Proceedings of the 1985 Winter Simulation Conference
D. Gantz, G. Blais, S. Solomon (eds.)

# INTRODUCTION TO SANDIE

Arne Thesen
Department of Industrial Engineering
University of Wisconsin-Madison.

## ABSTRACT

This paper gives a short introduction to SANDIE, a statistical data analysis system designed to run on the IBM-PC and similar computers. Special care was made in the development of SANDIE to design a system that would serve as an aid in the development of an intuitive "sense of numbers". This goal was achieved by providing very fast response times, easy to use graphics and commands that encourage the casual exploration of properties of data sets.

## WHAT IS SANDIE?

SANDIE is an interactive tool for statistical analysis of data on an IBM-PC. Designed for an educational environment, SANDIE provides a combination of capabilities that differs somewhat from those provided by several of the main frame based systems that now are migrating towards the PC. On the positive side, SANDIE is substantially easier to use (menus and commands), provides several less commonly available capabilities related to simulation, and can comfortably be used on a standard PC with 256k bytes of memory and a double sided drive (ie. no hard disk required). On the negative side, SANDIE provides fewer statistical tests, and it does not have the report generating capability of some of the larger systems.

Among SANDIE's features are:

- Capacity for up to 16,000 observations;
- command and menu orientation;
- confidence intervals;
- data manipulation and transformation;
- eight random number generators;
- fitting of seven distributions;
- histograms and plots;
- one and two-way analysis of variance;
- probability formulae for three distributions;
- regression;
- runtest and autocorrelation;
- simulation of simple queuing systems.

Two versions of SANDIE are available. One uses the 8087 numeric co-processor, the other does not. Both versions provide the ability to program the alternate function keys and the ablility to execute macro instructions in an iterative fashion. Neither version is intended as a replacement for systems designed to analyze very large datasets. Rather, SANDIE is intended as an easy-to-use desk top aid in analyzing and exploring data. First time users of SANDIE should read Appendix 1 before attempting to use the system.

## BASIC CONCEPTS

Users of SANDIE must understand the following five fundamental constructs:

- columns - how data is stored;
- the current column - default data for operations;
- windows - how SANDIE talks to the user;
- commands - one way to tell SANDIE what to do;
- menus - another way to control SANDIE .

These constructs are briefly discussed in this section.

| row | Col 1 | Col2 | Col 3 | Col 4 | Col5 |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 206.51 | 23.00 | 8.85 |
| 2 | 0.12 | 3.00 | 187.02 | 56.00 | 22.51 |
| 3 | 0.70 | 1.00 | 197.04 | | 22.29 |
| 4 | 0.72 | 5.00 | 182.49 | | 1.07 |
| 5 | 0.00 | 2.00 | 201.53 | | 4.53 |
| 6 | 0.26 | 6.00 | 207.64 | | 19.06 |
| 7 | 0.25 | 8.00 | 188.74 | | 20.62 |
| 8 | 0.93 | 4.00 | 214.38 | | 0.64 |
| 9 | 0.52 | 1.00 | | | 14.86 |
| 10 | 0.69 | 2.00 | | | 2.23 |
| 11 | | | | | 17.78 |

Figure 1: Typical data sheet.

COLUMNS: SANDIE stores data in columns (Figure 1). Each column is assumed to contain observations about a single variable such as test scores or observations from an experiment. Separate columns may contain completely independent data or they may contain related data (such as the weight and height of subjects).

THE CURRENT COLUMN: This is the column that SANDIE works on unless told to use on another column. For example the command **HISTOGRAM** displays a histogram of the data in the current column. The index of the current column is always displayed on the screen, and its value is easily changed.

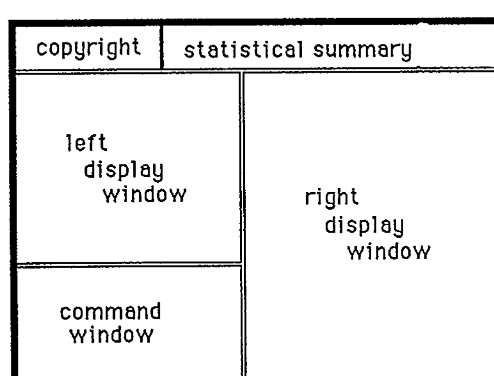



Figure 2: SANDIE's windows.

WINDOWS: The SANDIE display is partitioned into six independent windows (Figure 2). Four of these are extensively used:

- the *statistical summary* window contains a summary of the data in the current column.

- the *right* and *left display* windows are used by SANDIE to display results of calculations.

- the *command window* is where all user commands are echoed.

The following shows how the data in column one in Figure 1 would be summarized in the summary window:

```
Col obs mis    mean  stdv  m/s   min     max
 1   10   0    0.42  0.33 1.25  0.00     0.9
```

```
F1 Read/Write    F2 Random varia
F3 Transformat   F4 Simulation
F5 Test current  F6 Fit distribu
F7 Show <alt F>  F8 X-Y Plots
F9 Show commands F10 Bye
```

| | | |
|---|---|---|
| 7 Data | 8 Edit | 9 Labels |
| 4 Prev column | 5 Histo | 6 Next Column |
| 1 List | 2 Summary | 3 Box Plot |

Figure 3: Primary pop-up menus.

POP-UP MENUS: The inexperienced user is likely to control SANDIE by making selections from pop-up menus (Figure 3). These appear whenever the space bar is hit. Two different menus are provided:

- the function keys - mostly for secondary menus.
- the numeric keypads - mostly for data exploration.

It is seen that the <5> key on the keypad causes a histogram to be generated and the <F2> key brings up a random number generator menu. A summary of all of SANDIE's menus is given in Appendix 3.

COMMANDS: SANDIE may also be controlled through the use of conventional commands. While commands are not as easy to use as pop-up menus they have the advantage that the user has more freedom in specifying optional parameters. A summary of SANDIE's most important commands is given in Appendix 2.

## AN INTRODUCTORY EXAMPLE

Before we discuss SANDIE's capabilities in more detail, we will illustrate the use of SANDIE through a simple example. In this example, we will first read a datafile containing census data for a hospital, then we will plot a histogram of the census, finally we will compute a confidence interval about the mean census.

To read the file, we first hit the space bar to bring up the main function key menu :

```
F1 Read/Write menu  F2 Random variates
F3 Transformations  F4 Simulation
F5 Test current col F6 Fit distribution
F7 Show <alt F> keys F8 X-Y Plots
F9 Show commands    F10 Bye
```

We see that <F1> brings up the Read/Write menu. We hit this key:

```
F1 Read file    F2 Print Screen
F3 Save screen  F4 Save file
F5              F6
F7              F8
F9              F10
```

We now hit <F1> to tell SANDIE that we want to read a file. SANDIE responds by asking for the file name:

**Filename(SANDIE.DAT):_**

We type CENSUS80.DAT to indicate that we want to read a file on the current drive by this name. (this is a file containg hospital census data for 1980). After a short while the (red) statistical summary window on the top of the screen is updated to read as follows:

```
Col obs  mis  mean  stdv   m/s   min     max
 1  366    0 15.76  8.82  1.79  1.00   31.00
```

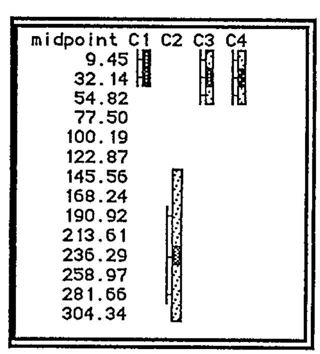
From this we see that column one of the data file contains 366 observations (1980 was a leap year) with a mean of 15.76, a standard deviation of 8.82 etc. The heading mis is used to indicate the number of "missing" observations in the column, no observations are missing in this particular column. However column one was not the only column in the data set that we just just read. A complete summary of the data in all of the columns read in is given in the right display window:

| Ci | obs | mean | var | stdv | max | min | mis |
|---|---|---|---|---|---|---|---|
| 1 | 366 | 15.76 | 77.86 | 8.82 | 31.00 | 1.00 | 0 |
| 2 | 366 | 233.37 | 556.16 | 23.58 | 293.00 | 142.00 | 0 |
| 3 | 366 | 32.20 | 118.05 | 10.87 | 62.00 | 6.00 | 0 |
| 4 | 366 | 32.17 | 82.53 | 9.08 | 61.00 | 10.00 | 0 |

In addition to reading in the data summarized above, SANDIE also reads any labels that may have been assigned to different data columns. Such labels are useful in describing the content of data columns. The labels defining the content of the above data are displayed by hitting the key:

| i | n | label |
|---|---|---|
| 1 | 366 | DAY OF MONTH |
| 2 | 366 | CENSUS |
| 3 | 366 | ADMISSIONS |
| 4 | 366 | DISCHARGES |

We now hit <3> to generate the following "boxplot" of our hospital data



To look at the daily census data in column two in more detail, we must change the current column from column 1 to column 2. One of the keypad keys is used for this purpose. To figure out which one to use, we hit the space bar to bring up the keypad menu:

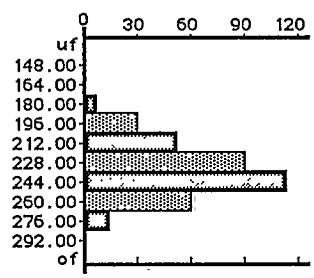
| | | |
|---|---|---|
| 7 Data | 8 Edit | 9 Labels |
| 4 Prev Column | 5 Histo | 6 Next Column |
| 1 List | 2 Summary | 3 Box Plot |

We see that the <6> key (right arrow) issues this command. We press this key. The following changes are observed:

- the prompt is changed to CO2:
- the statistical summary window is changed to:

```
Col obs mis  mean  stdv  m/s   min    max
 2 366   0 233.37 23.58 9.90 142.00 293.00
```

Now we want to see a histogram of the data in this column. From the keypad menu we see that the <5> key causes a histogram to be displayed. We hit this key. The following histogram is displayed.




NOTE: It is possible that absolutely nothing happens when the <5> key is hit. If this happens to you, press the <NUM LOCK> key and try again. (Shift <5> sends no signal to the computer.)

Finally we want to develop confidence intervals for the true means of our four data columns. This is done as follows:

- Hit the spacebar to bring up the main menu:

```
F1 Read/Write menu   F2 Random variates
F3 Transformations   F4 Simulation
F5 Test current col  F6 Fit distribution
F7 Show <alt F> keys F8 X-Y Plots
F9 Show commands     F10 Bye
```

- Hit <F3> to bring up the Test current col menu:

```
F1 Autocor 15      F2  Conf interval
F3 Autocor 80      F4 Part auto 15
F5 Batch mean  5 F6 Part auto 80
F7 Batch mean 10 F8 Runtest
F9 Batch mean 50 F10 Save to C5
```

Hit <F2> to compute confidence interval. The following report is displayed:

| | | 95% range for mean | | | |
|---|---|---|---|---|---|
| Col | obs | low | x-bar | high | stdv |
| 1 | 366 | 15.00 | 15.76 | 16.52 | 8.82 |
| 2 | 366 | 231.34 | 233.37 | 235.40 | 23.58 |
| 3 | 366 | 31.26 | 32.20 | 33.13 | 10.87 |
| 4 | 366 | 31.39 | 32.17 | 32.95 | 9.08 |

We note that the 95% confidence interval for the true mean of the census is (231.34 - 235.40).

SANDIE can also be operated as a command driven system. This approach gives the user more control and flexibility in specifying optional parameters. The <F9> key causes the most important commands to be displayed. The following (underlined) commands would replicate the above example:

```
C01: READ CENSUS80.DAT
C01: C2
C02: HISTO
C02: CONF
```

A summary of all of SANDIE's commands is given in Appendix 2.

## OVERVIEW OF CAPABILITIES

Now that we have seen how easy it is to use SANDIE, it is time to discuss some of the many tasks that SANDIE can perform.

In the exploratory phase of data analysis, the user might want to use SANDIE as a tool to edit, summarize, display, and plot the data. Some of the capabilities provided by SANDIE for this purpose are:

| Activity | Function key | Command |
|---|---|---|
| Boxplots | <3> | (none) |
| Edit data | <8> | **Edit** *{row}* |
| Enter data | <7> | **Data** *values* |
| Histograms | <5> | **Histo** *{lo {wide }}* |
| Read file | <F1> <F1> | **Read** *filename* |
| Show data | <1> | **List** |
| Timeseries plot | <3> | **Plot** *Ci* |
| X-Y plots | <3> | **Plot** *Cx* $Cy_1\{ .. Cy_n\}$ |

Various transformations of the original data could also be explored at this time. Nine different transformations (such as log, sin , cosine etc) are provided by the <F3> Transformations menu:

```
F1 Ci = Abs(Cj) F2 Ci = e**(Cj)
F3 Ci = Cum(Cj) F4 Ci = Roun(Cj)
F5 Ci = Cos(Cj) F6 Ci = Sin(Cj)
F7 Ci = Diff    F8 Ci = Sqrt(Cj)
F9 Ci = Log(Cj) F10
```

In addition, several more elaborate transformations are available in the command mode. Among these are:

| Activity | Command | Example |
|---|---|---|
| Append other column | **Appe** *Cfrom* | **APPE C3** |
| Column arithmetic | *= expression* | **= C3 / C4** |
| Compute finite diff. | **DIff** *Cfrom lag* | **DIFF C3 1** |
| Moving average | **Mova** *Cfrom lag* | **MOVA C1 3** |

Following the exploratory phase, the user may explore the nature of a single variable. The <F5> Test current col menu is provided for this purpose:

```
F1 Autocor 15      F2  Conf interval
F3 Autocor 80      F4  Part auto 15
F5 Batch mean   5  F6  Part auto 80
F7 Batch mean  10  F8  Runtest
F9 Batch mean  50  F10 Save to Ci
```

Some of these menu selections (such as auto correlation and batch means) yield results that may warrant further study. Such values are saved in an empty column if <F10> is hit prior to making one of these selections. The index of the column used for this purpose is indicated in SANDIE's report.

The command equivalents of these menu selections give the user more freedom in specifying optional parameter values:

| Activity | Command | Example |
|---|---|---|
| Autocorrelation | **AC** *{lag}* | **AC 23** |
| Batch means | **BAtch** n d | **BATCH 50** |
| Confidence intervals | **Conf** | **CONF** |
| Partial autocorrelation. | **Part** *{lag }* | **PART 31** |
| Run test | **Runt** | **RUNT** |

At this point it might also be useful to see if the data could have been generated by one of several different probability distributions. The <F6> Fit distribution menu is provided for this purpose:

```
F1 Beta          F2  Normal
F3 Erlang        F4  Poisson
F5 Exponential   F6  Uniform
F7 Gamma         F8
F9               F10 Exit
```

A drawback to the menu approach to fitting of distributions is the fact that no distribution parameters can be specified (they are estimated from the data). This restriction does not apply when the command mode is used:

| Distribution | Command | Example |
|---|---|---|
| Beta. | **FITB** *{a {b }}* | **FITB 3 5** |
| Erlang | **FITE** *{mean {shape }}* | **FITE 3 1** |
| Gamma | **FITG** *{mean {shape }}* | **FITG 3 1.5** |
| Normal | **FITN** *{mean {stdv }}* | **FITN 9 0.6** |
| Poisson | **FITP** *{mean }* | **FITP 1.4** |
| Uniform | **FITU** *{lo {hi }}* | **FITU 0 1** |
| Weibull | **FITW** *loc shape scale* | **FITW 0 2 5** |

And, some of the commands (no menu provided) that provide insights into the possible relationships between variables are:

| Procedure | Command | Example |
|---|---|---|
| Mann-Whitney u test for means | **Mann** *Ci Cj* | **MANN C1 C2** |
| One way ANOVA | **Onew** *Cdata Ctreat* | **ONEW C1 C4** |
| Regression | **Regr**$CyCx_1..Cx_n$ | **REGR C1 C2** |
| Two way ANOVA | **Twow** *Cdata* $Ctr_1$ $Ctr_2$ | **TWOW C1 C2 C3** |

SANDIE also provides random number generators from eight different distributions. Seven of these can be accesed through the <F2> Random variates menu:

```
Anything else to exit
Space to generate 100 variates
F1 Beta          F2  Gamma
F3 Bernoulli     F4  Normal
F5 Exponentia    F6  Poisson
F7 New count     F8  Uniform
F9 0.00          F10  1.0
     low              high
```

All eight generators may be used in the command mode:

| Distribution | Command | Example |
|---|---|---|
| Beta . | **Beta** *{a {b }}* | **BETA 1 2** |
| Exponential | **Expo** *{mean }* | **EXPO 1.5** |
| Erlang | **Erlang** *{mean {shape }}* | **ERLA 1.5 3** |
| Gamma | **Gamma** *{mean {shape }}* | **GAMM 1 3.5** |
| Normal | **Normal** *{mean {stdv }}* | **NORM 1.4 3.9** |
| Poisson | **Poisson** *{mean }* | **POIS 2.4** |
| Uniform | **Unif** *{lo {hi }* | **UNIF 0 1** |
| Weibull | **Weib** *scale shape loc* | **WEIB 1 2 0** |

Finally SANDIE has a built in model for simulation of waiting lines. The <F4> Simulation menu gives access to this model:

```
---<space> to simulate one event-
F1 Incr stime      F2  Decr stime
F3 Incr schape     F4  Decr shape
F5 Incr servers    F6  Decr servers
F7 Departures      F8  Start simul
F9 Capture wait    F10 Exit
```

The use of this menu is illustrated in the following section.

## EXAMPLE - ANALYSIS OF SIMULATION OUTPUTS

To illustrate some of the unique features of SANDIE of special interest to simulators, we will first simulate the performance of a single-server queuing system, then we will use SANDIE to analyze the resulting customer waiting times.

We first hit **<F4>**. This brings up the simulation menu and an initial description of the system to be simulated:

```
---<space> to simulate one event-
F1 Incr stime      F2  Decr stime
F3 Incr schape     F4  Decr shape
F5 Incr servers    F6  Decr servers
F7 Departures      F8  Start simul
F9 Capture wait    F10 Exit
```

Command window

```
Avg arrival interv:  1.00   Seed[1].:11111
Departures........:  1000   Seed[2].:22222
Servers...........:  1
Mean service time.:  0.70   Shape:     1
Waitingtimes not saved
```

Right display window

SANDIE is now ready to simulate 1000 departures from a single server queuing system with exponentially distributed interarrival times with a mean of one and Erlang distributed service times with a mean of 0.7 and a shape of one.

We use the function keys to change the model as follows:

- the mean service time is changed from 0.7 to 0.9 in four increments of 0.05 each by hitting **<F1>** four times;

- the shape of the service time distribution is increased to 2 by hitting <F3> once;

- finally <F9> is hit save the wait times of departing customers in a currently empty column.

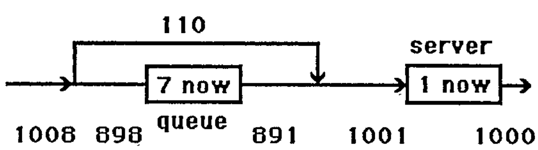
The right display window now looks as follows:

```
Avg arrival interv:  1.00  Seed[1].:11111
Departures........:  1000  Seed[2].:22222
Servers...........:  1
Mean service time.:  0.90  Shape:    2
Waiting times placed in C1

              Right display window
```

We now hit <F8> to start the simulation run. A trace of every twentieth event is given in the left display window, and after approximately one minute (30 seconds for the 8087 version) the following report is displayed in the right display window:

```
               System   Queue     Server(s)
--------       ------   ------   ----------
    in          1008      898       1001
    out         1000      891       1000
    now            8        7          1

    avg-n       4.96     4.07       0.90
    max-n         20    19.00       1.00
    avg-t       4.82     4.41       0.87
    max-t      18.94    17.92       4.64
    std-t       3.86
```

As shown in the following figure, 1008 customers entered the system; of these, 898 entered the queue and 110 were immediately served without waiting. At the time the simulation ended, one customer was receiving service and seven customers were waiting for service.




To further analyze the performance of the simulated queuing system, we return to the main menu to use SANDIE's other tools to analyze the transit times saved in column one.

We hit <F10> to exit the simulation part of SANDIE. At this time, the summary window at the top of the screen reflects the fact that 1000 departure times is stored in column 1:

| Col | n | mis | mean | stdv | m/s | min | max |
|---|---|---|---|---|---|---|---|
| 1 | 1000 | 0 | 4.82 | 3.86 | 1.25 | 0.04 | 19.9 |

We note that this summary data agrees with the data diplayed above. (For example the mean of the 1000 observations in column 1 is 4.82 as was the observed mean wait time for the 1000 simulated departures.) As this mean is a realization of a random process, we may wish to estimate a confidence interval about this mean. However this is not easily done as the data we are dealing with is highly auto correlated.

The batch means technique has been suggested as a way to deal with this problem. This technique as well as tools to determine if the underlying assumptions are violated are offered in the Test current col menu. This menu is brought up:

```
F1 Autocor 15      F2  Conf interval
F3 Autocor 80      F4  Part auto 15
F5 Batch mean   5  F6  Part auto 80
F7 Batch mean  10  F8  Runtest
F9 Batch mean  50  F10 Save to Ci
```

We hit <F10> to tell SANDIE to place resulting batch means in an available column. Then we hit <F9> to perform a 50 bin batch means analysis (the 1000 original observations are grouped into 50 bins, and the means of the 50 bins are used to estimate a variance about the mean. ) SANDIE responds as follows:

```
Means saved in C2
Observations.= 1000
Batches..... =   50
Batch size...=   20
Discards.....=    0

Smallest mean=   1.03
Average mean.=   4.82
Largest mean.=  14.76
Stdv of means=   3.48

+-----------------------------------------+
| 90% confidence interval for batch mean  |
+-------------+-------------+-------------+
|  low bound  |    mean     | upper bound |
+-------------+-------------+-------------+
|    4.01     |    4.82     |    5.63     |
+-------------+-------------+-------------+
```

At this time we type <6> to change the default column from column one to column two. The summary window now reads as follows:

| Col | obs | mis | mean | stdv | m/s | min | max |
|---|---|---|---|---|---|---|---|
| 2 | 50 | 0 | 4.82 | 3.48 | 1.39 | 1.03 | 14.8 |

The theory of the batch means confidence interval is based on the premise that the means collected in column 2 are uncorrelated and normally distributed. Let us see if these assumptions are satisfied.

First let us look for autocorrelation. To do this we bring back up the <F5> Test current col menu, then we press <F1> to perform an autocorrelation analysis that considers the first 15 lags. The resulting report is shown in Figure 2:

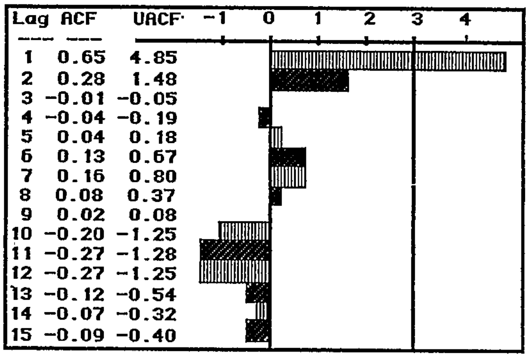
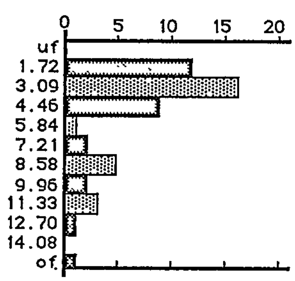



Figure 2: Correlation between successive batch means.

If there were no correlation between successive batch means, then the resulting normalized autocorrelation coefficient would be a normally distributed random variable with a mean of zero and a variance of one; and, there would be no systematic relationship between the coefficients for different lags. This is not the situation shown in Figure 2. Here a definite pattern is shown, and the first order autocorrelation coefficient is more that four standard deviations away from the mean. From this we conclude that the assumption that our batch means are independent random variables is violated.

Let us now see if the means could be from the normal distribution. First we hit <5> to get a histogram of the means:




Distribution of batch means

This histogram does not seem to describe a random variable from the normal distribution. However, to be sure, we use SANDIE to apply a chi-square test to see if the difference between a normal distribution and the observed distribution are statistically significant. From the pop-up menus we see that <F6> brings up the Fit distribution menu:

| | | | |
|---|---|---|---|
| F1 | Beta | F2 | Normal |
| F3 | Erlang | F4 | Poisson |
| F5 | Exponential | F6 | Uniform |
| F7 | Gamma | F8 | Weibull |
| F9 | | F10 | Exit |

Now we hit <F2> to see if the assumption of normally distributed means is reasonable. The following report is produced:

Fitting Normal with mean: 4.82 Stdv: 3.48

| Bin | Low | High | Exp. | Obs | Err |
|---|---|---|---|---|---|
| 1 | -999.99 | 0.37 | 5.0 | 0 | 5.0 |
| 2 | 0.37 | 1.90 | 5.0 | 7 | -2.0 |
| 3 | 1.90 | 3.01 | 5.0 | 17 | -12.0 |
| 4 | 3.01 | 3.94 | 5.0 | 5 | 0.0 |
| 5 | 3.94 | 4.82 | 5.0 | 5 | 0.0 |
| 6 | 4.82 | 5.70 | 5.0 | 2 | 3.0 |
| 7 | 5.70 | 6.64 | 5.0 | 1 | 4.0 |
| 8 | 6.64 | 7.75 | 5.0 | 1 | 4.0 |
| 9 | 7.75 | 9.28 | 5.0 | 5 | 0.0 |
| 10 | 9.28 | 9999.99 | 5.0 | 7 | -2.0 |

Chi-square= 43.60
Pr{Chi-square>43.60| df= 7} = 0.000

We get a Chi-square of 43.60. SANDIE tells us that there is a probability of 0.000 that a normally distributed data set would have resulted in a chi-square value greater than or equal to 43.60. Again, we must conclude that the assumptions used to justify the batch means analysis procedure did not hold for the case studied here.

## DISCUSSION - MENU VERSUS COMMAND ORIENTATION

SANDIE's menus are designed to make the system easy to use for the inexperienced user. As a consequence default parameter assignments are made whenever possible. When this is not possible, parameters are estimated from the data ( the Fit distribution menu), or, separate menu selections may be offered for different parameter values (the Test current col menu). This approach maximizes ease of use at the cost of user control. For example:

- the number of histogram bins and their location cannot be specified,
- autocorrelation studies can only be performed for lags of 15 and 80,
- batch means analysis can only be performed for 5, 10 or 50 batches,
- plots have default origins.
- parameters of distributions to be fitted are estimated from the data.

These limitations do not apply when the command approach is used. In addition, certain commands that depend heavily on user specified parameters do not have equivalent menu access.

A final advantage of the command approach is that the alternate function keys may be programmed by the user. Commands provided for this purpose include:

| Command | Example | Effect |
|---|---|---|
| **Do F*i*** | **DO F9** | Transfer control from the "program" in the current function key to the program in the <alt> <Fi > key |
| **F*i*** *text* | **F3 HI** | Assign command(s) to <alt><Fi > |
| **LOOP *n*** | **LOOP 8** | Exit loop after *n* passes |

ACKNOWLEDGEMENTS

Many individuals have had a crucial role in the development of SANDIE. The first version of SANDIE was written for a 48K CP/M system by J. Leung, and important sections of the current code were written by T.J.Wang, L.Lei and H.G.Chen. Significant credit should also go to P.Brennan and other early users. SANDIE would never be finished were it not for their willingness to put up with several generations of bugs.

## APPENDIX 1: USING SANDIE FOR THE FIRST TIME

Here we give step by step instructions to users that are running the SANDIE system for the first time.

### Step 1: Make sure the hardware configuration is appropriate.

SANDIE will run on any IBM-PC or equivalent computer with at least 256k bytes of memory and at least one double sided disk drive.

SANDIE'S display looks particularly nice when a high resolution color monitor is used. However any type monitor will yield acceptable displays. A printer is recommended, but not required. SANDIE translates IBM graphics characters to standard ASCII characters. Any printer should therefore give acceptable results.

An 8087 numeric co-processor is required to run the 8087 version. The system will hang if you try to run this version without one. The 8087 is an extra cost option. If you do not know if your computer has this option, then it probably does not. We recommend that you get one.

### Step 2: Boot the system.

SANDIE runs under PC-DOS 2.0 or newer. This is most likely the standard operating system provided with your computer. Before you can run SANDIE, you must boot the system using your copy of the original DOS operating systems disk. (The ıSANDIE distribution disk is not a bootable disk.) We assume that you are ıfamiliar with your computer and your operating system.

### Step 3: Run SANDIE

ıBefore you run SANDIE, you should change the defalt drive to the one you ıplan to use for SANDIE. For example, if you plan to insert the SANDIE disk into drive **B:**, change the default drive to drive **B** (by typing **B:<return>**); before you run SANDIE.

To run SANDIE, insert the SANDIE disk into the default drive and type **SANDIE** if you are running the standard version or type **SANDIE87** if you are running the enhanced 8087 version. After a very short time you should see a display such as the one shown in Figure 1. ıIf this display does not appear in 15 seconds or less, then something is wrong. The most likely problems are:

- your computer has less than 256k of memory;
- you did not place the SANDIE disk in the default drive (this will get you back to the operating system with an obscure error message);
- you are running the 8087 version on a system that does not have the 8087 numeric co-processor.

## APPENDIX 2: FUNCTION KEY MENUS

```
F1 Read/Write menu   F2 Random variates
F3 Transformations   F4 Simulation
F5 Test current col  F6 Fit distribution
F7 Show <alt F> keys F8 X-Y Plots
F9 Show commands     F10 Bye
```

Main Menu

```
F1 Read file     F2  Print Screen
F3 Save screen   F4  Save file
F5               F6
F7               F8
F9               F10
```

<u><F1> Read/Write</u> Menu

```
 Anything else to exit
 Space to generate 100 variates
 F1  Beta          F2   Gamma
 F3  Bernoulli     F4   Normal
 F5  Exponentia    F6   Poisson
 F7  New count     F8   Uniform
 F9  0.00          F10   1.0
      low                high
```

<u><F2> Random variates</u>

```
SANDIE for the 8087 | Col obs mis  mean stdv  m/s min  max
(c)1985 Arne Thesen |  1   0
PROGRAMABLE FUNCTION KEYS
-------------------------
<Alt-F1>
<Alt-F2>
<Alt-F3>
<Alt-F4>
<Alt-F5>                           Free memory = 575312 bytes
<Alt-F6>   MANN C1 C2
<Alt-F7>   ONEWAY C1 C2            7.       8.        9.
<Alt-F8>   TWOWAY C1 C2 C3          Data     Edit      Labels
<Alt-F9>   REGRESS C1 C2
F1 Read/Write      F2 Random variates   4 Prev  5        6 Next
F3 Transformatioı  F4 Simulation         column  Histo     Column
F5 Test current    F6 Fit distribution
F7 Show <alt F> .  F8 X-Y Plots        1.       2.        3. Box
F9 Show commands   F10 Bye              List     Summary     Plot
C01:_
```

Figure 1: SANDIE's Initial Screen

```
F1 Ci = Abs(Cj) F2 Ci = e**(Cj)
F3 Ci = Cum(Cj) F4 Ci = Roun(Cj)
F5 Ci = Cos(Cj) F6 Ci = Sin(Ci)
F7 Ci = Diff    F8 Ci = Sqrt(Cj)
F9 Ci = Log(Cj) F10
```

<F3> Transformations

```
---<space> to simulate one event-
F1 Incr stime     F2 Decr stime
F3 Incr schape    F4 Decr shape
F5 Incr servers   F6 Decr servers
F7 Departures     F8 Start simul
F9 Capture wait F10 Exit
```

<F4> Simulation

```
F1 Autocor 15      F2  Conf interval
F3 Autocor 80      F4 Part auto 15
F5 Batch mean   5 F6 Part auto 80
F7 Batch mean  10 F8 Runtest
F9 Batch mean  50 F10 Save to Ci
```

<F5> Test current col

```
F1  Beta          F2  Normal
F3  Erlang        F4  Poisson
F5  Exponential  F6  Uniform
F7  Gamma         F8  Weibull
F9                F10 Exit
```

<F6> Fit distribution

```
F1 Ymax = 294.00  F2  Xmax =  31.00
F3 Ymin = 142.00  F4  Xmin =   1.00
F5 Y1   =  C2     F6  Xaxis C1
F7 Y2   = none    F8  Y3    = none
```

Typical <F8> X-Y Plots menu

## APPENDIX 3: COMMANDS

Changing the current column

| | |
|---|---|
| *Ci* | Switch to column *i* |
| C\ | Switch to the next available column |

Data Displays

| | |
|---|---|
| **Hbins** *count* | Set count of bins for current histogram. |
| **Label** *text* | Assign a label to the current column. |
| **Low** *value* | Location of underflow bin for current histogram. |
| **Wide** *value* | Bin width for current histograms. |

Fitting of distributions to data

| | |
|---|---|
| **Bins** *count* | Bincount used for chi-square test |
| **Fitw** *a,b,c* | Fit data to Weibull distribution. |

Macros

| | |
|---|---|
| **DO** *Fi* | Transfer control to command string in <Alt> <*Fi*> |
| *Fi cmd;* | Assign command(s) to<Alt> <*Fi*> |
| **Loop** *i* | Return to command window (ie exit loop) after this statement has been traversed *i* times. |

Probability distributions

| Command | Displays probability that |
|---|---|
| **Pchi** *x df* | $X^2 > x$ if $X^2$ is chi-square distributed with *df* degrees of freedom. |
| **PF** $x\ df_1\ df_2$ | z is greater that *x* if z is from the F distribution with $df_1$ and $df_2$ degrees of freedom. |
| **PN** *x mn st* | z is greater than *x* iif z is from a normal distribution with mean *mn* and standard deviation *st*. |

The pseudorandom number generator

**Constant** *value*
**MUltiplier** *value*
**Seed** *value*
**STREam** *index*

Transformation of data

**Append** *Ci*
**Copy** *Ci*
**Fill** *Xlow* {*dx* { *Xhi* {*xfirst*}}}
**Pick** *Csource Creference lo* {*hi*}
= *expression*

ARNE THESEN
Dr. Arne Thesen, a professor of Industrial Engineering and Computer Sciences at the University of Wisconsin-Madison, maintains an active research program in areas related to decision support systems and computer simulation with particular emphasis on microcomputer based applications in the manufacturing area. His research has appeared in journals such as Management Science, Networks, Naval Research Logistics Quarterly, Simulation, Software, and Project Management. He is the author of "Computer Methods in Industrial Engineering" and co-author of "Systems Tools for Development Planning." Dr. Thesen serves as consultant to a number of institutions both in the public and private sector in the United States, Europe and Asia. He is a member of The Institute of Management Sciences, The Operations Research Society of America, and The Association for Computing Machinery.

Department of Industrial Engineering
University of Wisconsin-Madison
1513 University Avenue
Madison, WI 53706