and cut down on the amount of time the user spent with it.

We chose SAS graph to produce the graphs because of the ability to control the output and the high quality of the finished product. By using the SAS macro environment, we were able to create automatic loops that allowed users to produce multiple graphs with the touch of a few keys. Finally, the SAS macro windows helped us produce a user-friendly, flexible interface to the data base that allows users to choose quickly and easily the variables and graphs that will be most useful to them.

Methods

Overall, we needed to pull the data from an Informix database, ask the users what they wanted to graph, graph it, and then either present the graphs in a window or transfer the finished file to a VAX system for printing. Figure 1 outlines the hardware connections and figure 2 overviews the program set up. In figure 2, underlined entries represent user choices, the outlined entries represent tasks performed automatically by the program, and circled entries represent internal choices.

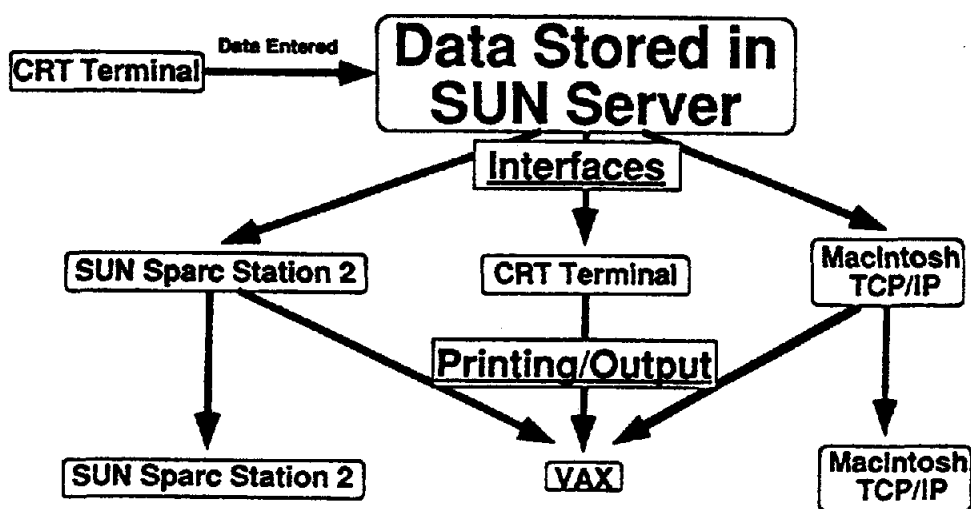


Figure 1--An overview of the hardware network.

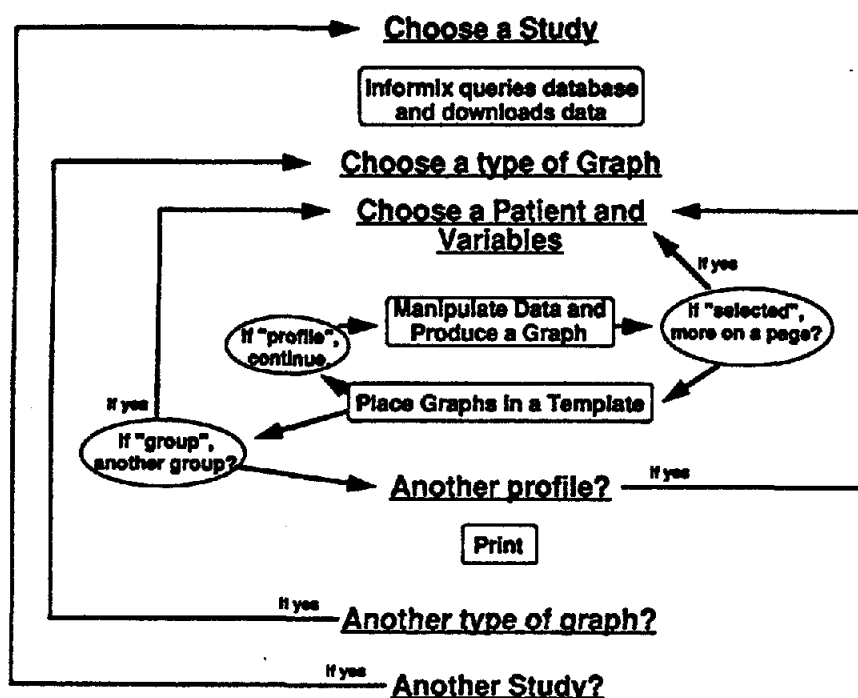


Figure 2--An overview of the graphics program where underlined entries represent user choices, the outlined entries represent automatic tasks performed by the program, and circled entries represent internal choices.

Interface

The data are stored in Informix SQL databases, with several studies located in each. Individual protocols need to be extracted separately and downloaded to an ASCII file. This is accomplished by using a preprogrammed, interactive Informix query which allows the user to select a particular study and a particular drug within the study. It is then downloaded to an ASCII file that can be read into the SAS graphing macros.

After this, a shell script menu appears, allowing the user to choose between two types of graphs. Once chosen, the appropriate macro is loaded and run.

Windows

The graphics macro first obtains the list of variables to be graphed through a series of windows. These windows need to be generalizable to different devices, simple to use, and easy to follow. They need to be translatable to many different devices, since the data managers use simple terminals with

basic CRT screens and the investigators are partial to MacIntoshes. Therefore, we developed the program in an X-windows environment on a Sun SPARC station. It needs to be simple to use since the users did not know SAS, and it needs to be easy to follow since there are so many variables to choose from, for any given patient.

SAS windows is a good tool for extracting information from the users. The windows are set up in the default monochrome for the terminals. Although plain, they are clearly legible for all the different types of devices. The SAS windows are easy to follow as well--by including instructions to choose a number and then hit enter, even a novice can follow along. To solve the problem of potential confusion, the windows break down the variables into biologically based groups. The first window shows all of the groups and once one is chosen, the next window shows all of the variables contained in that group. By grouping these variables in logical packets, we have avoided overwhelming the user with choices.

Macro

The goal of the whole application was to automate the production of quality graphs for the users' choices of variables. To make it easier, we have automated as much of the process as possible without losing the flexibility to choose specific variables. In addition, we present the graphs in a logical, easy to read format.

We gave the user an "all of the above" choice for all variables in a group or all groups for a patient or protocol. In order to automate this, we built a hierarchy of macros around a central core that produces a single graph. In the lowest level of the hierarchy, there is little automation and the user chooses the variables one by one to be graphed and in the order in which they will be presented. At the next level, the user selects a group of variables to be graphed. The user can also produce a profile containing all of the variables in a study by invoking the group macro for all groups. For the patient tracking graph, the user can choose to graph all variables by group for up to five patients, thus almost completely automating the graphing process.

Underneath these levels of automation is the core macro which reads in the data and creates the graph. Depending on the type of graph, it rescales dates, finds medians, sorts dates, restructures the data set, or flags data points which are out of range. The graph produced is either a connected line graph for patient tracking or a scatterplot for the outlier graphs. It also creates an annotated set specific to the type of graph being generated. For the patient tracking graph, the annotated data set draws vertical arrows at each treatment date, horizontal dotted lines at extreme values of the variable, and two different warnings about missing or incorrect laboratory dates. The annotated set for the outlier detection graphs adds the patient number next to the plotted point for values that are deemed to be too far from the centroid of the scatterplot. All graphs are labeled with the name of the variable and other pertinent information, such as the patient number.

Presentation

To present the graphs, another macro places them into an appropriate template of one, four, six or eight graphs to a page. The "all group variables" selection places all variables for that group on one page and labels it according to the group. The full profile of a patient or a study consists of the eight pages of group graphs. If a user selects variables one by one, the macro automatically puts them into the right sized template and labels them as "Selected" instead of by a group label.

The graphs are then printed or shown on the screen in a window on either the SUN SPARC station or the MacIntosh. Due to time limitations, the graphs are not able to be viewed on the CRT terminals. We chose to print on the VAX system since it has printers designed to print large numbers of pages. In order to accomplish this, we link Informix query language to SAS macros via shell script and then pass the file and print commands using ftp and kermi.

Examples:

The resulting program walks the users through a series of menu windows (see fig. 3), allowing them to choose specific graphs for specific studies. The two types of graphs, patient tracking and outlier, are presented in figures 4 and 5. The patient tracking graphs (fig. 4) show all blood chemistry values

for a particular patient over time. This allows the investigator to view a patient's progress over time. Several patients' can simultaneously be examined for a particular laboratory result, to detect possible adverse effects or double check safety margins. Arrows on the graphs show treatment times for the patient so associations between drug administration and a laboratory test can be evaluated. In addition, tolerance limits are shown on the graph as horizontal lines. Printed warnings occur for laboratory dates which are out of range and for graphs that are missing data for longer than two weeks. For this example, patient #8984 showed a sharp decline in sodium levels after the fourth treatment. Although these levels were not classified as life threatening, the sudden decline and slow recovery could mean drug induced toxicities of which the clinicians need to be aware.

PHASE I GRAPHING MENU

1. Outlier Detection Graphs
2. Patient Tracking Graphs
3. Raw Data Check

Please select a number (e to exit):

Figure 3--An example of the shell script menu for choosing a type of graph.

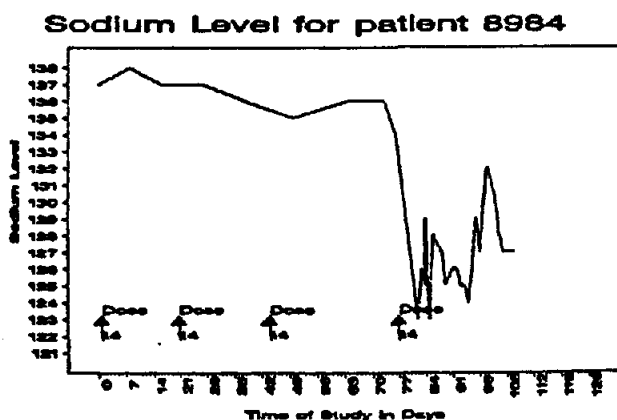


Figure 4--An example of the patient tracking graph.

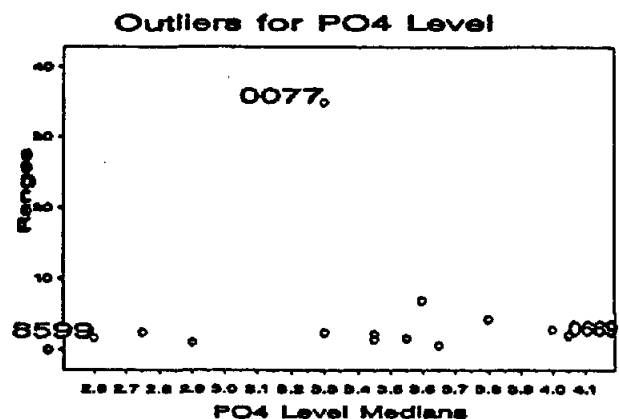


Figure 5--Example of the outlier detection graph.

The outlier detection graphs (fig. 5), on the other hand, are more oriented towards data management than patient management. They simplify the search for misread or miskeyed values by highlighting possible problem values. Each point on the graph is the median sodium level vs. the range of sodium levels for a specific patient in a protocol, so any patient with an unlikely value will show up on the outer reaches of the graph. These points are conveniently labeled with the patient number to make follow up simpler. For example, patient #0077 can be found in the extreme top of the PO4 level graph, indicating a possible problem for PO4 level. After looking at the appropriate raw data, we see an impossible value for PO4, which could be the result of a miskey or mistake on the laboratory form. These graphs provide the nurse data managers with a quick quality control check.

Discussion

The users seem quite happy with the choice of graphs and have been using them to detect and follow up on various problem points. Although it has not yet been placed at the disposal of the intended users, pretests indicate that the windows are easy to follow and that the general set up is simple to use.

The biggest problem has been efficiency and the amount of time required to run the program. We are currently investigating the consolidation and replacement of certain data steps in an effort to improve the speed because a great deal of time is lost in printing the finished graphs to a file. Different device drivers require different lengths of time and based on this and the available printers, we have chosen to use the Hewlett-Packard Laser printer II driver known as HPLJ300.

By making a user-friendly, flexible interface with the database, we have put investigators and data managers in closer contact with their data. They can look at the data in a number of ways that would have been difficult if not impossible by considering the raw data alone. If used regularly, the phase 1 quality control graphics program can help identify data problems and drug-related toxicities, resulting in improved quality control of these important clinical trials.