



N 70 36368
CR 108584

STANFORD RSL Technical Report 70-11

1969/70 STANFORD SPECTRAL DATA MANAGEMENT SYSTEM

by

Michael Heathman

Systems Programmer
Stanford Remote Sensing Laboratory

Report Prepared Under

NASA Contract NAS9-7313
"Infrared Spectrometry Studies"
Phase III, Tasks 3.4, 3.5.

**CASE FILE
COPY**

June 1970

REMOTE SENSING LABORATORY
SCHOOL OF EARTH SCIENCES

STANFORD UNIVERSITY • STANFORD, CALIFORNIA

STANFORD RSL Technical Report 70-11

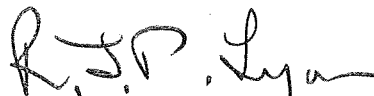
1969/70 STANFORD SPECTRAL DATA MANAGEMENT SYSTEM

by

Michael Heathman
Systems Programmer

Report Prepared Under

NASA Contract NAS9-7313
"Infrared Spectrometry Studies"
Phase III, Tasks 3.4, 3.5.

A handwritten signature in dark ink, appearing to read "R.J.P. Lyon". The signature is written in a cursive, flowing style. A horizontal line is drawn across the page just below the signature.

R.J.P. Lyon, Principal Investigator
Dept. of Mineral Engineering
Stanford University
Stanford, California 94305

June 1970

STANFORD RSL TECHNICAL REPORT SERIES

Already Issued

- 67-1 "Field Infrared Analysis of Terrain-Spectral Correlation Program" Part I (by R.J.P. Lyon)
- 67-2 "Field Infrared Analysis of Terrain-Spectral Correlation Program" Part II and Part III (by R.J.P. Lyon)
- 67-3 "Statistical Analysis of IR Spectra--Stanford Programs Applied to USGS Spectra in Tech Letter #13" (by R.J.P. Lyon)
- 67-4 "Computer Reduction and Analysis of an Infrared Image" (by Keenan Lee)
-
- 68-1 "Infrared Exploration for Coastal and Shoreline Springs" (by Keenan Lee)
- 68-2 "Re-evaluation of the Normative Minerals of Sonora Pass Rock Standards-University of Nevada Reports #7 and #12" (by Attila Kilinc)
- 68-3 "Nearest Neighbor--A New Non-parametric Test Used for Classifying Spectral Data" (by Paul Switzer)
- 68-4 Final Report "Field Analysis of Terrain"--NGR-05-020-115 (contains all the drawings and logic diagrams for Stanford CVF Spectrometer and Digital Data Recording System (by R.J.P. Lyon)
-
- 69-1 "Mission 78--Flights 1 and 2 Ninety-Day Report" (by R.J.P. Lyon)
- 69-2 "Quantitative Geological Analysis of Multiband Photography from the Mono Craters Area, California" (by G.I. Ballew)
- 69-3 "The Stanford Infrared Spectra Processing Package" (by John Moore with subsequent modifications by Michael Heathman)
- 69-4 "Thermal Dynamics at the Earth-Air Interface: The Implications for Remote Sensing of the Geologic Environment" (by Rossman Smith)
- 69-5 "Infrared Spectrometry Studies: Final Report Phase I" (by R.J.P. Lyon)

- 69-6 "Infrared Spectrometry Studies: Final Report Phase II"
(by R.J.P. Lyon and Attila Kilinc)
- 69-7 "Infrared Exploration for Shoreline Springs at Mono Lake,
California, Test Site" (by Keenan Lee)
-
- 70-1 "Detailed Ground Study of 8-13 μ m Infrared Imagery, Carrizo
Plains, Calif." (by R. Campbell, Jr., L. D. Hoover, and
F. Querol-Suné)
- 70-2 "Terms in Geologic Remote Sensing; A Condensed Glossary"
(by R.J.P. Lyon)
- 70-3 "Geologic Interpretation of Airborne Infrared Thermal Imagery
of Goldfield, Nevada" (by I.A. Kilinc and R.J.P. Lyon)
- 70-4 "Surface Heat Flow Studies for Remote Sensing of Geothermal
Resources" (by H. Hase, Geol. Survey of Japan)
- 70-5 "Geophysical Analysis of Thermal Infrared Imagery near Goldfield
Nevada: Experimental and Correlation Analysis" (by W.B. Irvine)
- 70-6 "The Multiband Approach to Geologic Mapping from Orbiting
Satellites: Is it Redundant or Vital?" (by R.J.P. Lyon) to be
published by AIAA
- 70-7 "Airborne Geological Mapping Using Infrared Emission Spectra"
(by R.J.P. Lyon and J. Patterson) to be published in Proc.
6th Remote Sensing Conference
- 70-8 "Pseudo-Radar: Very High Contrast Aerial Photography at Low
Sun Angles" (by R.J.P. Lyon, Jose Mercado, and Robt. Campbell,
Jr.) to be published in Photogrammetric Engineering
- 70-9 "Remote Sensing in Exploration for Mineral Deposits" (by
R.J.P. Lyon and Keenan Lee) to be published in Economic Geology
- 70-10 "Phenomena and Properties of Geologic Materials Affecting
Microwaves--A Review" (by D. Oberste-Lehn)
- 70-11 "1969/70 Stanford Spectral Data Management System" (by Michael
Heathman)

Table of Contents

	<u>Page</u>
I. Introduction.	1
II. Program Environment - The Stanford Computer Center. . .	3
III. Data Flow.	4
IV. Data Format.	7
A. NASA Format.	7
B. Truck Format.	9
C. Standard Format.	10
V. Program Descriptions.	12
A. TASK.	12
B. NEAREST NEIGHBOR.	16
C. BMD07M.	17
VI. Program Listings.	18
A. TASK Program.	18
B. NEAREST NEIGHBOR Spectrum Matching Program. . .	18
VII. Program Examples.	38
A. Housekeeping.	39
B. Plotting and Banding.	39
C. NEAREST NEIGHBOR Run.	39
D. BMD07M Run.	39

APPENDICES:

Appendix A: Smoothing and Differentiation of Data
by Simplified Least Squares Procedures

Table of Contents (cont.)

APPENDICES (cont.):

Appendix B: Block Radiometer Conversion Formulas

Appendix C: BMD07M--Stepwise Discriminant Analysis

DISTRIBUTION LIST

I. Introduction

This is a report on the data analysis system used by Stanford Remote Sensing Laboratory to read in, modify, and reduce the data obtained from both the NASA IR airborne spectrometer, and Stanford's SG-4 field spectrometer. In this write up are sections covering the general overview of the system, data flow and formats within the system, program explanations, and specific program-use instructions. The appendices contain system flow diagrams, references, program listings, and example program runs.

The system described here is not completely current because Stanford's new digital data system is not fully activated. Accordingly, the subroutine for that format has not been tested and inserted into the appropriate program. Also, the NASA data format described here is one used by NASA in 1968, which is not the same as that for 1969.

General questions may be answered by careful reading of Stanford RSL Technical Report 69-1 (Mission 78 - Ninety Day Report) which deals in a detailed manner with the pre-1968 data format. It is strongly advised. Copies may be obtained from the address below. The reader is also directed to Stanford RSL Technical Report 69-3 (The Stanford Spectral Processing Package) as a pre-cursor to this report.

Any questions on this report should be addressed to:

Dr. R. J. P. Lyon
Department of Mineral Engineering
School of Earth Sciences
Stanford University
Stanford, California 94305

II. Program Environment - The Stanford Computer Center

All the data processing for this project is done on Stanford's IBM 360/67. This machine is the center of Stanford's rather full and complex computational system whose special features are used to speed the data flow through our system. The core capacity for the machine is large, around 350K bytes, enabling the use of large programs to sacrifice size for speed and flexibility. The secondary storage system is also large and relatively cheap. There are 12 publicly available IBM 2314 disk volumes, each holding 1,000 K bytes, as well as one 2314 unit reserved for interchangeable private disk packs. This large storage capacity enables the data format to be designed with flexibility in mind, rather than size (resulting in an eventual space-saving by avoiding duplication of data).

Stanford also has a well-developed time sharing system which is instrumental in the data management of our system. The text editor, WYLBUR, is used to input and modify programs, delete or modify data files, and input jobs into the time share stream.

III. Data Flow

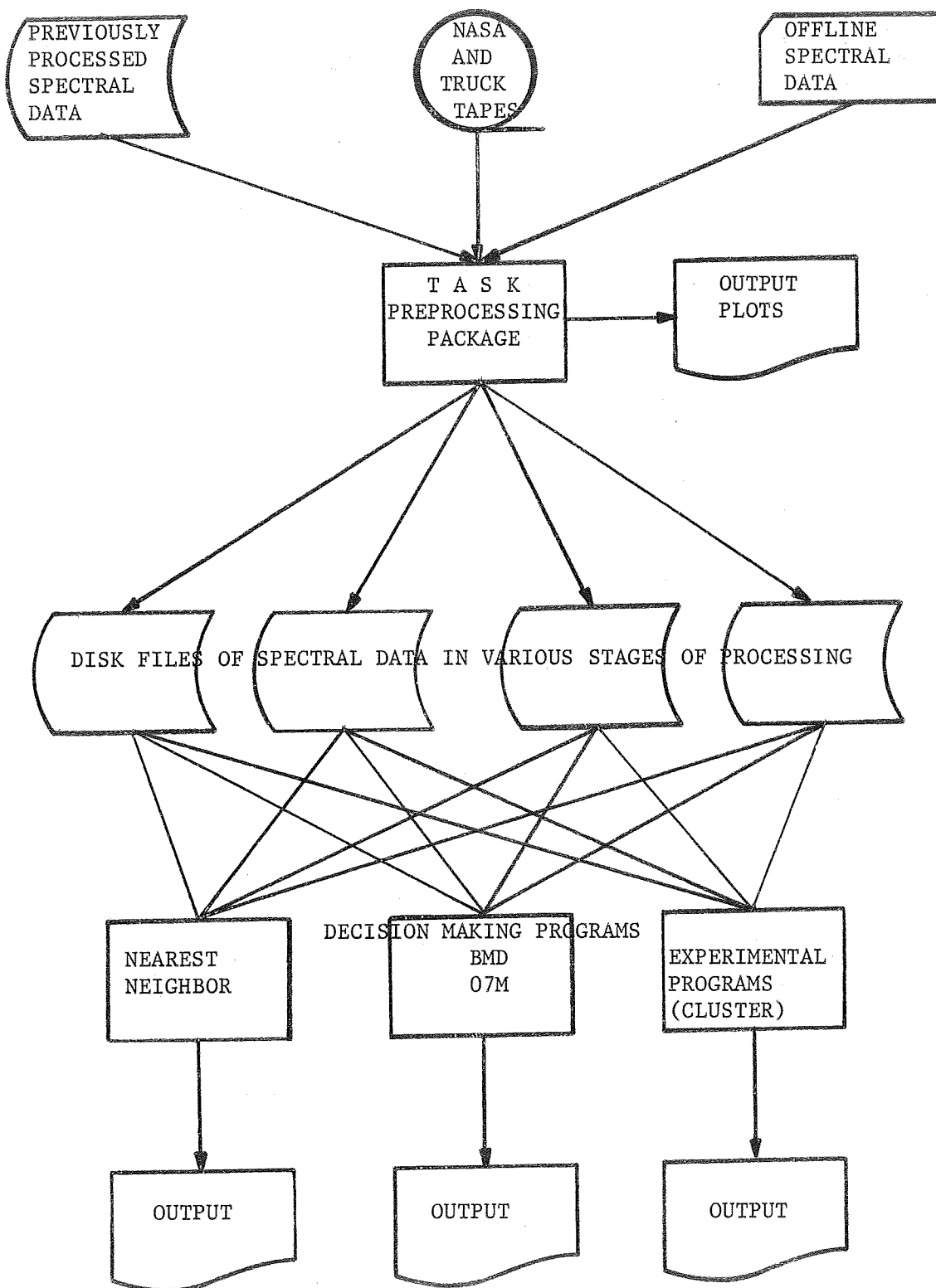
The normal data flow through our system starts with spectral information obtained from either NASA's airborne spectrometer, or our own SG-4 truck-mounted spectrometer. In either case, the information is introduced into the system via magnetic tape.

The data is either transferred directly to disk for later processing, or it is pre-processed on its way to disk. This pre-processing is done by the program TASK and normally consists of the smoothing, ratioing to a standard blackbody spectrum, and normalization of each incoming data spectrum (more detailed information on these pre-processing steps can be found in the section on TASK). TASK will also produce plots of individual or of group average spectra, and perform conversion-to-degrees on the radiometric temperature data accompanying each spectrum.

TASK produces data files of standard format (but perhaps from different pre-processing steps) from either NASA's or our own tapes. These files can then be concatenated as needed (via IBM's job control language JCL) for use by the next stage of processing--decision making. The programs of this stage actually decide (or attempt to decide) which spectra correspond to which rock type.

Several decision making programs have been tried, although only two have been successful enough to warrant our using them for any

length of time. These two, our own NEAREST NEIGHBOR and UCLA's BMD07M, are both similar statistical analysis programs. Both programs require spectra whose rock type are known--training spectra. In these programs, all spectra, both unknown and training, are projected into n-dimensional space (n being the number of different wavelengths sampled). NEAREST NEIGHBOR then decides what each unknown spectra is on the basis of what are a pre-determined number of its nearest training spectra neighbors. BMD07M, on the other hand, makes its decision on the basis of which detail on both of these programs can be found in their respective sections of this report.) The data flow then stops with printed results of these programs.



IV. Data Format

As the system stands now, data is handled by the system in only 3 different formats: NASA, TRUCK, and standard. The system can handle as input data any of the three forms, but outputs data (from the pre-processing step) in standard format only. In all description of the data format, standard IBM 360 terminology (8 bits=1 byte, 4 bytes=1 word) is used unless otherwise stated. In all three formats, each logical (and in this case, physical) record contains spectral data from only one sweep of the spectrometer.

A. NASA Format

With digital data in the NASA/Stanford format, each dataset has only one file, with variable length type records (the actual length of each record is the same throughout). This means that the first two words of each record contain information describing the length of the record, and in the following description of NASA format, these will be ignored (since they are transparent to the FORTRAN programmer).

The first 12 bytes of each record contain a header which identifies the following spectral data as to day, mission, time, line, etc. The first 8 bytes are eight different logical binary integers which

contains:

Byte	
1	-mission
2	-day
3	-line
4	-run
5	-site
6	- (this one is blank)
7	-error (zero if no error in creating data)
8	-ramp (1=high ramp, 2=low ramp)

The next word is a twos-complement integer that contains the time of day in milliseconds from 00:00:00 (MIDNIGHT). Following this are 90 half-word two's complement integers, which are the sample values of the infrared spectrum at 90 different frequencies (counter-points). Finally come 54 halfword two's complement integers containing 18 groups of 3 data values pertaining to the 18 radiometer temperature samples taken during the spectrum sweep.

B. Truck Format

Digital data in TRUCK format is recorded, one field to a dataset (on tape only), in undefined length type records. This means that there are no record information words as in variable length format type, and physical and logical records are identical. Since this format is used only on 7 track tape, all bytes referred to in this section are 6 bit bytes.

The first 12 bytes, again, contain a header. The first halfword contains 3 BCD digits, an unsigned integer, giving the day of the year.

The next word consists of 6 BCD digits giving the time in HH:MM:SS: format. The next 3 halfwords are like the first, unsigned integer in 3 BCD digits, but giving the site, sample, and spectrum count numbers respectively. Following these come a halfword whose 12 bits are all ones, used for synchronization purposes to check for (and enable one to correct) errors made in writing the record. Next come 192 halfwords, each halfword containing a 10 bit two's complement integer, and two synchronization bits. They are ordered as follows:

Tape Channel	First Recorded Byte	Second Byte
1	2**0	2**5
2	2**1	2**6
4	2**2	2**7
8	2**3	2**8
A	2**4	sign bit
B	sync bit-on	sync bit-off

These 192 halfwords contain, alternatingly, the spectral samples over 96 different frequencies and the 96 radiometer samples taken during the spectrum sweep. At the end, there are 6 halfwords, each a 10 bit integer described above, giving the values of 6 different general input multiplex channels for that spectrum sweep. These contain ground meteorological information.

C. Standard Format

The standard format is similar to the NASA format in that there is only one field per dataset, and the records, on tape or disk, are again in variable length type. The first 16 bytes contain header information which varies in composition in accordance with the origin of the spectrum. If the spectrum was made on the NASA airborne spectrometer, the first 12 bytes are identical in format to the first 12 bytes of the

NASA FORMAT. If the spectrum originated from the SG-4, the first 8 bytes are 4 halfwords two's-complement integers containing day, site, sample, and spectrum count respectively. The next 4 bytes are an integer word containing the time in milliseconds of the spectrum. In any case, bytes 13-16 of the header contain two halfword integers specifying spectrum-processing history and spectrum origin.

Their contents are:

Spectrum Processing History			Spectrum Origin
bit		meaning	
1	-	temp conv	1=NASA
2	-	smoothing	2=Truck
3	-	ratioing	
4	-	normalizing	
5	-	banding	

Following the header come between 1 to 96 real words (These are normalized hexadecimal floating point as defined for the IBM 360); these are the spectral sample points. After them come 1 to 102 halfword integers which are the radiometer values. If the spectrum originated with the SG-4, the last 6 halfwords are the multiplex data.

V. Program Descriptions

A. Task

Task is a multipurpose pre-processing program which readies the raw data for the decision making programs. This program, which accepts input from either NASA or TRUCK format, performs, individually or in any combination specified; clipping, normalization, smoothing, ratioing, banding, radiometer data conversion on each individual spectra, as well as plotting averaged groups of spectra. Organizationally, the program is divided into groups of tasks called cycles. Each spectrum inputted goes through a cycle of tasks (normalizing, smoothing, etc.) before finally being outputted. More than one cycle may be batched together in one run. The following are general descriptions of the operation of each task. For actual input parameters, see the listing of TASK in the pre-program listings.

Normalization

In the normalization task, the spectrum mean is set to zero, and its standard deviation to one. This can be expressed mathematically by the formula:

$$y(i) = (x(i) - \bar{x}) / sd$$

where $y(i)$ is the new value of the spectrum at frequency i , $x(i)$ is the old value, $\bar{x}$ the mean of $x(i)$, and sd the standard deviation.

Clipping

The clipping task deleted sample points from the beginning and

end of each spectrum; all following tasks are performed on only the remaining sample points. This is occasionally useful to delete unwanted portions of each spectral sample (such as a water vapor absorption band) from affecting further computations. In all NASA data, the first point of every spectrum is dropped (see 90-day Report).

Smoothing

The smoothing task uses a curve-fitting technique to reduce the noise on the spectrum. The value of a given spectral point is changed to fit into an Nth degree curve composed of itself and the Ith points on either side of it. Both N and I can be specified, but the normal values used are a quartic curve using 4 points on either side. A more detailed write up on this smoothing process can be found in Appendix A.

Ratioing

The ratioing task consists of two parts, a pre-cycle part, and the normal cycle task. The task ratios the input spectrum selected as a "blackbody". This process diminishes spectrum distortion due to atmospheric absorption. The pre-cycle part of this task finds the blackbody spectra and averages them into one spectrum, clipping and smoothing if necessary. This is done before the cycle is started.

Input

The input task reads in spectra in one of four different formats (STANDARD, NASA, TRUCK, and CARD), and changes it to a standard internal format for processing by the other task steps.

Output

The output task writes the processed spectrum on the device specified in the JCL in STANDARD format.

Plot

This task gives a line printer plot of either every spectrum, or the average of these spectra, inside a given time period. The Y scale may be specified, or the task can be left to seek its own.

Banding

The banding task performs one of two operations of a spectrum. In its first mode, it averages a group of spectral points into one point. In its second mode, it averages every n points together, resulting in a spectrum whose length is divided by n. In both cases, the global spectral points parameter is reset to the proper value.

The second mode of this task is useful in reducing the amount of information given to the decision-making programs, while eliminating only a minimum of spectrum characteristic. This enables these latter programs to operate more quickly.

Temperature Conversion

This task converts the 18 radiometer values appended to the NASA form of the spectrum to degrees centigrade. As received from NASA, these radiometer values are in millivolts; there exists a formula (T. Barnett) which relates the value in millivolts to degrees, it is:

$$\overline{B} = \int \Delta\lambda \text{ filter PHI}(\lambda) B(\lambda, T) d\lambda$$

where PHI is a filter function given by the manufacturers of the radio-

meter, and lambda is wavelength. In the program, radiance is converted to temperature (and vice versa) by a table look-up routine.

Before the cycle starts, this task either has to be given the conversion factor, R, or computes it. To compute R, the program finds an average voltage V over a time period whose temperature is known (while viewing a calibrating external balckbody), and substitutes into the above formula for R. Then, in the cycle this task takes each radio-meter voltage, substitutes it into the formula, and finds Btarg, which is then changed into degrees via the table look-up. The temperature in degrees x 10 is then put back into the radiometer part of the spectrum. The relevent filter factors and table generation programs for this task can be found in Appendix B.

Temperature Cutting

This task examines the radiometric part of the spectrum, and deletes the spectrum from the process stream if any one of its temperature sample points is greater than a specified range from the temperature mean of that spectrum. The "distance" can be given either absolutely, or in terms of the standard deviation of temperature points for that spectrum. NOTE: The temperature conversion task must have been applied to the spectrum before this task is used. In other words, the temperature values must be in degrees centigrade.

B. NEAREST NEIGHBOR

The NEAREST NEIGHBOR program is one of the two programs used for identifying rock types in a set of "unknown" spectra. To do this, it first reads in several groups of known, or training spectra, and

does cross-correlation on them, computing group averages, and selecting the Ith most useful points in the spectrum (where I is specified externally), and the weighting factors for these points. The training spectra are then filtered by these weighting factors, and inter-group distances are computed. After this, the program reads in the unknown spectra, one at a time, weights the spectrum, and using the I most important points, finds the distance between the unknown spectrum and all the training spectra in Ith dimensional space. Then NEAREST NEIGHBOR prints out the N nearest neighbors and the training group name of the majority of these neighbors. Finally, the program punches a summary card for each spectrum.

The control cards necessary to use this program can be found in the comment section of the NEAREST NEIGHBOR listing. For a more detailed explanation of the theory behind the program, refer to Stanford RSL Technical Report 68-3.

C. BMD07M

The BMD07M program is a Stepwise Discriminant Analysis program, one of a series of bio-medical statistical analysis programs written by the University of California, Los Angeles, Health Sciences Computing Facility. For all information about this program, refer to Appendix C.

VI. Program Listings

A.) TASK Program

See Computer Print-Out Text VI-1 to VI-15.

B.) NEAREST NEIGHBOR Spectrum Matching Program

See Computer Print-Out Text VI-16 to VI-19.

1 (1012,332,5,6,10,9999,...X),HEATHMAN TASK',MSCLEVEL=1
RTNCL
SIN DR

L PURPOSE SPECTRUM PREPARATION TASK LANGUAGE PROGRAM

EN BY MICHAEL HEATHMAN
DR. LYON
DEPARTMENT OF MINERAL ENGINEERING
STANFORD UNIVERSITY
FEBRUARY 1970

PROGRAM PERFORMS 8 DIFFERENT TASKS, IN ANY ORDER SPECIFIED, ON
SPECTRA GENERATED BY EITHER NASA'S OR DR. LYON'S SPECT-
ROMETER. A MORE DETAILED EXPLANATION OF THIS PROGRAM AND THE SYSTEM
WHICH IT OPERATES CAN BE HAD FROM:

DR. R. J. P. LYON
DEPARTMENT OF MINERAL ENGINEERING
SCHOOL OF EARTH SCIENCES
STANFORD UNIVERSITY
STANFORD, CALIFORNIA 94305

C THE TASKS ARE PERFORMED ON THE SPECTRAL DATA IN GROUPS CALLED CYCLES.
C A CYCLE IS DEFINED AS A SERIES OF TASKS PERFORMED SEQUENTIALLY ON
C EACH SPECTRUM OF A SET OF SEQUENTIALLY ORGANIZED DATA. IN THE INPUT
C STREAM, EACH CYCLE IS DELIMITED BY A 'START' AND A 'FINISH' CARD; AND
C WHILE THERE MAY BE AS MANY CYCLES AS WANTED, THERE CAN ONLY BE 20
C TASKS PER CYCLE.

C *****

TASK DESCRIPTIONS

C IN EACH TASK DESCRIPTION, AN EXAMPLE OF THE ACTUAL CONTROL CARDS
C IS GIVEN. EACH CONTROL CARD IS NUMBERED IN COLUMN 5 AS TO WHICH
C CARD FOR THAT TASK IT IS; IF THE NUMBER IS FOLLOWED BY A LETTER,
C THAT CARD MUST BE USED ONLY IF NECESSARY. IF MORE THAN ONE
C EXAMPLE CARD IS GIVEN FOR A CONTROL CARD, THE SECOND AND FOLLOWING
C EXAMPLE CARDS GIVE FURTHER OPTIONS FOR THAT POSITION ON THE CONTROL
C CARD. PARAMETERS ENCLOSED IN BRACKETS ARE OPTIONAL. FLOATING
C POINT FIELDS ARE INDICATED BY A STRING OF PERIODS OR STARS; ALL
C FLOATING POINT INPUT IS F10.3. SINCE THESE EXAMPLE CARDS ARE
C COMMENT CARDS, THEY START IN COLUMN 8 INSTEAD OF COLUMN 1; SO
C TO GET THE ACTUAL COLUMN WHERE THE PARAMETERS SHOULD GO -- SUBTRACT
C 7 FROM THE COLUMNS THEY ARE IN ON THESE CARDS.

START -- 2 CARDS

C 1 START NDMP IBN 90 54
C 1 DUMP NIBN
C 2 TITLE CARD

C PARAMETERS: DIAGNOSTIC DUMP, IBM TAPE ERROR RECOVERY

CONTROL, SPECTRAL POINTS, RADIOMETRIC DATA POINTS, TITLE

C THIS TASK INITIALIZES THE COMPILATION STAGE OF EACH CYCLE, GLOBAL
C NPTS AND RPTS ARE SET UP

INPUT -- 2,3 CARDS

C 1 INPUT <90 54>
C 2 DOWN SEQL DISK 14 00 00 00000 99 99 99999 < 9000>
C 2 UP NSQL TAPE
C 2 TRCK
C 2 CARD

C 2A RUN TIME FORMAT CARD, USED ONLY IF INPUT FORMAT IS 'CARD'

C PARAMETERS: RAMP, TIME ORGANIZATION OF DATA SET, INPUT FORMAT,
C LOGICAL NUMBER, TIME (HH MM MILSEC HH MM MILSEC), SKIP RECORDS

C THE RUN TIME FORMAT MUST READ: 8 INTEGER*2, 1 TO 100 REAL*4,
C 1 TO 102 INTEGER*2

C *** NOTE: *** AT THIS TIME THE 'TRCK' OPTION IS NOT YET IMPLE-
C MENTED. IN OTHER WORDS, THE SG-4 TAPES CANNOT YET BE READ.

CLIP -- 2 CARDS

C 1 CLIP NPMT <90 54>
C 1 PTIM
C 1 PVAL
C 2 10 10

C PARAMETERS: PRINT TASK IDENTIFIER OPTION, <NPTS AND RPTS>, LEFT
C CLIP POINTS, RIGHT CLIP POINTS

C THIS TASK CLIPS THE CALLED FOR NUMBER OF DATA POINTS OFF THE LEFT
C (BEGINNING) AND RIGHT (END) OF THE SPECTRUM (IN ARRAY SPECT).
C GLOBAL NPTS (<NPTS>) IS REDUCED ACCORDINGLY

SMOOTH -- 2 CARDS

C 1 SMOOTH NPMT <90 54>
C 1 PTIM
C 1 PVAL
C 2 -9

C PARAMETERS: PRINT TASK IDENTIFIER OPTION, <NPTS AND RPTS>,
C SMOOTHING FACTOR

C THIS TASK SMOOTHS THE SPECTRUM, USING QUADRATIC CURVE FITTING
C SMOOTHING (NUMBER OF POINTS IS CONTROLLED BY THE SMOOTHING FACTOR)

RATIO -- 3,5 CARDS

C 1 RATIO NPMT <90 54>
C 1 PTIM

VI-2

```

C 1 PVAL
C 2 AVER DOWN SEQL DISK 10 -9 10 10 00 00 00000 99 99 99999 <90>
C 2 UNXX UP NSCL TAPE
C 2 MXED CARD
C 2A RUN TIME FORMAT CARD, USED ONLY IF INPUT FORMAT IS 'CARD'
C 2 RPT DISK 10
C 1 RPT CARD
C 2A RUN TIME FORMAT CARD, USED ONLY IF OUTPUT FORMAT IS 'CARD'
C
C PARAMETERS: PRINT TASK IDENTIFIER OPTION, <NPTS AND RPTS>, BLACK
C BODY TYPE, RAMP, DATA CHRONOLOGICAL ORGANIZATION, INPUT FORMAT,
C LOGICAL NUMBER, SMOOTHING, LEFT CLIP, RIGHT CLIP, TIME,
C NUMBER OF DATA POINTS IN RAW BLACK BODY, <FORMAT CARD IF
C NEEDED>, AVERAGED BLACK BODY WRITE CONTROL, OUTPUT FORMAT,
C OUTPUT LOGICAL NUMBER, <OUTPUT FORMAT CARD, IF NEEDED>
C
C THE INPUT RUN TIME FORMAT MUST READ: 8 INTEGER*2, 1 TO 100
C REAL*4
C THE OUTPUT RUN TIME FORMAT MUST WRITE: 8 INTEGER*2, 1 TO 100
C REAL*4
C
C THIS TASK READS IN A BLACK BODY, ALREADY AVERAGED, UNAVERAGED ON
C A DATASET WITH OTHER SPECTRA, OR UNAVERAGED ALONE ON A DATASET
C THEN IT RATIOS EACH SPECTRUM IN THE CYCLE TO THIS AVERAGED BLACK
C BODY. THE OBTAINING OF THE BLACK BODY IS DONE BEFORE THE CYCLE
C STARTS. THE AVERAGED BLACK BODY IS ALSO WRITTEN IF DESIRED.
C IF THIS TASK IS CALLED MORE THAN ONCE PER CYCLE, ALL CALLS BUT
C THE FIRST ONE MUST USE ONLY CARD 1.
C
C-----
C NORM -- 1 CARD
C
C 1 NORM NPNT <90 54>
C 1 PTIM
C 1 PVAL
C
C PARAMETERS: PRINT TASK IDENTIFIER CONTROL
C
C THIS TASK NORMALIZES THE DATA; THAT IS SETS THE MEAN OF EACH
C SPECTRUM TO ZERO, AND ITS STANDARD DEVIATION TO ONE. THIS IS DONE
C BY SUBTRACTING THE MEAN FROM EACH DATA POINT AND THEN DIVIDING BY
C THE STANDARD DEVIATION.
C
C-----
C BAND -- 2 CARDS
C
C 1 BAND NPNT <90 54>
C 1 PTIM
C 1 PVAL
C 2 ONCE 10<10>
C 2 REPT
C
C PARAMETERS: PRINT TASK IDENTIFIER OPTION, <NPTS AND RPTS>, BANDING
C TYPE, LEFT EDGE OF BAND AREA (IF BANDING OPTION IS ONCE) OR NPTS
C DIVISOR (IF BANDING OPTION IS REPT), <RIGHT EDGE OF BANDING AREA>
C
C THIS TASK, IN THE 'ONCE' MODE TAKES THE POINTS BETWEEN THOSE
C SPECIFIED BY THE LEFT AND RIGHT BANDING EDGES, AND AVERAGES THEM
C INTO ONE POINT (<NPTS IS CHANGED ACCORDINGLY>). IN THE 'REPT'
C
C
C WDOF, THE NUMBER OF SPECTRAL POINTS IS DIVIDED BY THE NPTS DIVISOR
C BY EVERY GROUP OF N POINTS INTO ONE POINT, WHERE N IS THE NPTS
C DIVISOR. MNPTS IS ALSO CHANGED ACCORDINGLY.
C
C-----
C TEMP -- 3,4 CARDS
C
C 1 TEMP NPNT <90 54>
C 1 PTIM
C 1 PVAL
C 2 MXED DOWN SEQL DISK 10 00 00 00000 99 99 99999 <90 54>
C 2 UNXX UP NSCL TAPE
C 2 AVER CARD
C 2A RUN TIME FORMAT CARD, USED ONLY IF INPUT FORMAT IS 'CARD'
C 1 *****
C
C PARAMETERS: PRINT TASK IDENTIFIER OPTION, <NPTS AND RPTS>,
C CONVERSION FACTOR DATA BASE ORGANIZATION, RAMP, DATA SET
C CHRONOLOGICAL ORGANIZATION, INPUT FORMAT, LOGICAL NUMBER,
C DATA BASE TIME LIMITS, <NPTS AND RPTS>, IF DIFFERENT FROM NPTS
C AND RPTS, <RUN TIME FORMAT CARD, NEEDED IF INPUT FORMAT IS
C CARD>, RUN TIME INTERNAL BLACK BODY TEMPERATURE,
C CALIBRATION INTERNAL BLACK BODY TEMPERATURE (OR IF CONVERSION
C FACTOR DATA BASE ORGANIZATION IS AVER, THEN CONVERSION FACTOR),
C EXTERNAL BLACK BODY TEMPERATURE FOR CALIBRATION.
C
C THE INPUT RUN TIME FORMAT MUST READ: 8 INTEGER*2, 1 TO 100
C REAL*4, 1 TO 102 INTEGER*2
C
C BEFORE THE CYCLE STARTS, THIS TASK FINDS A CONVERSION FACTOR,
C EITHER FROM THE CONTROL CARD (AVER) OR BY COMPUTING IT GIVEN
C A CALIBRATING SET OF SPECTRA OF KNOWN TEMPERATURE (EXTERNAL
C BLACK BODY TEMPERATURE OVER THE TIME SPECIFIED). DURING CYCLE
C RUN EACH RADIOMETER VALUE IS CONVERTED FROM MILLIVOLTS TO
C DEGREES CENTIGRADE. INTERNAL BLACKBODY TEMPERATURE MUST BE
C SUPPLIED FOR BOTH CALIBRATION AND RUNNING.
C
C-----
C TCUT
C
C 1 TCUT NPNT <90 54>
C 1 PTIM
C 1 PVAL
C 2 NCUT PREV .....
C 2 CUT ABSL .....
C
C PARAMETERS: PRINT TASK IDENTIFIER OPTION, <NPTS AND RPTS>,
C SPECTRUM CUT PARAMETER, ACCEPTABLE ERROR TYPE, ACCEPTABLE
C ERROR
C
C THIS TASK REMOVES WHOSE TEMPERATURE VARIANCE IS TOO GREAT FROM
C FURTHER COMPUTATION. ANY SPECTRUM WITH A RADIOMETRIC VALUE
C THAT EXCEEDS A GIVEN LEVEL (EITHER IN PERCENT STANDARD
C DEVIATION OF THE RAW READINGS FOR THAT SPECTRUM --PREV-- OR
C IN AN ABSOLUTE VALUE --ABSL-- IS EITHER CUT (CUT) FROM THE
C COMPUTATION CYCLE, OR MERELY NOTED (NCUT).
C
C-----
C PLOT -- 2 CARDS

```

```

C 1 PLOT      APMT      <90 54>
C 1          PTIM
C 1          PVAL
C 2 EACH 10 00 00 00000 99 99 99999<.....>*****>
C 2 AVER
C
C PARAMETERS: PRINT TASK IDENTIFIER OPTION,<NPIS AND RPIS>,PLOTING
C TYPE, OUTPUT LOGICAL, TIME LIMIT,<MAX SCALE VALUE, MIN SCALE VALUE>
C
C THIS TASK PLOTS EITHER A SERIES OF INDIVIDUAL SPECTRA (EACH) OR THE
C AVERAGE OF A GROUP (AVER). THE Y SCALE CAN BE SET BY MAX AND MIN
C SCALE PARAMETERS. THE AVERAGE FUNCTION OF THIS TASK CAN ONLY BE
C CALLED ONCE, ALL FURTHER CALLS TO IT ARE IGNORED.
C
C -----
C
C OUTPUT -- 2,3 CARDS
C
C 1 OUTPUT      NPMT      <90 54>
C 1            PTIM
C 1            PVAL
C 2 DISK 10
C 2 CARD
C 2A RUN TIME FORMAT CARD, USED ONLY IF OUTPUT FORMAT IS 'CARD'
C
C PARAMETERS: PRINT TASK IDENTIFIER OPTION, <NPIS AND RPIS>, OUTPUT
C FORMAT, OUTPUT LOGICAL, RUN TIME FORMAT CARD IF NECESSARY (CARD)
C
C OUTPUT RUN TIME FORMAT MUST WRITE: 8 INTEGER*2, 3 INTEGER*4,
C 1 TO 100 REAL*4, 1 TO 102 INTEGER*2
C
C THIS TASK WRITES THE PROCESSED DATA ON THE APPROPRIATE DEVICE.
C
C -----
C
C REWIND -- 2 CARDS
C
C 1 REWIND
C 2 4 10 11 12 13
C
C PARAMETERS: NUMBER OF LOGICALS TO REWIND, LOGICALS
C
C THIS TASK REWINDS ALL CALLED FOR LOGICAL AT THE BEGINNING OF EACH
C CYCLE RUN IN WHICH IT WAS CALLED.
C
C -----
C
C FINISH -- 1 CARD
C
C 1 FINISH      <NRUN>
C 1            ABND
C
C PARAMETERS: COMPILE ONLY (NRUN), EXIT WITH ABNORMAL RUN END (NRUN)
C THIS OPTION WRITES ON A MINUS LOGICAL AS ITS FINAL STEP.
C
C THIS TASK SEPARATES ONE CYCLE FROM ANOTHER, AND STARTS EXECUTION
C OF THE COMPILED CYCLE.
C
C ***** DECLARATIONS AND INITIALIZATIONS *****

```

```

C
C IMPORTANT VARIABLES:
C
C RAT      LOGICAL*1 .TRUE. IF RATIO IN CYCLE
C TMP      LOGICAL*1 .TRUE. IF TEMPERATURE CONVERSION IN CYCLE
C PLCT     LOGICAL*1 .TRUE. IF AVERAGE PLCT IN CYCLE
C LABMD    LOGICAL*1 .TRUE. IF ABNORMAL END REQUIRED
C CMPFLG   LOGICAL*1 .TRUE. IF DEBUG DIAGNOSTICS REQUIRED
C TPL      LOGICAL*1 .TRUE. IF TRUCK TAPE TO BE READ
C SENSE     INTEGER*2 ERROR RETURN OF SUBROUTINE REED (TRUCK
C                TAPE READ)
C NPPTS     INTEGER*2 BLACK BODY SPECTRAL POINTS
C RPIS      INTEGER*2 SPECTRUM RADIOMETRIC POINTS
C TRPTS     INTEGER*2 RPTS FOR TEMPERATURE CONV DATA BASE SPECTRA
C TNPTS     INTEGER*2 NPTS FOR TEMPERATURE CONV DATA BASE SPECTRA
C NPAT      INTEGER  NUMBER OF SPECTRAL DATA POINTS
C NZ(150)   INTEGER*2 READ BUFFER FOR INPUT FORMAT = 'TAPE'
C HSPCT(100) INTEGER*2 UNCONVERTED INPUT FROM SUBROUTINE REED
C RAD(102)  INTEGER*2 RADIOMETRIC DATA PART OF SPECTRUM
C RLTT(18)  INTEGER*2 TITLE PART OF SPECTRUM
C HFX(206)  INTEGER*2 UNCONVERTED INPUT FROM SUBROUTINE REED
C ***SEQ(20) INTEGER*2 SERIES OF ARRAYS TO PASS INFO FROM COMPILE
C                SECTION TO EXECUTE SECTION
C ILCG(10)  INTEGER*2 DITTO
C NCLL(20)  INTEGER*2 DITTO
C NCLR(20)  INTEGER*2 DITTO
C PRNTOP(20) INTEGER*2 DITTO
C LEND(20)  INTEGER*2 DITTO
C REND(20)  INTEGER*2 DITTO
C PREAD(40) INTEGER*2 DITTO
C MET(20)   INTEGER  DITTO
C MST(20)   INTEGER  DITTO
C FMT(20,20) INTEGER  DITTO
C TIM       INTEGER  TIME PART OF RLTT
C NTIM      INTEGER  TIME PART OF NZ
C RADTR(14) REAL  TABLE USED IN TEMP CONV
C SPECT(100) REAL  SPECTRUM
C BASPECT(100) REAL BLACK BODY SPECTRUM
C
C LOGICAL*1 RAT,TPL,LABMD, FALSE, CMPFLG,NCAPM,PLCT,TMP,OTMP
C INTEGER*2 SENSE,NPPTS,RPTS,PRSD, BBLCD,BRRCD,TRPTS,TNPTS,NPTP
C *LT,ASPEC,IZ
C INTEGER*2 NZ(150),HSPCT(100),RAD(102),RLTT(18),HFX(206),ILCG(10),TS
C *SEQ(20),SEQSEQ(20),NPSEQ(20),LNGSEQ(20),SMOSEQ(20),NCLL(20),NCLR
C *(20),RPTSSEQ(20),PRNTOP(20),STP(20),LEND(20),REND(20),PREAD(40),CUT
C *SEQ(20)
C INTEGER TIM,STATUS,PRCNT,ST,ET,PRAMP,PAMP, TASK,PAPAM,ARM,E
C *RSDMP,RTYPE,BUTYPE,RCUT,BUTYPE,RSQC,SEQ,ICTYPE,BSM,PEH,IMTYPE,IMR
C *AMP,ISQC,ITYPE,ISM,TEM,ITCONT,J/O,TLICG,BILCG,RCLOG,PLCG,RRSMC
C INTEGER IST(3),IST(3),VFT(20),MST(20),FMT(20,20),RPFMT(20),BIFMT(2
C *0),TIFMT(20),RPFMT(20),ICT(3),IPST(3)
C INTEGER STAP,STAP,OUTPUT,OUTPUT,INPUT,INPUT,PATIC/RATIC,S
C 1*OOTHS/SMOHS,NCRH/NCWH,CLID/CLID,FINISH/FINISH,DUMP/DUMP
C 27,APND/ABND,DISK/DISK,TEMP/TEMP,PLD/CLD,TRCK/TRCK
C 37,CARD/CARD,SEQ/SEQ,PRNT/PRNT,UNMX/UNMX,WXED/WXED
C 47,AVER/AVER,NCLT/CUT,PVAL/PVAL,PTIV/PTIV,NPNT/NPNT
C 57,NIN/NIN,RPOT/RCUT,TAPE/TAPE,UP/UP,DOWN/DOWN,NS
C 67,INQL/PLTT,PLCT/PLCT,BAND/BAND,REWIND/REWIND,PEPT/PEPT,
C 77,INFE/TCUT,TCUT/TCUT,DOEV/DOEV,ARSL/ARSL,FACH/FACH,NN
C 87,INXX/CCCC/CCCC/CCCC/CCCC,RRRR/RRRR,TTTT/TTTT,RRD

```

```

00/*****
REAL RACT3(141),TCTPCT(20),SPFCT(100),RSPCT(100),DSPCT(100),TLT(2
*0),MEAN(100),SD(100),PF(100),XMIN(20),XMAX(20),ZERO(100)/100*0.0/
REAL AA (71)/C.715934E-03,0.722103E-03,C.728405E-03,0.734739E-03,
*0.741104E-03,0.747502E-03,0.753933E-03,C.760396E-03,C.766891E-03,
*0.773418E-03,0.779979E-03,0.786572E-03,0.793197E-03,0.799855E-03,
*0.806545E-03,0.813269E-03,0.820023E-03,0.826810E-03,0.833630E-03,
*0.840484E-03,0.847369E-03,0.854287E-03,C.861237E-03,C.868220E-03,
*0.875236E-03,C.882284E-03,0.889364E-03,0.896478E-03,0.903624E-03,
*0.910803E-03,C.918013E-03,0.925256E-03,C.932533E-03,C.939841E-03,
*0.947182E-03,0.954556E-03,0.961963E-03,0.969401E-03,0.976873E-03,
*0.984377E-03,C.991913E-03,0.999483E-03,C.100708E-02,0.101472E-02,
*0.102239E-02,0.103009E-02,0.103782E-02,C.104558E-02,C.105338E-02,
*0.106121E-02,C.106907E-02,0.107697E-02,0.108489E-02,0.109285E-02,
*0.110084E-02,C.110887E-02,0.111693E-02,C.112502E-02,0.113314E-02,
*0.114129E-02,0.114948E-02,0.115770E-02,C.116555E-02,C.117423E-02,
*0.118255E-02,C.119090E-02,0.119928E-02,C.120769E-02,0.121614E-02,
*0.122462E-02,0.123313E-02/
REAL CC(701)/0.126167E-02,0.125025E-02,0.125886E-02,0.126749E-02,
*0.127617E-02,C.128487E-02,0.129361E-02,0.130238E-02,0.131118E-02,
*0.132001E-02,0.132887E-02,0.133777E-02,0.134670E-02,C.135566E-02,
*0.136466E-02,0.137368E-02,0.138274E-02,0.139183E-02,C.140095E-02,
*0.141010E-02,0.141929E-02,0.142850E-02,0.143775E-02,C.144704E-02,
*0.145635E-02,0.146569E-02,0.147507E-02,0.148448E-02,0.149392E-02,
*0.150339E-02,C.151289E-02,0.152243E-02,0.153199E-02,0.154159E-02,
*0.155122E-02,0.156088E-02,0.157058E-02,C.158030E-02,C.159006E-02,
*0.159985E-02,0.160967E-02,0.161952E-02,0.162940E-02,0.163931E-02,
*0.164926E-02,0.165924E-02,0.166924E-02,0.167928E-02,0.168935E-02,
*0.169946E-02,0.170959E-02,0.171975E-02,0.172995E-02,C.174017E-02,
*0.175043E-02,0.176072E-02,0.177104E-02,0.178139E-02,0.179177E-02,
*0.180218E-02,0.181263E-02,0.182310E-02,C.183361E-02,C.184414E-02,
*0.185471E-02,0.186531E-02,0.187594E-02,0.188660E-02,0.189729E-02,
*0.190801E-02/
REAL*8 V,SV,B
EQUIVALENCE (AA(1),RADTB(1)),(CC(1),RADTB(72))
EQUIVALENCE (TIM,RTLT(51)),(NTIM,MZ(5))
COMMON /PARAM/ LCNTRL,RTLT,HSPCT,RAD,HEX,STATUS,ECBW,CSW,SENSE
*****
*
*          PROGRAM          START
*
*****
***** COMPILATION SECTION *****
*****
CYCLE COMPILATION PARAMETER INITIALIZATION
10 ITSK=0
DMPFLG=.FALSE.
NCPARM=.FALSE.
TPL=.FALSE.
RAT=.FALSE.
PLOT=.FALSE.
QTMP=.FALSE.
TMP=.FALSE.
NEND=0
ISUB=0
MBND=0
NIFMT=0
NILCG=0
NISTP=0
NISFO=0

NICL=0
NIT=0
NRPTS=0
NISMC=0
NPOP=0
NSPCT=0
NIPRD=-1
NIPCT=0
NRND=0
NICUT=0
IF(J.EQ.1) GO TO 100

***** CYCLE COMPILATION START *****

READ AND ANALYZE OPERATOR
20 READ(5,1001,END=939) TASK,PARAM,FRDMP,AIRM,NPTS,RPTS
1001 FORMAT(A4,6X,3(A4,1X),2I3)
21 IF(TASK.EQ.START) GO TO 100
IF(TASK.EQ.REWIND) GO TO 109

INCREMENT TASK COUNTER AND GET GLOBAL NPTS AND RPTS CHANGE (IF ANY)
ITSK=ITSK+1
IF(ITSK.GT.20) GO TO 10017
IF(NPTS.NE.C) MNPTS=NPTS
IF(RPTS.NE.C) MRPTS=RPTS
NFTSEQ(ITSK)=MNPTS
IF(TASK.EQ.TEMP) GO TO 110
IF(TASK.EQ.OUTPUT) GO TO 101
IF(TASK.EQ.INPUT) GO TO 102
IF(TASK.EQ.RATIO) GO TO 103
IF(TASK.EQ.SMOOTH) GO TO 104
IF(TASK.EQ.NORM) GO TO 105
IF(TASK.EQ.CLIP1) GO TO 106
IF(TASK.EQ.BAND) GO TO 108
IF(TASK.EQ.CUT) GO TO 341
IF(TASK.EQ.PLOT) GO TO 780
IF (TASK.EQ.FINISH) GO TO 107
GO TO 10000

TASK=START
100 READ(5,1303) TLT
1303 FORMAT(20A4)
WRITE(6,1100) PARAM,FRDMP,AIRM,NPTS,RPTS,TLT
1100 FORMAT('SPECTRUM PREPARATION CYCLE START. PARAMETERS ARE: ',3(A4,
A1X),14,' = NPTS.',14,' = RPTS.'/) ' ',20A4)
IRM=1
LCNTRL=1

SET DIAGNOSTIC MESSAGE DUMP FLAG FOR MAIN AND SUBROUTINE PEED
IF(FRDMP.NE.DUMP) GO TO 157
DMPFLG=.TRUE.
LCNTRL=3
157 IF(AIRM.EQ.AIRM) IRM=0

INITIALIZE GLOBAL NPTS AND RPTS

```

VRPTS=PPPTS
MNPTS=NPPTS
GO TO 20

VI-5

```
C
C TASK=OUTPUT
C
101 READ(5,1402) IOTYPE,K
102 FORMAT(44,1X,12)
WRITE(6,1104) ITSK,PARAM,IOTYPE,K,MNPTS,MRPTS
1104 FORMAT(/' ',T5,'TASK',I3,' -- OUTPUT',T30,A4,1X,A4,13,I4,' = NPPTS.
1',I4,' = RPTS.')
```

```
1105 NLOG=NLOG+1
LOGSEQ(NLOG)=K
PPPTS=MRPTS+1
WRITE(6,1105) ITSK,PARAM,MNPTS,MRPTS
IF(IOTYPE.NE.DISK) GO TO 111
TSKSEQ(ITSK)=1
GO TO 90
111 IF(IOTYPE.NE.CARD) GO TO 10001
TSKSEQ(ITSK)=4
NIFMT=NIFMT+1
READ(5,1303) (FMT(NIFMT,K),K=1,20)
WRITE(6,1103) (FMT(NIFMT,K),K=1,20)
GO TO 90
C
C TASK=INPUT
C
102 READ(5,1002) PAMP,SEC,IOTYPE,K,IST,IET,IPREAD
1002 FORMAT(3(A4,1X),I2,1X,2(2I3,I6),I10)
WRITE(6,1102) ITSK,RAMP,SEC,IOTYPE,K,IST,IET,IPREAD,MNP
ITS,MRPTS
1102 FORMAT(/' ',T5,'TASK',I3,' -- INPUT',T30,3(A4,1X),I2,1X,2(2I3,I6)
A,I10,' = PREAD',I4,' = NPPTS. ',I3,' = RPTS.')
```

```
IF(IOTYPE.NE.DISK) GO TO 151
J=5
GO TO 152
151 IF(IOTYPE.NE.TAPE) GO TO 154
J=2
GO TO 152
154 IF(IOTYPE.NE.TRCK) GO TO 155
TPL=.TRUE.
J=7
GO TO 152
155 IF(IOTYPE.NE.CARD) GO TO 10001
J=8
NIFMT=NIFMT+1
READ(5,1303) (FMT(NIFMT,K),K=1,20)
WRITE(6,1103) (FMT(NIFMT,K),K=1,20)
1103 FORMAT(' ',T10,'FORMAT: ',20A4)
152 NISEQ=NISEQ+1
SEQSEQ(NISEQ)=1
IF(SEQ.EQ.NSQL) SEQSEQ(NISEQ)=2
IF((SEQ.EQ.NSQL).AND.(J.LE.5)) REWIND K
IF(IPREAD.EQ.0) GO TO 142
NIPRD=NIPRD+2
PREAD(NIPRD)=LOGSEQ(NLOG)
PREAD(NIPRD+1)=IPREAD
142 TSKSEQ(ITSK)=J
IF(RAMP.NE.UP) GO TO 191
JJ=0
GO TO 116
191 IF(RAMP.NE.DOWN) GO TO 192
JJ=1
GO TO 116
192 JJ=3
116 NLOG=NLOG+1
LOGSEQ(NLOG)=K
NISTP=NISTP+1
STP(NISTP)=JJ
MRPTS=MRPTS+1
RPTSSEQ(NRPTS)=MRPTS
NIT=NIT+1
NIST(NIT)=IST(1)*3600000+IST(2)*60000+IST(3)
NET(NIT)=IET(1)*3600000+IET(2)*60000+IET(3)
OTMP=.FALSE.
GO TO 20
```

```
C
C TASK=RATIO
C
103 TSKSEQ(ITSK)=9
IF(PAT) GO TO 115
READ(5,1004) BRTYPE,BRRAMP,RSEQ,BITYPE,BLOG,BBSMO,BBLCP,BBRCP,IST
I,IET,MRPTS
1004 FORMAT(4(A4,1X),I2,3I3,1X,2(2I3,I6),I4)
IF(BNPTS.EQ.0) BNPTS=MNPTS
WRITE(6,1105) ITSK,PARAM,MNPTS,BRTYPE,BRRAMP,RSEQ,BITYPE,BLOG,BBS
LMO,BBLCP,BBRCP,IST,IET,MRPTS
1105 FORMAT(/' ',T5,'TASK',I3,' -- RATIO',T30,A4,I4,' = NPPTS. '/' ',T10,
A4(A4,1X),I2,3I3,1X,2(2I3,I6),I4,' = BNPTS.')
```

```
IF(BITYPE.NE.CARD) GO TO 1115
READ(5,1303) RIFMT
WRITE(6,1103) RIFMT
1115 READ(5,1014) BOUT,BCTYPE,BOLOG
1014 FORMAT(2(A4,1X),I2)
WRITE(6,1015) BOUT,BCTYPE,BOLOG
1015 FORMAT(' ',T10,2(A4,1X),I2)
IF(BCTYPE.NE.CARD) GO TO 1116
READ(5,1303) RCFMT
WRITE(6,1103) RCFMT
1116 BSM=IST(1)*3600000+IST(2)*60000+IST(3)
BSM=IET(1)*3600000+IET(2)*60000+IET(3)
RAT=.TRUE.
GO TO 90
115 WRITE(6,1105) ITSK,PARAM,MNPTS
GO TO 90
C
C TASK=SMOOTH
C
104 READ(5,1005) K
1005 FORMAT(I3)
WRITE(6,1107) ITSK,PARAM,K,MNPTS
1107 FORMAT(/' ',T5,'TASK',I3,' -- SMOOTH',T30,A4,I3,' = SMOOTHING ',I3
A,' = NPPTS.')
```

```
1108 NISM=NISM+1
SMSEQ(NISM)=K
TSKSEQ(ITSK)=10
GO TO 90
C
C TASK=NOOP
C
105 WRITE(6,1108) ITSK,PARAM,MNPTS
```

```

1108 FCPVAT(/' ,T5,'TASK',I3,' -- N0PM',T30,A4,I3,' = NPTS.')
      TSKSEQ(ITSK)=11
      GO TO 90
C
C   TASK=CLIP
C
106 READ(5,1007) J,K
      IF(NPTS.EQ.0) MNPTS=MNPTS-J-K
      WRITE(6,1109) ITSK,PARAM,J,K,MNPTS
1007 FCPVAT(I2I3)
1109 FCPVAT(/' ,T5,'TASK',I3,' -- CLIP',T30,A4,I4,' = LEFT CLIP
C
C   J AND K ARE SPECTRAL COUNTED POINT BOUNDS WHICH AREN'T AFFECTED BY
C   PREVIOUS ROUNDOFFS. LEND AND REND CONTAIN ACTUAL SPECT APPAY SUBSCRIPT
C   WHICH ARE AFFECTED. ISUP IS THE CORRECTION FACTOR.
C
      I,I3,' = RIGHT CLIP ',I3,' = NPTS.')
      TSKSEQ(ITSK)=12
      NICL=NICL+1
      MCCL(NICL)=J
      KCL(NICL)=K
      ISUP=0
      GO TO 90
C
C   TASK=RAND
C
108 NEND=NEND+1
      REAC(5,1166) IOTYPE,J,K
1166 FCPVAT(A4,I3,2I3)
      IF(IOTYPE.NE.ONCE) GO TO 1453
      WRITE(6,1441) ITSK,PARAM,IOTYPE,J,K,MNPTS
1441 FCPVAT(/' ,T5,'TASK',I3,' -- BAND',T30,A4/' ,T10,A4,I4,' = LEFT
      IBOUND INCLUSIVE ',I3,' = RIGHT BOUND INCLUSIVE ',I3,' = NPTS.')
      MNPTS=MNPTS-K+J
      TSKSEQ(ITSK)=6
      LEND(NEND)=J-ISUP
      REND(NEND)=K-ISUP
      ISUP=ISUP+K-J
      GO TO 90
1453 IF(ICTYPE.NE.REPT) GO TO 10007
      WRITE(6,1457) ITSK,PARAM,IOTYPE,J,MNPTS
1457 FCPVAT(/' ,T5,'TASK',I3,' -- BAND',T30,A4/' ,T10,A4,I4,' = DIVIS
      IOR ',I3,' = NPTS.')
      K=MOD(MNPTS,J)
      MNPTS=MNPTS/J
      IF(K.NE.0) MNPTS=MNPTS+1
      ISUP=0
      TSKSEQ(ITSK)=14
      LEND(NEND)=J
      REND(NEND)=K
      GO TO 90
C
C   TASK=TEMP
C
110 IF(OTMP) GO TO 10011
      IF(TMP) GO TO 1350
      READ(5,1351) TMTYPE,TMRAMP,TSEQ,TITYPE,TILOG,IST,IET,NPTS,RPTS,TRR
      AEF,TIREF,TRR
1351 FCPVAT(4(A4,I3),I2,I3,2(I2I3,I6),I4,I3/3F10.2)
      TRPTS=RPTS
      IF(RPTS.EQ.0) TRPTS=MRPTS

      TNPTS=NPTS
      IF(NPTS.EQ.0) TNPTS=MNPTS
      WRITE(6,1352) ITSK,PARAM,TMTYPE,TMRAMP,TSEQ,TITYPE,TILOG,IST,IET,
      A,NPTS,TRPTS,TRREF,TIREF,TRR
1352 FCPVAT(/' ,T5,'TASK',I3,' -- TEMP CONV',T30,5(A4,I3),I2,I3,2(I2I3,
      A,I6),I4,' = TNPTS.',I4,' = TRPTS.',I4,' = T10,F10.3,' = RUN REF TEMP ',
      A,F10.3,' = INIT REF TEMP OR RADIANCE FACTOR ',F10.3,' = EXT RB TEM
      C')
      IF(TITYPE.NE.CARD) GO TO 1353
      READ(5,1303) TIFMT
      WRITE(6,1103) TIFMT
1353 TSM=IST(1)*3600000+IST(2)*60000+IST(3)
      TEM=IET(1)*3600000+IET(2)*60000+IET(3)
      TMP=.TRUE.
      IF(TIREF.NE.C.0) RPRM=TIREF
1354 TSKSEQ(ITSK)=15
      NRPTS=NRPTS+1
      RPTSEQ(NRPTS)=MRPTS
      CTMP=.TRUE.
      MRPTS=(MRPTS+2)/3
      GO TO 90
1350 WRITE(6,1352) ITSK,PARAM
      GO TO 1354
C
C   TASK=TCUT
C
341 READ(5,1610) PAMP,ICTYPE,PERCENT
1610 FCPVAT(2(A4,I3),F10.3)
      WRITE(6,1613) ITSK,PARAM,PAMP,ICTYPE,PERCENT
1613 FCPVAT(/' ,T5,'TASK',I3,' -- TEMP CUT',T30,3(A4,I3),F10.3,
      *' = PERCENT')
      TSKSEQ(ITSK)=16
      NRPTS=NRPTS+1
      RPTSEQ(NRPTS)=MRPTS
      NIPCT=NIPCT+1
      IF(ICTYPE.NE.PDEV) GO TO 1611
      TCTPCT(NIPCT)=PERCENT
      GO TO 1612
1611 IF(ICTYPE.NE.ABSL) GO TO 10013
      TCTPCT(NIPCT)=PERCENT
1612 NICUT=NICUT+1
      IF(RAMP.NE.NCUT) GO TO 1614
      CUTSEQ(NICUT)=0
      GO TO 90
1614 IF(RAMP.NE.CUT) GO TO 10016
      CUTSEQ(NICUT)=1
      GO TO 90
C
C   TASK=PLOT
C
780 READ(5,1904) IOTYPE,K,IST,IET,P,PR
1904 FCPVAT(A4,I3,I2,I3,2(I2I3,I6),F10.3)
      WRITE(6,1905) ITSK,PARAM,IOTYPE,K,IST,IET,P,PR,MNPTS
1905 FCPVAT(/' ,T5,'TASK',I3,' -- PLOT',T30,2(A4,I3),I2,I3,2(I2I3,I6),F
      *I0.3,' = XVAL ',F10.3,' = YVAL ',I3,' = NPTS.')
      J=IST(1)*3600000+IST(2)*60000+IST(3)
      L=IET(1)*3600000+IET(2)*60000+IET(3)
      IF(ICTYPE.NE.AVER) GO TO 781
      IF(PLOT) GO TO 10014
      NPTPLT=MNPTS
      NOST=J

```

VI-6

VI-7

```

MPFT=L
DO 5050 I=1,3
  IPFT(I)=IT(I)
5050 IPST(I)=IST(I)
  PXMIN=0
  PXMAX=PR
  NLG=K
  TSKSEQ(ITSK)=3
  PLDT=.TRUE.
  GO TO 90
781 IF(ITYPE.NE.FACH) GO TO 10015
  NIT=NIT+1
  NST(NIT)=J
  NET(NIT)=L
  NILOG=NILOG+1
  LCGSEQ(NILOG)=K
  NBND=NBND+1
  XMIN(NBND)=R
  XMAX(NBND)=RR
  TSKSEQ(ITSK)=17
  GO TO 20
C
C   TASK=REWIND
C
109 READ(5,1167) J,(ILOG(I),I=1,J)
1167 FORMAT(11I3)
  WRITE(6,1168) (ILOG(I),I=1,J)
1168 FORMAT(/,1,15,'TASK -- REWIND'/,1,T10,'LOGICALS ARE: ',10I3)
  DO 1169 I=1,J
    JK=ILOG(I)
1169 REWIND JK
  GO TO 20
C
C   TASK=FINISH
C
107 TSKSEQ(ITSK)=13
  IF(PARAM.EQ.ABND) LABND=.TRUE.
  IF(PARAM.NE.NRUN) GO TO 141
  J=0
  GO TO 10
C
C   FINISH COMPILATION OF TASK WITH PRINT PARAMETER ANALYSIS
C
9C J=0
  IF(PARAM.EQ.PVAL) J=3
  IF(PAPAM.EQ.PTIM) J=2
  IF(PAPAM.EQ.NPNT) J=1
  IF(J.EQ.0) GO TO 10006
  NPOP=NPOP+1
  PRNTCP(NPOP)=J
  GO TO 20
C
C ***** END COMPILATION SECTION *****
C
C   PRE READ INPUT DATA SET
C
141 IF(NIPRD.EQ.-1) GO TO 143
  DO 145 I=1,NIPRD,2
    J=PREAD(I)
    K=PREAD(I+1)
    DO 146 JK=1,K
146 READ(J)
145 CONTINUE
143 IF(.NOT.TPL) GO TO 135
C
C   OPEN TAPEIN IF TRUCK TAPE IS TO BE READ
C
  CALL STREAD(IRM,STATUS)
  IF(STATUS.NE.1) GO TO 11000
C
C   GET AVERAGE BLACK BODY
C
135 IF(.NOT.RAT) GO TO 122
  BBCCNT=0
  DO 200 I=1,BNPTS
    200 BBSPT(I)=0.0
C
C   ANALYZE BB INPUT TYPE
C
  IF(ITYPE.NE.DISK) GO TO 201
  J=1
  GO TO 202
201 IF(ITYPE.NE.TAPE) GO TO 203
  K=BNPTS+6
  J=2
  GO TO 202
203 IF(ITYPE.NE.CARD) GO TO 10002
  J=3
C
C   GET CLIPPED BNPTS (JJ), AND ANALYZE BB TYPE
C
202 JK=BNPTS-BBLCP
  JJ=JK-BBRCP
  BPSQ=1
  IF(RBTYPE.NE.MXED) GO TO 204
  IF(RSEQ.NE.SEQ1) BBSQ=2
  GO TO 205
204 IF(RBTYPE.NE.UNMX) GO TO 206
  RSM=-2147483647
  BEM=-2147483647
  GO TO 205
206 IF(RBTYPE.NE.AVER) GO TO 277
  RSM=-2147483647
  REM=-2147483647
  GO TO 205
277 IF(RBTYPE.NE.CLD) GO TO 10003
  GO TO 278
C
C   ANALYZE PB PARAM
C
205 IF(BBAMP.NE.UP) GO TO 207
  IRAMP=0
  GO TO 208
207 IF(BBAMP.NE.DOWN) GO TO 209
  IRAMP=1
  GO TO 208
208 IRAMP=4
C
C   START PR READ
C
209 GO TO (210,212,211),J

```

```

210 READ(RILCG,END=223) RTLT,(SPECT(I),I=1,BNPTS)
  IF(TIM,LT,BSM) GO TO 210
  IF(TIM,GT,BEW) GO TO (217,210),BBSQ
  GO TO 213
211 READ(RILCG,BIFMT,END=223) RTLT,(SPECT(I),I=1,BNPTS)
  IF(TIM,LT,BSM) GO TO 211
  IF(TIM,GT,BEW) GO TO (217,211),BBSQ
  GO TO 213
212 READ(BILCG,END=223) (MZ(I),I=1,K)
  IF(NTIM,LT,BSM) GO TO 212
  IF(NTIM,GT,BEW) GO TO (217,212),BBSQ
  DO 214 I=1,BNPTS
214 SPECT(I)=MZ(I+6)
  DO 215 I=1,6
215 RTLT(I)=MZ(I)
  RTLT(R)=1
213 IF(BBTYPE,EQ,AVER) GO TO 218
  IF(RTLT(R),EQ,2) GO TO 216
  IF(RTLT(4),EQ,IPAMP) GO TO 208
C
C   IF SPECT IS ACCEPTABLE, CLIP + SMOOTH
C
216 BBCCNT=BBCCNT+1
  IF(BRLCP,EQ,0) GO TO 227
  DO 228 I=1,JK
228 SPECT(I)=SPECT(I)+BRLCP
227 CALL SM(SPECT,OSPCT,JJ,STATUS,BBSMO)
  IF(STATUS,NE,0) GO TO 219
  DO 220 I=1,JJ
220 BBSPECT(I)=OSPCT(I)+BBSPECT(I)
  GO TO 208
219 DO 222 I=1,JJ
222 BBSPECT(I)=SPECT(I)+BBSPECT(I)
  GO TO 208
C
C   IF ECF IS HIT, GET LAST BB TIME
C
223 CALL XLATE (TIM,IST)
  WRITE(6,2000) IST
2000 FORMAT(/' END OF BLACKBODY DATASET, LAST TIME -- ',2I3,16)
217 IF((J,NE,3).AND.(BBTYPE,EQ,MXED)) REWIND BILOG
  IF(BBCONT,EQ,C) GO TO 10005
  BNPTS=JJ
C
C   GET AVERAGE OF BB SPECTRA
C
  DO 224 I=1,BNPTS
224 BBSPECT(I)=BBSPECT(I)/BBCCNT
  BBRCP=BNPTS+BBLCF
  BBLCF=BBLCF+1
278 WRITE(6,2223) BNPTS,BBLCF,BBRCP,BBCCNT
2223 FORMAT(/' BLACK BODY HAS ',I3,' POINTS, BETWEEN COUNTER POINTS ',I
  A3,' AND ',I3,' INCLUSIVE. NUMBER OF BB SPECTRA PROCESSED IS ',I5)
  GO TO 271
C
C   IF BBTYPE IS ALREADY AVERAGED TRANSFER DIRECTLY TO BBSPECT
C
218 DO 272 I=1,BNPTS
272 BBSPECT(I)=SPECT(I)
271 IF(BOU,NE,BBOT) GO TO 122
C

C   WRITE BB AVERAGE
C
  J=0
  IF(BBSMO,EQ,-9) J=2
  RTLT(7)=J
  IF(BOTYPE,NE,CARD) GO TO 225
  WRITE(BOLOG,ROFMT) RTLT,(BBSPECT(I),I=1,BNPTS)
  GO TO 122
225 IF(BOTYPE,NE,DISK) GO TO 10004
  WRITE(BOLCG) RTLT,(BBSPECT(I),I=1,BNPTS)
C
C   GET AVERAGE RADIANCE REFERENCE VALUES
C
122 IF(.NOT,TMP) GO TO 136
  BBSQ=1
  IF(TMTYPE,NE,AVER) GO TO 1355
  KK=1
  TT=TRREF
C
C   RAD TABLE INTERPOLATION ROUTINE, KK CONTROLS RE-ENTRY POINT
C   TT INPUT PARAM, RR OUTPUT PARAM
C
1361 R=TT*2.+1.
  J=P
  SQTNSP=J
  RR=RANTR(JJ)
  RR=(R-SQTNSP)*(RADT(J+1)-RP)+RR
  GO TO (1356,1357,1358,1359),KK
1356 RREF=RR
  GO TO 136
1355 IF(TMTYPE,NE,MXF) GO TO 1360
  TSM=-2147483647
  TEM=2147483647
1363 KK=2
  TT=TRREF
  GO TO 1361
1357 RREF=RR
  KK=3
  TT=TRREF
  GO TO 1361
1358 RREF=RR
  KK=4
  TT=TRREF
  GO TO 1361
1359 RREF=RR
  GO TO 1362
1360 IF(TMTYPE,NE,MXF) GO TO 10008
  IF(TSEQ,NE,SECL) BBSQ=2
  GO TO 1363
1362 IF(TMTYPE,NE,DISK) GO TO 1364
  J=1
  GO TO 1365
1364 IF(TMTYPE,NE,TAPE) GO TO 1366
  K=96+TRPTS
  J=2
  GO TO 1365
1366 IF(TMTYPE,NE,CARD) GO TO 10009
  J=3
1365 IF(TMRAMP,NE,UP) GO TO 1367
  TRAMP=0
  GO TO 1368

```

VI-9

```

1367 IF (TMPAMP.NE.D0KN) GO TO 1369
  TRAMP=1
  GC TO 1368
1368 TRAMP=4
  TTCNT=0
  V=0.0
  SV=0.0
  NRAD=0
1371 GO TO (1372,1373,1374),J
1372 READ(TILCG,END=1375) PTLT,(SPECT(I),I=1,TNPTS),(RAD(I),I=1,TNPTS)
  IF(TIM.LT.TSM) GO TO 1372
  IF(TIM.GT.TEM) GO TO (1377,1372),RRSQ
  GO TO 1378
1373 READ(TILCG,TIFMT,END=1375) PTLT,(SPECT(I),I=1,TNPTS),(RAD(I),I=1,T
  ARPTS)
  IF(TIM.LT.TSM) GO TO 1373
  IF(TIM.GT.TEM) GO TO (1377,1373),RRSQ
  GO TO 1378
1374 READ(TILCG,END=1375) (MZ(I),I=1,KK)
  IF(NTIM.LT.TSM) GO TO 1374
  IF(NTIM.GT.TEM) GO TO (1377,1374),RRSQ
  DO 1380 I=1,TRPTS
1380 RAD(I)=MZ(96+I)
  DO 1381 I=1,6
1381 RTLT(I)=MZ(I)
  RTLT(8)=1
C
C ANALYSE SPECTRA
C
1378 IF(RTLT(8).EQ.2) GO TO 10012
  IF(PTLT(4).EQ.TRAMP) GC TO 1371
  TTCNT=TTCNT+1
  DO 1383 I=1,TRPTS,3
  NRAD=NRAD+1
  B=RAD(I)
  V=V+B
1383 SV=SV+R*R
  GO TO 1371
1375 CALL XLATE (TIM,IST)
  WRITE(6,1384) IST
1384 FORMAT(/' END OF RAD REF DATASET, LAST TIME -- ',2I3,16)
1377 IF((J.NE.3).AND.(TMTYPE.EQ.MXED)) REWIND TILCG
  IF(TTCNT.EQ.0) GC TO 10010
C
C PRODESS AVERAGE RADIANCE LEVEL FOUND.
C
  R=(NRAD*SV-V*V)/(NRAD*NRAD-NRAD)
  RR=V/NRAD
  R=SQRT(R)
  RPRM=(RR/1000.-.970)/(BIPEF-BBB)
  WRITE(6,1385) RPRM,TTCNT,NRAD,RR,R,TREF,TBB
1385 FORMAT(/' REFERENCE RADIANCE IS ',F15.7,5X,I6,' SPECTRA WERE READ'
  A,I7,' RADIOMETER VALUES WERE USED'/' MEAN = ',F15.7,' STD. DEV. =
  B,F15.7/' INTERNAL REF TEMP = ',F4.1,' BLACK BODY TEMP = ',F4.1)
136 WRITE(6,1600)
C
C ***** RUN SECTION *****
C
C
C INITIALIZE RUN TIME PARAMETERS
C
  NSPROC=0
  IORECD=0
  IORECC=0
  LRCRD=0
  IF(.NOT.PLOT) GC TO 127
  DO 134 I=1,100
  MEAN(I)=0.0
134 SD(I)=0.0
C
C INITIALIZE CYCLE RUN PARAM
C
127 ITSK=0
  NIPCT=1
  NEND=1
  NRPTS=1
  NIFMT=1
  NISTP=1
  NPPD=1
  NILOG=1
  NICL=1
  NISEQ=1
  NIT=1
  NISMD=1
  NICUT=1
  NBNM=1
C
C FIND NEW TASK
C
128 ITSK=ITSK+1
  I=TSKSEQ(ITSK)
  GO TO (51, 52, 53,54,55, 56,57,58,59,60,61,62,127,63,64,65,66),I
C
C INPUT, DISK
C
55 ET=MET(NIT)
  ST=MST(NIT)
  NIT=NIT+1
  LOG=LOGSEQ(NILOG)
  NILOG=NILOG+1
  K=STP(NISTP)
  NISTP=NISTP+1
  NPTS=NPTSEQ(ITSK)
  RPTS=RPSEQ(NRPTS)
  NRPTS=NRPTS+1
  J=SEQSEQ(NISEQ)
  NISEQ=NISEQ+1
1301 READ(LCG,END=9999) RTLT,(SPECT(I),I=1,NPTS),(RAD(I),I=1,RPTS)
  LRCRD=LRCRD+1
  IF(TIM.LT.ST) GC TO 301
  IF(TIM.GT.ET) GC TO (11,301),J
  IF(K.EQ.0) RTLT(4) GO TO 301
  NSPROC=NSPROC+1
  GO TO 129
C
C INPUT, TAPE NA FORMAT
C
52 ET=MET(NIT)
  ST=MST(NIT)
  NIT=NIT+1
  LOG=LOGSEQ(NILOG)
  NILOG=NILOG+1

```

```

      KK=STP(NISTP)
      NISTP=NISTP+1
      J=SEFSEQ(NISEQ)
      NISEQ=NISEQ+1
      NPTS=NPTSEQ(ITSK)
      RPTS=RPTSEQ(NRPTS)
      NRPTS=NRPTS+1
      K=NPTS+6+RPTS
800 READ(LOG,END=9999,ERR=5252) (MZ(I),I=1,K)
      LRCRD=LRCRD+1
      IF (NTIM.LT.ST) GO TO 800
      IF (NTIM.GT.ET) GO TO (11,800), J
      I=MZ(4)
      IF(I.GT.21) GO TO 800
      IF(I.EQ.KK) GO TO 800
      DO 803 I=1,4
803  RTLT(I)=MZ(I)
      RTLT(7)=0
      RTLT(8)=1
      TIM=NTIM
      DO 802 I=1,NPTS
802  SPECT(I)=MZ(I+6)
      J=NPTS+6
      DO 804 I=1,RPTS
804  RAD(I)=MZ(I+J)
      NSPROC=NSPROC+1
      GO TO 129
5252 WRITE(6,5251) LOG
5251 FORMAT(' *** INPUT ERROR ON LOGICAL ',I2,'. *RECORD SKIPPED')
      READ(LOG,END=9999) (MZ(I),I=1,K)
      GO TO 800

C
C INPUT, TAPE TRUCK FORMAT
C
57 ET=MET(NIT)
   ST=MST(NIT)
   NIT=NIT+1
   LOG=LOGSEQ(NILOG)
   NILOG=NILOG+1
   NPTS=NPTSEQ(ITSK)
   RPTS=RPTSEQ(NRPTS)
   NRPTS=NRPTS+1
   J=SEQSEQ(NISEQ)
   NISEQ=NISEQ+1
454 K=1
450 CALL REED(69999)
      LRCRD=LRCRD+1
      GO TO (451,452,452,452,453,453,453),STATUS
451 IF(TIM.LT.ST) GO TO 450
      IF(TIM.GT.ET) GO TO (11,450),J
      RTLT(7)=0
      RTLT(8)=2
      DO 461 I=1,NPTS
461  SPECT(I)=HSPECT(I)/51.2
      NSPROC=NSPROC+1
      GO TO 129
453 K=2
452 WRITE(6,460) STATUS,ECWB,CSWH,SENSE
460 FORMAT(/' ***ERROR IN READING TRUCK TAPE, STATUS= ',I2,'EC9W= ',
      A28,' CSH(FIRST HALF)= ',Z8,' SENSE BYTES= ',Z4)
      GO TO (451,454),K

C
C INPUT CARD
C
58 ET=MET(NIT)
   ST=MST(NIT)
   NIT=NIT+1
   LOG=LOGSEQ(NILOG)
   NILOG=NILOG+1
   K=STP(NISTP)
   NISTP=NISTP+1
   NPTS=NPTSEQ(ITSK)
   RPTS=RPTSEQ(NRPTS)
   NRPTS=NRPTS+1
   J=SEQSEQ(NISEQ)
   NISEQ=NISEQ+1
      DO 350 I=1,20
350  PFMT(I)=FMT(NIFMT,I)
      NIFMT=NIFMT+1
351 READ(LOG,PFMT,END=9999) RTLT,(SPECT(I),I=1,NPTS),(RAD(I),I=1,RPTS)
      LRCRD=LRCRD+1
      IF(TIM.LT.ST) GO TO 351
      IF(TIM.GT.ET) GO TO (11,351),J
      IF(K.EQ.9TLT(4)) GO TO 351
      NSPROC=NSPROC+1
      GO TO 129

C
C OUTPUT, DISK
C
51 RPTS=RPTSEQ(NRPTS)
   NPTS=NPTSEQ(ITSK)
   NRPTS=NRPTS+1
   LOG=LOGSEQ(NILOG)
   NILOG=NILOG+1
      WRITE(LOG) RTLT,(SPECT(I),I=1,NPTS),(RAD(I),I=1,RPTS)
      IORECD=IORECD+1
      IOCNT=IORECD
      GO TO 92

C
C OUTPUT, CARD
C
54 RPTS=RPTSEQ(NRPTS)
   NPTS=NPTSEQ(ITSK)
   NRPTS=NRPTS+1
   LOG=LOGSEQ(NILOG)
   NILOG=NILOG+1
      DO 500 I=1,20
500  PFMT(I)=FMT(NIFMT,I)
      NIFMT=NIFMT+1
      WRITE(LOG,PFMT) RTLT,IST,(SPECT(I),I=1,NPTS),(RAD(I),I=1,RPTS)
      IORECC=IORECC+1
      IOCNT=IORECC
      GO TO 92

C
C PATION
C
50 NPTS=NPTSEQ(ITSK)
      DO 550 I=1,NPTS
550  SPECT(I)=SPECT(I)/8859.7(I)
      J=PTLT(7)
      IF(MOD(J,8).EQ.4) GO TO 556
      PTLT(7)=J+4

```

```

556 LETTER=RRRR
GO TO 91

C
C NCRM
C
61 NPTS=NPTSEQ(IITSK)
SUM=0.0
SSQ=0.0
DO 600 I=1,NPTS
R=SPECT(I)
SLM=SUM+R
600 SSQ=SSQ+R**P
R=NPTS
SSQ=(P*SSQ-SUM*SUM)/(P-P)
SUM=SUM/P
SSQ=SQRT(SSQ)
DO 601 I=1,NPTS
601 SPECT(I)=(SPECT(I)-SUM)/SSQ
J=PTLT(7)
IF(MOD(J,16).GE.8) GO TO 607
RTLT(7)=J+8
607 LETTER=NNNN
GO TO 91

C
C CLIP
C
62 J=NCLL(NICL)
NPTS=NPTSEQ(IITSK)-NCLR(NICL)-J
NICL=NICL+1
DO 650 I=1,NPTS
650 SPECT(I)=SPECT(I+J)
LETTER=CCCC
GO TO 91

C
C SMOOTH
C
60 NPTS=NPTSEQ(IITSK)
J=SMCSEQ(NISMC)
NISMC=NISMC+1
CALL SM(SPECT,DSPCT,NPTS,STATUS,J)
IF(STATUS.EQ.0) GO TO 700
WRITE(6,701)
701 FORMAT(' ***ERROR IN SMOOTHING, SMOOTHING DELETED')
GO TO 128
700 DO 709 I=1,NPTS
709 SPECT(I)=DSPCT(I)
J=RTLT(7)
IF(MOD(J,4).GE.2) GO TO 710
RTLT(7)=J+2
710 LETTER=SSSS
GO TO 91

C
C BANDING ONCE
C
56 NPTS=NPTSEQ(IITSK)
J=LEND(NEND)
K=REND(NEND)
NEND=NEND+1
JK=K-J+1
SUM=0.0
DO 900 I=J,K

900 SUM=SUM+SPECT(I)
SPECT(J)=SUM/JK
K=K+1
JK=JK-1
DO 901 I=K,NPTS
901 SPECT(I-JK)=SPECT(I)
J=RTLT(7)
IF(MOD(J,32).GE.16) GO TO 905
RTLT(7)=J+16
905 LETTER=RRRR
GO TO 91

C
C BANDING REPEATED
C
63 NPTS=NPTSEQ(IITSK)
J=LEND(NEND)
K=REND(NEND)
NEND=NEND+1
JK=0
ISUB=NPTS-K
DO 950 I=1,ISUB,J
JK=JK+1
SUM=0.0
KK=I+J-1
DO 951 JJ=I,KK
951 SUM=SPECT(JJ)+SUM
950 SPECT(JK)=SUM/J
IF(K.EQ.0) GO TO 953
JK=JK+1
J=I+K-1
SUM=0.0
DO 952 JJ=I,J
952 SUM=SUM+SPECT(JJ)
SPECT(JK)=SUM/K
GO TO 953
953 J=RTLT(7)
IF(MOD(J,32).GE.16) GO TO 957
RTLT(7)=J+16
957 LETTER=RRRB
GO TO 91

C
C TEMP
C
64 RPTS=RPPTSEQ(NRPTS)
NRPTS=NRPTS+1
JJ=PRNTCP(NPCP)
NOPP=NOPP+1
IF(PTLT(R).EQ.2) GO TO 128
N=0
RBSQ=0
DO 810 I=1,RPTS,3
N=N+1
R=READ(I)
R=REF-(R/1000.-.970)/PPPM
J=1
K=141
KK=71
811 TIRFF=PADTP(KK)
IF(RQ-TIRFF) R12,R13,R14
814 JK=(K-KK+1)/2+KK
J=KK+1

```

UI-II

```

      IF(KK.EQ. JK) GO TO 815
      KK=JK
      GO TO 811
812 JK=-(KK-J+1)/2+KK
      K=KK-1
      IF(KK.EQ. JK) GO TO 815
      KK=JK
      GO TO 811
813 RAD(N)=((TREF-1.)/2.)*10.
      GO TO 810
815 IF(J.EQ.142.OF.K.EQ.0) GO TO 820
      JK=KK+1
      IF(PP,LT,TREF) JK=KK-1
      TRB=RADTR(JK)
      TREF=(PP-TREF)/ABS(TRB-TREF)
      TREF=TREF+KK
      RAD(N)=((TREF-1.)/2.)*10.
      GO TO 810
820 WRITE(6,821) A,IST,P,PR
821 FORMAT(/' *** RADIOMETER VALUE ',I3,' IN SPECTRUM',2I3,16,
* ' IS OUTSIDE RADIANCE TABLE LIMITS. (0-70 DEG. CENT.)'/
* ' RADIOMETER VALUE = ',F6.0,' RADIANCE VALUE = ',E13.6)
      RAD(N)=R
      PRSQ=1
810 CONTINUE
      IF(BRSQ.NE.0) GO TO 825
      J=RTLT(7)
      IF(MOD(J,2).GE.1) GO TO 825
      RTLT(7)=J+1
825 IF(JJ-2) 128,816,816
816 WRITE(6,817) IST,NSPROC
817 FORMAT(/' ',2I3,16,' T',I10)
      IF(JJ-3) 128,818,818
818 WRITE(6,819) (RAD(I),I=1,N)
819 FORMAT(4(/' ',25I5))
      GO TO 128

C
C TCUT
C
65 RPTS=PPTSEQ(NRPTS)
      NRPTS=NRPTS+1
      TREF=0.0
      TRREF=0.0
      DO 960 I=1,RPTS
      TRB=RAD(I)
      TREF=TREF+TRB
960 TRREF=TRREF+TRB*TRB
      TRREF=(RPTS*TRREF-TREF*TREF)/(RPTS*NRPTS-RPTS)
      TREF=TREF/RPTS
      TRREF=SQRT(TRREF)
      TRB=TCTPCT(NIPCT)
      NIPCT=NIPCT+1
      LETTER=TTTT
      IF(TRB) 970,971,971
971 J=TREF-TRREF*TRB
      K=TREF+TRREF*TRB
963 DO 961 I=1,RPTS
      JK=RAD(I)
      IF(JK,LT,J) GO TO 962
961 IF(JK,GT,K) GO TO 962
      GO TO 91

128 JJ=PRNTCP(NPCP)
      IF(JJ.NE.0) WRITE(6,966) IST,I,JK,K,J
966 FORMAT(/' ',2I3,16,' RAC VALUE NO. ',I3,' IS',I4,' MAX =',I4,
* ' MIN =',I4)
      L=CUTSEQ(NICUT)
      NICUT=NICUT+1
      IF(L) 91,91,964
964 IF(JJ.NE.0) WRITE(6,965)
965 FORMAT(' SPECTRUM CUT')
      GO TO 127
970 J=TREF+TRB
      K=TRREF-TRB
      GO TO 963

C
C PLOT
C
53 DO 751 I=1,NPTPLT
      R=SPECT(I)
      MEAN(I)=MEAN(I)+R
751 SD(I)=SD(I)+R*R
      NSPCT=NSPCT+1
      ICONT=0
      GO TO 92

C
C PLOT, EACH
C
66 IZ=1
      CALL PLCT(NPTSEQ(ITSK),SPECT,ZERO,IZ,IST,IST,TLT,XMIN(NBND),
* XMAX(NBND),LOGSEQ(NILCG))
      NRND=NRND+1
      NILCG=NILCG+1
      GO TO 128

C
C ROUTINE TO TRANSLATE TIME AFTER EACH INPUT TASK
C
129 IF(RTLT(5).EQ.1) TIM=NTIM
      IST(1)=TIM/3600000
      K=MOD(TIM,3600000)
      IST(3)=MOD(K,60000)
      IST(2)=K/60000
      GO TO 128

C
C ROUTINE TO PRINT SPECTRUM AFTER EACH TASK, IF REQUIRED
C
91 J=PRNTCP(NPCP)
      NPCP=NPCP+1
      IF(J-2) 128,954,954
954 WRITE(6,903) IST,LETTER, NSPROC
903 FORMAT(/' ',2I3,16,1X,A1,I10)
      IF(J-3) 128,955,955
955 WRITE(6,655) (SPECT(I),I=1,JK)
655 FORMAT(10(/' ',10F10.3))
      GO TO 128

C
C ROUTINE TO PRINT SPECTRA AFTER EACH OUTPUT TASK, IF REQUIRED
C
92 J=PRNTCP(NPCP)
      NPCP=NPCP+1
      IF(J-2) 128,752,752
752 WRITE(6,753) IST,RTLT,ICONT
753 FORMAT(/' ',2I3,16,5X,4(7,1X),I10)

```

VI-12

```

      IF(J-3) 128,754,754
      754 WRITE(6,655) (SPCT(I),I=1,NPTPLT)
      GO TO 128
C
C ***** END RUN SECTION *****
C
C EXIT FROM CYCLE RUN
C
9999 IF((TASKSFC(ITSK),LE,5).AND.(J.EQ,2)) REWIND LOG
      11 IF(NSPPCC.NE,C) GO TO 172
      WRITE(6,776) LRCPO
      776 FORMAT('END OF TASK CYCLE, NO SPECTRA WITHIN TIME LIMITS ON',
      A' DATASET//' NUMBER OF RECORDS READ = ',I7)
      J=0
      GO TO 10
      172 IF(.NOT.PLOT) GO TO 171
C
C PLOT AVERAGE
C
C COMPUTE MEAN, SD AND RE
C
      IF(NSPCT.LE,1) GO TO 3302
      FNSPCT=NSPCT
      SQTNSP=SQRT(FNSPCT)
      DO 3001 I=1,NPTPLT
      SD(I)=SD(I)-MEAN(I)*MEAN(I)/NSPCT
      SD(I)=SQRT(SD(I)/(NSPCT-1))
      MEAN(I)=MEAN(I)/NSPCT
      3001 RE(I)=SD(I)/(MEAN(I)*SQTNSP)
      GO TO 3402
      3302 DO 3401 I=1,NPTPLT
      SD(I)=0.0
      3401 RE(I)=0.0
C
C INITIALIZE MEAN, SD, AND RE TOTALS TO ZERO
C
      3402 TMEAN=0.0
      TSD=0.0
      TRE=0.0
C
C WRITE MEAN, SD, AND RE
C
      WRITE (PLOG,1903) TLT, NSPCT
      1903 FORMAT('1',20A4//',',NUMBER OF SPECTRA IS',I10//',',4X,'COUNTER',
      117X,'AVE',13X,'STD DEV',6X,'RELATIVE ERROR')
      DO 3002 I=1,NPTPLT
      WRITE(PLOG,1908) I, MEAN(I),SD(I),RE(I)
      1908 FORMAT(' ',I11,2F20.4,F20.8)
      TMEAN=MEAN(I)+TMEAN
      TSD=SD(I)+TSD
      3002 TRE=RE(I)+TRE
      TMEAN=TMEAN/NPTPLT
      TSD=TSD/NPTPLT
      TRE=TRE/NPTPLT
      WRITE(PLOG,1907) TMEAN,TSD,TRE
      1907 FORMAT(' ',7X,'MEAN',2F20.4,F20.8)
C
C PLOT MEAN AND SD
C
      I2=PLOG
      CALL PLOTT(NPTPLT,MEAN,SD,NSPCT,IPST,TPET,TLT,PXMIN,PXMAX,I2)
C
C
C
      171 WRITE(6,777) NSPPCC,LRCPO,IORECD,IORECC
      777 FORMAT('END OF TASK CYCLE, NUMBER OF SPECTRA PROCESSED = ',I7//', N
      1UMBER OF RECORDS READ = ',I7//', NUMBER OF RECORDS WRITTEN ON DISK =
      2',I7//', NUMBER OF RECORDS WRITTEN ON CARDS = ',I7)
      J=0
      GO TO 10
C
C ERROR ROUTINE
C
10000 WRITE(6,1200) TASK
      1200 FORMAT('*** TASK ERROR -- TASK OPERATOR IS ',A4,' CYCLE DELETED
      1')
      GO TO 130
10001 WRITE(6,1200) ICTYPE
      1201 FORMAT('*** ICTYPE ERROR -- ICTYPE IS ',A4,' CYCLE DELETED')
      GO TO 130
10002 WRITE(6,1202) BCTYPE
      1202 FORMAT('*** BCTYPE ERROR -- BCTYPE IS ',A4,' CYCLE DELETED')
      GO TO 130
10003 WRITE(6,1203) BBTYPE
      1203 FORMAT('*** BBTYPE ERROR -- BBTYPE IS ',A4,' CYCLE DELETED')
C
C SEARCH FOR START OF NEXT CYCLE
C
      130 READ(5,1001,END=999) TASK,PARAM,ERRDMP,AIBN,NPTS,RPTS
      IF(TASK.NE,START) GO TO 130
      J=1
      GO TO 10
11000 WRITE(6,1204) STATUS
      1204 FORMAT('*** ERROR, TRUCK TAPE DATA SET UNABLE TO BE OPENED, CYC
      1CLE DELETED. STATUS IS ',I5)
      GO TO 130
10004 WRITE(6,1205) BCTYPE
      1205 FORMAT('*** BCTYPE ERROR -- BCTYPE IS ',A4,' CYCLE DELETED')
      GO TO 130
10005 WRITE(6,1206)
      1206 FORMAT('*** ERROR, THERE WERE NO BLACKBODY SPECTRA ON THE DATA
      1ASET WHICH WERE INSIDE TIME LIMITS, CYCLE DELETED')
      GO TO 130
10006 WRITE(6,1207) PARAM
      1207 FORMAT('*** ERROR, PRINT PARAM FOR TASK WAS INVALID, PRINT PARA
      1M IS ',A4,' -- CYCLE DELETED.')
      GO TO 130
10007 WRITE(6,1208) ICTYPE
      1208 FORMAT('*** ERROR, MULTIPLICITY PARAMETER FOR BANDING IS ',A4,'
      1. CYCLE DELETED.')
      GO TO 130
10008 WRITE(6,1209) TTYPE
      1209 FORMAT('*** TTYPE ERROR, TTYPE IS ',A4,' CYCLE DELETED')
      GO TO 130
10009 WRITE(6,1210) TITYPE
      1210 FORMAT('*** TITYPE ERROR, TITYPE IS ',A4,' CYCLE DELETED')
      GO TO 130
10010 WRITE(6,1211)
      1211 FORMAT('*** ERROR, THERE WERE NO REFERENCE RADIANCE SPECTRA
      1ON THE DATASET WHICH WERE INSIDE TIME LIMITS, CYCLE DELETED')
      GO TO 130
10011 WRITE(6,1212)
      1212 FORMAT('*** ERROR, RADIOMETER VALUES ALREADY CONVERTED FROM AA
      1*FORMAT -- TASK IGNORED')
      READ(5,10112)

```

UL-13

```

10112 FORMAT(/)
      GO TO 20
10012 WRITE(6,1213)
1213 FORMAT(/' *** ERROR, TRUCK TAPE AS INPUT TO TEMP REFERENCE RADIA
      *NCE -- CYCLE DELETED.')
      GO TO 130
10013 WRITE(6,1214) ICTYPE
1214 FORMAT(/' *** ERROR, TWTYPF IS INVALID IN TASK TCUT. TWTYPF IS ',
      *A4,' -- CYCLE DELETED.')
      GO TO 130
10014 WRITE(6,1215)
1215 FORMAT(/' *** ERROR, AVERAGE PLOT TASK ALREADY CALLED -- TASK IGM
      *NCE.')
      GO TO 20
10015 WRITE(6,1216) ICTYPE
1216 FORMAT(/' *** PLOT PARAMETER ERROR. ICTYPE IS ',A4,'. CYCLE DELET
      *ED.')
      GO TO 130
10016 WRITE(6,1217) PAMP
1217 FORMAT(/' *** ERROR, TCUT PARAMETER IN ERROR. PAMP IS ',A4,'. CYC
      *LE DELETED.')
      GO TO 130
10017 WRITE(6,1218)
1218 FORMAT(/' ***ERROR -- GREATER THAN 20 TASKS IN THIS CYCLE. CYCLE
      * DELETED.')
      GO TO 130

```

```

C
C   END
C

```

```

939 WRITE(6,1600)
      J=-4
      IF(LABND) WRITE(J,1600)
      RETURN
1630 FORMAT('1')
      END
      SUBROUTINE XLATE(TIM,IST)
      INTEGER IST(3),TIM
      IST(1)=TIM/3600000
      K=MOD(TIM,360000)
      IST(3)=MOD(K,40000)
      IST(2)=K/40000
      RETURN
      END
      SUBROUTINE PLOT(NPTS,MEAN,SD,NSPCT,IST,IET,TLT,MIN,MAX,LOG)

```

```

C
C   PLOT SUBROUTINE
C
C   THIS SUBROUTINE MUST BE SUPPLIED: NUMBER OF POINTS PER SPECTRUM,
C   NUMBER OF SPECTRUM, BEGINNING AND END TIMES, TITLE, AND
C   VECTORS CONTAINING THE MEAN AND STANDARD DEV FOR EACH POINT
C   OF THE SPECTRUM
C

```

```

      REAL MEAN(100),SD(100),YCORD(11),TLT(20),MAX,MIN
      INTEGER*2 NPTS,NSPCT,LCG
      INTEGER IST(3),IET(3),GRAPH(101),PLOG
      INTEGER YBD(26)/'*****','**','*****','*****','*****','*****',
1**,'*****','*****','*****','*****','*****','*****','*****',
2**,'*****','*****','*****','*****','*****','*****','*****',
      INTEGER DOT/'...','STAR/'****'/,PLUS/'+++'/,MINUS/'---'/

```

```

      INTEGER BLNK/' ',XXXX/'XXXX'/
C
C   WRITE PLOT HEADING
C
      PLOG=LOG
      XMAX=MAX
      XMIN=MIN
      IF(NSPCT.EQ.1) GO TO 46
      WRITE(PLOG,1000)
1090 FORMAT('1',19X,'PLOT OF MEANS AND STANDARD DEVIATION OF AVERAGES F
      ROM TIMES')
      WRITE(PLOG,1001) IST,IET
1001 FORMAT(' ',25X,2I3,I6,' TC',2I3,I6)
      WRITE(PLOG,1002) TLT
1002 FORMAT(' ',19X,20A4)
      WRITE(PLOG,1003) NSPCT
1003 FORMAT(' ',19X,'NUMBER OF SPECTRA IS',I5)
      GO TO 47
46 WRITE(PLOG,6000) IST
6000 FORMAT('1',19X,'PLOT OF TIME ',2I3,I6/)
      WRITE(PLOG,1002) TLT

```

```

C
C   COMPUTE XMAX,XMIN,DELTA
C
      47 IF(XMIN.LE.XMAX) GO TO 42
      XMAX=10.0E-65
      XMIN=10.0E65
      DO 3000 I=1,NPTS
      IF (XMAX.GE.(MEAN(I)+SD(I))) GO TO 10
      XMAX=MEAN(I)+SD(I)
10 IF (XMIN.LE.(MEAN(I)-SD(I))) GO TO 3000
      XMIN=MEAN(I)-SD(I)
3000 CONTINUE
      42 DELTA=(XMAX-XMIN)/100.

```

```

C
C   WRITE GRAPH INFORMATION
C
      WRITE(PLOG,1004) XMAX,XMIN,DELTA
1004 FORMAT(/' ',19X,'XMAX IS',18X,'XMIN IS',19X,'DELTA IS'/19X,F12.4,
      I13X,F12.4,I15X,F12.4)
      WRITE(PLOG,1005) XMAX,XMIN
1005 FORMAT(' ',19X,'HIGHEST X IS',I3,' LOWEST X IS',I3)
C
C   COMPUTE AND PRINT Y COORDINATES
C
      DO 3001 I=1,11
      YCORD(I)=XMIN+(DELTA*(10*(I-1)))
      WRITE(PLOG,1006) YCORD
1006 FORMAT(/' ',7X,I11(F9.2,1X))
      WRITE(PLOG,1007) YBD
1007 FORMAT(' ',15X,26A4)

```

```

C
C   COMPUTE *,-,* POSITIONS AND FORM GRAPH LINE
C
      DO 3002 I=1,NPTS
C
C   INITIALIZE GRAPH LINE
C
      IYST=MOD(I,10)

```

VI-15

```

      IF (IMT, H, C) GO TO 61
      DO 3004 I=1, 101
3004  GRAPH(I)=NOT
      GO TO 74
      DO 3005 I=1, 101
3005  GRAPH(I)=NOT
      COMPUTE M, +, + POSITION S
      78  SM=MEAN(I)-XMIN
      ISM=(S/DLTA)+1
      ISH=(ISM+SD(I))/DELTA+1
      ISL=(ISM-SD(I))/DELTA+1
      IF (NOT (ISM, GE, 2, AND, ISM, LE, 101)) GO TO 3403
      IF (ISM, EQ, 1) GO TO 3403
      L=ISM-1
      DO 3003 J=1, L
3003  GRAPH(J)=XXXX
3403  IF (ISM, GE, 1, AND, ISH, LE, 101) GRAPH(ISH)=PLUS
      IF (ISL, GE, 1, AND, ISL, LE, 101) GRAPH(ISL)=MINUS
      IF (ISM, GE, 1, AND, ISM, LE, 101) GRAPH(ISM)=STAR
      WRITE GRAPH LINE
      WRITE (PLOG, 1008) MEAN(I), I, GRAPH, I
1008  FORMAT(' ', F10.5, I4, I4, 101A1, I4)
3007  CONTINUE
      WRITE (PLOG, 1007) YBC
      WRITE (PLOG, 1006) YCRC
      939  RETURN
      79  WRITE (PLOG, 1015) ISL, ISM, ISH
1015  FORMAT('/', '***GRAPH ERROR: ISL, ISM, OR ISH ARE OUT OF BOUNDS!')
      ISL = 'I4', ISM = 'I4', ISH = 'I4'
      GO TO 939
      END
      SUBROUTINE STREAN
      RETURN
      END
      SUBROUTINE REED
      RETURN
      END
      SUBROUTINE SM(MDATA, MDATA, M, IEP, NMP)
      SMOOTHING SUBROUTINE WRITTEN BY J.R. MOORE
      MDATA=INPUT SPECTRUM
      MDATA=OUTPUT SPECTRUM
      N=NUMBER OF POINTS
      IER=ERROR MESSAGE--0 IF OK, -1 IF NOT
      NMP=SMOOTHING TYPE
      REAL MDATA(100), MDATA(100), NP(20)
      IF (N, GT, 100, OR, NMP, LT, -20, OR, NMP, GT, 20) GO TO 900
      NNP=NMP
      IF (NNP, LT, 0) NNP=-NNP
      NXP=NNP
      IF (NNP, EQ, -1) NNP=3
      MM=NNP-1
      DO 20 I=1, N
20  MDATA(I)=MDATA(I)
      DO 10 I=2, NNP
      J=I-1
10  NP(I)=MDATA(J)
      DO 200 J=1, MM
      J1=J+MM
      DO 11 K=1, MM
      KA=K+1
11  NP(K)=NP(KA)
      NP(MNP)=MDATA(J)
      IF (NNP, LT, 0) GO TO 100
      GO TO (300, 500, 900, 900, 101, 900, 102, 900, 103, 900, 104, 900,
      1401, 900, 900, 900, 402), NNP
101  SUM=17*NP(3)+12*(NP(2)+NP(4))-3*(NP(1)+NP(5))
      MDATA(I+2)=SUM/35
      GO TO 200
401  SUM=NP(1)+NP(2)+NP(3)+NP(4)+NP(5)+NP(6)+NP(7)+NP(8)+NP(9)+
      NP(10)+NP(11)+NP(12)+NP(13)
      MDATA(I+6)=SUM/13
      GO TO 200
402  SUM=-21*(NP(1)+NP(17))-6*(NP(2)+NP(16))+7*(NP(3)+NP(15))+
      118*(NP(4)+NP(14))+27*(NP(5)+NP(13))+34*(NP(6)+NP(12))+
      139*(NP(7)+NP(11))+42*(NP(8)+NP(10))+43*NP(9)
      MDATA(I+8)=SUM/323
      GO TO 200
403  SUM=105*(NP(1)+NP(17))-195*(NP(2)+NP(16))-260*(NP(3)+NP(15))-
      1117*(NP(4)+NP(14))+135*(NP(5)+NP(13))+415*(NP(6)+NP(12))+
      2660*(NP(7)+NP(11))+825*(NP(8)+NP(10))+883*(NP(9))
      MDATA(I+8)=SUM/4199
      GO TO 200
102  SUM=-2*(NP(1)+NP(7))+3*(NP(2)+NP(6))+6*(NP(3)+NP(5))+7*NP(4)
      MDATA(I+3)=SUM/21
      GO TO 200
103  SUM=-21*(NP(1)+NP(9))+14*(NP(2)+NP(8))+39*(NP(3)+NP(7))+
      154*(NP(4)+NP(6))+59*NP(5)
      MDATA(I+4)=SUM/231
      GO TO 200
300  SUM=NP(1)+NP(2)+NP(3)
      MDATA(I+1)=SUM/3
      GO TO 200
104  SUM=-36*(NP(1)+NP(11))+9*(NP(2)+NP(10))+44*(NP(3)+NP(9))+
      167*(NP(4)+NP(8))+84*(NP(5)+NP(7))+89*NP(6)
      MDATA(I+5)=SUM/429
      GO TO 200
      CONTINUE
      GO TO (300, 500, 900, 900, 101, 900, 106, 900, 107, 900, 108,
      1900, 900, 900, 500, 900, 403), NXP
106  SUM=5*(NP(1)+NP(7))-3*(NP(2)+NP(6))+75*(NP(3)+NP(5))+131*NP(4)
      MDATA(I+3)=SUM/231
      GO TO 200
107  SUM=15*(NP(1)+NP(9))-55*(NP(2)+NP(8))+30*(NP(3)+NP(7))+
      1135*(NP(4)+NP(6))+179*NP(5)
      MDATA(I+4)=SUM/429
      GO TO 200
108  SUM=1*(NP(1)+NP(11))-4*(NP(2)+NP(10))-10*(NP(3)+NP(9))+
      16*(NP(4)+NP(8))+12*(NP(5)+NP(7))+14*NP(6)
      MDATA(I+5)=SUM/429
      CONTINUE
      IER=0
      RETURN
      100  IER=-1

```

-33-

```

      //LXFF, SYSNAME DD JSI=J32, SYSLIN(TASK), DISP=(CLD,KEEP),
      //VOLUME=SYS12, UNIT=2114, SPACE=(TRK, (10, 10, 2), RLSE)
      //

```

```

1. //COPYL JCL (JCL2,312,1,C,101,'HEATHMAN NEWFIGHT',MSGLEVEL=1
2. // EXEC FORTFCL
3. //FORT,SYSLN DD *
4. C+++++
5. C
6. C
7. C
8. C
9. C STANFORD NEAREST NEIGHBOR
10. C -----
11. C
12. C SPECTRAL MATCHING PROGRAM
13. C -----
14. C
15. C
16. C WRITTEN BY J. R. MOORE FOR R. LYNN
17. C DEPARTMENT OF GEOPHYSICS
18. C STANFORD UNIVERSITY
19. C DECEMBER 1968
20. C
21. C
22. C
23. C
24. C+++++
25. C
26. C
27. C
28. C
29. C
30. C-----INPUT CONTROL DATA-----
31. C-----CARD 1 TITLE CCLS 1/72
32. C-----CARD 2 NPTS,IP,IVOT,ISM,ITR,INCC,ICLIP IN 7F10.0 FORMAT
33. C APTS=NUMBER OF POINTS IN SPECTRA ACTUALLY USED
34. C IP=NUMBER OF USEFUL POINTS TO BE SELECTED
35. C IVOT=NUMBER OF "VOTES" USED IN DECISION
36. C ISM=1 IF WEIGHTING FACTORS TO BE READ IN
37. C ITR=NUMBER OF SPECTRA READ FOR EACH TRAINING
38. C TIME SPECIFIED
39. C INOC=1 IF NC VARIANCE IS TO BE COMPUTED
40. C ICLIP=NUMBER OF POINTS CLIPPED OFF FRONT OF
41. C EACH SPECTRUM
42. C NOTE:NPTS+ICLIP MUST NOT EXCEED NUMBER OF
43. C POINTS IN STORED SPECTRA
44. C-----CARDS 2,1/2,ANN (OPTIONAL) SPACE VECTORS UP AND DOWN
45. C IN 10F5.0 FORMAT
46. C-----TRAINING CONTROL CARDS:
47. C COLS 1/5 "TRAIN"
48. C CCLS 9/16 GROUP LABEL IF NEW GROUP
49. C COLS 17/28 START TIME FOR TRAINING GROUP
50. C COLS 29/40 BLANK
51. C CCLS 41/43 "NEW" IF TIME REPRESENTS START OF NEW
52. C TRAINING GROUP, BLANK IF TIME IS ADDING TO
53. C GROUP LAST READ IN
54. C NOTE: EACH TRAINING GROUP MUST BE COMPLETELY SPECIFIED
55. C BEFORE A NEW ONE IS STARTED. IF A NEW GROUP
56. C STARTS WITH A TIME WHICH IS PREVIOUS ON THE
57. C DISK TO THE LAST TIME READ IN THE PREVIOUS
58. C GROUP, THEN COLUMNS 45/50 MUST BE "REWIND"
59. C-----TEST CASE CONTROL CARDS:
60. C COLS 1/4 "TEST"

61. C COLS 9/16 TEST GROUP LABEL IF DESIRED
62. C COLS 17/28 START TIME FOR TEST GROUP
63. C CCLS 29/40 END TIME FOR TEST GROUP (213,16)
64. C REMAINING COLUMNS INSIGNIFICANT
65. C-----STOP CARD
66. C TO TERMINATE EXECUTION, USE "STOP" IN COLS 1/4
67. C-----CARD SEQUENCE IS:
68. C TITLE CARD
69. C CONTROL CONSTANTS CARD
70. C (OPTIONAL) SPACE VECTORS CARD
71. C TRAINING CONTROL CARDS
72. C TEST CONTROL CARDS
73. C STOP CARD
74. C
75. C*****
76. C G E N E R A L P R O G R A M M I N G H I N T S
77. C MAXIMUM NUMBER OF TRAINING SPECTRA GROUPS IS 10
78. C MAXIMUM NUMBER OF INDIVIDUALS PER TRAINING GROUP IS 20
79. C MAXIMUM NUMBER OF POINTS IS 100
80. C THIS PROGRAM WILL HANDLE ONLY DATA WITH TIME CODED
81. C INFORMATION AND WHICH HAS A LOW-HIGH RAMP MARKER
82. C IN THE DATA IDENTIFICATION BLOCK. IT DOES NOT REQUIRE
83. C THAT THERE BE AN EVEN ALTERNATION OF HIGH AND LOW
84. C RAMPS
85. C*****
86. C
87. C
88. C MODIFIED SPRING 1970 BY MIKE HEATHMAN
89. C FOR USE WITH 69/70 VERSION OF DATA SYSTEM
90. C
91. C NOTE: F107F001 IS NORMALLY DUMMIED, UNLESS CARDS
92. C ARE WANTED.
93. C
94. C
95. C+++++
96. C REAL*4 X(100)/100*0./,SS(100,2)/200*0./,DIST(2,10,20)/400*0./,
97. C ITLT(1,20)
98. C INTEGER*4 ISC(10)/10*0/,NSEL(100,2)/20*0/,ISB(3),IEB(3),LAPA(100)
99. C REAL*8 TLT(0),TRAIN/'TRAIN' //,TEST/'TEST' //,TXT,
100. C ISTOP/'STOP' //,TLT(10),TXT,TRWN,IREW/'REWIND' //
101. C INTEGER*2 RTLT(12)
102. C INTEGER*4 IS(3),UPLCW(2)/'LCW','HIGH'//,INFW/'NEW' //,
103. C IKBNA/' //,KTIE/'*TIE'//
104. C EQUIVALENCE (ITIME,RTLT(5))
105. C COMMON T(100,2,10,20),S(100,2),TH(100,2,10),IT(10,2),IALT
106. C INTEGER CLECK1
107. C READ(5,1) TLT
108. C IALT=1
109. C
110. C READ IN CONTROL CONSTANTS
111. C
112. C DO 3000 I=1,10
113. C IT(1,1)=0
114. C 3000 IT(1,2)=0
115. C 1 FORMAT(SAE)
116. C WRITE(6,11) TLT
117. C 11 FORMAT('11 STANFORD NEAREST NEIGHBOR PROCESSOR 1/720,000')
118. C READ(5,2) XPTS,XP,XMT,ISM,XTC,XNCC,XCLIP
119. C ITI=CLECK1(3)
120. C 2 FORMAT(7F10.0)
121. C INCC=XNCC

```

01-17

```

122. ICLIP=XCLIP
123. NPTS=NPTS
124. IP=XIP
125. IVCT=XYCT
126. ISM=XSM
127. ITR=XTR
128. WRITE(6,12) NPTS,IP,IVCT,ISM,ITR,ICLIP
129. 12 FORMAT('NPTS=',I2,' IP=',I2,' IVCT=',I2,' ISM=',I2,' ITR=',
130. 114,' ICLIP=',I2)
131. NT=0
132. C
133. C READ IN SPACE VECTOR IF REQUIRED
134. C
135. IF (ISM.EQ.1) READ(5,3)((S(I,J),I=1,NPTS),J=1,2)
136. 3 FORMAT(10F5,0)
137. WRITE(6,1006)
138. 1006 FORMAT('1')
139. 100 NT=NT+1
140. C
141. C READ IN TRAINING CONTROL INFORMATION
142. C
143. READ(5,4) TXT,TTLT(NT),ISB,IEB,INEH,TRHW
144. IF (IP.NE.EQ.IREW) REWIND 12
145. 1 STAR=ISB(1)*360000+ISB(2)*60000+ISB(3)
146. 1 IENE=IEB(1)*360000+IEB(2)*60000+IEB(3)
147. IF (TXT.EQ.TEST) GO TO 500
148. IF (TXT.EQ.STOP) GO TO 900
149. IF (TXT.NE.TRAIN) GO TO 901
150. IF (INEH.NE.INEW) NT=NT+1
151. WRITE(6,13) TXT,TTLT(NT),NT,ISB,IEB,INEH,TRHW
152. 4 FORMAT(2A8,2(2I3,16),A4,A8)
153. 13 FORMAT('BEGINNING STEP ',2A8,1X,I2,' TIMES: ',2(2I3,16),1X,A4,A8)
154. ICCUNT=C
155. C
156. C READ IN TRAINING DATA AND TEST TIMES
157. C
158. 101 IF (ICLIP.LE.0,OP.ICLIP+NPTS.GT.100) GO TO 7000
159. READ(12,EAC=906,ERR=903) RTLT,(ZZZ,I=1,ICLIP),(X(I),I=1,NPTS)
160. GO TO 7001
161. 7000 REAC(12,END=906,ERR=903) RTLT,(X(I),I=1,NPTS)
162. 7001 IF (ISB(1).EQ.0) GO TO 102
163. IF (ISTAR.GT.ITIME) GO TO 101
164. 102 NNN=NNN+1
165. IF (NNN.NE.1.AND.NNN.NE.2) GO TO 910
166. IF (NNN.EQ.2) IALT=2
167. ICCUNT=ICCUNT+1
168. ITINT,INN=ITINT,NNN)+1
169. DO 103 I=1,NPTS
170. T(I,NNN,NT,IT(NT,NNN))=X(I)
171. CALL TLT(TIME,IS)
172. WRITE(6,5) TXT,TTLT(NT),ICCUNT,IS(1),IS(2),IS(3),UPLOW(NNN)
173. 5 FORMAT(1X,2A8,14,1X,2I3,16,1X,A4)
174. IF (ICCUNT.LT.ITR) GO TO 101
175. GO TO 100
176. 500 NT=NT+1
177. T1J=CLCK1(3)
178. PPP=(T1J-T11)/100.
179. WRITE(6,5000) PPP
180. 5000 FORMAT(' ***ELAPSED TIME IS ',F10.3,' SECONDS****')
181. IF (INOC.NE.1) CALL COVAR(NT,IP,NPTS,ISM)
182. C

```

```

183. C TRAINING INPUT COMPLETED...COMPUTE WEIGHTING VECTOR
184. C
185. WRITE(6,1000)
186. 1000 FORMAT('***COVAR CALL COMPLETED..STARTING PROCESSING..')
187. 110 CONTINUE
188. DO 111 K=1,IALT
189. DO 111 I=1,IP
190. STST=100
191. DO 112 J=1,NPTS
192. IF (S(I,K).LT.STST) GO TO 112
193. ISEL=J
194. STST=S(I,K)
195. 112 CONTINUE
196. SS(ISEL,K)=S(ISEL,K)
197. S(ISEL,K)=0
198. C
199. C OUTPUT COMPUTED TRAINING VALUES AND STATISTICS
200. C
201. 111 NSEL(I,K)=ISEL
202. WRITE(6,2000) ((NSEL(I,J),I=1,IP),J=1,IALT)
203. 2000 FORMAT('SELECTION COMPLETED NSEL=',9(/I6))
204. WRITE(6,2001) IT
205. 2001 FORMAT(' IT=',10(/1X,I5))
206. DO 1001 I=1,IALT
207. 1001 WRITE(6,1002) UPLOW(I),((S(NSEL(J,I),I),J=1,IP)
208. 1002 FORMAT('0',A4,' WEIGHTING FACTORS ARE: ',10(/10F12.3))
209. ITRW=0
210. DO 6010 I=1,100
211. DO 6010 K=1,IALT
212. IF (SS(I,K).EQ.0) SS(I,K)=1
213. C
214. C SELECT BEST POINTS AND STORE COUNTER IN NSEL
215. C
216. DO 114 I=1,NPTS
217. DO 114 K=1,IALT
218. DO 114 J=1,NT
219. TM(I,K,J)=TM(I,K,J)+SS(I,K)
220. II=IT(I,K)
221. DO 114 L=1,II
222. 116 T(I,K,J,L)=T(I,K,J,L)+SS(I,K)
223. WRITE(6,1006)
224. WRITE(6,2002)
225. 2002 FORMAT('SELECTION COMPLETE')
226. DO 120 J=1,NT
227. DO 120 K=1,IALT
228. 120 WRITE(6,14) TTLT(J),UPLOW(K),(TM(I,K,J),I=1,NPTS)
229. T11=CLCK1(3)
230. PPP=(T11-T1J)/100.
231. WRITE(6,5000) PPP
232. 14 FORMAT('GROUP MEANS ',A8,1X,A4,10(/1X,10F12.7))
233. WRITE(6,6003)
234. DO 6003 K=1,IALT
235. DO 6003 L=1,NT
236. DO 6003 J=1,NT
237. DO 6003 I=1,IP
238. DO 6003 I=1,IP
239. DO 6001 I=1,IP
240. DO 6001 I=1,IP
241. 6000 WRITE(6,6002) TTLT(J),UPLOW(K),TTLT(II),UPLOW(K),DO
242. 6003 FORMAT('1 DISTANCES BETWEEN GROUP MEANS',120(' ')//)
243. 6002 FORMAT('DIST FROM ',A8,1X,A4,' TO ',A8,1X,A4,' IS ',F16.5)

```

01-18

```

244. 501 REWIND 12
245. GC TO 502
246. 503 REAL(5,4) TXT,TTLT(NT+1),ISB,IFB,INEX,TRWK
247. IF(1PWK.EQ.1PFW1) REWIND 12
248. IF(1XT.EQ.1STGP) GC TO 500
249. IF(1XT.NE.TEST) GC TO 504
250. NEWST=ISB(1)*360000+ISF(2)*60000+ISB(3)
251. IF(NEWST.LE.1STAB) GC TO 501
252. 502 ISTAT=NEWST
253. WRITE(6,13) TXT,TTLT(NT+1),NT,ISP,IFB,INEX,TRWK
254. IFB=IFB(1)*360000+IFB(2)*60000+IFB(3)
255. C
256. C REAL IN TEST SPECTRA AND CHECK TIMES
257. C
258. 510 IF(1CLIP.LE.0.DF.1CLIP+APTS.GT.100) GC TO 7002
259. READ(12,END=905,EPR=903) RTLT,(Z2,I=1,1CLIP),(X(I),I=1,NPTS)
260. GC TO 7002
261. 7002 READ(12,FND=905,EPR=903) RTLT,(X(I),I=1,NPTS)
262. 7003 IF(1SE(1).EQ.0) GC TO 520
263. IF(1STAP.GT.1TIME) GC TO 510
264. IF(1TIME.GT.1END) GC TO 503
265. 520 AAA=NNN+1
266. IF(NNN.NE.1.AND.NNN.NE.2) GC TO 910
267. C
268. C WEIGHT X VALUES WITH WEIGHTING VECTOR
269. C
270. DC 800 I=1,1P
271. 800 X(NSEL(I,NNN))=X(NSEL(I,AAA))*SS(ASFL(I,AAA),NNN)
272. IF(AAA.EQ.1) WRITE(6,1006)
273. CALL IXLT(1TIME,1S)
274. WRITE(6,15) TTLT(NT+1),IS(1),IS(2),IS(3),LFLOW(NNN)
275. 15 FCRMAT('0',2,(120(' ')))// NEAREST NEIGHBOR DECISION FOR 'A8,
276. 1213,16,44,120(' ')
277. WRITE(6,1004) (NSEL(I,NNN),X(NSEL(I,NNN)),I=1,1P)
278. 1004 FCRMAT('0',T20,'WEIGHTED X VALUES:',20(15(3X,13,''),F12.4))
279. WRITE(6,1005)
280. 1005 FCRMAT(1/X,120(' '))
281. WRITE(6,666)
282. 666 FCRMAT('0',T5,'GROUP',T15,'NUMBER',T25,'DISTANCE',T40,'DECISION'/
283. 11X,120(' '))
284. C
285. C COMPUTE DISTANCES FOR SINGLE UNKNOWN SPECTRA
286. C
287. DC 627 I=1,NT
288. II=IT(I,NNN)
289. DC 627 J=1,II
290. DIST(NNN,I,J)=0
291. DO 627 JJJ=1,1P
292. 627 DIST(NNN,I,J)=DIST(NNN,I,J)+(X(NSEL(JJJ,NNN))-X(NSEL(JJJ,NNN),NNN,
293. 11,J))**2
294. DO 667 I=1,1C
295. 667 ISC(I)=0
296. C
297. C SELECT AND OUTPLT CHOICES
298. C
299. DO 625 JJ=1,IVGT
300. DEC=100000000.
301. DC 626 I=1,NT
302. II=IT(I,NNN)
303. DC 626 J=1,II
304. IF(DEC.LT.DIST(NNN,I,J))GC TO 626

```

```

305. IGR=I
306. IIV=J
307. DEC=DIST(NNN,I,J)
308. 626 CONTINUE
309. ISC(IGR)=ISC(IGR)+1
310. IF(JJ.EQ.1) CS=DEC
311. DIST(NNN,IGR,IIV)=10000000.
312. 625 WRITE(6,16) TTLT(IGR),IIV,DEC
313. 16 FCRMAT(15,A8,T15,16,T25,E16.5)
314. C
315. C DETERMINE WINNER OF VOTING AND CHECK FOR TIE
316. C
317. IDEC=0
318. KTY=KBLA
319. DO 630 I=1,NT
320. IF(ISC(I).LT.IDEC) GO TO 630
321. ITIE=IGR
322. IGP=I
323. IF(ISC(I).NE.IDEC) ITIE=0
324. IDEC=ISC(I)
325. 630 CONTINUE
326. WRITE(6,17) TTLT(IGR)
327. 17 FCRMAT(1736,'*****',A8,'*****')
328. IF(ITIE.NE.0) KTY=KTIE
329. IF(ITIE.NE.C) WRITE(6,18) TTLT(ITIE)
330. 18 FCRMAT(1730,'*****TIE ',A8,' TIE*****')
331. XXX=INVT
332. SSS=ISC(IGR)
333. SSS=SSS/XXX
334. WRITE(6,1005)
335. C
336. C PUNCH SUMMARY CAPC FOR EACH SPECTRUM
337. C
338. WRITE(7,19) IS(1),IS(2),IS(3),TTLT(IGR),SSS,OS,KTY
339. 19 FCRMAT(1213,16,2X,A8,2X,FR.4,2X,FR.4,2X,A4)
340. DO 640 I=1,NT
341. SSS=0
342. DC 641 J=1,1P
343. SSS=SSS+(X(NSEL(J,NNN))-X(NSEL(J,AAA),AAA,I))**2
344. 640 WRITE(6,24) TTLT(I),SSS
345. 24 FCRMAT(11,DISTANCE FROM GRUP ',A8,' MEAN IS ',F12.7)
346. GC TO 510
347. C
348. C EVALUATE AND REPORT CONTROL EPOCHS
349. C
350. 502 WRITE(6,20)
351. 20 FCRMAT(11,'***END OF DATA ON LOGICAL 12***')
352. RETURN
353. 906 WRITE(6,1002) TTLT(NT)
354. 1003 FCRMAT(11,'***END OF FILE ON LOGICAL 12***')
355. 11 TRAINING GROUP ',A8,' HAS FEWER MEMBERS')
356. IF(1(NT,1).EQ.0.OR.1(NT,2).EQ.0) NT=NT-1
357. REWIND 12
358. GC TO 100
359. 905 WRITE(6,20)
360. REWIND 12
361. GC TO 503
362. 503 WRITE(6,21)
363. 21 FCRMAT(11,'***EPOCHS IN READING***')
364. RETURN
365. 510 WRITE(6,201) AAA

```

U1-19

VII. Program Examples

The following pages list the input and outputs of programs involved in a normal process run.

First, TASK was run on the raw data, which had previously been read off of tape and stored on disk for easier access. Three datasets were concatenated together for input, and a fourth contained the pre-averaged blackbody input. Output was of two types and in two forms on four different datasets.

Spectra were read in off the input datasets, reduced to 89 data points, smoothed, and ratioed to an average blackbody spectrum which had been averaged earlier. Then the spectra were normalized, clipped of 18 points on either side, outputted to a temporary dataset, banded together by every three points, and outputted in standard and card formats to different output datasets for the two later programs to work on.

Then the temporary dataset was rewound, the partially processed data read in again, banded together by 5 points each, and outputted on two different output datasets.

After this pre-processing, NEAREST NEIGHBOR and BMD07M, the two identification programs, were run on the data. Both these used as the two training groups, andesite and granite. The unknown group in the BMD program is labeled by a negative group size.

The input data for BMD was assembled by the Stanford text

editing system, WYLBUR.

A.) Housekeeping

See Computer Print-Out Text VII-1 to VII-5.

B.) Plotting and Banding

See Computer Print-Out Text VII-6 to VII-9.

C.) NEAREST NEIGHBOR Run

See Computer Print-Out Text VII-10 to VII-32.

D.) BMD07M Run

See Computer Print-Out Text VII-33 to VII-42.

VII-1

- 40 -

TASK 1 -- INPUT DOWN SEQ DISK 8 0 0 0 99 99 99999 0 = PFEAD 90 = NPTS. 1 = RPTS.
 TASK 2 -- CLIP NPNT 1 = LEFT CLIP 0 = RIGHT CLIP 89 = NPTS.
 TASK 3 -- SMOOTH NPNT -9 = SMOOTHING 89 = NPTS.
 TASK 4 -- RATIO NPNT 89 = NPTS.
 AVER DOWN SEQ DISK 12 0 0 0 0 0 0 0 0 89 = NPTS.
 NPNT DISK 0
 TASK 5 -- MORE NPNT 89 = NPTS.
 TASK 6 -- CLIP NPNT 18 = LEFT CLIP 18 = RIGHT CLIP 53 = NPTS.
 TASK 7 -- OUTPUT PTIM DISK 13 53 = NPTS. 1 = RPTS.
 TASK 8 -- RATIO NPNT
 PEPT 3 = DIVISOR 53 = NPTS.
 TASK 9 -- PLOT NPNT AVER 6 0 0 0 99 99 99999 0.0 = XMIN 0.0 = XMAX 18 = NPTS.
 TASK 10 -- OUTPUT PTIM DISK 9 18 = NPTS. 1 = RPTS.
 TASK 11 -- OUTPUT NPNT CARD 10 18 = NPTS. 1 = RPTS.
 FORMAT: (B24/213,16,21/8F10.4)/2F10.4,I10)

19 53 3169	4E01 0F01 1300 0000 0444 45C1 000E 0001	1
19 53 3169	4E01 0F01 1300 0000 0444 45C1 001E 0001	2
19 53 3473	4E01 0F01 1300 0000 0444 46F1 000E 0001	3
19 53 3473	4E01 0F01 1300 0000 0444 46F1 001E 0001	4
19 53 3777	4E01 0F01 1300 0000 0444 4821 000E 0001	5
19 53 3777	4E01 0F01 1300 0000 0444 4821 001E 0001	6
19 53 4081	4E01 0F01 1300 0000 0444 4951 000E 0001	7
19 53 4081	4E01 0F01 1300 0000 0444 4951 001E 0001	8
19 53 4383	4E01 0F01 1300 0000 0444 4A7F 000E 0001	9
19 53 4383	4E01 0F01 1300 0000 0444 4A7F 001E 0001	10
19 53 4686	4E01 0F01 1300 0000 0444 4BAE 000E 0001	11
19 53 4686	4E01 0F01 1300 0000 0444 4BAE 001E 0001	12
19 53 4991	4E01 0F01 1300 0000 0444 4CDF 000E 0001	13
19 53 4991	4E01 0F01 1300 0000 0444 4CDF 001E 0001	14
19 53 5293	4E01 0F01 1300 0000 0444 4E0D 000E 0001	15
19 53 5293	4E01 0F01 1300 0000 0444 4E0D 001E 0001	16
19 53 5596	4E01 0F01 1300 0000 0444 4F3C 000E 0001	17
19 53 5596	4E01 0F01 1300 0000 0444 4F3C 001E 0001	18
19 53 5899	4E01 0F01 1300 0000 0444 506B 000E 0001	19
19 53 5899	4E01 0F01 1300 0000 0444 506B 001E 0001	20
19 53 6205	4E01 0F01 1300 0000 0444 519D 000E 0001	21
19 53 6205	4E01 0F01 1300 0000 0444 519D 001E 0001	22
19 53 6507	4E01 0F01 1300 0000 0444 52C9 000E 0001	23
19 53 6507	4E01 0F01 1300 0000 0444 52C9 001E 0001	24
19 53 6811	4E01 0F01 1300 0000 0444 53FB 000E 0001	25
19 53 6811	4E01 0F01 1300 0000 0444 53FB 001E 0001	26
19 53 7115	4E01 0F01 1300 0000 0444 552A 000E 0001	27
19 53 7115	4E01 0F01 1300 0000 0444 552A 001E 0001	28
19 53 7417	4E01 0F01 1300 0000 0444 5659 000E 0001	29
19 53 7417	4E01 0F01 1300 0000 0444 5659 001E 0001	30

19 53 7720	4E01 0F01 1300 0000 0444 5788 000E 0001	31
19 53 7720	4E01 0F01 1300 0000 0444 5788 001E 0001	32
19 53 8024	4E01 0F01 1300 0000 0444 5888 000E 0001	33
19 53 8024	4E01 0F01 1300 0000 0444 5888 001E 0001	34
19 53 8328	4E01 0F01 1300 0000 0444 59E8 000E 0001	35
19 53 8328	4E01 0F01 1300 0000 0444 59E8 001E 0001	36
19 53 8630	4E01 0F01 1300 0000 0444 5B16 000E 0001	37
19 53 8630	4E01 0F01 1300 0000 0444 5B16 001E 0001	38
19 53 8935	4E01 0F01 1300 0000 0444 5C47 000E 0001	39
19 53 8935	4E01 0F01 1300 0000 0444 5C47 001E 0001	40
19 53 9239	4E01 0F01 1300 0000 0444 5D77 000E 0001	41
19 53 9239	4E01 0F01 1300 0000 0444 5D77 001E 0001	42
19 53 9541	4E01 0F01 1300 0000 0444 5EA5 000E 0001	43
19 53 9541	4E01 0F01 1300 0000 0444 5EA5 001E 0001	44
19 53 9845	4E01 0F01 1300 0000 0444 5FD5 000E 0001	45
19 53 9845	4E01 0F01 1300 0000 0444 5FD5 001E 0001	46
19 53 10148	4E01 0F01 1300 0000 0444 6104 000E 0001	47
19 53 10148	4E01 0F01 1300 0000 0444 6104 001E 0001	48
19 53 10452	4E01 0F01 1300 0000 0444 6234 000E 0001	49
19 53 10452	4E01 0F01 1300 0000 0444 6234 001E 0001	50
19 53 10754	4E01 0F01 1300 0000 0444 6362 000E 0001	51
19 53 10754	4E01 0F01 1300 0000 0444 6362 001E 0001	52
19 53 11058	4E01 0F01 1300 0000 0444 6492 000E 0001	53
19 53 11058	4E01 0F01 1300 0000 0444 6492 001E 0001	54
19 53 11361	4E01 0F01 1300 0000 0444 65C1 000E 0001	55
19 53 11361	4E01 0F01 1300 0000 0444 65C1 001E 0001	56
19 53 11666	4E01 0F01 1300 0000 0444 66F2 000E 0001	57
19 53 11666	4E01 0F01 1300 0000 0444 66F2 001E 0001	58
19 53 11969	4E01 0F01 1300 0000 0444 6821 000E 0001	59
19 53 11969	4E01 0F01 1300 0000 0444 6821 001E 0001	60

19 53 12272	4E01 0F01 1300 0000 0444 6950 000E 0001	61
19 53 12272	4E01 0F01 1300 0000 0444 6950 001E 0001	62
19 53 12576	4E01 0F01 1300 0000 0444 6A80 000E 0001	63
19 53 12576	4E01 0F01 1300 0000 0444 6A80 001E 0001	64
19 53 12879	4E01 0F01 1300 0000 0444 6BAF 000E 0001	65
19 53 12879	4E01 0F01 1300 0000 0444 6BAF 001E 0001	66
19 53 13182	4E01 0F01 1300 0000 0444 6CDE 000E 0001	67
19 53 13182	4E01 0F01 1300 0000 0444 6CDE 001E 0001	68
19 53 13485	4E01 0F01 1300 0000 0444 6E0D 000E 0001	69
19 53 13485	4E01 0F01 1300 0000 0444 6E0D 001E 0001	70
19 53 13789	4E01 0F01 1300 0000 0444 6F3D 000E 0001	71
19 53 13789	4E01 0F01 1300 0000 0444 6F3D 001E 0001	72
19 53 14397	4E01 0F01 1300 0000 0444 719D 000E 0001	73
19 53 14397	4E01 0F01 1300 0000 0444 719D 001E 0001	74
19 53 14701	4E01 0F01 1300 0000 0444 72CD 000E 0001	75
19 53 14701	4E01 0F01 1300 0000 0444 72CD 001E 0001	76
19 53 15003	4E01 0F01 1300 0000 0444 73FB 000E 0001	77
19 53 15003	4E01 0F01 1300 0000 0444 73FB 001E 0001	78
19 53 15610	4E01 0F01 1300 0000 0444 765A 000E 0001	79
19 53 15610	4E01 0F01 1300 0000 0444 765A 001E 0001	80
19 53 15914	4E01 0F01 1300 0000 0444 778A 000E 0001	81
19 53 15914	4E01 0F01 1300 0000 0444 778A 001E 0001	82
20 1 29267	4E01 1201 1300 0000 044B FER3 000E 0001	83
20 1 29267	4E01 1201 1300 0000 044B FER3 001E 0001	84
20 1 29572	4F01 1201 1300 0000 044B FFE4 000E 0001	85
20 1 29572	4F01 1201 1300 0000 044B FFE4 001E 0001	86
20 1 29875	4E01 1201 1300 0000 044C 0113 000E 0001	87
20 1 29875	4E01 1201 1300 0000 044C 0113 001E 0001	88
20 1 30177	4F01 1201 1300 0000 044C 0241 000E 0001	89
20 1 30177	4E01 1201 1300 0000 044C 0241 001E 0001	90
20 1 30480	4E01 1201 1300 0000 044C 0370 000E 0001	91

U11-4

20	1	30480	4E01 1201 1300 0000 044C 0370 001E 0001	92
20	1	30785	4E01 1201 1300 0000 044C 04A1 000E 0001	93
20	1	30785	4E01 1201 1300 0000 044C 04A1 001E 0001	94
20	1	31087	4E01 1201 1300 0000 044C 05CF 000E 0001	95
20	1	31087	4E01 1201 1300 0000 044C 05CF 001E 0001	96
20	1	31390	4E01 1201 1300 0000 044C 06FE 000E 0001	97
20	1	31390	4E01 1201 1300 0000 044C 06FE 001E 0001	98
20	1	31693	4E01 1201 1300 0000 044C 082D 000E 0001	99
20	1	31693	4E01 1201 1300 0000 044C 082D 001E 0001	100
20	1	31999	4E01 1201 1300 0000 044C 095F 000E 0001	101
20	1	31999	4E01 1201 1300 0000 044C 095F 001E 0001	102
20	1	33818	4E01 1201 1300 0000 044C 107A 000E 0001	103
20	1	33818	4E01 1201 1300 0000 044C 107A 001E 0001	104
20	1	34122	4E01 1201 1300 0000 044C 11AA 000E 0001	105
20	1	34122	4E01 1201 1300 0000 044C 11AA 001E 0001	106
20	1	34424	4E01 1201 1300 0000 044C 12D8 000E 0001	107
20	1	34424	4E01 1201 1300 0000 044C 12D8 001E 0001	108
20	1	34727	4E01 1201 1300 0000 044C 1407 000E 0001	109
20	1	34727	4E01 1201 1300 0000 044C 1407 001E 0001	110
20	1	35033	4E01 1201 1300 0000 044C 1539 000E 0001	111
20	1	35033	4E01 1201 1300 0000 044C 1539 001E 0001	112
20	1	35336	4E01 1201 1300 0000 044C 1668 000E 0001	113
20	1	35336	4E01 1201 1300 0000 044C 1668 001E 0001	114
20	1	35639	4E01 1201 1300 0000 044C 1797 000E 0001	115
20	1	35639	4E01 1201 1300 0000 044C 1797 001E 0001	116
20	1	35942	4E01 1201 1300 0000 044C 18C6 000E 0001	117
20	1	35942	4E01 1201 1300 0000 044C 18C6 001E 0001	118
20	1	36246	4E01 1201 1300 0000 044C 19F6 000E 0001	119
20	1	36246	4E01 1201 1300 0000 044C 19F6 001E 0001	120
20	1	36549	4E01 1201 1300 0000 044C 1825 000E 0001	121
20	1	36549	4E01 1201 1300 0000 044C 1825 001E 0001	122

MISSION 78 DAY 1 53 POINTS 3 AND 5 POINT BANDING

VII-5

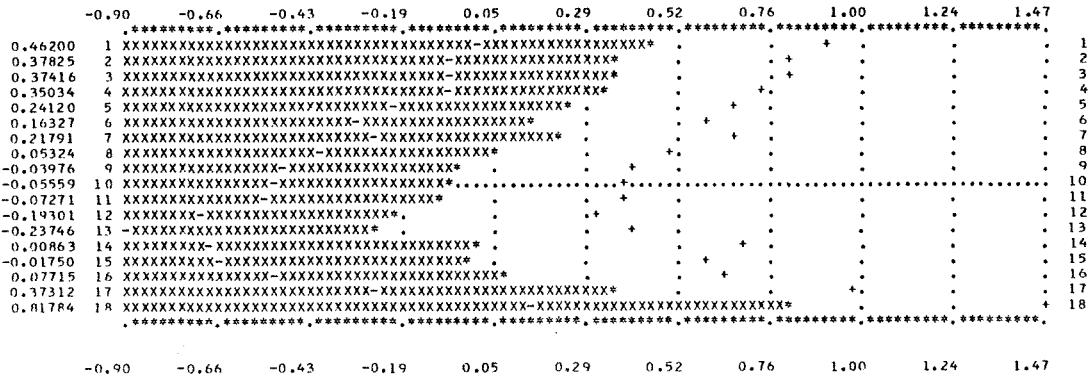
NUMBER OF SPECTRA IS

61

COUNTER	AVE	STD DEV	RELATIVE ERROR
1	0.4620	0.4619	0.12523121
2	0.3782	0.4314	0.14603031
3	0.3742	0.4389	0.15019232
4	0.3503	0.4028	0.14719367
5	0.2412	0.4344	0.23056555
6	0.1633	0.4474	0.35083085
7	0.2179	0.4557	0.26774651
8	0.0532	0.4523	1.08779144
9	-0.0398	0.4499	-1.44860172
10	-0.0556	0.4555	-1.04919624
11	-0.0727	0.4679	-0.82396668
12	-0.1930	0.5047	-0.33478868
13	-0.2375	0.6630	-0.35747927
14	0.0086	0.6847	10.15949631
15	-0.0175	0.6340	-4.63830185
16	0.0772	0.5883	0.97632068
17	0.3731	0.6171	0.21175091
18	0.8178	0.6549	0.10253221
MEAN	0.1612	0.5130	0.29463023

PLOT OF MEANS AND STANDARD DEVIATION OF AVERAGES FROM TIMES
0 0 0 TO 99 99 99999
MISSION 78 DAY 1 53 POINTS 3 AND 5 POINT BANDING
NUMBER OF SPECTRA IS 61

XMAX IS 1.4728 XMIN IS -0.9004 DELTA IS 0.0237
HIGHEST X IS 18 LOWEST X IS 13



END OF TASK CYCLE, NUMBER OF SPECTRA PROCESSED = 61
 NUMBER OF RECORDS READ = 120
 NUMBER OF RECORDS WRITTEN ON DISK = 122
 NUMBER OF RECORDS WRITTEN ON CARDS = 61

VII-6

SPECTRUM PREPARATION CYCLE START. PARAMETERS ARE: NDMP IBM 53 = NPTS. 1 = RPTS.
 5 POINT BANDING

TASK -- REWIND
 LOGICALS ARE: 13

TASK 1 -- INPUT DOWN SEOL DISK 13 0 0 0 99 99 99999 0 = PREAD 53 = NPTS. 1 = RPTS.

TASK 2 -- BAND NPNT
 REPT 5 = DIVISOR 53 = NPTS.

TASK 3 -- OUTPUT PTIM DISK 14 11 = NPTS. 1 = RPTS.

TASK 4 -- OUTPUT NPNT CARD 15 11 = NPTS. 1 = RPTS.
 FORMAT: (8Z4/2I3,16/8F10.4/3F10.4,110)

TASK 5 -- PLOT NPNT AVER 6 0 0 0 99 99 99999 0.0 = XMIN 0.0 = XMAX 11 = NPTS.

- 45 -

011-7

19 53 3169	4E01 0F01 1300 0000 0444 45C1 001F 0001	1
19 53 3473	4FC1 0F01 1300 0000 0444 46F1 001E 0001	2
19 53 3777	4F01 0F01 1300 0000 0444 4821 001E 0001	3
19 53 4081	4E01 0F01 1300 0000 0444 4951 001F 0001	4
19 53 4383	4E01 0F01 1300 0000 0444 4A7F 001E 0001	5
19 53 4686	4E01 0F01 1300 0000 0444 4BAF 001E 0001	6
19 53 4991	4E01 0F01 1300 0000 0444 4CDF 001F 0001	7
19 53 5293	4E01 0F01 1300 0000 0444 4E0D 001E 0001	8
19 53 5596	4E01 0F01 1300 0000 0444 4F3C 001E 0001	9
19 53 5899	4E01 0F01 1300 0000 0444 5068 001E 0001	10
19 53 6205	4E01 0F01 1300 0000 0444 519D 001E 0001	11
19 53 6507	4E01 0F01 1300 0000 0444 52C8 001F 0001	12
19 53 6811	4E01 0F01 1300 0000 0444 53FB 001E 0001	13
19 53 7115	4E01 0F01 1300 0000 0444 552B 001E 0001	14
19 53 7417	4E01 0F01 1300 0000 0444 5659 001E 0001	15
19 53 7720	4E01 0F01 1300 0000 0444 5788 001E 0001	16
19 53 8024	4E01 0F01 1300 0000 0444 5888 001E 0001	17
19 53 8328	4E01 0F01 1300 0000 0444 59E8 001E 0001	18
19 53 8630	4E01 0F01 1300 0000 0444 5B16 001E 0001	19
19 53 8935	4E01 0F01 1300 0000 0444 5C47 001E 0001	20
19 53 9239	4E01 0F01 1300 0000 0444 5D77 001E 0001	21
19 53 9541	4E01 0F01 1300 0000 0444 5EA5 001E 0001	22
19 53 9845	4E01 0F01 1300 0000 0444 5FD5 001E 0001	23
19 53 10148	4E01 0F01 1300 0000 0444 6104 001E 0001	24
19 53 10452	4E01 0F01 1300 0000 0444 6234 001E 0001	25
19 53 10754	4E01 0F01 1300 0000 0444 6362 001E 0001	26
19 53 11058	4E01 0F01 1300 0000 0444 6492 001E 0001	27
19 53 11361	4E01 0F01 1300 0000 0444 65C1 001F 0001	28
19 53 11666	4E01 0F01 1300 0000 0444 66F2 001E 0001	29
19 53 11969	4E01 0F01 1300 0000 0444 6821 001E 0001	30

19 53 12272	4E01 0F01 1300 0000 0444 6950 001E 0001	31
19 53 12576	4E01 0F01 1300 0000 0444 6A80 001E 0001	32
19 53 12879	4E01 0F01 1300 0000 0444 6BAF 001E 0001	33
19 53 13182	4E01 0F01 1300 0000 0444 6CDE 001F 0001	34
19 53 13485	4E01 0F01 1300 0000 0444 6E0D 001E 0001	35
19 53 13789	4E01 0F01 1300 0000 0444 6F3D 001E 0001	36
19 53 14397	4E01 0F01 1300 0000 0444 719D 001E 0001	37
19 53 14701	4E01 0F01 1300 0000 0444 72CD 001E 0001	38
19 53 15003	4E01 0F01 1300 0000 0444 73FB 001E 0001	39
19 53 15610	4E01 0F01 1300 0000 0444 765A 001E 0001	40
19 53 15914	4E01 0F01 1300 0000 0444 778A 001E 0001	41
20 1 29267	4E01 1201 1300 0000 044B FER3 001E 0001	42
20 1 29572	4E01 1201 1300 0000 044B FFE4 001E 0001	43
20 1 29875	4E01 1201 1300 0000 044C 0113 001E 0C01	44
20 1 30177	4E01 1201 1300 0000 044C 0241 001E 0001	45
20 1 30480	4E01 1201 1300 0000 044C 0370 001F 0001	46
20 1 30785	4E01 1201 1300 0000 044C 04A1 001E 0001	47
20 1 31087	4F01 1201 1300 0000 044C 05CF 001E 0001	48
20 1 31390	4E01 1201 1300 0000 044C 06FE 001E 0001	49
20 1 31693	4E01 1201 1300 0000 044C 082D 001E 0001	50
20 1 31999	4E01 1201 1300 0000 044C 095F 001E 0001	51
20 1 33818	4E01 1201 1300 0000 044C 107A 001E 0001	52
20 1 34122	4F01 1201 1300 0000 044C 11AA 001F 0001	53
20 1 34424	4F01 1201 1300 0000 044C 1208 001E 0001	54
20 1 34727	4E01 1201 1300 0000 044C 1407 001F 0001	55
20 1 35033	4F01 1201 1300 0000 044C 1539 001E 0001	56
20 1 35336	4E01 1201 1300 0000 044C 1668 001E 0001	57
20 1 35639	4E01 1201 1300 0000 044C 1797 001E 0001	58
20 1 35942	4F01 1201 1300 0000 044C 18C6 001E 0001	59
20 1 36246	4E01 1201 1300 0000 044C 19F6 001E 0001	60

-46-

5 POINT BANDING

NUMBER OF SPECTRA IS

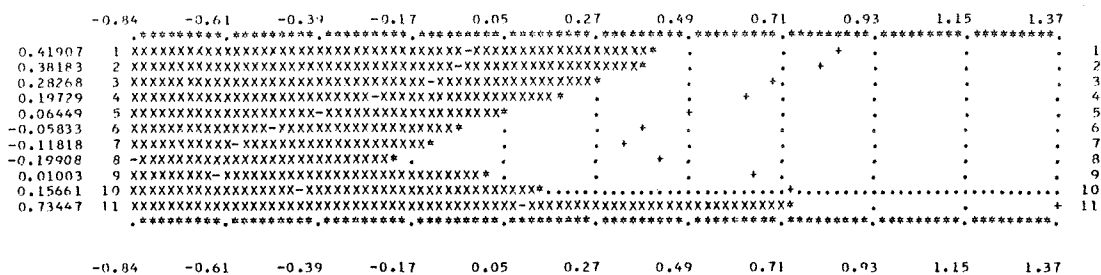
61

COUNTER	AVE	STD DEV	RELATIVE ERROR
1	0.4191	0.4380	0.13381279
2	0.3818	0.4283	0.14360881
3	0.2827	0.4111	0.18621463
4	0.1973	0.4449	0.28873807
5	0.0645	0.4468	0.88705057
6	-0.0583	0.4438	-0.97408921
7	-0.1182	0.4683	-0.50733346
8	-0.1991	0.6361	-0.40907168
9	0.0100	0.6452	8.23297119
10	0.1566	0.5918	0.48379689
11	0.7345	0.6386	0.11133134

MEAN	0.1701	0.5084	0.77972984
------	--------	--------	------------

-47-

VII-9



```

END OF TASK CYCLE, NUMBER OF SPECTRA PROCESSED =      61
NUMBER OF RECORDS READ =      61
NUMBER OF RECORDS WRITTEN ON DISK =      61
NUMBER OF RECORDS WRITTEN ON CARDS =      61

```

U11-10

STANFORD NEAREST NEIGHBOR PROCESSOR
NEAREST NEIGHBOR ON 11 POINT 5 BANDED LOW ALT BKHK UNKNOWN

```
NPTS= 11
IP= 8
IVOT= 5
ISM= 0
ITR= 10ICLIP= 0
```

- 49 -

U11-11

```

BEGINNING STEP TRAIN GRANITE 2 TIMES: 20 1 33817 20 1 36549 NEW
TRAIN GRANITE 1 20 1 33818 LOW
TRAIN GRANITE 2 20 1 34122 LOW
TRAIN GRANITE 3 20 1 34424 LOW
TRAIN GRANITE 4 20 1 34727 LOW
TRAIN GRANITE 5 20 1 35033 LOW
TRAIN GRANITE 6 20 1 35336 LOW
TRAIN GRANITE 7 20 1 35639 LOW
TRAIN GRANITE 8 20 1 35942 LOW
TRAIN GRANITE 9 20 1 36246 LOW
TRAIN GRANITE 10 20 1 36549 LOW
*****ELAPSED TIME IS 5.050 SECONDS*****
**COVAR CALL COMPLETED..STARTING PROCESSING..

```

SELECTION COMPLETED NSFL=

11
9
10
8
1
2
3
4

 $\Gamma =$

10
10
0
0
0
0
0
0
0

0
0
0
0
0
0
0
0

LOW WEIGHTING FACTORS ARE:

1.000	0.959	0.956	0.904	0.876	0.818	0.805	0.795
-------	-------	-------	-------	-------	-------	-------	-------

SELECTION COMPLETED SELECTION COMPLETED SELECTION COMPLETED SELECTION COMPLETED SELECTION COMPLETED

VII-12

GROUP MEANS ANDESITE LOW

0.3932663 0.2776964 0.1655771 0.1216741 0.0778306 0.0444211 -0.0192370 0.1313328 0.4020879 0.5293124
0.9249996

GROUP MEANS GRANITE LOW

0.4660694 0.2993004 0.2388952 0.2368486 0.1658325 -0.0000843 -0.1574887 -0.4100167 -0.1568868 0.1845412
0.9746788

ELAPSED TIME IS 0.230 SECONDS

DISTANCES BETWEEN GROUP MEANS

DIST FROM ANDESITE LOW TO ANDESITE LOW IS 0.0
DIST FROM GRANITE LOW TO ANDESITE LOW IS 0.86675E 00
DIST FROM ANDESITE LOW TO GRANITE LOW IS 0.86675E 00
DIST FROM GRANITE LOW TO GRANITE LOW IS 0.0

-51-

VII-13

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 3169LOW

WEIGHTED X VALUES:
 11) 1.0787 9) 0.4511 10) 0.5211 8) 0.2889 1) 0.3072
 2) 0.2845 3) 0.1611 4) 0.0191

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	9	0.71427E-01	
ANDESITE	3	0.77378E-01	
ANDESITE	1	0.10316E 00	
ANDESITE	4	0.27635E 00	
ANDESITE	10	0.31768E 00	

ANDESITE

DISTANCE FROM GROUP ANDESITE MEAN IS 0.0689072
 DISTANCE FROM GROUP GRANITE MEAN IS 1.0611620

U1-14

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 3473LOW

WEIGHTED X VALUES:
 11) 0.0197 9) -0.7860 10) -0.6012 8) -0.6236 1) 0.7992
 2) 0.8549 3) 0.5772 4) 0.3408

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 5 0.65890E 00
 GRANITE 2 0.11116E 01
 GRANITE 6 0.14510E 01
 GRANITE 9 0.19524E 01
 GRANITE 8 0.20591E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 4.7943125
 DISTANCE FROM GROUP GRANITE MEAN IS 2.5155792

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 3777LOW

WEIGHTED X VALUES:
 11) 1.4850 9) 0.6790 10) 0.9038 8) 0.3697 1) 0.0088
 2) -0.0472 3) -0.1983 4) -0.1212

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 8 0.30561E 00
 ANDESITE 7 0.55075E 00
 ANDESITE 1 0.62599E 00
 ANDESITE 9 0.62659E 00
 ANDESITE 3 0.71176E 00

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 1.0321903
 DISTANCE FROM GROUP GRANITE MEAN IS 2.7330799

-53-

VII-15

NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 4081LOW

WEIGHTED X VALUES:
11) -0.1926 9) 0.0809 10) -0.4848 8) 0.1568 1) 0.4482
2) 0.4789 3) 0.5139 4) 0.5508

.....
GROUP NUMBER DISTANCE DECISION
.....
GRANITE 6 0.49197E 00
GRANITE 2 0.75814E 00
ANDESITE 2 0.94540E 00
GRANITE 9 0.16673E 01
ANDESITE 6 0.19515E 01

GRANITE **

.....
DISTANCE FROM GROUP ANDESITE MEAN IS 2.7304201
DISTANCE FROM GROUP GRANITE MEAN IS 2.3953609

NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 4383LOW

WEIGHTED X VALUES:
11) 1.0776 9) 0.0881 10) 0.2752 8) -0.1394 1) 0.4878
2) 0.3752 3) 0.1399 4) -0.1926

.....
GROUP NUMBER DISTANCE DECISION
.....
ANDESITE 4 0.17500E 00
GRANITE 10 0.27376E 00
GRANITE 1 0.34547E 00
GRANITE 9 0.36681E 00
ANDESITE 9 0.44745E 00

GRANITE **

.....
DISTANCE FROM GROUP ANDESITE MEAN IS 0.3776101
DISTANCE FROM GROUP GRANITE MEAN IS 0.3525515

-54-

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 4686LOW

WEIGHTED X VALUES:
 11) 0.4010 9) -0.9299 10) -0.3932 8) -0.9652 1) 0.7508
 2) 0.5003 3) 0.0962 4) -0.2228

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 5 0.60416E 00
 GRANITE 9 0.15310E 01
 GRANITE 10 0.16781E 01
 GRANITE 8 0.16784E 01
 GRANITE 4 0.17385E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 4.4031048
 DISTANCE FROM GROUP GRANITE MEAN IS 1.9218254

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 4991LOW

WEIGHTED X VALUES:
 11) 1.3900 9) 0.9638 10) 0.8206 8) 0.5116 1) 0.0589
 2) -0.0606 3) 0.0267 4) 0.0310

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 8 0.21214E 00
 ANDESITE 7 0.33468E 00
 ANDESITE 3 0.55706E 00
 ANDESITE 1 0.56489E 00
 ANDESITE 9 0.70938E 00

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 1.0148602
 DISTANCE FROM GROUP GRANITE MEAN IS 3.0649614

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 5293LOW

WEIGHTED X VALUES:
 11) 1.3904 9) -0.0160 10) 0.3640 8) -0.1299 1) 0.2111
 2) 0.2788 3) 0.1935 4) -0.1010

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 4 0.34184E 00
 GRANITE 1 0.45249E 00
 ANDESITE 9 0.46796E 00
 GRANITE 7 0.47459E 00
 GRANITE 10 0.50694E 00

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 0.5705184
 DISTANCE FROM GROUP GRANITE MEAN IS 0.4850007

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 5596LOW

WEIGHTED X VALUES:
 11) -0.2214 9) -0.7609 10) -0.7375 8) -0.6545 1) 1.0393
 2) 0.9090 3) 0.6437 4) 0.4357

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 5 0.11677E 01
 GRANITE 2 0.16588E 01
 GRANITE 6 0.16762E 01
 GRANITE 9 0.28572E 01
 GRANITE 8 0.31314E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 6.0322104
 DISTANCE FROM GROUP GRANITE MEAN IS 3.6090994

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 5899LOW

WEIGHTED X VALUES:
 11) 0.0141 9) -0.0734 10) -0.2617 8) 0.0165 1) 0.0764
 2) 0.5106 3) 0.9953 4) 1.2632

.....
 GROUP NUMBER DISTANCE DECISION

 GRANITE 2 0.73758E 00
 GRANITE 6 0.97256E 00
 ANDESITE 2 0.18242E 01
 GRANITE 9 0.30775E 01
 GRANITE 8 0.32869E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 3.8407640
 DISTANCE FROM GROUP GRANITE MEAN IS 3.1326933

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 6205LOW

WEIGHTED X VALUES:
 11) 1.4704 9) 1.0160 10) 0.9962 8) 0.9780 1) -0.6347
 2) -0.4746 3) -0.4921 4) -0.5068

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 8 0.90489E 00
 ANDESITE 7 0.26304E 01
 ANDESITE 3 0.29714E 01
 ANDESITE 9 0.31372E 01
 ANDESITE 1 0.32949E 01

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 4.0592022
 DISTANCE FROM GROUP GRANITE MEAN IS 7.1045246

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 6507LOW

WEIGHTED X VALUES:
 11) 0.3682 9) 0.4272 10) 0.0616 8) 0.3676 1) 0.7193
 2) 0.5560 3) 0.4143 4) 0.2501

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	2	0.13213E 00	
ANDESITE	6	0.58244E 00	
GRANITE	6	0.80300E 00	
ANDESITE	4	0.94166E 00	
GRANITE	9	0.98192E 00	

ANDESITE**

DISTANCE FROM GROUP ANDESITE MEAN IS 0.8473530
 DISTANCE FROM GROUP GRANITE MEAN IS 1.4897614

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 6811LOW

WEIGHTED X VALUES:
 11) 0.5149 9) 0.7631 10) 0.2709 8) 0.5855 1) 0.0864
 2) 0.1464 3) 0.1737 4) 0.3097

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	3	0.37553E 00	
ANDESITE	2	0.74058E 00	
ANDESITE	9	0.79233E 00	
ANDESITE	7	0.83764E 00	
ANDESITE	1	0.89662E 00	

ANDESITE**

DISTANCE FROM GROUP ANDESITE MEAN IS 0.7183629
 DISTANCE FROM GROUP GRANITE MEAN IS 2.2334337

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 7115LOW

WEIGHTED X VALUES:
 11) 0.0044 9) 0.1066 10) 0.0039 8) 0.0050 1) 0.4857
 2) 0.3662 3) 0.4437 4) 0.3793

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 2 0.37832E 00
 GRANITE 6 0.47047E 00
 GRANITE 2 0.55403E 00
 ANDESITE 6 0.80301E 00
 GRANITE 9 0.82750E 00

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 1.3869219
 DISTANCE FROM GROUP GRANITE MEAN IS 1.2828131

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 7417LOW

WEIGHTED X VALUES:
 11) 0.3289 9) -0.6030 10) -0.5412 8) -0.6424 1) 0.8754
 2) 0.8616 3) 0.6119 4) 0.3390

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 5 0.36812E 00
 GRANITE 2 0.99338E 00
 GRANITE 6 0.12211E 01
 GRANITE 8 0.13888E 01
 GRANITE 9 0.14506E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 3.9299088
 DISTANCE FROM GROUP GRANITE MEAN IS 1.8300037

NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 7720LOW

WEIGHTED X VALUES:
 11) 0.8747 9) 0.1597 10) 0.3644 8) 0.0056 1) 0.2114
 2) 0.5523 3) 0.5817 4) 0.6276

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	2	0.42406E 00	
GRANITE	5	0.59860E 00	
ANDESITE	4	0.81713F 00	
ANDESITE	10	0.82709E 00	
GRANITE	8	0.90054F 00	

ANDESITE**

DISTANCE FROM GROUP ANDESITE MEAN IS 0.6417823
 DISTANCE FROM GROUP GRANITE MEAN IS 0.7143880

NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 8024LOW

WEIGHTED X VALUES:
 11) -0.1705 9) -0.7968 10) -0.7229 8) -0.8626 1) 0.8255
 2) 0.6282 3) 0.4626 4) 0.2016

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	5	0.94258E 00	
GRANITE	2	0.16945E 01	
GRANITE	6	0.21742E 01	
GRANITE	9	0.23734E 01	
GRANITE	8	0.27616F 01	

GRANITE **

DISTANCE FROM GROUP ANDESITE MEAN IS 5.5979261
 DISTANCE FROM GROUP GRANITE MEAN IS 3.0380001

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 8328LOW

WEIGHTED X VALUES:
 11) 1.3367 9) 1.1012 10) 1.0667 8) 0.9332 1) -0.4044
 2) -0.1831 3) -0.0965 4) -0.1143

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	8	0.23151E 00	
ANDESITE	7	0.12243E 01	
ANDESITE	3	0.16416E 01	
ANDESITE	1	0.17838E 01	
ANDESITE	9	0.19891E 01	

 DISTANCE FROM GROUP ANDESITE MEAN IS 2.5609168
 DISTANCE FROM GROUP GRANITE MEAN IS 5.5189867

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 8630LOW

WEIGHTED X VALUES:
 11) 1.1984 9) 0.4939 10) 0.6739 8) -0.1317 1) 0.3094
 2) 0.1783 3) -0.0903 4) -0.2166

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	9	0.22688E 00	
ANDESITE	1	0.42278E 00	
ANDESITE	3	0.51839E 00	
GRANITE	1	0.53191E 00	
ANDESITE	10	0.54770E 00	

 DISTANCE FROM GROUP ANDESITE MEAN IS 0.3701146
 DISTANCE FROM GROUP GRANITE MEAN IS 1.1436291

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 8935LOW

WEIGHTED X VALUES:
 11) 1.9147 9) 0.5706 10) 0.5485 8) 0.4086 1) 0.2028
 2) 0.0994 3) 0.1617 4) 0.1733

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 3 0.20557E-01
 ANDESITE 9 0.13084E 00
 ANDESITE 1 0.14486E 00
 ANDESITE 7 0.24820E 00
 ANDESITE 4 0.50629E 00

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 0.1844921
 DISTANCE FROM GROUP GRANITE MEAN IS 1.4528351

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 9239LOW

WEIGHTED X VALUES:
 11) 1.3871 9) 0.5136 10) 0.5951 8) 0.2554 1) 0.2948
 2) 0.3250 3) 0.1904 4) 0.0801

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 1 0.12750E 00
 ANDESITE 9 0.22249E 00
 ANDESITE 3 0.26504E 00
 ANDESITE 10 0.28699E 00
 ANDESITE 7 0.52229E 00

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 0.2599865
 DISTANCE FROM GROUP GRANITE MEAN IS 1.2879848

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 9541L7W

WEIGHTED X VALUES:
 11) 1.3001 9) 0.7033 10) 0.8876 8) 0.2874 1) 0.1913
 2) 0.1440 3) -0.0061 4) -0.1433

.....

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	1	0.24143E 00	
ANDESITE	9	0.32959E 00	
ANDESITE	7	0.33170E 00	
ANDESITE	3	0.37075E 00	
ANDESITE	8	0.51706E 00	

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 0.5424773
 DISTANCE FROM GROUP GRANITE MEAN IS 2.1306705

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 9845L0W

WEIGHTED X VALUES:
 11) 1.0428 9) 0.3986 10) 0.6174 8) 0.0441 1) 0.6179
 2) 0.4251 3) 0.2383 4) -0.0677

.....

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	10	0.86441E-01	
ANDESITE	1	0.26559E 00	
ANDESITE	9	0.29985E 00	
ANDESITE	6	0.38383E 00	
ANDESITE	3	0.43430E 00	

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 0.1426240
 DISTANCE FROM GROUP GRANITE MEAN IS 0.8383213

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 10148LOW

WEIGHTED X VALUES:
 1) 0.9960 9) 0.5187 10) 0.5204 8) 0.3677 1) 0.2374
 2) 0.2315 3) 0.0836 4) -0.0166

GROUP NUMBER DISTANCE DECISION
 ANDESITE 3 0.32145E-01
 ANDESITE 9 0.83890E-01
 ANDESITE 1 0.15231E 00
 ANDESITE 4 0.33890E 00
 ANDESITE 7 0.40873E 00

ANDESITE**

DISTANCE FROM GROUP ANDESITE MEAN IS 0.1268762
 DISTANCE FROM GROUP GRANITE MEAN IS 1.3198004

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 10452LOW

WEIGHTED X VALUES:
 1) 1.6000 9) 0.8584 10) 0.9794 8) 0.5672 1) 0.0353
 2) 0.0010 3) -0.0983 4) -0.2075

GROUP NUMBER DISTANCE DECISION
 ANDESITE 8 0.15353E 00
 ANDESITE 7 0.71590E 00
 ANDESITE 1 0.84807E 00
 ANDESITE 3 0.98209E 00
 ANDESITE 9 0.10195E 01

ANDESITE**

DISTANCE FROM GROUP ANDESITE MEAN IS 1.4390135
 DISTANCE FROM GROUP GRANITE MEAN IS 3.5942011

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 10754LOW

WEIGHTED X VALUES:
 11) 1.2787 9) 0.6629 10) 0.8141 8) 0.3546 1) 0.3190
 2) 0.2647 3) 0.0967 4) 0.0259

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 1 0.85224E-01
 ANDESITE 3 0.23941E 00
 ANDESITE 7 0.24695E 00
 ANDESITE 9 0.27607E 00
 ANDESITE 10 0.39196E 00

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 0.3436835
 DISTANCE FROM GROUP GRANITE MEAN IS 1.8330479

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 11058LOW

WEIGHTED X VALUES:
 11) 0.7839 9) 0.5572 10) 0.4574 8) 0.4878 1) 0.2730
 2) 0.2209 3) 0.2050 4) 0.1718

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 3 0.57938E-01
 ANDESITE 9 0.26808E 00
 ANDESITE 1 0.28056E 00
 ANDESITE 7 0.44187E 00
 ANDESITE 4 0.48251E 00

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 0.1979795
 DISTANCE FROM GROUP GRANITE MEAN IS 1.4756746

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 11361LOW

WEIGHTED X VALUES:
 11) 0.6237 9) 0.1018 10) 0.1323 8) 0.1037 1) 0.5844
 2) 0.6663 3) 0.4229 4) 0.0648

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 2 0.20923E 00
 GRANITE 9 0.38294E 00
 ANDESITE 6 0.41164E 00
 ANDESITE 4 0.47134E 00
 GRANITE 8 0.60307E 00

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 0.5964222
 DISTANCE FROM GROUP GRANITE MEAN IS 0.6689354

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 11666LOW

WEIGHTED X VALUES:
 11) -0.8603 9) -0.6036 10) -0.4137 8) -0.5684 1) -0.1645
 2) -0.1319 3) -0.2627 4) -0.2237

.....
 GROUP NUMBER DISTANCE DECISION

 GRANITE 2 0.39219E 01
 GRANITE 9 0.40153E 01
 ANDESITE 5 0.43994E 01
 ANDESITE 6 0.48289E 01
 GRANITE 6 0.51866E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 6.3591042
 DISTANCE FROM GROUP GRANITE MEAN IS 4.9968576

U11-27

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 11969L7W

WEIGHTED X VALUES:
 11) -0.5923 9) -0.6158 10) -0.7774 8) -0.5698 1) -0.3694
 2) -0.2381 3) -0.2172 4) -0.2360

.....

GROUP	NUMBER	DISTANCE	DECISION
GRANITE	9	0.41609E 01	
GRANITE	2	0.41443E 01	
ANDESITE	5	0.47709E 01	
ANDESITE	4	0.51699E 01	
ANDESITE	6	0.53371E 01	

ANDESITE**

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 6.6592703
 DISTANCE FROM GROUP GRANITE MEAN IS 5.0350847

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 12272LOW

WEIGHTED X VALUES:
 11) -0.8180 9) -0.9006 10) -0.6828 8) -0.6154 1) -0.2118
 2) 0.0694 3) -0.1669 4) -0.2527

.....

GROUP	NUMBER	DISTANCE	DECISION
GRANITE	2	0.40764E 01	
ANDESITE	5	0.43480E 01	
GRANITE	9	0.44356E 01	
GRANITE	6	0.55077E 01	
GRANITE	8	0.57460E 01	

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 7.4221659
 DISTANCE FROM GROUP GRANITE MEAN IS 5.4781017

-67-

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 12576LOW

WEIGHTED X VALUES:
 11) 1.8259 9) -0.3569 10) 0.1990 8) -0.6444 1) -0.7480
 2) -0.4361 3) -0.2606 4) 0.0339

.....
 GROUP NUMREP DISTANCE DECISION

 GRANITE 7 0.11554E 01
 GRANITE 3 0.15057E 01
 GRANITE 5 0.25874E 01
 GRANITE 1 0.29762E 01
 ANDESITE 4 0.33092E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 4.0997238
 DISTANCE FROM GROUP GRANITE MEAN IS 3.1250610

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 12879LOW

WEIGHTED X VALUES:
 11) 0.8001 9) -1.2442 10) -0.8045 8) -1.5072 1) 0.5806
 2) 0.5388 3) 0.3399 4) 0.1580

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 5 0.18171E 01
 GRANITE 4 0.28525E 01
 GRANITE 8 0.30600E 01
 GRANITE 10 0.33206E 01
 GRANITE 3 0.34576E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 7.3248291
 DISTANCE FROM GROUP GRANITE MEAN IS 3.4817190

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 13182LOW

WEIGHTED X VALUES:
 11) 0.5450 9) -0.1435 10) 0.1725 8) -0.7828 1) 0.2388
 2) -0.2418 3) -0.1057 4) -0.1097

.....
 GROUP NUMBER DISTANCE DECISION

 GRANITE 3 0.93980E 00
 GRANITE 9 0.95666E 00
 GRANITE 1 0.10994E 01
 GRANITE 10 0.11422E 01
 GRANITE 5 0.12424E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 1.8259659
 DISTANCE FROM GROUP GRANITE MEAN IS 0.9072018

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 13485LOW

WEIGHTED X VALUES:
 11) 0.1264 9) -0.8166 10) -0.6182 8) -1.1018 1) 0.8825
 2) 0.4862 3) 0.1573 4) -0.1865

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 5 0.84220E 00
 GRANITE 9 0.21032E 01
 GRANITE 10 0.24745E 01
 GRANITE 8 0.24867E 01
 GRANITE 2 0.26081E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 5.3383589
 DISTANCE FROM GROUP GRANITE MEAN IS 2.6720533

NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 13789LOW

WEIGHTED X VALUES:
 11) 0.0789 9) -0.3610 10) -0.4972 8) -0.3255 1) 0.4584
 2) 0.6893 3) 0.9694 4) 1.2438

GROUP	NUMBER	DISTANCE	DECISION
GRANITE	2	0.61235E 00	
GRANITE	6	0.85044E 00	
ANDESITE	5	0.22857E 01	
ANDESITE	2	0.23511E 01	
GRANITE	9	0.29023E 01	

GRANITE **

DISTANCE FROM GROUP ANDESITE MEAN IS 4.6409874
 DISTANCE FROM GROUP GRANITE MEAN IS 3.0161753

NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 14397LOW

WEIGHTED X VALUES:
 11) 0.3978 9) -1.0987 10) -0.7184 8) -0.9994 1) 0.7347
 2) 1.2204 3) 1.2404 4) 1.0431

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	5	0.25801E 01	
GRANITE	2	0.26730E 01	
GRANITE	6	0.33001E 01	
GRANITE	8	0.41044E 01	
GRANITE	9	0.50402E 01	

GRANITE **

DISTANCE FROM GROUP ANDESITE MEAN IS 8.3749752
 DISTANCE FROM GROUP GRANITE MEAN IS 4.9559937

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 14701LOW

WEIGHTED X VALUES:
 11) -0.6504 9) -1.3231 10) -1.0464 8) -1.3348 1) 0.9983
 2) 0.9889 3) 0.8973 4) 0.7172

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 5 0.37826E 01
 GRANITE 2 0.45280E 01
 GRANITE 6 0.50547E 01
 GRANITE 9 0.71053E 01
 GRANITE 8 0.71744E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 11.8527403
 DISTANCE FROM GROUP GRANITE MEAN IS 7.7945890

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 15003LOW

WEIGHTED X VALUES:
 11) 0.7306 9) -0.6811 10) -0.6652 8) -0.7899 1) 0.5727
 2) 0.3821 3) 0.2243 4) 0.0813

.....
 GROUP NUMBER DISTANCE DECISION

 ANDESITE 5 0.69688E 00
 GRANITE 9 0.10558E 01
 GRANITE 8 0.10618E 01
 GRANITE 10 0.12410E 01
 GRANITE 4 0.15971E 01

GRANITE **

.....
 DISTANCE FROM GROUP ANDESITE MEAN IS 3.5348406
 DISTANCE FROM GROUP GRANITE MEAN IS 1.2433958

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 15610LOW

WEIGHTED X VALUES:
 11) 0.7432 9) -0.6252 10) -0.2213 8) -1.1431 1) 0.7911
 2) 0.4729 3) 0.2903 4) 0.2211

GROUP	NUMBER	DISTANCE	DECISION
ANDESITE	5	0.34145E 00	
GRANITE	4	0.82201E 00	
GRANITE	10	0.10196E 01	
GRANITE	8	0.10806E 01	
GRANITE	9	0.12197E 01	

GRANITE **

DISTANCE FROM GROUP ANDESITE MEAN IS 3.4977875
 DISTANCE FROM GROUP GRANITE MEAN IS 1.1137066

 NEAREST NEIGHBOR DECISION FOR UNKNOWN 19 53 15914LOW

WEIGHTED X VALUES:
 11) 1.0840 9) -0.3433 10) -0.0580 8) -0.7508 1) 0.6131
 2) 0.3890 3) 0.2825 4) 0.1358

GROUP	NUMBER	DISTANCE	DECISION
GRANITE	10	0.20175E 00	
GRANITE	8	0.28419E 00	
GRANITE	4	0.34585E 00	
GRANITE	9	0.49206E 00	
ANDESITE	5	0.65231E 00	

GRANITE **

DISTANCE FROM GROUP ANDESITE MEAN IS 1.7785826
 DISTANCE FROM GROUP GRANITE MEAN IS 0.2634534
 *****STOP CARD ENCOUNTERED*****

0.3196	0.1806	0.1530	0.3250	0.1921	-0.1156	-0.1533	-0.4027
-0.5313	-0.2326	-0.1046	-0.1306	-0.1730	-0.0894	0.0106	0.2967
0.7283	1.4695	2422					
19 53 6596							
1.2553	1.1072	1.1174	1.0066	0.6755	0.5287	0.4932	0.3143
0.2713	0.1642	0.1171	-0.2569	-0.7896	-0.6052	-0.8732	-0.8934
-0.5270	-0.1293	2744					
19 53 5899							
0.0247	0.2610	0.4085	0.9493	1.4023	1.4935	1.7105	1.4117
0.0705	0.0280	0.2659	0.1446	0.0031	-0.0756	-0.0854	-0.2917
-0.1979	0.0717	2910					
19 53 6205							
-0.6373	-0.8016	-0.5666	-0.5144	-0.6735	-0.6182	-0.6845	-0.4702
-0.1476	0.1403	0.4957	0.8702	1.0946	1.1869	0.9813	1.0336
1.1272	1.5632	2915					
19 53 6507							
0.7641	0.9114	0.6202	0.6225	0.4411	0.2326	0.4012	0.2996
0.4621	0.6327	0.5900	0.5267	0.3296	0.6303	0.3173	-0.0060
0.2419	0.3595	2988					
19 53 6811							
0.2061	0.0342	0.2036	0.1486	0.2297	0.3190	0.5186	0.5989
0.4927	0.7537	0.8358	0.5236	0.6787	0.7485	0.7809	0.4284
0.1311	0.6414	2866					
19 53 7115							
0.5917	0.4839	0.3761	0.6610	0.4761	0.4161	0.4851	0.1740
0.1489	0.1394	0.2293	0.1509	-0.1207	0.3917	-0.0964	0.0394
-0.0225	-0.0085	2915					
19 53 7417							
0.9890	1.0637	1.0289	0.9757	0.6305	0.3821	0.4565	0.1497
-0.2850	-0.0425	-0.1574	-0.5747	-0.7176	-0.7084	-0.6073	-0.6165
-0.3168	0.4787	2929					
19 53 7720							
0.2678	0.3082	0.6781	0.8051	0.6922	0.7572	0.8114	0.8048
0.6836	0.7622	0.5029	0.0942	-0.0768	0.2801	0.0964	0.3474
0.4882	1.0113	2905					
19 53 8024							
0.9991	0.9197	0.6904	0.7963	0.4028	0.2060	0.2851	-0.2450
-0.5936	-0.6042	-0.7870	-0.7793	-1.0053	-0.7878	-0.9125	-0.9715
-0.3608	-0.1468	2861					
19 53 8328							
-0.4342	-0.4334	-0.2053	-0.2147	-0.0547	-0.2050	-0.0318	0.0585
0.1159	0.4863	0.7292	0.7767	1.0526	1.2035	1.1095	1.0859
1.1840	1.3832	2841					
19 53 8630							
0.3891	0.2854	0.2020	0.1480	-0.2593	-0.3068	-0.2712	-0.5408
-0.4444	-0.6145	-0.3019	-0.2736	-0.1962	0.4610	0.4544	0.6322
0.9140	1.2400	2832					
19 53 8935							
0.2750	0.1575	0.1193	0.1554	0.2160	0.1087	0.3382	0.1101
0.0466	0.3126	0.1590	0.2376	0.4432	0.6314	0.5549	0.5089
0.7412	1.0804	2558					
19 53 9239							
0.3945	0.3107	0.4070	0.2672	0.2378	0.1195	0.0697	0.0476
-0.0569	0.0886	0.2886	0.1292	0.2389	0.6643	0.4137	0.6351
0.7428	1.5694	2705					
19 53 9541							
0.1540	0.2948	0.1784	0.0532	-0.0356	-0.2064	-0.1737	-0.2206
-0.2155	-0.0692	0.0885	0.0977	0.3065	0.7001	0.7244	0.7628
1.2169	1.3006	2558					
19 53 9845							
0.7576	0.6454	0.4801	0.4365	0.2153	-0.1195	-0.0464	-0.2424

-0.3247	-0.2621	-0.3826	-0.4740	0.0872	0.4181	0.3768	0.5915
0.8199	1.0608	2631					
19 53 10148							
0.3351	0.1879	0.2943	0.2486	0.0305	0.0064	-0.1153	-0.2228
-0.1483	-0.1341	0.1180	0.2310	0.3787	0.6403	0.4496	0.5338
0.6443	1.0872	2631					
19 53 10452							
0.0181	0.1029	-0.0496	-0.0304	-0.1753	-0.3128	-0.1783	-0.2384
-0.0547	0.2121	0.1139	0.4011	0.6076	0.8613	0.9073	0.8866
1.3354	1.6271	2612					
19 53 10754							
0.4048	0.3457	0.3171	0.2082	0.0706	0.0464	-0.0061	0.0417
-0.0568	-0.0581	0.0608	0.0312	0.4419	0.6195	0.7096	0.8163
0.9651	1.3739	2587					
19 53 11058							
0.3025	0.2980	0.2114	0.3647	0.2174	0.1582	0.2536	-0.1134
0.0671	0.0502	0.1259	0.2562	0.5153	0.8024	0.4505	0.3886
0.6758	0.7749	2719					
19 53 11361							
0.7314	0.6955	0.8509	0.6053	0.4620	0.1558	-0.0395	-0.1416
-0.2057	-0.1314	-0.1119	-0.1036	0.1757	0.1121	0.0769	0.0980
0.2900	0.6993	2939					
19 53 11666							
-0.1647	-0.1893	-0.1669	-0.1579	-0.4465	-0.3919	-0.2003	-0.4489
-0.4483	-0.4989	-0.6185	-0.6565	-0.6739	-0.6163	-0.6079	-0.3200
-0.6506	-0.9160	4062					
19 53 11969							
-0.4345	-0.2862	-0.3447	-0.2537	-0.3185	-0.3547	-0.2064	-0.5528
-0.7685	-0.6383	-0.8197	-0.4107	-0.6442	-0.8523	-0.5407	-0.7809
-0.7910	-0.5625	4145					
19 53 12272							
-0.2179	-0.2110	0.1156	0.0705	-0.3146	-0.3257	-0.3584	-0.5809
-0.6054	-0.4126	-0.5581	-0.6777	-0.6478	-1.0252	-0.8634	-0.7435
-0.7466	-0.7765	4130					
19 53 12576							
-0.9157	-0.7012	-0.5602	-0.2854	-0.3836	-0.0752	0.2593	0.4695
0.5648	0.3797	0.4327	-0.1315	-0.8662	-0.3835	-0.4645	0.0719
0.7109	2.0849	3447					
19 53 12879							
0.6404	0.7792	0.6605	0.2885	0.5376	0.1664	0.2552	0.0264
-0.5980	-0.8170	-1.0735	-1.3872	-1.7162	-1.5776	-1.1813	-1.9061
-0.3444	1.1231	3442					
19 53 13182							
0.4759	-0.1140	-0.3521	-0.0359	-0.2300	-0.2109	-0.0706	-0.3709
-0.6965	-0.8322	-0.7164	-0.0850	-0.0731	-0.1777	-0.2113	0.1851
0.2429	0.6264	3144					
19 53 13485							
1.0593	0.8915	0.5017	0.3436	0.1096	-0.2451	-0.2775	-0.7732
-1.1030	-1.1464	-1.3356	-1.1025	-1.2772	-0.0962	-0.7933	-0.7651
-0.3431	0.2360	3110					
19 53 13789							
0.6111	0.4666	0.8542	1.1023	1.2490	1.4315	1.3033	1.5943
1.0691	0.7399	0.4001	-0.2465	-0.4438	-0.1191	-0.5743	-0.4402
-0.4856	0.2075	3298					
19 53 14397							
0.7437	1.1215	1.5125	1.5317	1.5426	1.4044	1.1900	0.8175
0.3905	0.0284	-0.4947	-0.7362	-1.1624	-1.2366	-1.0770	-0.7694
-0.4970	0.6185	3740					
19 53 14701							
1.2664	1.0048	1.2456	1.1008	1.0642	0.9843	0.6256	0.2920
-0.0007	-0.7916	-0.5317	-1.1402	-1.5954	-1.1680	-1.5698	-1.0321

-1.1670 -0.4121 3344
 19 53 15003 0.6490 0.6583 0.4213 0.3499 0.2541 0.0700 0.1313 -0.2699
 -0.5584 -0.7679 -0.7413 -0.5911 -0.9556 -0.7775 -0.7143 -0.7492
 -0.3218 0.9637 3066
 19 53 15610 0.7841 0.7888 0.5116 0.5507 0.2346 0.1519 0.4377 0.0599
 0.0168 -0.5292 -0.6641 -0.7794 -1.4790 -0.7133 -0.6913 -0.3766
 0.1136 0.9390 3081
 19 53 15914 0.8371 0.5127 0.5076 0.3549 0.3319 0.1439 0.2419 -0.0259
 -0.3950 -0.5579 -0.4362 -0.7285 -0.9028 -0.4904 -0.3389 -0.1547
 0.3047 1.2495 2928
 SOURCE 08 VF501020304050607080910111213141516171819
 FILE 154
 /6

VII-35

PROBLEM CODE L04TST
NUMBER OF VARIABLES 18
NUMBER OF GROUPS 3
NUMBER OF CASES IN EACH GROUP 10 10 -41
VARIABLE FORMAT (2(F10.4),2(F10.4))

DATA INPUT FROM CARDS

MEANS (THE LAST COLUMN CONTAINS THE GRAND MEANS OVER THE GROUPS USED IN THE ANALYSIS)

VARIABLE	GROUP ANDES	GRANIT	UNKNOWN	
1	0.48582	0.60925	0.42029	0.54753
2	0.39188	0.41305	0.36644	0.40246
3	0.33897	0.36168	0.38579	0.35032
4	0.26170	0.33167	0.37651	0.29668
5	0.17852	0.27559	0.24811	0.22705
6	0.11733	0.27848	0.14638	0.19790
7	0.17069	0.29808	0.20988	0.23438
8	0.06444	0.16501	0.02325	0.11472
9	0.07879	0.05927	-0.04282	0.06903
10	0.02769	-0.02783	-0.08266	-0.00007
11	0.00804	-0.08167	-0.09022	-0.03681
12	-0.05080	-0.21635	-0.19760	-0.18357
13	0.13849	-0.50900	-0.26292	-0.18525
14	0.41363	-0.23763	-0.03009	0.08800
15	0.39926	-0.14655	-0.08768	0.12635
16	0.44399	0.04387	-0.00420	0.24393
17	0.78113	0.52677	0.23613	0.65395
18	0.93338	1.08845	0.72366	1.01091

STANDARD DEVIATIONS

VARIABLE	GROUP ANDES	GRANIT	UNKNOWN
1	0.34628	0.31450	0.49971
2	0.33659	0.30335	0.48216
3	0.33593	0.43897	0.46873
4	0.27004	0.30312	0.45109
5	0.19475	0.41207	0.48469
6	0.20391	0.42702	0.49569
7	0.17235	0.42578	0.51164
8	0.15962	0.43233	0.50497
9	0.15821	0.51166	0.47975
10	0.13518	0.44633	0.50952
11	0.24260	0.35304	0.53404
12	0.32045	0.40366	0.55982
13	0.42037	0.34258	0.73399
14	0.47596	0.30275	0.75755
15	0.42745	0.24749	0.70390

16	0.42657	0.20927	0.65394
17	0.38644	0.36699	0.66370
18	0.30685	0.53594	0.72584

WITHIN GROUPS COVARIANCE MATRIX

VARIABLE	1	2	3	4	5	6	7	8	9
1	0.10941								
2	0.09522	0.10266							
3	0.10245	0.11242	0.15277						
4	0.04725	0.05752	0.09242	0.08240					
5	0.03183	0.04557	0.09189	0.07605	0.10386				
6	0.01890	0.01739	0.06040	0.06632	0.09532	0.11196			
7	0.01838	0.01855	0.04907	0.04928	0.07137	0.09869	0.10550		
8	0.02692	0.02499	0.05642	0.04726	0.06837	0.09584	0.10140	0.10619	
9	0.00875	-0.00003	0.02018	0.03214	0.02839	0.08580	0.09547	0.10062	0.14341
10	-0.00875	-0.00342	0.01726	0.02332	0.02919	0.06220	0.07595	0.08288	0.11261
11	-0.02920	-0.02344	0.00111	0.01935	0.02985	0.05540	0.05406	0.06247	0.08650
12	-0.06647	-0.04529	-0.02351	0.00918	0.02504	0.04507	0.04637	0.04686	0.06874
13	-0.06413	-0.04836	-0.02528	0.00158	0.03668	0.04236	0.04733	0.04799	0.04338
14	-0.06668	-0.06714	-0.06370	-0.02646	0.00160	0.02819	0.04383	0.04359	0.04923
15	-0.05719	-0.05998	-0.06494	-0.04549	-0.01496	-0.00266	0.01985	0.01916	0.01589
16	-0.05428	-0.06641	-0.07127	-0.04160	-0.02573	-0.01655	-0.00401	-0.00838	-0.00858
17	-0.07319	-0.08183	-0.11745	-0.07854	-0.07800	-0.07335	-0.05503	-0.06117	-0.04879
18	-0.08404	-0.08035	-0.12093	-0.08210	-0.08870	-0.09937	-0.08479	-0.09437	-0.08366

VARIABLE	10	11	12	13	14	15	16	17	18
10	0.10874								
11	0.08615	0.09175							
12	0.08234	0.08900	0.13281						
13	0.06188	0.08120	0.11682	0.14703					
14	0.05487	0.06612	0.09696	0.13494	0.15910				
15	0.02947	0.03071	0.06446	0.09919	0.12601	0.12198			
16	0.00246	0.01069	0.02315	0.06329	0.09513	0.09262	0.11288		
17	-0.03220	-0.02249	-0.00705	0.01854	0.05853	0.07115	0.10176	0.14201	
18	-0.05407	-0.04551	-0.01485	-0.01120	0.02178	0.04865	0.08471	0.14706	0.19070

WITHIN GROUPS CORRELATION MATRIX

VARIABLE	1	2	3	4	5	6	7	8	9
1	1.00000								
2	0.89852	1.00000							
3	0.79242	0.80771	1.00000						
4	0.49761	0.62543	0.82370	1.00000					
5	0.29861	0.44135	0.72947	0.82203	1.00000				
6	0.17078	0.16216	0.46183	0.69048	0.79119	1.00000			
7	0.17112	0.17827	0.38655	0.52855	0.68181	0.90811	1.00000		
8	0.24972	0.27934	0.44294	0.50524	0.65105	0.87890	0.95796	1.00000	
9	0.06986	-0.00026	0.13635	0.29562	0.23264	0.67712	0.77620	0.81538	1.00000
10	-0.08019	-0.03236	0.13392	0.24636	0.27471	0.56367	0.70915	0.77122	0.90173
11	-0.20144	-0.26152	0.00934	0.22252	0.30578	0.54666	0.54945	0.63293	0.75408
12	-0.55142	-0.28789	-0.16505	0.08779	0.21322	0.36960	0.39178	0.39454	0.49804
13	-0.50563	-0.35361	-0.16884	0.01436	0.29681	0.33012	0.38001	0.38409	0.29874
14	-0.50541	-0.52537	-0.40457	-0.23110	0.01244	0.21120	0.33831	0.33535	0.32594
15	-0.49502	-0.53600	-0.47569	-0.45375	-0.13295	-0.02276	0.17499	0.16837	0.12016

VII-37

16	-0.48839	-0.61690	-0.54269	-0.43129	-0.23758	-0.14723	-0.03672	-0.07658	-0.06746
17	-0.58717	-0.67771	-0.79741	-0.72600	-0.64229	-0.58174	-0.44957	-0.49811	-0.34191
18	-0.58191	-0.57427	-0.70851	-0.65495	-0.63028	-0.68007	-0.59780	-0.66318	-0.50590

VARIABLES									
	10	11	12	13	14	15	16	17	18
VARIABLE									
10	1.00000								
11	0.86253	1.00000							
12	0.63514	0.80625	1.00000						
13	0.48040	0.69913	0.83595	1.00000					
14	0.41714	0.54725	0.66702	0.88226	1.00000				
15	0.25592	0.29028	0.50645	0.74065	0.90451	1.00000			
16	0.02223	0.10508	0.18906	0.49129	0.70990	0.78928	1.00000		
17	-0.25908	-0.19705	-0.05132	0.12833	0.38938	0.54058	0.80372	1.00000	
18	-0.37551	-0.34404	-0.09332	-0.06686	0.12502	0.31901	0.57738	0.89362	1.00000

SUBPROBLEM 1
 F-LEVEL FOR INCLUSION 0.0
 F-LEVEL FOR DELETION 0.0
 TOLERANCE LEVEL 0.0
 CONTROL VALUES 1111111111111111

STEP NUMBER 0
 VARIABLE ENTERED
 VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 18

1	0.6962	4	0.2971	7	0.7691	10	0.1417	13	14.2567	16	7.0914
2	0.0218	5	0.4536	8	0.4762	11	0.4386	14	13.3295	17	2.2780
3	0.0169	6	1.1597	9	0.0133	12	2.6547	15	12.2113	18	0.6305

STEP NUMBER 1
 VARIABLE ENTERED 13
 VARIABLES INCLUDED AND F TO REMOVE - DEGREES OF FREEDOM 1 18

13 14.2567
 VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 17

1	0.8179	4	0.1893	7	3.2921	10	1.5004	14	0.2430	17	0.5627
2	1.1172	5	1.8605	8	2.8321	11	4.0315	15	0.5687	18	0.1553
3	0.1394	6	3.1928	9	0.5935	12	4.0803	16	0.4535		

U-STATISTIC 0.55802 DEGREES OF FREEDOM 1 1 18
 APPROXIMATE F 14.25670 DEGREES OF FREEDOM 1 18.00

F MATRIX - DEGREES OF FREEDOM 1 18

GROUP
 ANDES
 GROUP
 GRANIT 14.25669
 FUNCTION
 ANDES GRANIT
 VARIABLE
 13 0.94189 -3.46179
 CONSTANT
 -0.06522 -0.88103

NUMBER OF CASES CLASSIFIED INTO GROUP -
 ANDES GRANIT

GROUP		
ANDES	9	1
GRANIT	1	9
UNKNOWN	22	19

STEP NUMBER 2
VARIABLE ENTERED 12

VII-38

VARIABLES INCLUDED AND F TO REMOVE - DEGREES OF FREEDOM 1 17

12 4.0803 13 15.9212

VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 16

1 0.1034	4 0.0189	7 1.7906	10 0.0041	15 0.0157	18 0.2159
2 0.5191	5 1.7119	8 1.4980	11 0.8365	16 0.0669	
3 0.0611	6 1.5960	9 0.9343	14 0.0027	17 0.0216	

U-STATISTIC 0.45001 DEGREES OF FREEDOM 2 1 18
APPROXIMATE F 10.38834 DEGREES OF FREEDOM 2 17.00

F MATRIX - DEGREES OF FREEDOM 2 17

GROUP
ANDES
GROUP
GRANIT 10.38830
FUNCTION
ANDES GRANIT
VARIABLE
12 -4.02051 2.20107
13 4.13617 -5.21054
CONSTANT
-0.38853 -0.97793

NUMBER OF CASES CLASSIFIED INTO GROUP -
ANDES GRANIT

GROUP		
ANDES	9	1
GRANIT	1	9
UNKNOWN	18	23

STEP NUMBER 3
VARIABLE ENTERED 7

VARIABLES INCLUDED AND F TO REMOVE - DEGREES OF FREEDOM 1 16

7 1.7906 12 2.4817 13 14.4529

VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 15

1 1.1953	4 0.4510	8 0.0189	11 0.1234	16 0.2995
2 1.6312	5 0.2675	9 3.3009	14 0.0001	17 0.3683
3 0.9710	6 0.0165	10 1.2475	15 0.0026	18 2.2783

U-STATISTIC 0.40472 DEGREES OF FREEDOM 3 1 18
APPROXIMATE F 7.84453 DEGREES OF FREEDOM 3 16.00

F MATRIX - DEGREES OF FREEDOM 3 16

GROUP
ANDES
GROUP
GRANIT 7.84450
FUNCTION
ANDES GRANIT
VARIABLE
7 1.82688 5.01041
12 -4.42116 1.10226
13 3.86644 -5.95030

CONSTANT
-0.53594 -2.08675

NUMBER OF CASES CLASSIFIED INTO GROUP -
ANDES GRANIT

GROUP		
ANDES	9	1
GRANIT	0	10
UNKNOWN	20	21

STEP NUMBER 4
VARIABLE ENTERED 9

VARIABLES INCLUDED AND F TO REMOVE - DEGREES OF FREEDOM 1 15

7 5.3054 9 3.3009 12 5.3355 13 19.3398

VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 14

1 0.6024	3 1.8156	5 0.5767	8 0.4901	11 4.1022	15 0.0779	17 0.7226
2 2.0549	4 1.2913	6 0.0467	10 0.1114	14 0.6126	16 0.7493	18 1.2836

U-STATISTIC 0.33172 DEGREES OF FREEDOM 4 1 18
APPROXIMATE F 7.55467 DEGREES OF FREEDOM 4 15.00

F MATRIX - DEGREES OF FREEDOM 4 15

GROUP
ANDES
GROUP
GRANIT 7.55467
FUNCTION
ANDES GRANIT
VARIABLE
7 0.96235 10.05937
9 1.03712 -6.02760
12 -5.11578 9.15894
13 4.39208 -9.02012

CONSTANT
-0.55686 -2.80022

NUMBER OF CASES CLASSIFIED INTO GROUP -
 ANDES GRANIT

GROUP		
ANDES	9	1
GRANIT	0	10
UNKNOWN	18	23

U11-39

STEP NUMBER 5
 VARIABLE ENTERED 11

VARIABLES INCLUDED AND F TO REMOVE - DEGREES OF FREEDOM 1 14

7 7.3499 4 7.9055 11 4.1022 12 1.6711 13 25.8002

VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 13

1 0.5964 3 3.4543 5 6.2729 8 0.0376 14 5.5396 16 1.9662 18 2.1868
 2 2.1206 4 3.1392 6 1.2155 10 0.1513 15 3.4401 17 1.9435

U-STATISTIC 0.25655 DEGREES OF FREEDOM 5 1 18
 APPROXIMATE F 8.11412 DEGREES OF FREEDOM 5 14.00

F MATRIX - DEGREES OF FREEDOM 5 14

	GROUP	
	ANDES	
GROUP		
GRANIT	8.11406	
	FUNCTION	
	ANDES GRANIT	
VARIABLE		
7	0.35769	13.01744
9	2.56081	-13.50622
11	-2.64763	12.95256
12	-4.67217	2.98874
13	5.24543	-13.19482

CONSTANT -0.60266 -3.89628

NUMBER OF CASES CLASSIFIED INTO GROUP -
 ANDES GRANIT

GROUP		
ANDES	9	1
GRANIT	0	10
UNKNOWN	19	22

STEP NUMBER 6
 VARIABLE ENTERED 5

VARIABLES INCLUDED AND F TO REMOVE - DEGREES OF FREEDOM 1 13

5 6.2729 7 14.1206 9 16.5628 11 10.9342 12 1.5075 13 38.2115

VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 12

1 0.8327 3 0.0732 6 0.0155 10 0.4463 15 0.1052 17 0.1994
 2 0.0192 4 0.4021 8 0.0554 14 0.5492 16 0.0646 18 0.0261

U-STATISTIC 0.17305 DEGREES OF FREEDOM 6 1 18
 APPROXIMATE F 10.35399 DEGREES OF FREEDOM 6 13.00

F MATRIX - DEGREES OF FREEDOM 6 13

	GROUP	
	ANDES	
GROUP		
GRANIT	10.35393	
	FUNCTION	
	ANDES GRANIT	
VARIABLE		
5	5.17418	-15.25835
7	-7.68829	36.74455
9	9.59629	-34.25343
11	-8.98750	28.99440
12	-5.07504	4.17677
13	7.79312	-20.70781

CONSTANT -0.81976 -5.78427

NUMBER OF CASES CLASSIFIED INTO GROUP -
 ANDES GRANIT

GROUP		
ANDES	10	0
GRANIT	0	10
UNKNOWN	13	28

STEP NUMBER 7
 VARIABLE ENTERED 1

VARIABLES INCLUDED AND F TO REMOVE - DEGREES OF FREEDOM 1 12

1 0.8327 5 6.1903 7 13.4692 9 13.8356 11 11.3091 12 2.2822 13 38.4694

VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 11

2 1.2626 4 0.2592 8 0.0943 14 0.5410 16 0.3335 18 0.0467
 3 0.8101 6 0.2326 10 0.9772 15 0.0624 17 0.0004

U-STATISTIC 0.16182 DEGREES OF FREEDOM 7 1 18
 APPROXIMATE F 8.87961 DEGREES OF FREEDOM 7 12.00

F MATRIX - DEGREES OF FREEDOM 7 12

	GROUP	
	ANDES	
GROUP		
GRANIT	8.87959	

VARIABLE	FUNCTION	
	ANDES	GRANIT
1	9.91169	16.76385
5	-6.70707	-35.35336
7	6.86992	61.76719
9	-7.16032	-62.59514
11	3.07477	47.87340
12	1.88333	15.94562
13	3.64524	-27.72321

CONSTANT
-2.33012 -10.10476

NUMBER OF CASES CLASSIFIED INTO GROUP -
ANDES GRANIT

GROUP	ANDES	GRANIT
ANDES	10	0
GRANIT	0	10
UNKNOWN	12	29

STEP NUMBER 8
VARIABLE ENTERED 2

VARIABLES INCLUDED AND F TO REMOVE - DEGREES OF FREEDOM 1 11

1	2.0926	5	3.8873	9	13.2447	12	3.4170
2	1.2626	7	10.9031	11	8.8644	13	30.5527

VARIABLES NOT INCLUDED AND F TO ENTER - DEGREES OF FREEDOM 1 10

3	0.0302	6	0.0761	10	0.2336	15	0.0120	17	0.0133
4	0.5091	8	0.0600	14	0.1317	16	0.0252	18	0.0527

U-STATISTIC 0.14516 DEGREES OF FREEDOM 8 1 18
APPROXIMATE F 8.09751 DEGREES OF FREEDOM 8 11.00

F MATRIX - DEGREES OF FREEDOM 8 11

GROUP
ANDES
GROUP
GRANIT 8.09736

VARIABLE	FUNCTION	
	ANDES	GRANIT
1	13.43883	35.03853
2	-3.62212	-18.76675
5	-6.23638	-32.91466
7	7.20807	63.11919
9	-8.27002	-68.34207
11	3.21335	48.59138
12	3.46885	24.16043
13	2.75715	-32.32455

CONSTANT

-2.40315 -12.06506

GROUP WITH SQUARE OF DISTANCE FROM AND POSTERIOR
LARGEST PROB. PROBABILITY FOR GROUP -

GROUP ANDES CASE		ANDES		GRANIT	
1	ANDES	6.643	1.000,	47.355	0.000,
2	ANDES	6.177	0.930,	11.365	0.070,
3	ANDES	3.839	0.999,	18.048	0.001,
4	ANDES	8.325	0.998,	21.078	0.002,
5	ANDES	9.022	0.999,	22.713	0.001,
6	ANDES	3.591	1.000,	27.119	0.000,
7	ANDES	7.836	1.000,	26.556	0.000,
8	ANDES	5.145	1.000,	33.215	0.000,
9	ANDES	7.140	1.000,	40.788	0.000,
10	ANDES	3.690	1.000,	25.175	0.000,

GROUP GRANIT CASE		ANDES		GRANIT	
1	GRANIT	30.559	0.000,	11.768	1.000,
2	GRANIT	37.673	0.000,	10.694	1.000,
3	GRANIT	30.849	0.000,	8.274	1.000,
4	GRANIT	39.134	0.000,	6.904	1.000,
5	GRANIT	35.117	0.000,	9.013	1.000,
6	GRANIT	17.849	0.013,	9.226	0.987,
7	GRANIT	36.968	0.000,	8.923	1.000,
8	GRANIT	21.054	0.000,	5.093	1.000,
9	GRANIT	21.848	0.002,	9.522	0.998,
10	GRANIT	23.541	0.000,	3.172	1.000,

GROUP UNKNOWN CASE		ANDES		GRANIT	
1	GRANIT	36.152	0.000,	16.088	1.000,
2	GRANIT	37.795	0.000,	11.499	1.000,
3	GRANIT	16.409	0.305,	14.758	0.695,
4	GRANIT	46.714	0.000,	14.549	1.000,
5	ANDES	9.075	0.812,	12.901	0.188,
6	GRANIT	16.125	0.092,	11.545	0.908,
7	GRANIT	31.757	0.000,	15.195	1.000,
8	GRANIT	29.022	0.000,	6.319	1.000,
9	GRANIT	57.600	0.000,	14.317	1.000,
10	GRANIT	64.528	0.002,	51.866	0.998,
11	ANDES	34.462	1.000,	52.755	0.000,
12	GRANIT	24.955	0.114,	20.850	0.886,
13	GRANIT	48.717	0.000,	23.231	1.000,
14	GRANIT	47.480	0.000,	9.424	1.000,
15	GRANIT	88.441	0.000,	33.332	1.000,
16	GRANIT	23.702	0.013,	15.059	0.987,
17	GRANIT	64.127	0.000,	19.581	1.000,
18	ANDES	16.712	0.998,	31.760	0.001,
19	GRANIT	11.065	0.403,	10.279	0.597,
20	GRANIT	11.576	0.305,	9.299	0.695,
21	GRANIT	14.035	0.075,	9.009	0.925,
22	ANDES	5.837	1.000,	28.501	0.000,
23	ANDES	9.568	1.000,	53.850	0.000,
24	ANDES	6.066	0.998,	19.609	0.001,
25	ANDES	8.000	1.000,	49.465	0.000,

VII-41

26	ANDRES	2.870	1.000,	36.999	0.000,
27	ANDRES	1.954	1.000,	22.666	0.000,
28	ANDRES	9.654	1.000,	49.573	0.000,
29	ANDRES	23.810	0.672,	25.246	0.328,
30	GRANIT	41.177	0.005,	30.547	0.995,
31	ANDRES	22.185	0.956,	28.331	0.044,
32	GRANIT	115.491	0.000,	76.657	1.000,
33	GRANIT	61.413	0.000,	39.394	1.000,
34	GRANIT	85.273	0.000,	30.032	1.000,
35	GRANIT	36.679	0.001,	22.141	0.999,
36	GRANIT	118.657	0.000,	53.737	1.000,
37	ANDRES	88.775	0.999,	102.314	0.001,
38	GRANIT	59.534	0.000,	24.436	1.000,
39	GRANIT	44.082	0.000,	11.787	1.000,
40	GRANIT	61.634	0.000,	16.682	1.000,
41	GRANIT	66.640	0.000,	14.925	1.000,

NUMBER OF CASES CLASSIFIED INTO GROUP -		
GROUP	ANDRES	GRANIT
ANDRES	10	0
GRANIT	0	10
UNKNOWN	13	28

SUMMARY TABLE

STEP NUMBER	VARIABLE		F VALUE TO ENTER OR REMOVE	NUMBER OF VARIABLES INCLUDED	U-STATISTIC			
	ENTERED	REMOVED						
1	13		14.2567	1	0.5580			
2	12		4.0803	2	0.4500			
3	7		1.7906	3	0.4047			
4	9		3.3009	4	0.3317			
5	11		4.1022	5	0.2565			
6	5		6.2729	6	0.1730			
7	1		0.8327	7	0.1618			
8	2		1.2626	8	0.1452			
EIGENVALUES								
	5.88901	0.00001	0.00000	0.00000	-0.00000	-0.00000	-0.00000	-0.00001
CUMULATIVE PROPORTION OF TOTAL DISPERSION								
	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CANONICAL CORRELATIONS								
	0.92458	0.00282	0.00207	0.00068	0.00053	0.00087	0.00146	0.00260
COEFFICIENTS FOR CANONICAL VARIABLE -								
ORIGINAL VARIABLE	1	2	3	4	5	6	7	8
1	4.69108	-8.19959	4.30950	0.05371	-0.82087	0.35626	2.05859	-2.20241
2	-3.28916	7.84717	-3.44238	-0.20154	-1.15108	0.44247	1.31847	-1.51104
5	-5.79408	-2.84003	-1.07916	-1.27580	-1.29061	-1.24721	-1.18548	7.80060
7	12.14300	2.80037	0.99595	0.90061	-0.88840	2.00256	-2.12290	-9.31728
9	-13.04668	-1.83178	-3.77232	-1.15999	2.26805	1.62716	0.17601	8.33065
11	9.85539	5.50979	6.41820	1.69064	-1.11350	-0.05004	1.38167	-3.37252
12	4.49389	-4.91521	-2.25782	3.20927	-1.72795	-1.51059	2.02610	-1.92751
13	-7.61921	0.12636	0.89359	-1.52587	0.23219	0.86694	-1.18755	-0.12628
GROUP CANONICAL VARIABLES EVALUATED AT GROUP MEANS								
1	-2.30220	0.00000	0.00000	0.00000	0.00000	-0.00000	-0.00000	0.00000
2	2.30220	-0.00000	-0.00000	-0.00000	-0.00000	0.00000	0.00000	-0.00000
3	1.21597	0.69359	-0.24147	0.12245	-0.16089	-0.44264	-0.32083	-0.40410
CHECK ON FINAL U-STATISTIC			0.14516					

POINTS PLOTTED ON THE FOLLOWING GRAPH

U11-42

Y = FIRST CANONICAL VARIABLE
X = SECOND CANONICAL VARIABLE
CASE NUMBER FOLLOWED BY * INDICATES THE POINT IS OFF THE GRAPH

GROUP ANDES MEAN COORDINATES -2.302 0.000

CASE	X	Y
1	-4.421	1.014
2	-0.563	-0.647
3	-1.543	-0.510
4	-1.385	2.038
5	-1.487	0.120
6	-2.555	-1.197
7	-2.033	-0.527
8	-3.048	0.137
9	-3.654	-0.492
10	-2.333	0.064

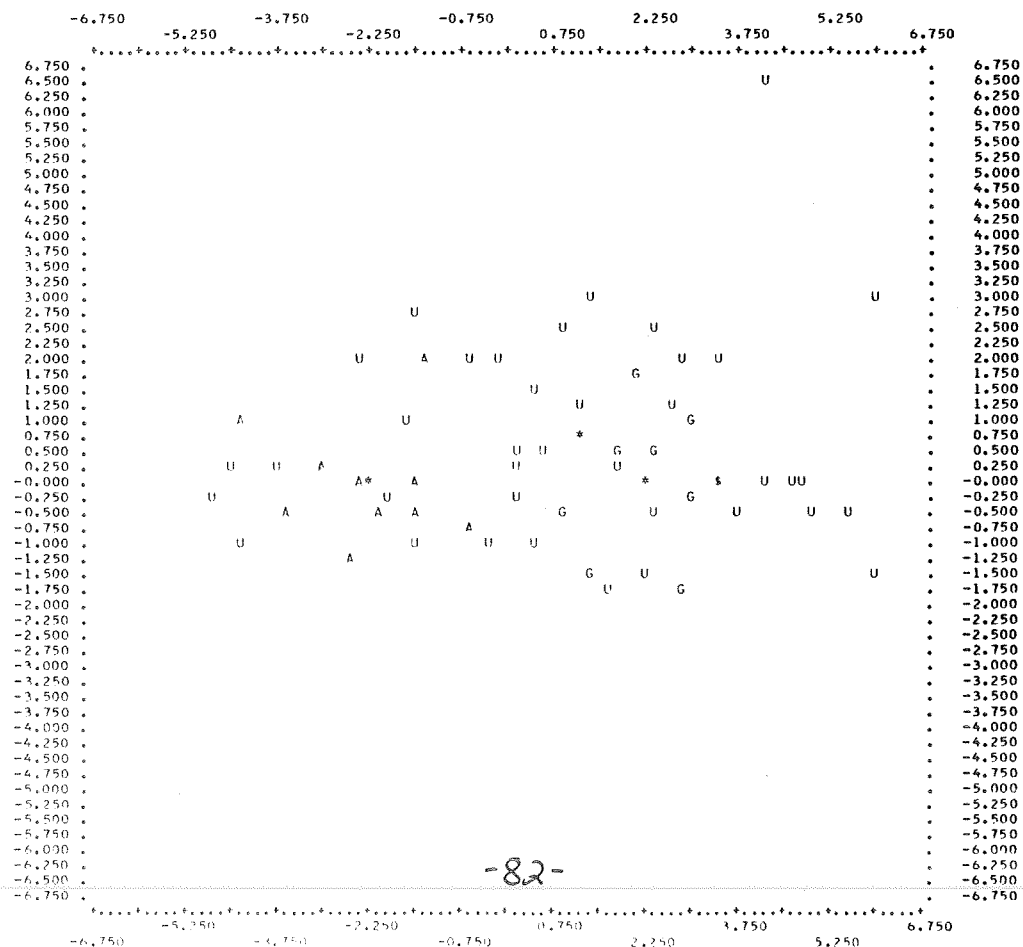
GROUP GRANIT MEAN COORDINATES 2.302 -0.000

CASE	X	Y
1	2.041	1.768
2	2.930	1.107
3	2.451	0.413
4	3.500	0.119
5	2.835	-1.762
6	0.936	-0.408
7	3.046	-0.274
8	1.733	0.458
9	1.338	-1.434
10	2.212	0.014

GROUP UNKNOW MEAN COORDINATES 1.216 0.694

CASE	X	Y	CASE	X	Y	CASE	X	Y	CASE	X	Y
1	2.179	-1.575	11	-1.986	-0.261	21	0.546	0.583	31	-0.667	2.074
2	2.856	2.048	12	0.446	1.385	22	-2.461	1.876	32	4.217	6.492
3	0.179	0.463	13	2.713	1.363	23	-4.809	-0.287	33	2.391	2.604
4	3.493	2.061	14	4.133	-0.044	24	-1.471	-1.078	34	5.999	-1.524
5	-0.318	-1.038	15	5.984	2.884	25	-4.492	0.261	35	1.579	-1.696
6	0.497	-0.999	16	0.939	2.509	26	-3.706	0.288	36*	7.050	2.821
7	1.799	0.324	17	4.837	-0.096	27	-2.249	0.098	37	-1.470	2.750
8	2.465	-0.391	18	-1.634	1.117	28	-4.335	-0.900	38	3.811	-0.575
9	4.700	-0.061	19	0.085	0.267	29	-0.156	1.897	39	3.507	-0.016
10	1.375	2.951	20	0.179	-0.236	30	1.154	1.242	40	4.881	-0.595
									41	5.616	-0.549

OVERLAP IS INDICATED BY \$, GROUP MEANS BY *.



APPENDICIES

Appendix A: Smoothing and Differentiation of Data by
Simplified Least Squares Procedures

Appendix B: Block Radiometer Conversion Formulas

Appendix C: BMD07M--Stepwise Discriminant Analysis

Smoothing and Differentiation of Data by Simplified Least Squares Procedures

APPENDIX A

ABRAHAM SAVITZKY and MARCEL J. E. GOLAY

The Perkin-Elmer Corp., Norwalk, Conn.

► In attempting to analyze, on digital computers, data from basically continuous physical experiments, numerical methods of performing familiar operations must be developed. The operations of differentiation and filtering are especially important both as an end in themselves, and as a prelude to further treatment of the data. Numerical counterparts of analog devices that perform these operations, such as RC filters, are often considered. However, the method of least squares may be used without additional computational complexity and with considerable improvement in the information obtained. The least squares calculations may be carried out in the computer by convolution of the data points with properly chosen sets of integers. These sets of integers and their normalizing factors are described and their use is illustrated in spectroscopic applications. The computer programs required are relatively simple. Two examples are presented as subroutines in the FORTRAN language.

THE PRIMARY OUTPUT of any experiment in which quantitative information is to be extracted is information which measures the phenomenon under observation. Superimposed upon and indistinguishable from this information are random errors which, regardless of their source, are characteristically described as noise. Of fundamental importance to the experimenter is the

removal of as much of this noise as possible without, at the same time, unduly degrading the underlying information.

In much experimental work, the information may be obtained in the form of a two-column table of numbers, *A* vs. *B*. Such a table is typically the result of digitizing a spectrum or digitizing other kinds of results obtained during the course of an experiment. If plotted, this table of numbers would give the familiar graphs of %T vs. wavelength, pH vs. volume of titrant, polarographic current vs. applied voltage, NMR or ESR spectrum, or chromatographic elution curve, etc. This paper is concerned with computational methods for the removal of the random noise from such information, and with the simple evaluation of the first few derivatives of the information with respect to the graph abscissa.

The bases for the methods to be discussed have been reported previously, mostly in the mathematical literature (4, 6, 8, 9). The objective here is to present specific methods for handling current problems in the processing of such tables of analytical data. The methods apply as well to the desk calculator, or to simple paper and pencil operations for small amounts of data, as they do to the digital computer for large amounts of data, since their major utility is to simplify and speed up the processing of data.

There are two important restrictions on the way in which the points in the

table may be obtained. First, the points must be at a fixed, uniform interval in the chosen abscissa. If the independent variable is time, as in chromatography or NMR spectra with linear time sweep, each data point must be obtained at the same time interval from each preceding point. If it is a spectrum, the intervals may be every drum division or every 0.1 wavenumber, etc. Second, the curves formed by graphing the points must be continuous and more or less smooth—as in the various examples listed above.

ALTERNATIVE METHODS

One of the simplest ways to smooth fluctuating data is by a moving average. In this procedure one takes a fixed number of points, adds their ordinates together, and divides by the number of points to obtain the average ordinate at the center abscissa of the group. Next, the point at one end of the group is dropped, the next point at the other end added, and the process is repeated.

Figure 1 illustrates how the moving average might be obtained. While there is a much simpler way to compute the moving average than the particular one described, the following description is correct and can be extended to more sophisticated methods as will be seen shortly. This description is based on the concept of a convolute and of a convolution function. The set of numbers at the right are the data or ordinate values, those at the left, the abscissa information. The outlined

1800.0		705
1799.8		712
1799.6		717
1799.4		718
1799.2		721
1799.0		722
1798.8	$X_0 - 2$	C_{-2}
1798.6	$X_0 - 1$	C_{-1}
1798.4	X_0	C_0
1798.2	$X_0 + 1$	C_1
1798.0	$X_0 + 2$	C_2
1797.8		746
1797.6		750

Figure 1. Convolution operation

Abscissa points at left, tabular data at right. In box area the convolution integers, C_i . Operation is the multiplication of the data points by the corresponding C_i , summation of the resulting products, and division by a normalizer, resulting in a single convolute at the point X_0 . The box is then moved down one line, and the process repeated

block in the center may be considered to be a separate piece of paper on which are written a new set of abscissa numbers, ranging from -2 thru zero to $+2$. The C 's at the right represent the convoluting integers. For the moving average each C is numerically equal to one. To perform a convolution of the ordinate numbers in the table of data with a set of convoluting integers, C_i , each number in the block is multiplied by the corresponding number in the table of data, the resulting products are added and this sum is divided by five. The set of ones is the convoluting function, and the number by which we divide, in this case, 5, is the normalizing factor. To get the next point in the moving average, the center block is slid down one line and the process repeated.

The concept of convolution can be generalized beyond the simple moving average. In the general case the C 's represent any set of convoluting integers. There is an associated normalizing or scaling factor. The procedure is to multiply C_{-2} times the number opposite it, then C_{-1} by its number, etc., sum the results, divide by the normalizing factor, if appropriate, and the result is the desired function evaluated at the point indicated by C_0 . For the next point, we move the set of convoluting integers down and repeat, etc. The mathematical description of this process is:

$$Y_j^* = \frac{\sum_{i=-m}^{i=m} C_i Y_{j+i}}{N}$$

The index j represents the running index of the ordinate data in the original data table.

For the moving average, each C_i is equal to one and N is the number of convoluting integers. However, for many types of data the set of all 1's, which yields the average, is not particularly useful. For example, on going through a sharp peak, the average would tend to degrade the end of the peak. There are other types of smoothing functions which might be used, and a few of these are indicated in Figure 2.

Figure 2A illustrates the set where all values have the same weight over the interval—essentially the moving average.

The function in Figure 2B is an exponential set which simulates the familiar RC analog time constant—i.e., the most recent point is given the greatest weight, and each preceding point gets a lesser weight determined by the law of exponential decay. Future points have no influence. Such a function treats future and past points differently and so will obviously introduce a unidirectional distortion into the numerical results, as does the RC filter in an actual instrument.

When dealing with sets of numbers in hand, and not an actual run on an instrument where the data is emerging in serial order, it is possible to look ahead as well as behind. Then we can convolute with a function that treats past and future on an equal basis, such as the function in Figure 2D. Here the most weight is given to the central point, and points on either side of the center are symmetrically weighed exponentially. This function acts like an idealized lead-lag network, which is not practical to make with resistors, capacitors, and so on.

The usual spectrum from a spectrophotometer is the resultant of two convolutions of the actual spectrum of the material, first with a function representing the slit function of the instrument, which is much like the triangular convolute shown in Figure 2C, and then this first convolute spectrum is further convoluted with a function representing the time constant of the instrument. The triangular convoluting function could in many cases yield results not significantly different from the symmetrical exponential function.

Figure 3 illustrates the way in which each of these functions would act on a typical set of spectroscopic data. Curve 3A is replotted directly from the instrumental data. It is a single sharp band recorded under conditions which yield a reasonable noise level. The isolated point just to the right of the band has the value of 666 on the scale of zero to 1000 corresponding to approximately 0 to 100% transmittance. This point is introduced to illustrate the effect on these operations of a single point which has a gross error. The numbers along the bottom are the digital value at the

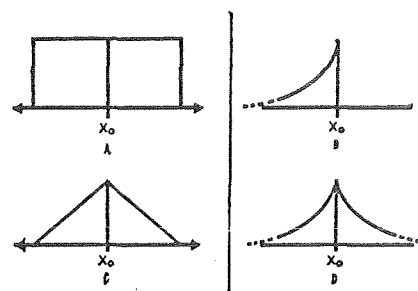


Figure 2. Various convolute functions

A. Moving average. B. Exponential function. C. Symmetrical triangular function, representing idealized spectrometer slit function. D. Symmetrical exponential function

lowest point of the plot, and one may consider that the peak goes down to 34.2% transmittance. The base line at the top is at about 79%.

Curve 3B is a nine-point moving average of the data. As expected, the peak is considerably shortened by this process. Especially interesting is the step introduced by the isolated error. In effect, it has the shape of the boxlike convolute in Figure 2A, which is exactly what one would expect from the convoluting process (3).

Curve 3C is for a triangular function which obviously forces both the peak itself and the isolated error into a triangular mold.

Curve 3D is the result of convoluting with the numerical equivalent of a conventional RC exponential time constant filter using only five points. The peak is not only shortened, but is also shifted to the right by one data point, or 0.002 micron and the isolated data point is asymmetric in the same manner. The convolution with a symmetrical lead-lag exponential, as in Figure 3E, does not distort the peak but does still reduce its intensity.

Note that while all of these functions have had the desired effect of reducing the noise level, they are clearly undesirable because of the accompanying degradation of the peak intensity.

METHOD OF LEAST SQUARES

The convoluting functions discussed so far are rather simple and do not extract as much information as is possible. The experimenter, if presented with a plot of the data points, would tend to draw through these points a line which best fits them. Numerically, this can also be done, provided one can adequately define what is meant by best fit. The most common criterion is that of least squares which may be simply stated as follows:

A set of points is to be fitted to some curve—for example, the curve $a_3x^3 + a_2x^2 + a_1x + a_0 = y$. The a 's are to be selected such that when each abscissa point is substituted into this equation,

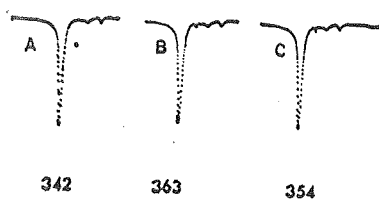


Figure 3. Spectral band convoluted by the various 9 point functions

The number at the bottom of each peak refers to the lowest recorded point, and is a measure of the ability to retain the shape of the peak. A. Raw data with single isolated error point. B. Moving average. C. Triangular function. D. Normal exponential function. E. Symmetrical exponential function. F. Least squares smoothing function

the square of the differences between the computed numbers, y , and the observed numbers is a minimum for the total of the observations used in determining the coefficients. All of the error is assumed to be in the ordinate and none in the abscissa.

Consider the block of seven data points enclosed by the left bracket in Figure 4. If these fall along a curve that can be described approximately by the equation shown, then there are specific procedures—which are described in most books on numerical analysis—to find the a 's. One then substitutes back into the resulting equation the abscissa at the central point indicated by the circle. The value which is obtained by this procedure is the best value at that point based on the least squares criterion, on the function which was chosen, and on the group of points examined.

This procedure can be repeated for

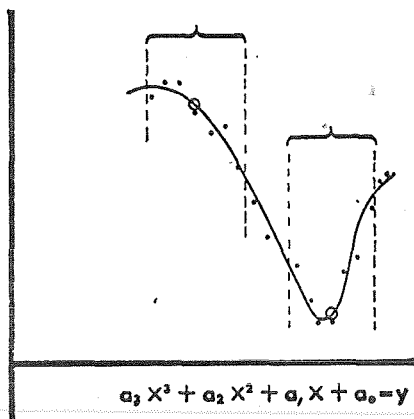


Figure 4. Representation of a 7-point moving polynomial smooth

each group of seven points, dropping one at the left and picking up one at the right each time. A somewhat later block is indicated at the right. In the usual case, there is found a different set of coefficients for each group of seven points. Even with a high-speed computer this is a tedious proposition at best.

Note, however, that finding the a coefficients is required only as a means for determining the final best value at just one point, the central point of the set. A careful study of the least squares procedure using these constraints, leads to the derivation of a set of integers which provide a weighting function. With this set of integers the central point can be evaluated by the convoluting procedure discussed above. This procedure is exactly equivalent to the least squares. It is not approximate.

The derivation is presented in Appendix I. For either a cubic or a quadratic function, the set of integers is the same, and the set for up to 25 points is shown in Table I of Appendix II with the appropriate normalizing factors. A most instructive exercise is to tabulate a simple function such as $y = x^3$ over any interval, apply these smoothing convolutes and compare these new values with the original. The answers will be found to be exact.

In Figure 3F this least squares convoluting procedure has been applied to the data of Figure 3A, using a 9-point cubic convolute. The value at the peak and the shape of the peak are essentially undistorted. As always, the isolated point assumes the shape of the convoluting function. The FORTRAN language computer program for performing this operation is presented in Program I of Appendix III.

Going beyond simple curve fitting, one can find in the literature on numerical analysis a variety of least squares procedures for determining the first derivative. These procedures are usually based on interpolation formulas and are for data at any arbitrary interval. Again, if we restrict ourselves to evaluating the function only at the center point of a set of equally spaced observations, then there exist sets of convoluting integers for the first derivative as well. (These actually evaluate the derivative of the least squares best function.)

A complete set of tables for derivatives up to the fifth order for polynomials up to the fifth degree, using from 5 to 25 points, is presented in Appendix II. These are more than adequate for most work, since, if the points are taken sufficiently close together, then practically any smooth curve will look more or less like a quadratic in the vicinity of a peak, or like a cubic in the vicinity of a shoulder. More complete tables can be found in the statistical literature (2, 4, 6, 9).

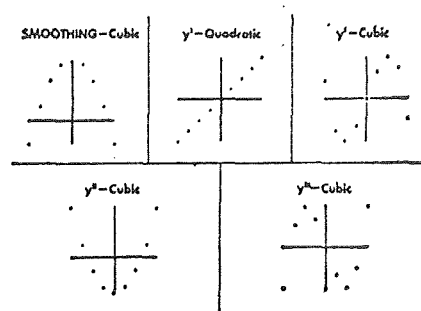


Figure 5. 9-Point convoluting functions (orthogonal polynomial) for smoothing and first, second, and third derivative

Program II of Appendix III shows the use of these tables to obtain the coefficients of a polynomial for finding the precise center of an infrared band.

The shapes of the 9-point convolutes for a few of the functions are illustrated in Figure 5. Of special interest is the linear relation of the first derivative convolute for a quadratic. This is quite unique operationally because in processing a table of data, only one multiplication is necessary for each convolution. The remainder of the points are found from the set calculated for the previous point by simple subtraction. In Figure 6B is shown the first derivative of the spectrum in Figure 6A, obtained using a 9-point convolute.

The derivatives are useful in cases such as our methods of band finding on a computer (7), in studies of derivative spectra, in derivative thermogravimetric analysis, derivative polarography, etc.

CONCLUSIONS

With the increase in the application of computers to the analysis of digitized data, the convolution methods described are certain to gain wider usage. With these methods, the sole function of the computer is to act as a filter to smooth the noise fluctuations and hopefully to introduce no distortions into the recorded data (3).

This problem of distortion is difficult to assess. In any of the curves of Figure 3, there remain small fluctuations in the

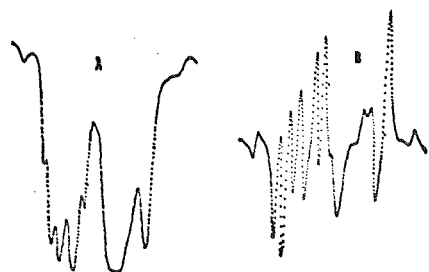


Figure 6. 17-Point first derivative convolute

A. Original spectrum. B. First derivative spectrum.

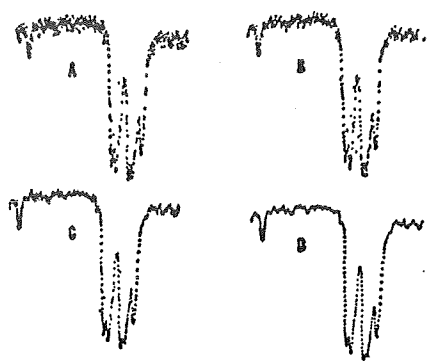


Figure 7. Square root relation between number of points and degree of smoothing

A. Raw data. B. 5-Point smooth. C. 9-Point smooth. D. 17-Point smooth

data. Are these fluctuations real, or, as is more likely, are they just a low frequency component of the noise level which could not be smoothed? The question cannot be answered by taking just the data from a single run. However, if one were to take more than one run, average these and then smooth, or smooth and then average, the computer-plus-instrument system could decide, since even low frequency noise will not recur in exactly the same place in different runs. Computer time is most efficiently used if the averaging is done prior to smoothing.

Recent work (1) has shown the utility of simple averaging of a large number of runs in the enhancement of signal-to-noise ratios. The use of combined smoothing and averaging can considerably reduce the instrument time required, throwing the burden onto the computer which operates in a wholly different time domain. A characteristic of both procedures is that the noise is reduced approximately as the square root of the number of points used. This is illustrated for the smoothing case by Figure 7. At the upper left is the raw data, at the upper right a 5-point smooth, lower left 9 points, and lower right 17 points. If it is desired to improve the signal-to-noise ratio by a factor of 10, simple averaging would require a total of 100 runs. Similar improvement could be achieved by making only 16 runs plus a 9-point least squares smooth (average of 16 runs $\cong 4 \times$ improvement, and 9-point smooth $\cong 3 \times$ improvement) or only 4 runs plus a 25-point smooth (average of 4 runs $\cong 2 \times$ improvement plus 25-point smooth $\cong 5 \times$ improvement). The distribution between the number of runs required and the number of points which may be used for the smoothing is a function of the experimental curve under examination. The minimum distortion will occur when the polynomial accurately describes the

analytical data, and will deviate as the polynomial departs from the true curve. The best results are obtained when the data are digitized at high densities—i.e., points very close together—and the number of points used in the convolute is chosen to be small enough so that no more than one inflection in the observed data is included in any convolution interval. Our results should be compared with those achievable using conventional instrument filtering. In a sense, we are substituting idealized filters and filter-networks for electronic hardware such as resistors, condensers, servos, etc. If one examines the time relationships, it probably takes longer to get the information using the digitizer and then the computer than with the analogous electronic network. There is, very definitely, the advantage in the computer of being able to vary the processing completely unfettered by the practical restriction of real circuitry and servo loops. Note too, that this processing can be done after the fact of data collection, and indeed several different procedures may be applied in order to assess the optimum. This is a real advantage in itself, and provides ample justification for use of the computer solely as a noise filter. However, the greatest utility of these methods comes in the pretreatment of data to be further processed, as in our bandfinding procedures, in any curvefitting operations, in quantitative analyses, etc.

This type of data processing, so far as computers are concerned, requires a relatively small amount of programming and relatively little use of the computer memory or of the computer's processing capability. Therefore, even accounting-type computers, such as the IBM 1401 can be used to process data in this way. Furthermore, on such computers there is generally a high-speed line printer which can be turned into a relatively crude point plotter. On each line an X can be placed at the appropriate position to 1% of value, and the actual value is printed at the edge of the paper. The rate can be on the order of 10 lines, or points, per second.

ACKNOWLEDGMENT

The authors appreciate the assistance of Harrison J. M. Kinyua in the re-compilation and checking of the convolute tables, of Robert Bernard of the Perkin-Elmer Scientific Computing Facility for his advice and assistance in program development, and of Joel Stutman, University of Maryland, for his advice on Appendix I.

APPENDIX I

The general problem is formulated as follows:

A set of $2m + 1$ consecutive values are to be used in the determination of

the best mean square fit through these values of a polynomial of degree n (n less than $2m + 1$). This polynomial is of the form:

$$f_i = \sum_{k=0}^{n} b_{nk} i^k = \quad \text{I}$$

$$b_{n0} + b_{n1}i + b_{n2}i^2 + \dots + b_{nn}i^n \quad \text{Ia}$$

The derivatives of this polynomial are:

$$\frac{df_i}{di} = b_{n1} + 2b_{n2}i + 3b_{n3}i^2 + \dots$$

$$+ nb_{nn}i^{n-1} \quad \text{Ib}$$

$$\frac{d^2f_i}{di^2} = 2b_{n2} + 3 \times 2b_{n3}i + \dots$$

$$+ (n-1)nb_{nn}i^{n-2} \quad \text{Ic}$$

$$\frac{d^nf_i}{di^n} = nb_{nn} \quad \text{Id}$$

Note that, in the coordinate system being considered, the value of i ranges from $-m$ to $+m$, and that $i=0$ at the central point of the set of $2m + 1$ values. Hence, the value of the s th derivative at that point is given by:

$$\left(\frac{d^sf_i}{di^s}\right)_{i=0} = sb_{ns} = a_n \quad \text{II}$$

where

$$f_0 = b_{n0} = a_{n0} \quad \text{IIa}$$

$$\frac{df_0}{di} = b_{n1} = a_{n1} \quad \text{IIb}$$

$$\frac{d^2f_0}{di^2} = 2b_{n2} = a_{n2} \quad \text{IIc}$$

The least squares criterion requires that the sum of the squares of the differences between the observed values, y_i , and the calculated, f_i , be a minimum over the interval being considered.

$$\frac{\partial}{\partial b_{nk}} \left[\sum_{i=-m}^{i=m} (f_i - y_i)^2 \right] = 0 \quad \text{III}$$

Minimizing with respect to b_{n0} , we have

$$\begin{aligned} \frac{\partial}{\partial b_{n0}} \left[\sum_{i=-m}^{i=m} b_{n0} + b_{n1}i + \dots \right. \\ \left. + (b_{nn}i^n - y_i)^2 \right] = \quad \text{IIIa} \\ 2 \sum_{i=-m}^{i=m} (b_{n0} + b_{n1}i + \dots \\ + b_{nn}i^n - y_i) = 0 \end{aligned}$$

and with respect to b_{ni} , we have

$$\begin{aligned} \frac{\partial}{\partial b_{ni}} \left[\sum_{i=-m}^{i=m} (b_{n0} + b_{n1}i + \dots + b_{nn}i^n - y_i)^2 \right] = 2 \sum_{i=-m}^{i=m} (b_{n0} + b_{n1}i + \dots \\ + b_{nn}i^n - y_i) i = 0 \quad \text{IIIb} \end{aligned}$$

and with respect to the general b_{nr} , we obtain

$$2 \sum_{i=-m}^{i=m} \left[\left(\sum_{k=0}^{k=n} b_{nk} i^k \right) - y_i \right] i^r = 0 \quad \text{IIIc}$$

or

$$\sum_{i=-m}^{i=m} \sum_{k=0}^{k=n} b_{nk} i^{k+r} = \sum_{i=-m}^{i=m} y_i i^r \quad \text{IVa}$$

Where r is the index representing the equation number which runs from 0 to n (there are $n+1$ equations). The summation indexes on the left side may be interchanged—i.e.,

$$\sum_{i=-m}^{i=m} \sum_{k=0}^{k=n} b_{nk} i^{k+r} = \sum_{k=0}^{k=n} \sum_{i=-m}^{i=m} b_{nk} i^{k+r} \quad \text{IVb}$$

and finally, since b_{nk} is independent of i ,

$$\sum_{k=0}^{k=n} b_{nk} \sum_{i=-m}^{i=m} i^{k+r} = \sum_{i=-m}^{i=m} y_i i^r = F_k \quad \text{Va}$$

or

$$\sum_{k=0}^{k=n} b_{nk} S_{r+k} = F_k \quad \text{Vb}$$

where

$$S_{r+k} = \sum_{i=-m}^{i=m} i^{r+k} \quad \text{Vc}$$

and

$$F_k = \sum_{i=-m}^{i=m} i^k y_i \quad \text{Vd}$$

Note that $S_{r+k} = 0$ for odd values of $r+k$. Since S_{r+k} exists for even values of $r+k$ only, the set of $n+1$ equations can be separated into two sets, one for even values of k and one for odd values. Thus, for a 5th degree polynomial, where $n=5$

$$S_0 b_{50} + S_2 b_{52} + S_4 b_{54} = F_0$$

$$S_2 b_{50} + S_4 b_{52} + S_6 b_{54} = F_2 \quad \text{VIa}$$

$$S_4 b_{50} + S_6 b_{52} + S_8 b_{54} = F_4$$

which can be used to solve for b_{50} , b_{52} , and b_{54} , while:

$$S_2 b_{51} + S_4 b_{53} + S_6 b_{55} = F_1$$

$$S_4 b_{51} + S_6 b_{53} + S_8 b_{55} = F_3 \quad \text{VIb}$$

$$S_6 b_{51} + S_8 b_{53} + S_{10} b_{55} = F_5$$

$$\frac{-14y_{-4} + 7y_{-3} + 13y_{-2} + 9y_{-1} + 0y_0 - 9y_1 - 13y_2 - 3y_3 + 14y_4}{1188}$$

which can be used to solve for b_{51} , b_{53} , and b_{55} . The set of equations in VIa has the same form for $n=4$, so that $b_{40} =$

b_{50} , $b_{42} = b_{52}$, and $b_{44} = b_{54}$ while the set VIb has the same form for $n=6$, so that $b_{51} = b_{61}$, $b_{53} = b_{63}$, and $b_{55} = b_{65}$. In other words, $b_{ns} = b_{n+1,s}$ for n and s both even or for n and s both odd. For example, to determine the third derivative for the best fit to a curve of third (or fourth) order, we would have:

$$S_2 b_{31} + S_4 b_{33} = F_1$$

$$S_4 b_{31} + S_6 b_{33} = F_3$$

$$\text{from which } b_{33} = \frac{S_2 F_3 - S_4 F_1}{S_2 S_6 - S_4^2}$$

When, for instance, $m=4$ ($2m+1=9$ points), we have from Vc that

$$S_2 = 60, S_4 = 708, S_6 = 9780$$

and

$$b_{33} = \frac{60F_3 - 708F_1}{60(9780) - (708)^2} = \frac{F_3 - 7F_1}{7128}$$

which reduces to:

The coefficients of y_i constitute the convoluting integers (Table VIII) for the third derivative of a cubic poly-

TABLE I

CONVOLUTES	SMOOTHING		QUADRATIC		CUBIC		A20	A30			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	-253										
-11	-138	-42									
-10	-33	-21	-171								
-09	62	-2	-76	-136							
-08	147	15	9	-51	-21						
-07	222	30	84	24	-6	-78					
-06	287	43	149	89	7	-13	-11				
-05	322	54	204	144	18	42	0	-36			
-04	387	63	249	189	27	87	9	9	-21		
-03	422	70	284	224	34	122	16	44	14	-2	
-02	447	75	309	249	39	147	21	69	39	3	-3
-01	462	78	324	264	42	162	24	84	54	6	12
00	467	79	329	269	43	167	25	89	59	7	17
01	462	78	324	264	42	162	24	84	54	6	12
02	447	75	309	249	39	147	21	69	39	3	-3
03	422	70	284	224	34	122	16	44	14	-2	
04	387	63	249	189	27	87	9	9	-21		
05	322	54	204	144	18	42	0	-36			
06	287	43	149	89	7	-13	-11				
07	222	30	84	24	-6	-78					
08	147	15	9	-51	-21						
09	62	-2	-76	-136							
10	-33	-21	-171								
11	-138	-42									
12	-253										
NORM	5175	8059	3059	2261	323	1105	143	429	231	21	35

TABLE II

CONVOLUTES	SMOOTHING			QUARTIC	QUINTIC		A40	A50			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	1265										
-11	-345	285									
-10	-1122	-114	11628								
-09	-1255	-285	-6460	340							
-08	-915	-285	-13005	-255	195						
-07	-255	-165	-11220	-420	-195	2145					
-06	590	30	-3940	-290	-260	-2860	110				
-05	1503	261	6378	18	-117	-2937	-198	18			
-04	2385	495	17655	405	135	-165	-160	-45	15		
-03	3155	705	28190	790	415	3755	110	-10	-55	5	
-02	3750	870	36660	1110	660	7500	390	60	30	-30	
-01	4125	975	42120	1320	825	10125	600	120	135	75	
00	4253	1011	44003	1393	883	11053	677	143	179	131	
01	4125	975	42120	1320	825	10125	600	120	135	75	
02	3750	870	36660	1110	660	7500	390	60	30	-30	
03	3155	705	28190	790	415	3755	110	-10	-55	5	
04	2385	495	17655	405	135	-165	-160	-45	15		
05	1503	261	6378	18	-117	-2937	-198	18			
06	590	30	-3940	-290	-260	-2860	110				
07	-255	-165	-11220	-420	-195	2145					
08	-915	-285	-13005	-255	195						
09	-1255	-285	-6460	340							
10	-1122	-114	11628								
11	-345	285									
12	1265										
NORM	30015	6555	260015	7429	4199	46189	2431	429	429	231	

TABLE III

CONVOLUTES	1ST DERIVATIVE			QUADRATIC			A21				
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	-12										
-11	-11	-11									
-10	-10	-10	-10								
-09	-9	-9	-9	-9							
-08	-8	-8	-8	-8	-8						
-07	-7	-7	-7	-7	-7	-7					
-06	-6	-6	-6	-6	-6	-6	-6				
-05	-5	-5	-5	-5	-5	-5	-5	-5			
-04	-4	-4	-4	-4	-4	-4	-4	-4	-4		
-03	-3	-3	-3	-3	-3	-3	-3	-3	-3	-3	
-02	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-01	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
00	0	0	0	0	0	0	0	0	0	0	0
01	1	1	1	1	1	1	1	1	1	1	1
02	2	2	2	2	2	2	2	2	2	2	2
03	3	3	3	3	3	3	3	3	3	3	3
04	4	4	4	4	4	4	4	4	4	4	4
05	5	5	5	5	5	5	5	5	5	5	5
06	6	6	6	6	6	6	6	6	6	6	6
07	7	7	7	7	7	7	7	7	7	7	7
08	8	8	8	8	8	8	8	8	8	8	8
09	9	9	9	9	9	9	9	9	9	9	9
10	10	10	10	10	10	10	10	10	10	10	10
11	11	11	11	11	11	11	11	11	11	11	11
12	12	12	12	12	12	12	12	12	12	12	12
NORM	1300	1012	770	570	408	280	182	110	60	28	10

TABLE IV

CONVOLUTES	1ST DERIVATIVE			CUBIC	QUARTIC		A31	A41			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	30866										
-11	8602	3938									
-10	-8525	815	84075								
-09	-20982	-1518	10032	6936							
-08	-29236	-3140	-43284	68	748						
-07	-33754	-4130	-78176	-4648	-98	12922					
-06	-35003	-4567	-96947	-7481	-643	-4121	1133				
-05	-33450	-4530	-101900	-8700	-930	-14150	-660	300			
-04	-29562	-4098	-95338	-8574	-1002	-18334	-1578	-294	86		
-03	-23806	-3350	-79564	-8179	-902	-17842	-1796	-532	-142	22	
-02	-16649	-2365	-56881	-5363	-673	-13843	-1489	-503	-193	-67	1
-01	-8558	-1222	-29592	-2816	-358	-7506	-832	-296	-126	-58	-8
00	0	0	0	0	0	0	0	0	0	0	0
01	8558	1222	29592	2816	358	7506	832	296	129	58	8
02	16649	2365	56881	5363	673	13843	1489	503	193	67	-1
03	23806	3350	79504	8179	902	17842	1796	532	142	-22	
04	29562	4098	95338	8574	1002	18334	1578	294	-86		
05	33450	4530	101900	8700	930	14150	660	-300			
06	35003	4567	96947	7481	643	4121	-1133				
07	33754	4130	78176	4648	98	-12922					
08	29236	3140	43284	-68	-748						
09	20982	1518	-10032	-6936							
10	8525	-815	84075								
11	-8602	-3938									
12	-30866										
NORM	1776060	197340	3634092	255816	23256	334152	24024	5148	1188	252	12

TABLE V

CONVOLUTES	1ST DERIVATIVE			QUINTIC	SEXIC	A51	A61				
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	-6356625										
-11	-11820675	-357045									
-10	-15593141	-654687	-15977364								
-09	-17062146	-840937	-28754154	-332684							
-08	-15896511	-878634	-35613829	-583549	-23945						
-07	-12139321	-752859	-34807914	-686099	-40483	-175125					
-06	-6301491	-478349	-26040033	-604484	-43973	-279975	-31380				
-05	544668	-106911	-10949942	-348823	-32306	-266401	-45741	-3084			
-04	6671883	265164	6402438	9473	-8671	-130506	-33511	-3776	-5758		
-03	9604353	489687	19052988	322378	16679	65229	-12	-1244	-4538	-90	
-02	6024183	359157	16649358	349928	24661	169819	27093	2166	2762	18	
-01	-8322182	-400653	-15033066	-255102	-14404	-78351	-14647	-573	-508	-2	
00	0	0	0	0	0	0	0	0	0	0	
01	8322182	400653	15033066	255102	14404	78351	14647	573	508	2	
02	-6024183	-359157	-16649358	-349928	-24661	-169819	-27093	-2166	-2762	-18	
03	-9604353	-489687	-19052988	-322378	-16679	-65229	12	1244	4538	90	
04	-6671883	-265164	-6402438	-9473	8671	130506	33511	3776	5758		
05	-544668	106911	10949942	348823	32306	266401	45741	3084			
06	6301491	478349	26040033	604484	43973	279975	31380				
07	12139321	752859	34807914	686099	40483	175125					
08	15896511	878634	35613829	583549	23945						
09	17062146	840937	28754154	332684							
10	15593141	654687	15977364								
11	11820675	357045									
12	6356625										
NORM	7153575	312455	5311735	81719	41990	20995	2431	143	143	1	

TABLE VI

CONVOLUTES	2ND DERIVATIVE			QUADRATIC		CUBIC	A22	A32			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	92										
-11	69	77									
-10	48	56	190								
-09	29	37	133	51							
-08	12	20	82	34	40						
-07	-3	5	37	19	25	91					
-06	-16	-8	-2	6	12	52	22				
-05	-27	-19	-35	-5	1	19	11	15			
-04	-36	-28	-62	-14	-8	-8	2	6	28		
-03	-43	-35	-83	-21	-15	-29	-5	-1	7	5	
-02	-48	-40	-98	-26	-20	-48	-10	-6	-8	0	2
-01	-51	-43	-107	-29	-23	-53	-13	-9	-17	-3	-1
00	-52	-44	-110	-30	-24	-56	-14	-10	-20	-4	-2
01	-51	-43	-107	-29	-23	-53	-13	-9	-17	-3	-1
02	-48	-40	-98	-26	-20	-48	-10	-6	-8	0	-2
03	-43	-35	-83	-21	-15	-29	-5	-1	7	5	
04	-36	-28	-62	-14	-8	-8	2	6	28		
05	-27	-19	-35	-5	1	19	11	15			
06	-16	-8	-2	6	12	52	22				
07	-3	5	37	19	25	91					
08	12	20	82	34	40						
09	29	37	133	51							
10	48	56	190								
11	69	77									
12	92										
NORM	26910	17710	33649	6783	3876	6188	1001	429	462	42	7

TABLE VII

CONVOLUTES	2ND DERIVATIVE			QUARTIC		QUINTIC	A42	A52			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	-429594										
-11	31119	-346731									
-10	298155	61845	-37791								
-09	413409	281979	11628	-96084							
-08	414786	358530	35802	45084	-121524						
-07	336201	331635	41412	105444	82251	-93093					
-06	207579	236709	34353	109071	153387	88803	-72963				
-05	54855	104445	19734	76830	137085	133485	98010	-10530			
-04	-100026	-39186	1878	26376	71592	95568	115632	20358	-4158		
-03	-239109	-172935	-15678	-27846	-11799	19737	53262	17082	12243	-117	
-02	-348429	-280275	-30183	-74601	-88749	-59253	-32043	117	4983	603	-3
-01	-418011	-349401	-39672	-105864	-141873	-116577	-99528	-15912	-6963	-171	48
00	-441870	-373230	-42966	-116820	-160740	-137340	-124740	-22230	-12210	-630	-90
01	-418011	-349401	-39672	-105864	-141873	-116577	-99528	-15912	-6963	-171	48
02	-348429	-280275	-30183	-74601	-88749	-59253	-32043	117	4983	603	-3
03	-239109	-172935	-15678	-27846	-11799	19737	53262	17082	12243	-117	
04	-100026	-39186	1878	26376	71592	95568	115632	20358	-4158		
05	54855	104445	19734	76830	137085	133485	98010	-10530			
06	207579	236709	34353	109071	153387	88803	-72963				
07	336201	331635	41412	105444	82251	-93093					
08	414786	358530	38802	45084	121524						
09	413409	281979	11628	-96084							
10	298155	61845	-37791								
11	31119	-346731									
12	429594										
NORM	4292145	2812095	245157	490314	478686	277134	160446	16731	4719	99	3

TABLE VIII

CONVOLUTES	3RD DERIVATIVE			CUBIC	QUARTIC		A33	A43			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	-506										
-11	-253	-77									
-10	-55	-35	-285								
-09	93	-3	-114	-204							
-08	196	20	12	-68	-28						
-07	259	35	98	28	-7	-91					
-06	287	43	149	89	7	-13	-11				
-05	285	45	170	120	15	35	0	-30			
-04	258	42	166	126	18	58	6	6	-14		
-03	211	35	142	112	17	61	8	22	7	-1	
-02	145	25	103	83	13	49	7	23	13	1	-1
-01	77	13	54	44	7	27	4	14	9	1	2
00	0	0	0	0	0	0	0	0	0		0
01	-77	-13	-54	-44	7	-27	-4	-14	-9	-1	-2
02	-149	-25	-103	-83	13	-49	-7	-23	-13	-1	1
03	-211	-35	-142	-112	17	-61	-8	-22	-7	1	
04	-258	-42	-166	-126	18	-58	-6	-6	14		
05	-285	-45	-170	-120	15	-35	0	30			
06	-287	-43	-149	-89	7	13	11				
07	-259	-35	-98	-28	-7	91					
08	-196	-20	-12	68	28						
09	-93	3	114	204							
10	55	35	285								
11	253	77									
12	506										
NORM	296010	32890	86526	42636	3876	7956	572	858	198	6	2

TABLE IX

CONVOLUTES	3RD DERIVATIVE		QUINTIC		SEXIC		A53	A63			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	118745										
-11	217640	23699									
-10	279101	42704	425412								
-09	290076	52959	749372	317655							
-08	244311	51684	887137	1113240	4915						
-07	144616	38013	787382	1231500	8020	93135					
-06	5131	13632	448909	932760	7975	141320	11260				
-05	-146408	-16583	-62644	259740	4380	113065	15250	1580			
-04	-266403	-43928	-598094	-589080	-1755	3800	8165	1700	2295		
-03	-293128	-55233	-908004	-1220520	-7540	-150665	-6870	-55	1280	65	
-02	-144463	-32224	-625974	-1007760	-7735	-260680	-16335	-2010	-2285	-40	
-01	284372	49115	748068	948600	5720	-169295	7150	645	500	5	
00	0	0	0	0	0	0	0	0	0	0	
01	-284372	-49115	-748068	-948600	-5720	-169295	-7150	-645	-500	-5	
02	144463	32224	625974	1007760	7735	260680	16335	2010	2285	40	
03	293128	55233	908004	1220520	7540	150665	6870	55	-1280	-65	
04	266403	43928	598094	589080	1755	-3800	-8165	-1700	-2295		
05	146408	16583	62644	-259740	-4380	-113065	-15250	-1580			
06	-5131	-13632	-448909	-932760	-7975	-141320	-11260				
07	-144616	-38013	-787382	-1231500	-8020	-93135					
08	-244311	-51684	-887137	-1113240	-4915						
09	-290076	-52959	-749372	-317655							
10	-279101	-42704	-425412								
11	-217640	-23699									
12	-118745										
NORM	5722860	749892	4249388	4247012	16796	2144809	9724	572	286	2	

TABLE X

CONVOLUTES	4TH DERIVATIVE			QUARTIC	QUINTIC		A44	A54			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	858										
-11	803	858									
-10	643	793	594								
-09	393	605	540	396							
-08	78	315	385	352	36						
-07	-267	-42	150	227	31						
-06	-597	-417	-130	42	17	621	84				
-05	-857	-747	-406	-168	-3	251	64	6			
-04	-982	-955	-615	-354	-24	-249	11	4	18		
-03	-897	-950	-680	-453	-39	-704	-54	-1	9	6	
-02	-517	-627	-510	-388	-39	-869	-96	-6	-11	1	
-01	253	133	0	-68	-13	-429	-66	-6	-21	-7	
00	1518	1463	969	612	52	1001	99	6	14	-3	
01	253	133	0	-68	-13	-429	-66	-6	-21	-7	
02	-517	-627	-510	-388	-39	-869	-96	-6	-11	1	
03	-897	-950	-680	-453	-39	-704	-54	-1	9	6	
04	-982	-955	-615	-354	-24	-249	11	4	18		
05	-857	-747	-406	-168	-3	251	64	6			
06	-597	-417	-130	42	17	621	84				
07	-267	-42	150	227	31	756					
08	78	315	385	352	36						
09	393	605	540	396							
10	643	793	594								
11	803	858									
12	858										
NORM	1430715	937365	408595	163438	8398	92378	4862	143	143	11	

TABLE XI

CONVOLUTES	5TH DERIVATIVE			QUINTIC	SEXIC		A55	A65			
POINTS	25	23	21	19	17	15	13	11	9	7	5
-12	-275										
-11	-500	-65									
-10	-631	-116	-1404								
-09	-636	-141	-2444	-44							
-08	-501	-132	-2819	-74	-55						
-07	-236	-87	-2354	-79	-88	-675					
-06	119	-12	-1063	-54	-83	-1000	-20				
-05	488	77	788	-3	-36	-751	-26	-4			
-04	753	152	2618	58	39	44	-11	-4	-9		
-03	748	171	3468	98	104	979	18	1	-4	-5	
-02	253	76	1938	68	91	1144	33	6	11	4	
-01	-1012	-209	-3876	-102	-104	-1001	-22	-3	-4	-1	
00	0	0	0	0	0	0	0	0	0	0	
01	1012	209	3876	102	104	1001	22	3	4	1	
02	-253	-76	-1938	-68	-91	-1144	-33	-6	-11	-4	
03	-748	-171	-3468	-98	-104	-979	-18	-1	4	5	
04	-753	-152	-2618	-58	-39	-44	11	4	9		
05	-488	-77	-788	3	36	751	26	4			
06	-119	12	1063	54	83	1000	20				
07	236	87	2354	79	88	675					
08	501	132	2819	74	55						
09	636	141	2444	44							
10	631	116	1404								
11	500	65									
12	275										
NORM	1300650	170430	1931540	29716	16796	83980	884	52	26	2	

PROGRAM 1

```

* SUBROUTINE SMOOTH -9 POINT 1
C 2
SUBROUTINE SMOOTH (N,NDATA,M,MDATA) 3
C 4
C INPUTS 5
C N NUMBER OF RAW DATA POINTS 6
C NDATA ARRAY OF N RAW DATA POINTS STORED IN MAIN PROGRAM 7
C DUMMY DIMENSION 8
C OUTPUTS 9
C M NUMBER OF SMOOTHED DATA POINTS = N-8 10
C MDATA ARRAY OF M SMOOTHED POINTS STORED IN MAIN PROGRAM 11
C MAY BE SAME REGION IN MAIN PROGRAM AS NDATA 12
C DUMMY DIMENSION 13
C 14
C DIMENSION NDATA(1000),MDATA(1000),NP(9) 15
C 16
C INITIALIZATION SEGMENT 17
C 18
M=N-8 19
DO 10 I=2,9 20
J=I-1 21
10 NP(I) = NDATA(J) 22
C 23
C SMOOTHING LOOP 24
C 25
DO 200 I=1,M 26
J=I+8 27
DO 11 K=1,8 28
KA = K+1 29
11 NP(K)=NP(KA) 30
NP(9) = NDATA(J) 31
NSUM=59*NP(5)+54*(NP(4)+NP(6))+39*(NP(3)+NP(7))+14*(NP(2)+NP(8))- 32
121*(NP(1)+NP(9)) 33
MDATA(I) = NSUM/231 34
200 CONTINUE 35
C 36
9999 RETURN 37
C 38
* END 39

```

nomial determined from a least squares fit to 9 points. Since the value of a_{33} is $3!b_{33}$, the denominator in the above expression must be divided by 6 to get the normalizer of 198 found in Table III.

In all of the above derivations, it has been assumed that the sampling interval is the same as the absolute abscissa interval—i.e., $\Delta x=1$. If not, the value of Δx must be included in the normalization procedure. Hence, to evaluate the s th derivative at the central point of a set of m values, based on an n th degree polynomial fit, we must evaluate

$$a_{ns,m} = s!b_{ns,m} = \frac{\sum_{i=-m}^{i=m} C_{i,m} y_i}{\Delta x^s N_{s,m}} \quad \text{VII}$$

Note that since $\Delta x^0=1$, the interval is of no concern in the case of smoothing.

Repeated Convolution. The process of convolution can be repeated if desired. For example, one might wish to further smooth a set of previously smoothed points, or to obtain the derivative only after the raw

data has been smoothed. Thus, if we convolute using p points the first time, and m points the second,

$$(f)_{s_1,s_2} = a_{s_1,s_2,m} = \frac{\sum_{i=-m}^{i=m} C_{i,s_2,m} a_{s_1,i,p}}{\Delta x^{(s_1+s_2)} N_{s_1,m}} \quad \text{VIII}$$

$$= \frac{\sum_{i=-m}^{i=m} C_{i,s_2,m} \sum_{j=-p}^{j=p} C_{j,s_1,p} F_j}{\Delta x^{(s_1+s_2)} N_{s_1,m} N_{s_1,p}} \quad \text{IX}$$

$$= \frac{\sum_{i=-m}^{i=m} \sum_{j=-p}^{j=p} C_{i,s_2,m} C_{j,s_1,p} F_{i+j}}{\Delta x^{(s_1+s_2)} N_{s_1,m} N_{s_1,p}} \quad \text{X}$$

thus:

$$(f)_{s_1,s_2} = \frac{\sum_{h=-(m+p)}^{h=(m+p)} d_h y_h}{N_h} \quad \text{XI}$$

where $h = i + j$

$$N_h = \Delta x^{(s_1+s_2)} N_{s_1,p} N_{s_2,m}$$

$$d_h = \sum_{i=-m}^{i=m} \sum_{j=-p}^{j=p} c_i c_j$$

Equation XI shows that one need not go through the convolution procedure twice, but can do a single convolution, using $2(m+p)+1$ points, and a table of new integers formed by combining the c 's.

For the case where a cubic smooth is to be followed by obtaining the quadratic first derivative using $m=2$ and $p=2$:

$$\begin{aligned}
 C_i &= -2, -1, 0, 1, 2, N_i = 10 \\
 C'_i &= -3, -12, 17, 12, -3, N'_i = 35 \\
 d_{-4} &= C_{-2} C'_{-2} = 6 \\
 d_{-3} &= C_{-2} C'_{-1} + C_{-1} C'_{-2} = -36 + 3 = -33 \\
 d_{-2} &= C_{-2} C'_0 + C_0 C'_{-2} + C_{-1} C'_{-1} = -34 - 12 = -46 \\
 d_{-1} &= C_{-2} C'_1 + C_1 C'_{-2} + C_{-1} C'_0 + C_0 C'_{-1} = -24 - 3 - 17 = -44 \\
 d_0 &= C_{-2} C'_2 + C_2 C'_{-2} + C_1 C'_{-1} - C_{-1} C'_1 + C_0 C'_0 = 0 \\
 d_1 &= \text{By symmetry} = 44 \\
 d_2 &= 46 \\
 d_3 &= 33 \\
 d_4 &= -6 \\
 N_h &= 350 \Delta x
 \end{aligned}$$

PROGRAM 2

```

* SUBROUTINE CENTER LORENZ
C
C SUBROUTINE CENTER(NPOINT,X,Y,TRANS,DENS)
C
C COMPUTATION OF PRECISE PEAK POSITION AND INTENSITY USING 9 POINTS
C TO A QUADRATIC, Y=A20+A21X+A22XSQ, X=(-A21/2A22). IN ORDER TO
C APPROXIMATE A LORENTZ CONTOUR, VALUES ARE CONVERTED TO ABSORBANCE
C AND THE RECIPROCAL ARE USED IN DETERMINING THE COEFFICIENTS BY
C ORTHOGONAL POLYNOMIALS
C
C DIMENSION NPOINT(25),DNS(9)
60 DO 61 I=1,9
    IP=I+8
61 DNS(I)=1./ALOG10F(1000./NPOINT(IP))
    P4=DNS(1)+DNS(9)
    P3=DNS(2)+DNS(8)
    P2=DNS(3)+DNS(7)
    P1=DNS(4)+DNS(6)
    A20=(-21.*P4)+(14.*P3)+(39.*P2)+(54.*P1)+(59.*DNS(5))
    A20=A20/231.
    A21=(4.*(DNS(9)-DNS(1)))+(3.*(DNS(8)-DNS(2)))+(2.*(DNS(7)-DNS(3)))
    A21=A21/60.
    A22=(28.*P4)+(7.*P3)-(8.*P2)-(17.*P1)-(20.*DNS(5))
    A22=A22/924.
    X=(-A21/(2.0*A22))
    Y=A20+X*(A21+X*A22)
    DENS=1.0/Y
    TRANS=1000./(10.0*DENS)
1000 RETURN
* END

```

APPENDIX II

The following eleven tables contain the convoluting integers for smoothing (zeroth derivative) through the fifth derivative for polynomials of degree two through five. They are in the form of tables of A_{ij} , where i is the degree of the polynomial and j is the order of the derivative. Thus, to obtain the third derivative over 17 points, assuming a fourth degree polynomial (A_{43}), one would use the integers in the column headed 17 of Table VIII.

APPENDIX III. COMPUTER PROGRAMS

The programming of today's high speed digital computers is still an art rather than a science. Different programmers presented with the same problem will, in general, write quite different programs to satisfactorily accomplish the calculation. Two programs are presented here as examples of the techniques discussed in this paper. They are written in the FORTRAN language, since this is one of the most widespread of the computer programming languages. Programmers using other languages should be able to follow the logic quite readily and make the appropriate translations. Each is

written as a subroutine for incorporation into a larger program as required.

Program 1 is a 9-point least squares smooth of spectroscopic data. The raw data has previously been stored by the main program in the region NDATA.

Lines 1 through 25 are explanatory and housekeeping to set up the initial conditions. The 9-point array NP contains the current set of data points to be smoothed. The main loop consists of lines 26 through 35. The inner loop, lines 28 through 30, moves the previous set of points up one position. The next point is added by line 31. In lines 32 and 33 the convoluting integers are multiplied by the corresponding data, and the products summed. In line 34, the sum is divided by the normalizing constant and the resulting smoothed point is stored.

Program 2 computes the precise peak position and intensity of a set of points which is known to contain a spectroscopic peak. In order to approximate a Lorentz contour (5), values are converted to absorbance and the reciprocals are used in determining the coefficients of a polynomial (6) having the form:

$$y = a_{20} + a_{21}x + a_{22}x^2$$

The center is at the point where $x = -a_{21}/2a_{22}$.

The data points to be used are points 9 through 17 stored in the array NPOINT and may have any value from 30 through 999. In the loop lines 12 through 14, each of these points is converted to absorbance, the reciprocal taken, and the result stored in the array DNS.

Since for a_{20} and a_{22} , the convolute function is symmetric about the origin, forming the sums $P4$ through $P1$ in lines 15 through 18 shortens the computation. The first constant, a_{20} is found in line 19, using the values of the 9-point convolute from Table I, and normalized in line 20.

The value of a_{21} is computed in lines 21-22 using the convolute from Table III (first derivative-quadratic). Note that this is an antisymmetric function. Table VI furnished the constants for the computation of a_{22} in line 24. Note that the normalizing factor of 924 is 2! times the value given in the table. X is computed in line 26 and the corresponding value of y is computed in line 27 by substituting the appropriate values into the polynomial. The absorbance or optical density (DENS) is, of course, the reciprocal of y (line 28).

LITERATURE CITED

- (1) Allen, L. C., Johnson, L. F., *J. Am. Chem. Soc.* **85**, 2668 (1963).
- (2) Anderson, R. L., Houseman, E. F., "Tables of Orthogonal Polynominal Values Extended to $N = 104$," Agricultural Experiment Station, Iowa State College of Agriculture and Mechanical Arts, Ames, Iowa, 1942.
- (3) Cole, LaMont C., *Science* **125**, 874 (1957).
- (4) Guest, P. G., "Numerical Methods of Curve Fitting," p. 349 ff., University Press, Cambridge, England, 1961.
- (5) Jones, R. N., Seshadri, K. S., Hopkins, J. W., *Can. Jour. Chem.* **40**, 334 (1962).
- (6) Kerawala, S. M., *Indian J. Phys.* **15**, 241 (1941).
- (7) Savitzky, A., *ANAL. CHEM.* **33**, 25A (Dec. 1961).
- (8) Whitaker, S., Pigford, R. L., *Ind. Eng. Chem.* **52**, 185 (1960).
- (9) Whittaker, E., Robinson, G., "The Calculus of Observations," p. 291 ff., Blackie and Son, Ltd., London & Glasgow, 1948.

RECEIVED for review February 10, 1964.
Accepted April 17, 1964. Presented in part at the 15th Annual Summer Symposium on Analytical Chemistry, University of Maryland, College Park, Md., June 14, 1962.

APPENDIX B

Block Radiometer Conversion Formulas

1) $V-.97 = R' (\bar{B}_{\text{ref}} - \bar{B}_{\text{targ}})$

2) $\bar{B} = \int_{\Delta\lambda} \text{filter } \phi(\lambda) B(\lambda, T) d\lambda$

3) $B(\lambda, T) = \frac{11909}{\lambda^5 (\text{EXP} (\frac{14388}{\lambda T}) - 1)}$

4) $\phi(\lambda)$ is filter transmission as function of λ filter function:

9.9,15
10.0,16
10.1,18
10.2,25
10.3,41
10.4,54
10.5,65
10.6,78
10.7,82
10.8,81
10.9,79
11.0,75
11.1,72
11.2,71
11.3,70
11.4,68
11.5,65
11.6,63
11.7,61
11.8,61
11.9,62
12.0,55
12.1,40
12.2,26
12.3,18
12.4,16
12.5,15

BMD07M
STEPWISE DISCRIMINANT ANALYSIS

1. GENERAL DESCRIPTION

- a. This program performs a multiple discriminant analysis in a stepwise manner. At each step one variable is entered into the set of discriminating variables. The variable entered is selected by the first of the following equivalent criteria:

- (1) The variable with the largest F value (see computational procedure).
- (2) The variable which when partialled on the previously entered variables has the highest multiple correlation with the groups.
- (3) The variable which gives the greatest decrease in the ratio of within to total generalized variances.

A variable is deleted if its F value becomes too low. The program also computes canonical correlations and coefficients for canonical variables. It plots the first two canonical variables to give an optimal two-dimensional picture of the dispersion.

- b. The output consists of:

- (1) Group means and standard deviations
- (2) Within groups covariance matrix
- (3) Within groups correlation matrix
- (4) At each step:
 - (a) Variables included and F to remove
 - (b) Variables not included and F to enter
 - (c) U statistic and approximate F statistic to test equality of group means
 - (d) Matrix of F statistics to test the equality of means between each pair of groups
- (5) At certain specified steps and after the last step:
 - (a) Discriminant functions
 - (b) Classification matrix

(6) For each case:

- (a) The posterior probability of coming from each group
- (b) Square of the Mahalanobis distance from each group

(7) Summary table. For each step of the procedure the following is tabulated:

- (a) Variable entered or removed
- (b) F value to enter or remove
- (c) Number of variables included
- (d) U statistic

(8) Eigenvalues, canonical correlation, and coefficients of canonical variables

(9) Plot of the first canonical variable against the second

(10) Residuals and canonical coefficients (optional)

c. Limitations per problem:

- (1) p, number of variables ($1 \leq p \leq 80$)
- (2) t, total number of groups ($2 \leq t \leq 80$)
- (3) j, number of Variable Format Card(s) ($1 \leq j \leq 16$)

d. Estimation of running time and output pages per problem:

Number of seconds = $.0006 p^2(mp + 2n) + 60$ (for IBM 7094)

Number of pages = $.02n(m + 2k) + .01(pg^2 + p^2) + p + 10$

where p = number of variables

t = total number of groups

n = total number of cases

m = 1 if the canonical analysis is to be performed
0 otherwise

k = number of steps at which the cases are to be
classified

2. ORDER OF CARDS IN JOB DECK

Cards indicated by letters enclosed in parentheses are optional.
All other cards must be included in the order shown.

a. System Cards

[Introduction, IV]

b. Problem Card

c. Covariance Weight Card(s)--COVAR

- d. Sample-size Card(s)
- e. Group Label Card(s)
- f. F-type Variable Format Card(s) [Introduction, III-C]
- g. F-type variable format for alternate output
- (h.) Data Input Cards
(Place data input deck here
if data input is from cards.)
- i. Subproblem Card
- (j.) Control-Delete Card

Repeat g. and (h.) as specified on Problem Card

...

Repeat b. through (h.) as desired.

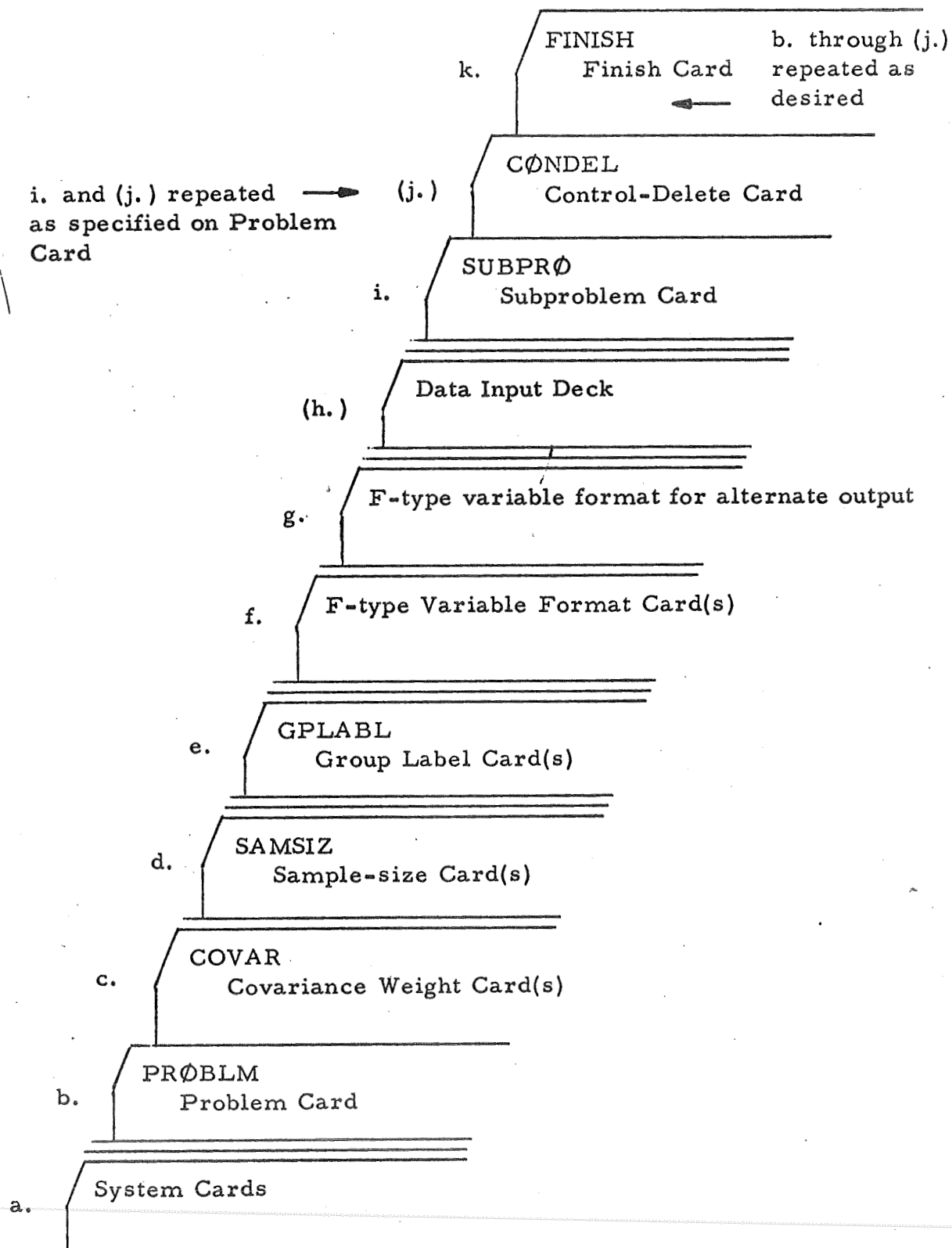
...
- k. Finish Card [Introduction, III-D]

3. CARD PREPARATION (SPECIFIC FOR THIS PROBLEM)

Preparation of the cards listed below is specific for this program.
All other cards listed in the preceding section are prepared according
to instructions in the Introduction.

- b. Problem Card (One Problem Card for each problem)

Col. 1-6	PRØBLM	(Mandatory)
Col. 7-12	Alphameric problem name	
Col. 13, 14	Number of variables ($1 \leq p \leq 80$)	
Col. 15, 16	Number of groups ($2 \leq t \leq 80$)	
Col. 17, 18	Number of Subproblem Cards	
Col. 19, 20	Number of Variable Format Cards ($1 \leq j \leq 16$)	
Col. 21, 22	Tape number if data is from tape ($\neq$ logical 1, 2 or 6); otherwise leave blank	

Example of Job Deck Set Up:

Col. 23, 24	Number of groups to be plotted on each page if the canonical analysis is to be done. Leave blank otherwise.
Col. 25-27	YES if group means are to be printed.
Col. 28-30	YES if standard deviations are to be printed.
Col. 31-33	YES if within groups covariance matrix is to be printed.
Col. 34-36	YES if within groups correlation matrix is to be printed.
Col. 37-39	YES if weighting of covariance matrix is desired.
Col. 40	Tape number of optional output for canonical variables.
Col. 41, 42	NO if alternate input tape is not to be rewound.
Col. 43	Tape number of optional output for coefficients of canonical variables.

c. Covariance Weight Card(s)

Col. 1-5	COVAR (Mandatory)
Col. 6	Blank
Col. 7-12	Weight for first group (keypunch decimal)
Col. 13-18	Weight for second group (keypunch decimal)
...	
Col. 67-72	Weight for eleventh group (keypunch decimal)

Additional cards may be used if needed.

d. Sample-size Card

Col. 1-6	SAMSIZE (Mandatory)
Col. 7-12	Number of cases in the first group.
Col. 13-18	Number of cases in the second group.
...	
Col. 67-72	Number of cases in the eleventh group.

If required, additional cards of the same form are used until all the groups are specified.

If the number of cases for a group is preceded by a minus sign, that group is deleted from the computation of everything except the group means and standard deviations, classification, and plotting. This allows classification of new cases.

e. Group Label Card

Col. 1-6	GPLABL (Mandatory)
Col. 7-12	Alphameric name of the first group
Col. 13-18	Alphameric name of the second group
...	
Col. 67-72	Alphameric name of the eleventh group

If required, additional cards of the same form are used until all the groups have been labeled. The first non-blank character of each group name is used for plotting.

i. Subproblem Card

Col. 1-6	SUBPRØ	(Mandatory)
Col. 7-10	Maximum number of steps (if blank, 2p is assumed)	
Col. 11-16	F value for inclusion (F to enter). Keypunch decimal. (if blank, .01 is assumed)	
Col. 17-22	F value for deletion (F to remove). Keypunch decimal. (if blank, .005 is assumed)	
Col. 23-28	Tolerance level. Keypunch decimal. (if blank, .0001 is assumed)	
Col. 29-31	YES if a Control-Delete Card is present.	
Col. 32-34	YES if the posterior probabilities are to be printed.	
Col. 35, 36 Col. 37, 38 Col. 71, 72	{ A list of integers. When the number of variables in the set of discriminating variables is equal to one of these numbers, the discriminant functions are printed, evaluated for each case and a classification matrix is printed.	

(j.) Control-Delete Card

Col. 1-6	CØNDEL	(Mandatory)
Col. 7	Control value for the first variable	
Col. 8	Control value for the second variable	
...		
Col. 72	Control value for the sixty-sixth variable	

If required, a second card of the same form is used to specify control values for the remaining variables. The control values specify the following:

0 or blank - Variable is not used for this subproblem

1 - Free variable

2 - Low level forced variable

.

.

9 - High level forced variable

If no Control-Delete Card is indicated on the Subproblem Card, a value of 1 will be assigned to each variable.

4. COMPUTATIONAL PROCEDURE

Notation: p = number of variables

g = number of groups used for the analysis. This excludes those with negative group size (see 3.d.)

t = total number of groups

n_m = number of cases in group m

n = total number of cases

x_{mki} = value of variable i for case k of group m

Assume for simplicity that the first g of the t groups are used for the analysis.

Step 1. The data are read and the following are formed:

$$\text{Means } \bar{x}_i = \frac{1}{n} \sum_{m=1}^g \sum_{k=1}^{n_m} x_{mki} \quad i = 1, 2, \dots, p$$

$$\checkmark \text{ Group means } \bar{x}_{mi} = \frac{1}{n_m} \sum_{k=1}^{n_m} x_{mki} \quad \begin{matrix} i = 1, 2, \dots, p \\ m = 1, 2, \dots, t \end{matrix}$$

Group standard deviations

$$s_{mi} = \sqrt{\frac{1}{n_m - 1} \sum_{k=1}^{n_m} (x_{mki} - \bar{x}_{mi})^2} \quad \begin{matrix} i = 1, 2, \dots, p \\ m = 1, 2, \dots, t \end{matrix}$$

Within and total cross-product matrices

$$W = \{w_{ij}\}; w_{ij} = \sum_{m=1}^g \sum_{k=1}^{n_m} (x_{mki} - \bar{x}_{mi})(x_{mkj} - \bar{x}_{mj})$$

$$T = \{t_{ij}\}; t_{ij} = \sum_{m=1}^g \sum_{k=1}^{n_m} (x_{mki} - \bar{x}_i)(x_{mkj} - \bar{x}_j)$$

$$i = 1, 2, \dots, p$$

$$j = 1, 2, \dots, p$$

Within groups covariance matrix

$$V = \{v_{ij}\}; v_{ij} = \frac{1}{n-g} w_{ij}$$

$$i = 1, 2, \dots, p$$

$$j = 1, 2, \dots, p$$

Within groups correlation matrix

$$R = \{r_{ij}\}; r_{ij} = \frac{w_{ij}}{\sqrt{w_{ii} w_{jj}}}$$

$$i = 1, 2, \dots, p$$

$$j = 1, 2, \dots, p$$

Step 2. At each step of the procedure the variables are divided into two disjoint sets; those included in the discriminant functions and those not included. Assume for simplicity that the first r are included.

$$\text{Let } W = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} \quad \text{and} \quad T = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix}$$

where W_{11} and T_{11} are $r \times r$.

$$\text{Let } A = \begin{bmatrix} W_{11}^{-1} & W_{11}^{-1} W_{12} \\ W_{21} W_{11}^{-1} & W_{22} - W_{21} W_{11}^{-1} W_{12} \end{bmatrix} = \{a_{ij}\}$$

$$\text{and } B = \begin{bmatrix} T_{11}^{-1} & T_{11}^{-1} T_{12} \\ T_{21} T_{11}^{-1} & T_{22} - T_{21} T_{11}^{-1} T_{12} \end{bmatrix} = \{b_{ij}\}$$

The coefficients and constant terms of the discriminant functions are computed:

$$c_{ki} = (n-g) \sum_{j=1}^r \bar{x}_{kj} a_{ij} \quad \begin{array}{l} i = 1, 2, \dots, r \\ k = 1, 2, \dots, g \end{array}$$

$$c_{k0} = -\frac{1}{2} \sum_{i=1}^r c_{ki} \bar{x}_{ki} \quad k = 1, 2, \dots, g$$

The following statistics are computed:

- a) F values for testing the differences between each pair of groups

$$F_{m\ell} = \frac{(n-g-r+1) n_m n_\ell}{r(n-g)(n_m + n_\ell)} \sum_{i=1}^r (c_{mi} - c_{\ell i}) (\bar{x}_{mi} - \bar{x}_{\ell i}) \quad \begin{array}{l} m = 1, 2, \dots, g \\ \ell = 1, 2, \dots, g \end{array}$$

with degrees of freedom r and $n-g-r+1$

- b) F values for each variable

- (1) If variable j has been entered

$$F_j = \frac{a_{jj} - b_{jj}}{b_{jj}} \frac{n-r-g+1}{g-1}$$

with degrees of freedom $g-1$ and $n-r-g+1$

- (2) If variable j has not been entered

$$F_j = \frac{b_{jj} - a_{jj}}{a_{jj}} \frac{n-r-g}{g-1}$$

with degrees of freedom $g-1$ and $n-g-r$

Under the usual normality assumptions these are the likelihood ratio tests of the equality over all g groups of the conditional distribution of variable j given the (remaining) entered variables.

- c) U statistic to test equality of group means

$$U = \text{Det}(W_{11}) / \text{Det}(T_{11})$$

with degrees of freedom $(r, g-1, n-g)$

- d) Approximate F statistic to test equality of group means

$$F = \frac{1 - U^{1/s}}{U^{1/s}} \frac{ms + 1 - rq/2}{rq}$$

$$\text{where } s = \sqrt{\frac{r^2 q^2 - 4}{r^2 + q^2 - 5}}, \text{ if } r^2 + q^2 \neq 5$$

$$s = 1, \text{ if } r^2 + q^2 = 5$$

$$m = n - \frac{r + q + 3}{2}$$

$$q = g - 1$$

its degrees of freedom are rq and $ms + 1 - rq/2$. If either r or q is 1 or 2 the approximation is exact.

Step 3. To move from one step to the next, one variable is added or removed from the discriminating set according to one of the following rules:

- a) If there are one or more variables which are entered, have a control value of 1 and an F value less than 'F to remove', the one with the smallest F will be deleted.
- b) If no variable satisfies a) and there are one or more variables which have not been included, pass the tolerance test, and have a control value greater than 1, the one with the largest control value and the largest F among all those with the same control value will be included.
- c) If no variable satisfies a) or b) and there are one or more variables not entered which pass the tolerance test, have a control value of 1 and an F value greater than 'F to enter', the one with the largest F will be entered.

If no variable satisfies a), b) or c), the process is terminated.

The tolerance test is used to maintain accuracy and may be controlled by the tolerance level on the Subproblem Card. A value of 10^{-n} specifies that roughly not more than n significant digits will be lost due to round-off.

Step 4. When the number of variables entered is equal to one of the numbers indicated on the Subproblem Card and after the last step the following are computed for $l = 1, 2, \dots, t$; $m = 1, 2, \dots, g$; $k = 1, 2, \dots, n_l$:

- a) Value of the m^{th} discriminant function evaluated at case k of group l

$$s_{lmk} = c_{mo} + \sum_{j=1}^r c_{mj} x_{mkj}$$

- b) Posterior probability of case k in group l having come from group m

$$p_{lmk} = \frac{\text{Exp}(s_{lmk})}{\sum_{i=1}^g \text{Exp}(s_{lil})}$$

- c) Square of Mahalanobis distance of case k in group m from group l

$$D_{lmk}^2 = \sum_{i=1}^r \sum_{j=1}^r (x_{mki} - \bar{x}_{li}) a_{ij} (x_{mkj} - \bar{x}_{lj})$$

This may be used as a chi-square variable with r degrees of freedom for classification purposed.

Step 5. At this point let p denote the number of variables which are included after the last step and let W and T be their within and total sum of product matrices. Let $B = T - W$. The eigenvalue problem

$$Bu_i = \lambda_i Wu_i \quad i = 1, 2, \dots, p$$

is solved to find coefficients, u_i , of canonical variables and the amount of dispersion λ_i explained by each canonical variable.

The vectors are normalized so that

$$u_i' W^{-1} u_j = \delta_{ij}$$

The canonical correlations of $\rho_1, \rho_2, \dots, \rho_p$ relative to the groups are then computed

$$\rho_i = \sqrt{\lambda_i / (1 + \lambda_i)}$$

For each case the first two canonical variables y and z are computed and plotted on a scattergram

$$y_{mk} = \sum_{j=1}^r u_{j1} (x_{mkj} - \bar{x}_j)$$

$$z_{mk} = \sum_{j=1}^r u_{j2} (x_{mkj} - \bar{x}_j)$$

5. REFERENCES

Anderson, T. W., Introduction to Multivariate Statistical Analysis, Wiley, 1958.

Efroymson, M. A., "Multiple Regression Analysis," Mathematical Methods for Digital Computers, Part V, (17). Edited by A. Ralston and H. S. Wilf, Wiley, 1960.

Rao, C. R., Advanced Statistical Methods in Biometric Research, Wiley, 1962.

This program was written by Paul Sampson, a member of the staff of Health Sciences Computing Facility, UCLA.

DISTRIBUTION LIST

<u>NASA-WASHINGTON</u>	<u>No of Copies</u>
T.A. George	1
J. Koutsandreas	1
Arch Park	1
J. Denoyer	1
 <u>NASA-HOUSTON</u>	
Tom Barnett	1
J.E. Dornbach	1
Charles M. Grant	1
W.E. Hensley (plus reproducible original)	4
Larry Korb	1
R. Piland	1
Ed Zeitler	
Retha Shirkay (Technical Information Dissemination Branch)	4
Al Watkins	1
 <u>NASA/MSFC-Huntsville</u>	
J. Fountain	1
 <u>NASA-GODDARD</u>	
Warren Hovis	1
William Nordberg	1
 <u>USGS</u>	
Bob Alexander, W.D.C.	1
Harry Smedes, Denver	1
R.W. Fary, W.D.C.	1
W.H. Hemphill, W.D.C.	1
Gordon Green, Denver	1
C.V. Robinove, W.D.C.	1
K. Watson, Denver	1
 <u>OTHERS</u>	
Len Bowden	1
Branner Library	1
R.N. Colwell	1
Michael Heathman	1
Dave Landgrebe	1

OTHERSNo. of Copies

R.K. Moore	1
Bob Peplies	1
Jack Quade	1
Dave Simonett	1
E.W.T. Whitten	1
Keenan Lee	1
R.B. MacDonald	1
V. Myers.	1