Income Distribution Within Groups,
Among Groups, and Overall:
A Technique of Analysis

by
Sherman Robinson

Research Program in Development Studies
Woodrow Wilson School
Princeton University
Princeton, New Jersey
August, 1976
Discussion Paper No. 65

Financial support for this work was provided by the World Bank through the Development Research Center and the Economic Analysis and Projections Department. The Bank does not necessarily agree with any views expressed in this paper.

This work was done as part of a joint project with Irma Adelman to build a model of income distribution in Korea. My thanks to her and also to Gregory Chow for discussions and assistance.

Note: Discussion papers of the Research Program in Development Studies are preliminary material circulated to stimulate discussion and critical comment. Please do not refer to discussion papers without permission of the author.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>II. Aggregating Group Distributions</td>
<td>3</td>
</tr>
<tr>
<td>III. Distribution Statistics</td>
<td>5</td>
</tr>
<tr>
<td>Mean and Variance of Income</td>
<td></td>
</tr>
<tr>
<td>Quantiles of the Distribution</td>
<td></td>
</tr>
<tr>
<td>Mean Incomes of Quantiles</td>
<td></td>
</tr>
<tr>
<td>Lorenz Curve and Gini Coefficient</td>
<td></td>
</tr>
<tr>
<td>Other Distribution Statistics</td>
<td></td>
</tr>
<tr>
<td>IV. Conclusion</td>
<td>15</td>
</tr>
</tbody>
</table>

## Appendices

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. User's Guide to the Income Distribution Program</td>
<td>16</td>
</tr>
<tr>
<td>Sample Output</td>
<td>19</td>
</tr>
<tr>
<td>B. Programmer's Guide</td>
<td>25</td>
</tr>
<tr>
<td>C. Program Listing</td>
<td>27</td>
</tr>
</tbody>
</table>
I. Introduction

The purpose of this paper is to describe a technique and an associated computer algorithm for deriving the overall distribution of income (the size distribution) given data on the distribution of income between and within aggregate groups in the economy. In particular, the technique can be used to map from the distribution of income by major factors of production such as labor, capital, and land (the functional distribution) into the overall size distribution. It can also be used with other group definitions such as by regions, sector of production, or socioeconomic categories. Any convenient definitions of groups will do so long as they are complete and mutually exclusive.

The technique was developed for an economy-wide model of income distribution in Korea and has also been applied in a very different model of socio-economic mobility and income distribution. See Adelman and Robinson (1976) and Robinson and Dervis (1974). Related approaches have been used by others -- see Rodgers, Hopkins and Wéry (1976) and Thorbecke and Sengupta (1972). The approach is clearly useful in economic models since it provides a way to take data economy-wide models usually generate such as factor incomes and, with some additional information, generate the overall income distribution.

In addition to its usefulness in models, the basic approach should also be very useful in the preparation of income distribution data for a country. It provides a framework within which one can integrate income distribution data from a number of different sources in order to generate the overall distribution. Using this technique, one can divide
the economy into any convenient complete set of disparate groups and then use data on each separate group to determine its within-group distribution. The data sources used to analyze each separate group might be entirely different. For example, one might use agricultural surveys for rural groups, household surveys for urban workers, tax data for the rich, and so forth. National accounts or input-output data can be used to analyze the between-group distribution and then the algorithm can be used to generate the overall size distribution. It thus can provide a method for reconciling income distribution data from a number of different sources.
II. Aggregating Group Distributions

The technique consists simply of numerically aggregating a number of different within-group income distributions whose functional form and parameters are known. The resulting aggregate distribution is thus completely specified numerically, even though it may not be able to be described by any convenient summary distribution function. All statistics of the aggregate distribution can be generated numerically.

Assume that there are $n$ different groups and that the distribution of income within each group is given by the probability distribution function:

$$f_i(y|\theta_i) \quad i = 1, \ldots, n$$

where $\theta_i$ is a vector of parameters for each distribution and $f_i$ reflects the functional form for the distribution within group $i$ (such as lognormal, Pareto, etc.). The overall distribution of income is given by:

$$f(y|\theta) = \sum_{i=1}^{n} w_i f_i(y|\theta_i) \quad (1)$$

where the weight $w_i$ is the population share of group $i$ and $\theta$ is the set of all parameters $\theta_i$ for all $i$. Note that since $0 \leq w_i \leq 1$ and $\sum w_i = 1$, the function $f(y|\theta)$ is indeed a probability distribution function. It is simply the weighted average of the separate within-group functions. Note also that since $f(y|\theta)$ is a sum of distributions and not the distribution of a sum of random variables, central limit theorems do not apply. The distribution $f(y|\theta)$ may in principal have any shape and be completely intractable analytically.

Given complete knowledge of $w_i$ and $\theta_i$, one can generate the overall distribution $f(y|\theta)$ numerically. While any statistics for the overall distribution are clearly a function of the parameter set $\theta$ and the population shares, $w_i$, it is possible to generate them numerically without attempting to deal analytically with the overall distribution function.
For the analysis of income distribution, a number of statistics are of interest which cannot be solved analytically but which must be calculated numerically. There are, of course, special cases of the distribution functions $f_1(y|\theta_1)$ which can yield a convenient and tractable form for the overall distribution function $f(y|\theta)$. The algorithm described here, however, always treats the overall distribution function numerically.

In the computer program described in the appendix, all the $f_1(y|\theta_1)$ distributions are specified as two parameter lognormal distributions. The lognormal distribution is commonly used to represent income distributions and seems to fit income distribution data very well.\(^1\) Note, however, that it is used here only to represent the within-group distributions. The overall distribution is thus a sum of lognormal distributions and may have any shape.\(^2\)

\(^1\)See Aitchison and Brown (1957), chapter 11.

\(^2\)If the within-group log variances are all the same and if the group mean incomes are distributed lognormally, then the overall distribution is also lognormal. See Aitchison and Brown (1957), p. 110. Usually, however, it is not reasonable to make either assumption when dealing with the functional distribution.
III. Distribution Statistics

Mean and Variance of Income

Assume that the mean income of each group is given by $\bar{y}_i$. Then
the overall mean income is:

\begin{equation}
(2) \quad \bar{y} = \sum_{i=1}^{n} w_i \bar{y}_i
\end{equation}

The overall variance of income, $s^2$, can be calculated from the decomposition
of variance formula:

\begin{equation}
(3) \quad s^2 = \sum_{i=1}^{n} w_i s_i^2 + \sum_{i=1}^{n} w_i (\bar{y}_i - \bar{y})^2
\end{equation}

where $s_i^2$ is the variance of incomes within group $i$ (assuming it is finite).

The variance of the logarithms of income is a commonly used
inequality measure and is especially interesting under the assumption of
lognormality since it is one of the parameters of the distribution. Denote
the two parameter lognormal cumulative distribution function for group $i$ as
$\Lambda(y | \mu_i, \sigma_i^2)$. Under the lognormal distribution, the variable $x = \log y$ is
distributed normally with mean $\mu_i$ and variance $\sigma_i^2$: $N(x | \mu_i, \sigma_i^2)$. Note that
in this case:

$$f_x(y | \theta_i) \, dy = d\Lambda(y | \mu_i, \sigma_i^2) = dN(x | \mu_i, \sigma_i^2)$$

when $y$ is distributed lognormally, the within-group mean and variance are
given by: \(^3\)

\begin{align}
(4) \quad \bar{y}_i &= \exp \left( \mu_i + \frac{1}{2} \sigma_i^2 \right) \\
(5) \quad s_i^2 &= \left[ \exp \left( 2 \mu_i + \sigma_i^2 \right) \right] \left[ \exp \left( \sigma_i^2 \right) - 1 \right]
\end{align}

\(^3\)See Aitchison and Brown (1957), p. 8.
The overall mean and variance are found by substituting (4) and (5) into (2) and (3). The overall log mean and log variance are given by:

\begin{align}
\mu &= \sum_{i=1}^{n} w_i \mu_i \\
\sigma^2 &= \sum_{i=1}^{n} w_i \sigma_i^2 + \sum_{i=1}^{n} w_i (\mu_i - \mu)^2
\end{align}

There is, of course, no presumption that the overall distribution is log-normal with parameters \( \mu \) and \( \sigma^2 \). Indeed, if one wished to fit a lognormal approximation to the overall distribution, it is not clear that these equations would even be a good way to estimate the parameters of the lognormal approximation. Equation (7), however, is the appropriate formula for calculating the overall log variance for use as an inequality measure.

**Quantiles of the Distribution**

Income distributions are commonly tabulated by quantiles such as quartiles or deciles. Quantiles are defined as the income ranges containing a specified share of the overall distribution, starting from zero. For example, the first decile is defined as the value of \( q_1 \) for which:

\[ \int_{0}^{q_1} f(y|\theta) \, dy = 0.1 \]

The second decile, \( q_2 \), is defined as:

\[ \int_{q_1}^{q_2} f(y|\theta) \, dy = 0.1 \]

and so forth to the last decile:

\[ \int_{q_9}^{\infty} f(y|\theta) \, dy = 0.1 \]

It requires nine quantile values to delineate ten decile income ranges.
For the lognormal distribution, the quantiles are easily calculated from the normal distribution. Define \( q_k^i \) as the \( k \)'th quantile for group \( i \) where group \( i \) has a within-group distribution which is lognormal. Define \( p_k \) as the share of the density associated with the choice of quantile (say, all equal 0.1 for deciles). The group quantiles can be defined iteratively as:

\[
\begin{align*}
(8) \quad \int_{q_{k-1}^i}^{q_k^i} dN(y|\mu_i, \sigma_i^2) &= p_k \\
&\quad k = 1, \ldots, 9
\end{align*}
\]

where \( q_0^i = 0 \). By a simple change of variable, these quantiles can be expressed in terms of the normal distribution. Let \( x = \log y \) and \( x_k^i = \log q_k^i \). Then

\[
\begin{align*}
(9) \quad \int_{x_{k-1}^i}^{x_k^i} dN(x|\mu_i, \sigma_i^2) &= p_k \\
&\quad k = 1, \ldots, 9
\end{align*}
\]

and \( x_0^i = -\infty \). Equation (9) is easily solved for the quantiles using standard algorithms for calculating the inverse normal integral.

The quantiles of the overall distribution, however, are not so easily calculated. It is necessary to solve the following equation for \( q_k \) given \( q_{k-1} \) and \( p_k \):

\[
(10) \quad \int_{q_{k-1}}^{q_k} f(y|\theta) \, dy = p_k \\
&\quad k = 1, \ldots, 9
\]

where \( q_0 = 0 \). There is certainly no simple relationship between \( q_k \) and the corresponding within-group quantiles \( q_k^i \). Note, however, that given \( q_{k-1} \) and \( q_k \), it is easy to use equation (10) to solve for \( p_k \) by simply summing the population shares within each group between the two incomes.
Equation (10) can be written as a non-linear algebraic equation in \( q_k \):

\[
Q(q_k) = \int_{q_{k-1}}^{q_k} f(y|\theta) \, dy - p_k = 0
\]

which can be evaluated and whose root we seek. This equation can be solved by a number of different techniques. In the computer program described in the appendix, a variant of Newton's method is used in which the derivative of \( Q(q_k) \) is calculated numerically.\(^4\) An initial guess for \( q_k \) is calculated by taking the weighted geometric mean of the corresponding within-group quantiles, \( q_k^{i} \). The technique/always converged within 5-10 iterations.

**Mean Incomes of Quantiles**

For both the within-group and overall distributions, it is interesting to know the mean income of people falling within given income ranges -- for example, the mean income of those in the lowest decile. Specify two incomes \( q_{k-1} \) and \( q_k \) corresponding to some quantile of the overall distribution. The mean income of those people whose income falls within the range \( q_{k-1} \leq y \leq q_k \) is simply the weighted average of the corresponding mean incomes of those within each group whose incomes are in the same range. Define the mean income of people within some quantile of the overall distribution as:

\[
m_k = \frac{1}{p_k} \int_{q_{k-1}}^{q_k} y f(y|\theta) \, dy
\]

where \( q_o = 0 \), and \( p_k \) is the share of the density defined by the quantile range:

\[
p_k = \int_{q_{k-1}}^{q_k} f(y|\theta) \, dy
\]

\(^4\)See Jarratt (1970) for a survey of the techniques available for solving such equations.
Similarly, for each group:

\[ m_k^i = \frac{1}{p_k^i} \int_{q_{k-1}^i}^{q_k^i} y f_i(y|\theta_i) \, dy \]

where \( q_{k-1} \) and \( q_k \) define a quantile of the overall distribution, \( q_0 = 0 \) and

\[ p_k^i = \int_{q_{k-1}^i}^{q_k^i} f_i(y|\theta_i) \, dy \]

It follows from the definition of \( f(y|\theta) \) that:

\[ m_k = \sum_{i=1}^{n} \frac{m_k^i}{\hat{w}_i} \frac{\hat{w}_i}{\sum_{j} \hat{w}_j p_k^j} \]

where \( \hat{w}_i = \frac{w_i f_k^i}{\sum_{j} w_j p_k^j} \)

Thus, to calculate the mean incomes of those falling within a specified quantile of the overall distribution defined by \( q_{k-1} \) and \( q_k \), one must calculate the mean income of those in each subgroup who fall within the same range.

For a lognormal distribution, it is possible to solve for \( m_k^i \) analytically. Equation (14) becomes:

\[ m_k^i = \frac{1}{p_k^i} \int_{q_{k-1}^i}^{q_k^i} y \, d\Lambda(y|\mu_i, \sigma_i^2) \]

\( q_0 = 0 \), and:

\[ p_k^i = \int_{q_{k-1}^i}^{q_k^i} d\Lambda(y|\mu_i, \sigma_i^2) \]

This integral can be solved in terms of the normal integral. Change variables \( x = \log y, y = e^x \)

and \( dx = \frac{1}{y} \, dy, \, dy = y \, dx \)

Thus \( x_k = \log q_k \) and \( x_0 = -\infty \).
From the definition of the lognormal distribution; equation (18) becomes:

\begin{equation}
\frac{1}{P_k} = \frac{x_k}{x_{k-1}} \int_{\frac{x_{k-1}}{x_k}}^1 \mathrm{d}N(x|\mu_i, \sigma_i^2)
\end{equation}

Equation (17) can also be written in terms of the normal distribution. Again, from the definition of the lognormal distribution.

\[ \frac{1}{\sigma_1 \sqrt{2\pi}} \exp\left[-\frac{1}{2} \left(\frac{\log(y) - \mu_1}{\sigma_1}\right)^2\right] \, \mathrm{d}y \]

Multiply both sides by \( y \), integrate between \( q_{k-1} \) and \( q_k \), change variables on the right hand side, and note that \( y = e^x \). The result is:

\[ \frac{1}{\sigma_1 \sqrt{2\pi}} \int_{q_{k-1}}^{q_k} y \cdot \frac{1}{\sigma_1 \sqrt{2\pi}} \int_{x_{k-1}}^{x_k} \exp \left[ x - \frac{1}{2} \left(\frac{x - \mu_1}{\sigma_1}\right)^2\right] \, \mathrm{d}x \]

Consider the term being exponentiated, \( \exp [-] \). Multiply it out and complete the square. The term becomes:

\[ -\frac{1}{2} \left[ \frac{x - (\mu_1 + \sigma_1^2)}{\sigma_1^2} \right]^2 + (\mu_1 + \frac{1}{2} \sigma_1^2) \]

Note that the second term does not involve \( x \) and can be factored out of the integral. Furthermore, from equation (4), \( \exp (\mu_1 + \frac{1}{2} \sigma_1^2) = \bar{y}_1 \). Thus the integral can be written:

\[ \bar{y}_1 \cdot \frac{1}{\sigma_1 \sqrt{2\pi}} \int_{x_{k-1}}^{x_k} \exp \left(-\frac{1}{2} \left[\frac{x - (\mu_1 + \sigma_1^2)}{\sigma_1}\right]^2\right) \, \mathrm{d}x \]

The integral can be seen to be a normal integral for a variable with mean \( (\mu_1 + \sigma_1^2) \) and variance \( \sigma_1^2 \). Thus, the quantile mean in equation (17) equals:

\begin{equation}
\frac{1}{n_k} = \frac{\int_{x_{k-1}}^{x_k} \mathrm{d}N(x|\mu_i, \sigma_i^2)}{\int_{x_{k-1}}^{x_k} \mathrm{d}N(x|\mu_i, \sigma_i^2)} \cdot \bar{y}_1
\end{equation}
If the within-group distributions are lognormal, equation (20) can be used to calculate the mean income of those with incomes in a specified range for each group. Equation (16) can then be used to calculate the mean income of those with incomes in the specified range for the overall distribution. This is the method used in the computer program described in the appendix.

**Lorenz Curve and Gini Coefficient**

A common way to present income distribution data is to plot the proportion of income receivers having income less than y along the horizontal axis (this is the cumulative distribution function) against the proportion of total income going to the same income receivers on the vertical axis. The resulting graph is a Lorenz curve, an example of which is given below.

![Lorenz Diagram](image)

If the distribution of income is perfectly equal, the Lorenz curve is the diagonal straight line. The more unequal is the distribution, the more the curve bows out from the diagonal and the greater is the shaded area. The Gini coefficient is a measure of income inequality based on the Lorenz curve and is equal to the ratio of the shaded area to the area...
of the triangle under the diagonal. Formally, for the overall distribution, the definition of the measure is:  

(21) \[ G = 1 - 2 \int_0^\infty \phi(y) dF(y) \]

where:

(22) \[ F(y) = \int_0^y f(t|\theta) \, dt, \text{ and} \]

(23) \[ \phi(y) = \frac{\int_0^y t f(t|\theta) \, dt}{\int_0^\infty t f(t|\theta) \, dt} \]

\( F(y) \) is simply the cumulative distribution function. Given that one can calculate the mean income of people falling within any income range, then it is straightforward to calculate \( \phi(y) \) numerically. Compare equations (12) and (23). The Lorenz diagram simply plots \( \phi(y) \) on the vertical axis against \( F(y) \) on the horizontal axis.

For the within-group distributions, which are assumed to be log-normal, the Gini coefficients can be derived analytically. They are given by:

(24) \[ G_i = 2 \cdot N \left( \frac{\sigma_i}{\nu_i^2} \right) \cdot 0.1 - 1 \quad \text{for } i = 1, \ldots, n \]

For the overall distribution, equation (21) must be evaluated numerically. In the computer program presented in the appendices, the calculation is done in the following steps. First, calculate quantiles (usually deciles) of the overall distribution and the corresponding incomes \( q_k \). Second, calculate the mean incomes of people falling within

---


\(^6\)See Aitchison and Brown (1957), p. 112.
each quantile. Third, using these means, calculate \( \phi(q_k) \) for all the quantiles. The points \( F(q_k) \) and \( \phi(q_k) \) fall on the Lorenz curve. Since the quantiles are chosen to contain equal proportions of the population (say, deciles), then the points \( F(q_k) \) are equally spaced along the horizontal axis. Fourth, integrate the Lorenz curve numerically based on the calculated points.

It is obvious that if the integration is done by simply connecting the points with straight lines, then the degree of inequality will be underestimated (as the diagram below shows).

![Diagram](image)

In the computer program described in the appendices, the straight line approximation is not used. Instead, an equal-interval quadrature method called Simpson's rule is used which fits a polynomial curve to successive sets of points and then integrates the fitted curve.\(^7\) The technique was tested by calculating the Gini coefficient numerically for a lognormal distribution (based on decile points) and comparing it with the value calculated from equation (24). The values are extremely close, within 0.5 percent of the actual value.

\(^7\)See Arden and Astill (1970), pp. 82-84.
Other Distribution Statistics

From the previous equations, one can generate empirically a complete description of all the within-group and overall distributions all as well as analyze the group composition of the over/distribution. Using this approach, it is interesting to calculate the composition by groups of quantiles of the overall distribution -- for example, the share of the people in the bottom decile who are rural. It is also interesting to all analyze the share of particular groups in quantiles of the over/distribution -- for example, the share of rural workers in the first decile of the overall distribution.

Given that the distribution can be generated empirically, it is easy to compute any aggregate measures of inequality based on it. A number of different measures have been used, generally based either on the frequency function or on the Lorenz curve. For a survey of such measures, see Sen (1973) or Szal and Robinson (1977). Certain aggregate statistics can be decomposed into within-group and between-group contributions. Equations (3) and (7) provide such a decomposition for the variance and log variance. Decompositions have also been devised for the Gini coefficient as well as for other measures.\(^8\)

\(^8\)See Sen (1973) and Szal and Robinson (1977).
IV. Conclusion

In the approach described here, the overall income distribution is generated given knowledge about the level and distribution of income within groups in the economy. Any convenient definition of groups can be used so long as it is complete and exclusive. The collection of groups must include the entire population and there can be no overlapping among groups. The approach can be used by model builders who face the problem that economy-wide economic models usually do not generate the size distribution but only the functional distribution. The approach can also be used as a framework for integrating and reconciling income distribution data from a variety of different sources.

In the computer program described in the appendices, all the within-group distributions are given by two-parameter lognormal distributions. The general approach described above certainly does not require this assumption and it would be possible to specify a completely different distribution for each group. The technique for computing the overall distribution does not depend on any particular specification of the within-group distributions.
Appendix A
User's Guide to the Income Distribution Program

This appendix describes how to use the program to aggregate a set of within-group income distributions. The program requires as input some control parameters; group names and a heading for labelling the output; and data on the population, mean income, and log variance for all the groups. In addition, the user may specify a set of absolute income ranges which will be used to generate an analysis of the within-group and overall distributions in the specified ranges.

The input data for each job are summarized in the following table, followed by a detailed discussion. As many jobs as desired may be done in the same run by simply including as many sets of parameter cards as desired, one for each run. After the last set of parameter cards, the user must include one card with the word END in columns 1-3.

Input Data for Each Job

<table>
<thead>
<tr>
<th>Number of cards</th>
<th>Variable</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HDG</td>
<td>A80</td>
</tr>
<tr>
<td>1</td>
<td>NG, NQ, NCL, NCODE</td>
<td>4I3</td>
</tr>
<tr>
<td>NG/8</td>
<td>XNAME</td>
<td>8A10</td>
</tr>
<tr>
<td>NG</td>
<td>Y, (\sigma^2), w</td>
<td>3F10.0</td>
</tr>
<tr>
<td>1</td>
<td>YCLS</td>
<td>4F10.0</td>
</tr>
</tbody>
</table>

HDG: a heading which will be printed at the top of each page of output. It can be up to 80 characters long (1 card). It must not have END as the first three characters.
NG: Number of groups. Must be less than 34.

NQ: Number of quantiles desired. Usually set to 10 (deciles). A separate analysis of the top five percent and one percent of the distribution will be generated in any case.

NCL: Number of specified income ranges for separate analysis. If NCL > 1, then the user must provide NCL - 1 values of YCLS to define the income ranges. If NCL ≤ 1, then no values of YCLS will be read and the YCLS card must be omitted. Note that it requires NCL - 1 incomes to define NCL quantile ranges. The bottom range is assumed to be from zero to the first value of YCLS and the top range is from the last value of YCLS to infinity. NCL must be less than or equal to 5.

NCODE: An integer code indicating the form in which the log variances are to be read in. If NCODE = 1, the log standard deviations, σ, must be provided. If NCODE = 0, then the log variances, σ², must be provided.

XNAME: Group names, up to 10 characters each. NG such names must be provided and will be read in 8 names per card, using as many cards as necessary.

Y: Mean income for each group.

σ²: Log variance for each group. If NCODE = 1, the log standard deviation σ, must be provided instead.

w: The population shares of each group. If desired, the absolute populations may be provided instead -- the program will automatically normalize the shares so they sum to one.

Y, σ², w: One card for each group (format 3F10.0). NG cards in all.

YCLS: NCL - 1 values of income to define NCL income ranges. If NCL ≤ 1, this card must be omitted.
To run multiple jobs in a single run, simply stack as many sets of parameter cards as desired, one for each run. At the end of the last set of parameter cards, the user must add one card with the word END in the first three columns.

The input cards for a sample program are listed below followed by a listing of the output from the program.

**Sample Job**

TEST RUN FOR DISTRIBUTION PROGRAM 8/9/76

<table>
<thead>
<tr>
<th>RICH</th>
<th>MIDDLE</th>
<th>POOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000.</td>
<td>0.45</td>
<td>15.0</td>
</tr>
<tr>
<td>300.</td>
<td>0.32</td>
<td>35.0</td>
</tr>
<tr>
<td>90.</td>
<td>0.50</td>
<td>50.0</td>
</tr>
<tr>
<td>50.0</td>
<td>200.0</td>
<td>800.0</td>
</tr>
<tr>
<td>END</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Test Run for Distribution Program 8/5/76

**Mean Incomes of Quantiles**

**Quantiles of 10.00 Percent.**

**CCL 11 is Overall Group Mean Incomes**

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCW</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 RICH</td>
<td>254.67</td>
<td>397.64</td>
<td>507.89</td>
<td>616.99</td>
<td>734.68</td>
<td>869.99</td>
<td>1036.29</td>
<td>1260.14</td>
<td>1615.02</td>
<td>2706.88</td>
<td>1000.00</td>
</tr>
<tr>
<td>2 MIDDLE</td>
<td>97.07</td>
<td>141.94</td>
<td>174.51</td>
<td>205.64</td>
<td>238.26</td>
<td>274.76</td>
<td>310.43</td>
<td>375.50</td>
<td>462.79</td>
<td>711.11</td>
<td>300.00</td>
</tr>
<tr>
<td>3 POOR</td>
<td>21.03</td>
<td>33.62</td>
<td>43.51</td>
<td>53.41</td>
<td>64.26</td>
<td>76.73</td>
<td>92.26</td>
<td>113.39</td>
<td>147.30</td>
<td>254.55</td>
<td>90.00</td>
</tr>
<tr>
<td>4 OVERALL</td>
<td>27.32</td>
<td>43.33</td>
<td>69.43</td>
<td>96.22</td>
<td>132.36</td>
<td>180.59</td>
<td>245.26</td>
<td>341.40</td>
<td>528.75</td>
<td>1330.26</td>
<td>300.00</td>
</tr>
</tbody>
</table>

Cols 1 - 10 are shares of income by quantiles of 10.00 percent.

CCL 11 is the Gini coefficient assuming lognormality.

CCL 12 is the Gini coefficient calculated numerically.

CCL 13 is the log variance.

CCL 14 is the geometric mean income, exp(xmu).

CCL 15 is mean income.

CCL 16 is group shares in total population.

CCL 17 is group shares in aggregate income.

Percent of total log variance due to within group variance = 34.790

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCW</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 OVERALL</td>
<td>0.9106</td>
<td>1.6109</td>
<td>2.3143</td>
<td>3.2075</td>
<td>4.4121</td>
<td>6.0199</td>
<td>8.1757</td>
<td>11.3804</td>
<td>17.6254</td>
<td>44.3432</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCW</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 RICH</td>
<td>0.3647</td>
<td>0.3632</td>
<td>0.4500</td>
<td>798.5164</td>
<td>1000.0000</td>
<td>15.0000</td>
<td>50.0000</td>
</tr>
<tr>
<td>2 MIDDLE</td>
<td>0.3108</td>
<td>0.3097</td>
<td>0.3200</td>
<td>255.6428</td>
<td>300.0000</td>
<td>35.0000</td>
<td>35.0000</td>
</tr>
<tr>
<td>3 POOR</td>
<td>0.3829</td>
<td>0.3811</td>
<td>0.5000</td>
<td>70.9290</td>
<td>90.0000</td>
<td>50.0000</td>
<td>15.0000</td>
</tr>
<tr>
<td>4 OVERALL</td>
<td>0.5679</td>
<td>0.5759</td>
<td>1.2345</td>
<td>158.7992</td>
<td>299.9998</td>
<td>100.0000</td>
<td>100.0000</td>
</tr>
</tbody>
</table>
### Percent Composition by Groups in Quantiles of Overall Distribution

Columns sum to one hundred

Row 4 is incomes of quantiles of 10.00 percent

<table>
<thead>
<tr>
<th>Column</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rich</td>
<td>0.00</td>
<td>0.01</td>
<td>0.04</td>
<td>0.21</td>
<td>0.81</td>
<td>2.39</td>
<td>5.96</td>
<td>14.55</td>
<td>39.40</td>
<td>86.62</td>
</tr>
<tr>
<td>Middle</td>
<td>0.15</td>
<td>1.42</td>
<td>6.03</td>
<td>18.07</td>
<td>39.42</td>
<td>61.82</td>
<td>75.40</td>
<td>76.86</td>
<td>57.74</td>
<td>13.09</td>
</tr>
<tr>
<td>Poor</td>
<td>99.85</td>
<td>98.57</td>
<td>93.93</td>
<td>81.72</td>
<td>59.77</td>
<td>35.79</td>
<td>18.65</td>
<td>8.56</td>
<td>2.85</td>
<td>0.28</td>
</tr>
<tr>
<td>Overall</td>
<td>38.63</td>
<td>58.26</td>
<td>81.55</td>
<td>112.48</td>
<td>154.24</td>
<td>209.53</td>
<td>285.74</td>
<td>409.56</td>
<td>700.34</td>
<td>0.0</td>
</tr>
</tbody>
</table>

### Percent Distribution of Groups in Quantiles of Overall Distribution

Columns sum to one hundred

<table>
<thead>
<tr>
<th>Column</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rich</td>
<td>0.03</td>
<td>0.00</td>
<td>0.03</td>
<td>0.14</td>
<td>0.54</td>
<td>1.59</td>
<td>3.97</td>
<td>9.76</td>
<td>26.27</td>
<td>57.75</td>
</tr>
<tr>
<td>Middle</td>
<td>0.04</td>
<td>0.41</td>
<td>1.72</td>
<td>5.16</td>
<td>11.26</td>
<td>17.66</td>
<td>21.54</td>
<td>21.96</td>
<td>16.50</td>
<td>3.74</td>
</tr>
<tr>
<td>Poor</td>
<td>99.97</td>
<td>99.71</td>
<td>98.79</td>
<td>96.34</td>
<td>91.95</td>
<td>97.16</td>
<td>93.73</td>
<td>92.22</td>
<td>96.37</td>
<td>96.06</td>
</tr>
</tbody>
</table>
TEST RUN FOR DISTRIBUTION PROGRAM 8/9/76

ANALYSIS OF TOP 5.00 PERCENT OF INCOME DISTRIBUTION

QUANTILE FOR OVERALL DISTRIBUTION IS 1090.4941
COL 1 IS MEAN INCOMES OF TOP 5.00 PERCENT OF EACH GROUP
COL 2 IS SHARES OF TOP 5.00 PERCENT IN GROUP INCOME
COL 3 IS PERCENT OF GROUP POPULATION IN TOP 5.00 PERCENT OF OVERALL INCOME DISTRIBUTION
COL 4 IS PERCENT COMPOSITION OF TOP 5.00 PERCENT OF OVERALL DISTRIBUTION

<table>
<thead>
<tr>
<th>ROW</th>
<th>1 RICH</th>
<th>2 MIDDLE</th>
<th>3 POOR</th>
<th>4 OVERALL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3300.36</td>
<td>16.50</td>
<td>32.11</td>
<td>96.33</td>
</tr>
<tr>
<td>2</td>
<td>841.55</td>
<td>14.03</td>
<td>0.52</td>
<td>3.62</td>
</tr>
<tr>
<td>3</td>
<td>313.53</td>
<td>17.42</td>
<td>0.01</td>
<td>0.05</td>
</tr>
<tr>
<td>4 OVERALL</td>
<td>1793.58</td>
<td>29.99</td>
<td>5.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

ANALYSIS OF TOP 1.00 PERCENT OF INCOME DISTRIBUTION

QUANTILE FOR OVERALL DISTRIBUTION IS 2164.7567

<table>
<thead>
<tr>
<th>ROW</th>
<th>1 RICH</th>
<th>2 MIDDLE</th>
<th>3 POOR</th>
<th>4 OVERALL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4890.85</td>
<td>4.89</td>
<td>6.68</td>
<td>99.73</td>
</tr>
<tr>
<td>2</td>
<td>1174.42</td>
<td>3.91</td>
<td>0.01</td>
<td>0.25</td>
</tr>
<tr>
<td>3</td>
<td>474.27</td>
<td>5.27</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>4 OVERALL</td>
<td>3045.32</td>
<td>10.15</td>
<td>1.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>
**TEST RUN FOR DISTRIBUTION PROGRAM 8/5/76**

**MEAN INCOMES OF GROUPS IN QUANTILE RANGE**
**QUANTILE INCOMES ARE GIVEN IN-final RCW**

<table>
<thead>
<tr>
<th>COLUMN</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCW</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 RICH</td>
<td>43.55</td>
<td>160.05</td>
<td>516.85</td>
<td>1499.21</td>
</tr>
<tr>
<td>2 MIDDLE</td>
<td>42.87</td>
<td>143.97</td>
<td>356.61</td>
<td>1006.99</td>
</tr>
<tr>
<td>3 POOR</td>
<td>33.58</td>
<td>97.10</td>
<td>282.47</td>
<td>972.69</td>
</tr>
<tr>
<td>4 OVERALL</td>
<td>33.62</td>
<td>110.26</td>
<td>383.75</td>
<td>1452.71</td>
</tr>
<tr>
<td>QUANTILES</td>
<td>50.00</td>
<td>200.00</td>
<td>800.00</td>
<td></td>
</tr>
</tbody>
</table>

**PERCENT SHARES OF GROUPS IN QUANTILE RANGE**
**QUANTIZED INCOMES ARE GIVEN IN-final RCW**

<table>
<thead>
<tr>
<th>COLUMN</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCW</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 RICH</td>
<td>0.00</td>
<td>1.95</td>
<td>48.16</td>
<td>49.89</td>
</tr>
<tr>
<td>2 MIDDLE</td>
<td>0.20</td>
<td>33.02</td>
<td>64.60</td>
<td>2.19</td>
</tr>
<tr>
<td>3 POOR</td>
<td>31.64</td>
<td>61.45</td>
<td>6.88</td>
<td>0.03</td>
</tr>
<tr>
<td>4 OVERALL</td>
<td>15.89</td>
<td>42.58</td>
<td>33.27</td>
<td>8.26</td>
</tr>
<tr>
<td>QUANTILES</td>
<td>50.00</td>
<td>200.00</td>
<td>800.00</td>
<td></td>
</tr>
</tbody>
</table>

**PERCENT COMPOSITION OF QUANTILES BY GROUPS**
**QUANTIZED INCOMES ARE GIVEN IN-final RCW**

<table>
<thead>
<tr>
<th>COLUMN</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCW</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 RICH</td>
<td>0.00</td>
<td>0.69</td>
<td>21.71</td>
<td>90.57</td>
</tr>
<tr>
<td>2 MIDDLE</td>
<td>0.43</td>
<td>27.15</td>
<td>67.95</td>
<td>5.26</td>
</tr>
<tr>
<td>3 POOR</td>
<td>99.57</td>
<td>72.17</td>
<td>10.34</td>
<td>0.17</td>
</tr>
<tr>
<td>4 OVERALL</td>
<td>100.00</td>
<td>100.00</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>QUANTILES</td>
<td>50.00</td>
<td>200.00</td>
<td>800.00</td>
<td></td>
</tr>
</tbody>
</table>
Appendix B
Programmer's Guide to the Income Distribution Program

A listing of the program is given in Appendix C. This appendix describes the basic architecture of the program. The program consists of a MAIN program and the following subprograms:

Group 1:  STATS
          CLASS

Group 2:  ADDER     MEANQ     PAGER
          CUMDF     MEANY     QUANT
          GINI      NORMAL    QUANT2
          LNORM     NORM1     QUANT3
          MATOUT    NORM2     QUANT4

Group 3:  PLOT
          RANKHI
          TSCALE
          PSSCALE

The Group 1 programs, STATS and CLASS, are the main driving routines for the program. STATS does the basic analysis of the within-group and overall distributions. CLASS does the analysis of the group and overall distributions using the income ranges specified in the vector YCLS.

The subprograms in group 2 are the main working routines of the program. Subroutine PAGER prints the heading at the top of each page and MATOUT is a utility program for printing a matrix.

Subroutine NORMAL is used to calculate the normal integral and its inverse. NORMAL, and only NORMAL, calls the two routines NORM1 and NORM2. Subprogram NORM1 calculates the standard normal integral and NORM2 calculates the inverse integral. Similar subroutines are usually part of the FORTRAN library at most computer installations. The appropriate library routines for an IBM 360 and a UNIVAC 1108 are indicated in subroutine NORMAL. The algorithms used in NORM1 and NORM2 are crude and it would be better to use a library routine, if available. The sample problem was done using
IBM library routines.

The subprograms in Group 3 are used to plot the distribution function and the Lorenz curve for the overall distribution. These subprograms are only called from subroutine STATS (which calls subroutine PLOT twice). If no plots are desired, all the programs in Group 3 can be deleted and the two calls to PLOT from STATS must also be deleted. In this case, the parameter IPRINT in STATS should be less than 2 to avoid printing the graph headings.

If it is desired to overlay the program, note that the groups of subprograms are hierarchical. No subprogram in a higher numbered group calls a subprogram from a lower numbered group. In addition, note that the Group 3 subprograms are only called by subroutine STATS.
Appendix C:

Program Listing
PROGRAM WRITTEN BY SHERMAN ROBINSON, PRINCETON UNIVERSITY,

MAIN PROGRAM TO RUN STATS, WHICH MAKES UP OVERALL DISTRIBUTION
FROM A SET OF GROUP DISTRIBUTIONS.

DIMENSION XBAR(35),XMU(35),SIGMA(35),W(35),
1 XNAME(35,3),XMEAN(35,5),POP(35,5),YCLS(5),SSUM(3)
DATA SSUM/0/,'&ALL',' /','XEND/',END'/
COMMON/PAGE/ IN MAIN,CLASS,HATOUT,PAGER,PLLOT,QUANT4,STATS
COMMON/PAGE/ NPAGE,NLINE,HDG(20)
COMMON/WORK/ IN MAIN,CLASS,GINI,QUANT3,QUANT4, AND STATS
COMMON /WORK/ NRDI,NCD,DUMY(35,20)

C
C INITIALIZE FIRST PAGE OF OUTPUT
NPAGE = 1
C
C DIMENSIONS FOR DUMMY, A WORK AREA
NRD = 35
NCD = 20
C
C FOR CARD READER, OR CONVERT TO TAPE UNIT
LR = 5
JB = 0

C READ ONE CARD (UP TO 80 CHARBS.) WHICH WILL BE A ONE LINE HEADING ON OUTPUT
100 READ LR,1000 (HDG(I),I=1,20)
1000 FORMAT(20A4)
   IF (HDG(1) .EQ. XEND) GO TO 9999
   JB = JB + 1
C
C READ NUMBER OF GROUPS TO BE AGGREGATED, NUMBER OF QUANTILES, AND
C NUMBER OF INCOME CLASSES.
READ LR,1001 NG,NQ,NCL,NCODE
1001 FORMAT(4I3)
   IF (NG .LT. 1 .OR. NG .GT. (NRD-2)) GC TO 900
   IF (NCL .GT. 5) GO TO 900
   IF (NQ .LE. 1 .OR. NQ .GT. 10) NQ = 10
C
C READ LABELS FOR GROUPS, UP TO 10 CHARACTERS
READ LR,1002 ((XNAME(I,J),J=1,3),I=1,NG)
1002 FORMAT(8(2A4,A2))
   DO 5 J = 1,3
5    XNAME(NG + 1,J) = SSUM(J)
C
C READ DATA
   DO 10 I = 1, NG
   READ LR,1003 XBAR(I),SIGMA(I),W(I)
1003 FORMAT(7F10.0)
C
NCODE .EQ. 1, READ STANDARD DEVIATION = SQRT(LOGVARIANCE) = SIGMA
NCODE .NE. 1, READ LOG VARIANCE, SIGMA ** 2
   IF (NCODE .NE. 1) SIGMA(I) = SQRT(SIGMA(I))
10 CONTINUE
C
KCL = NCL - 1
   IF (NCL .GT. 1) READ LR,1003 (YCLS(I),I=1,KCL)
   YCLS(NCL) = 0.0
   DO 20 I = 1, NG
   IF (XBAR(I) .LT. 1.E-8) GO TO 910
   XMU(I) = ALOG(XBAR(I)) - 0.5 * SIGMA(I) * SIGMA(I)
20 CONTINUE
   CALL STATS(XNAME,XBAR,XMU,SIGMA,W,NG,NQ,2,3)
C
   IF (NCL .GT. 1)
CALL CLASS(YCLS, YMEAN, POP, XBAR, XMU, SIGMA, W, NG, NCL, I, XNAME)

GO TO 100

WRITE(6,2000) JB, NG, NQ, NCL, NCODE
2000 FORMAT(1HO,'**** JOB ',I3,' ERROR IN PARAMETERS. NG = ',I4,', N
1Q = ',I4,', NCL = ',I4,', NCODE = ',I4,' ****')
STOP

WRITE(6,2001) JB, (XBAR(I), I=1,NG)
2001 FORMAT(1HO,'**** JOB ',I3,' NEGATIVE MEAN INCOME. XBAR(I) = ',
1 15E13.6, (/,40X,5E13.6))

STOP
END

SUBROUTINE STATS(XNAME,XMEAN,XMU,SIGMA,W,NP,N,INDX,IPRINT)

SUBROUTINE TO CALCULATE DISTRIBUTION STATISTICS AND PRINT THEM
WRITTEN BY S. ROBINSON AUGUST 1976.
INDX=1 ONLY CALCULATE QUANTILES AND MEAN INCOMES OF QUANTILES
INDX=2 ALSO CALCULATE QUANTILES AND MEANS OF OVERALL DISTRIBUTION, AS WELL AS NF GROUPS.
IPRINT=1 PRINT THE STATISTICS STORED IN DUMY.
IPRINT=2 IN ADDITION, CALCULATE AND PRINT A FREQUENCY DISTRIBUTION FOR THE OVERALL DISTRIBUTION.
(N-1) = NUMBER OF QUANTILES FOR N RANGES, SO N VALUES OF QMEAN ARE CALCULATED. IN DUMY, QUANTILES WILL BE STORED AS SHARES OF TOTAL INCOME RATHER THAN INCOME LIMITS.
QMEAN CONTAINS QUANTILE MEAN INCOMES FOR LAST GROUP.

DIMENSION W(1), QMEAN(35), XNAME(35,3), XMEAN(1), XMU(1), SIGMA(1),
1 SAVE(35), WW(35), XX(50), Y1(50), Y2(50),
2 Y3(50), XX2(1), WORK1(35), WORK2(35)
COMMON/PAGE/, NPAGE, NLINE, HDG(20)
COMMON/ WORK/, NRD, NCD, DUMY(35,20)
EQUIVALENCE (XX(1), DUMY(1,1)), (Y1(1), DUMY(1,3)), (Y2(1), DUMY(1,5)),
1 (Y3(1), DUMY(1,7)), (WORK1(1), DUMY(1,19)), (WORK2(1), DUMY(1,20))

UNIT = 1.0

DIMENSION OF DUMY
DIMENSION OF XNAME
INAME = 35

NORMALIZE WEIGHTS SO THAT THEY SUM TO 1.0, IF NECESSARY.

P = 100./FLOAT(N)
SUMW = 0.0
IF(NF.LE.0) GO TO 50
DO 4 IX=1,NP
4 SUMW = SUMW + W(IX)
IF(ABS(SUMW).LT.1.E-6) GO TO 50
DO 5 IX=1,NP
5 WW(IX) = W(IX)/SUMW

CALCULATE QUANTILES FOR EACH GROUP

NN = N + 1
CALL PAGER(100)
WRITE(6,900) P, NN, JJ J=1,NN
900 FORMAT(1H5,'MEAN INCOMES OF QUANTILES',//,1X,
1 'QUANTILES OF ',F5.2,' PERCENT.',',
2 //,1X,'COL ',I2,' IS OVERALL GROUP MEAN IN Comes',
3 //,2X,7HCOLUMN,7X,(11(I5,4X))
WRITE(6,901)
FORMAT (1H, 'ROW')
NLINE = NLINE + 7
IF(N.LE.1) GO TO 50
XMEAN(NF+1) = 0.0
DO 10 K=1,NP
  XMEAN(NF+1) = XMEAN(NF+1) + WW(K) * XMEAN(K)
  KK = K
  CALL QUANT2(SAVE,N,XMU(K),SIGMA(K),XXX)
  CALL MEANY(QMEAN,SAVE,XMU(K),SIGMA(K),XMEAN(K),N)
  CALL GINI(QMEAN,XMU(K),SIGMA(K),N,KK)
  DO 8 KK=1,N
    QMEAN(KK) = QMEAN(KK) * UNIT
    XXMEAN = XMEAN(KK) * UNIT
    WRITE(6,902) KK, (XNAME(KK,JJ), JJ=1,3), (QMEAN(JJ), J=1,N), XXMEAN
70 Q1 = Q2
  XXX = 0.0
  CALL QNORM(QMEAN, YMU, SIGMA, WW, NP, XXX)
  CALL LQNNORM(YMU, SIGMA, WW, NP, RATIO)
  DO 70 I=1, N
    Q2 = SAVE(I)
    IF(I.EQ.N) Q2 = 0.0
    CALL MEANY2(QMEAN(I), Q1, Q2, XM, SIGMA, WW, NP, WORK1, WORK2)
    SUM = SUM + QMEAN(I) / REAL(N)
    QMEAN(I) = QMEAN(I) * UNIT
 70 Q1 = Q2
  XXMEAN = XMEAN(NF+1) * UNIT
  KK = NP+1
  WRITE(6,902) KK, (XNAME(KK,JJ), JJ=1,3), (QMEAN(JJ), J=1,N), XXMEAN
  NLINE = NLINE + 7
  SUMY = 0.0
  DO 76 I = 1, NP
    DUMY(I,N+5) = XMEAN(I)
    DUMY(I,N+6) = WW(I) * 100.0
76 SUM = SUM + WW(I) * XMEAN(I)
  DO 77 I = 1, NP
    DUMY(I,N+7) = 100.0 * WW(I) * XMEAN(I) / SUM
    DUMY(NF+1,N+5) = XMEAN(NF+1)
    DUMY(NF+1,N+6) = 100.0
    DUMY(NF+1,N+7) = 100.0
    CALL GINI(QMEAN,YMU,SIGMA,N,NF+1)
100 CONTINUE
IF (IPRINT.LT.1) RETURN
NF2 = 0
IF(INDEX.EQ.2) NF2=1
N1 = N+1
N2 = N+7
NUMLIN = NLINE + 2 + NF + 22
CALL PAGER(NUMLIN)
IF (NLINE.GT.2) WRITE(6,1004)
FORMAT(1H0)
WRITE(6,1004) NP, (K,K=N1,N2), RATIO
1000 FORMAT(1H,'COLS 1 - ',I2,' ARE SHARES OF INCOME BY QUANTILES OF '1,F5.2,' PERCENT',/,'1X, 2 'COL ',I2,' IS THE GINI COEFFICIENT ASSUMING LOGNORMALITY',/,'1X, 3 'COL ',I2,' IS THE GINI COEFFICIENT CALCULATED NUMERICALLY',/,'1X, 4 'COL ',I2,' IS THE LOG VARIANCE',/,'1X,'COL ',I2, 5 ' IS THE GEOMETRIC MEAN INCOME, EXP(XMU)',/,'1X, 6 'COL ',I2,' IS MEAN INCOME',/,'1X, 7 'COL ',I2,' IS GROUP SHARES IN TOTAL POPULATION',/,'1X, 8 'COL ',I2,' IS GROUP SHARES IN AGGREGATE INCOME',/,'1X, 9 'PERCENT OF TOTAL LOG VARIANCE DUE TO WITHIN GROUP', 10 X 'VARIANCE = ',F6.3,/) 1005 FORMAT(1H,'PERCENT COMPOSITION BY GROUPS IN QUANTILES OF', 1 'OVERALL DISTRIBUTION',/,'1X,'COLUMN SUMS TO ONE HUNDRED',/,'1X,'ROW ',I2,' IS INCOMES OF QUANTILES OF ',F5.2,' PERCENT',/ ) 1015 FORMAT(1H,'PERCENT DISTRIBUTION OF GROUPS IN QUANTILES OF OVERALL DISTRIBUTION',/,'1X,'ROWS SUM TO ONE HUNDRED',/) 110 CONTINUE

CALL MATOUT(DUMY,XNAME,NF,NR,NCD,NR,NF,INAME,2,1)

CALL ADDER HERE TO GET ROW SUMS, NOT COI SUMS

CALL ADDER (DUMY,NF,NR,NCD)

DO 80 I = 1,NF
     DO 80 J = 1,N

80     DUMY(I,J) = DUMY(I,J) * 100.0 / DUMY(I,N+1)

CALL PAGER(NLNLIN)

IF (NLNLIN .EQ. 2) GO TO 81

WRITE(6,903)

WRITE(6,1015)

CALL MATOUT(DUMY,XNAME,NF,NR,NCD,NR,NF,INAME,2,0)

110 CONTINUE

CALL CALCULATE AND PRINT DISTRIBUTION STATISTICS FOR TOP QUANTILES

CALL QUANT4 (XNAME,XMEAN,XMU,SIGMA,WW,NF,INAME)

IF (IPRINT .LT. 2) RETURN

CALL CALCULATE AND PRINT A DENSITY FUNCTION OF THE OVERALL DISTRIBUTION

MN = 24

PROB = 0.85

CALL QUANT2(KK2,1,YMU,YSIGMA,PRCB)

DX = XX2(1) / FLOAT(NN)

N = NN+1

XX(1) = 0.0

Y1(1) = 0.0

Y2(1) = 0.0

F1 = 0.0

F2 = 0.0

DO 20 I=2,N

XX(I) = XX(I-1) + DX
Y = (ALOG(XX(I) - YMU) / YSIGMA
CALL NORMAL (F2, Y, IER)
Y2(I) = (F2 - F1) * 100.0
F1 = F2
CALL CUM DP (FF2, XX(I), YMU, SIGMA, WW, NF, 1)
Y1(I) = (FF2 - FF1) * 100.0
20 FF1 = FF2

REScale INCOME SO IT IS MEASURED IN UNITS.

DO 25 I = 1, NNN
25 XX(I) = XX(I) * UNIT
DDX = DX * UNIT

INSERT FOLLOWING CARD TO CHIT GRAPH OF FREQUENCY FUNCTION.
IF (IPRINT .GE. 2) GO TO 209

CALL PAGER(100)
WRITE(6, 1010) DDX, UNIT, YMU, YSIGMA
1010 FORMAT(1H 'OVERALL INCOME DISTRIBUTION', /, 'INCOME ',
1 'INTERVALS ARE ', 'F10.2', ' IN UNITS OF ', 'F6.0', ' ', /, '16X',
2 'FREQUENCIES ARE IN PERCENT', '/ ', '16X',
3 'SYMBOL * IS ACTUAL DISTRIBUTION', '/ ', '16X',
4 'SYMBOL * IS ESTIMATED DISTRIBUTION ASSUMING LOGNORMAL', '/ ', '15X',
5 'DISTRIBution WITH MU = ', 'F10.5', ' AND SIGMA = ', 'F10.5', ' / )

CALL PLOT(41, 101, XX, NNN, 2, Y1, Y2, Y1, Y1, 4, 1, X1, X2, X3, X4)
WRITE(6, 1011)
1011 FORMAT(1X, '/35X', 'ANNUAL INCOME')
NLINE = 100

CALL QUANT (Y3, 20, XMU, SIGMA, WW, NF, PROB)
SUM = 0.0
Q1 = 0.0
DO 210 I = 1, 20
Q2 = Y3(I)
IF (I .EQ. 20) Q2 = 0.0
CALL MEANY2(Y2(I), Q1, Q2, XMU, SIGMA, WW, NF, WORK1, WCRK2)
Q1 = Q2
XX(I) = 5.0 * FLOAT(I)
SUM = SUM + Y2(I)
Y1(I) = Y2(I)
IF (I .GT. 1) Y1(I) = Y1(I) * Y1(I - 1)
210 CONTINUE
DO 220 I = 1, 20
Y1(I) = 100.0 * Y1(I) / SUM
220 Y2(I) = XX(I)
XMAX = 100.0
XMIN = 0.0

USE THE FOLLOWING CARDS FOR A SMALL GRAPH

CALL PAGER(100)
WRITE(6, 1025)
CALL PLOT(36, 61, XX, 20, 2, Y2, Y1, Y3, 3, 1, 1, XMAX, XMIN, XMAX, XMIN)
WRITE(6, 1021)
CALL PAGER(100)
WRITE(6, 1025)
1025 FORMAT(1H 'LORENZ CURVE FOR THE OVERALL DISTRIBUTION', '/',
1 '1X, 'PERCENT OF INCOME')
CALL PLOT(51, 101, XX, 20, 2, Y2, Y1, Y3, Y3, 4, 1, XMAX, XMIN, XMAX, XMIN)
WRITE(6, 1021)
1021 FORMAT(1H0, 35X, 'PERCENT OF HOUSEHOLDS')
NLINE = 100
RETURN
50 WRITE(6, 1001) NF, N, SUMW
1001 FORMAT(1H0, '***ERROR IN PARAMETER IN STATS***', /, 5X, 'NF= ', I3, 
1 'N= ', I3, 'SUMW= ', E12.5)
NLINE = NLINE + 2
RETURN
END
SUBROUTINE CLASS(YCLS, YMEAN, POP, XMEAN, XMU, SIGMA, W, NG, NCL, IPRINT, 
1 XNAME)
SUBROUTINE TO CALCULATE POPULATION SHARE AND 
MEAN INCOME OF GROUPS FALLING WITHIN SPECIFIED 
INCOME RANGES GIVEN IN YCLS FOR NG GROUPS
NG = NUMBER OF GROUPS
XMU = XMU(NG) LOG MEANS
SIGMA = SIGMA(NG) LOG STANDARD DEVIATIONS
W = GROUP POPULATIONS OR SHARES
NCL = NUMBER OF INCOME CLASSES BY INCOME RANGE
YCLS = ARITHMETIC INCOMES OF INCOME CLASS YCLS(NC-1)
YMEAN = YMEAN(NG+1, NC) OUTPUT MEAN INCOME OF PEOPLE
IN INCOME CLASSES
POP = POP(NG+1, NC) POPULATION SHARES IN EACH CLASS. SHARES OF GROUP POP
IPRINT: PRINTING CODE. PRINT TABLE IF = 1
DIMENSION WW(35), YMEAN(35, 5), POP(35, 5), XNAME(35, 3), 
1 YCLS(NCL), XMU(NG), SIGMA(NG), W(NG), XMEAN(NG)
COMMON/PAGE/, NPAGE, NLINE, HDG(20)
COMMON /WORK/, NRD, NCD, DUMY(35, 20)
SET DIMENSIONS
NG IS NO. ROWS FOR YMEAN, POP
NCP IS NO. COLUMNS FOR YMEAN, POP
NRP = 35
NCP = 5
INAME = NRP
IF (NG .LT. 1 .OR. NG .GT. (NRP-2)) GO TO 900
IF (NCL .LT. 1 .OR. NCL .GT. NCP) GO TO 901
SUM = 0.0
5 DO 1 I = 1, NG
DO 6 I = 1, NG
1 WW(I) = W(I) / SUM
2 YCLS(NCL) = 0.0
3 NNG = NG + 1
4 Y1 = 0.0
5 DO 100 IC = 1, NCL
6 Y2 = YCLS(IC)
7 IF (IC .EQ. NCL) Y2 = 0.0
8 IF (IC .LT. NCL .AND. Y2 .LE. Y1) GO TO 910
9 CALL MEANZ2(X, Y1, Y2, XMU, SIGMA, WW, NG, YMEAN(1, IC), POP(1, IC))
10 YMEAN(NNG, IC) = X
11 F1 = 0.0
12 F2 = 1.0
13 IF (Y1 .GT. 1.0E-8) CALL CUMDF(F1, Y1, XMU, SIGMA, WW, NG, 1)
14 IF (Y2 .GT. 1.0E-8) CALL CUMDF(F2, Y2, XMU, SIGMA, WW, NG, 1)
15 POP(NNG, IC) = F2 - F1
100 IF (IPRINT .LT. 1) RETURN
NCLL = NCL - 1
RETURN
FORCE HEADER
CALL PAGER (100)
WRITE(6,1000)
1000 FORMAT(1H ,5X,'MEAN INCOMES OF GROUPS IN QUANTILE RANGES',/ ,5X,
1 'QUANTILE INCOMES ARE GIVEN IN FINAL ROW',/
) NLINE = NLINE + 3
CALL MATOUT(YMEAN,XNAME,NG,NCL,NPP,NCP,INAME,2,1)
WRITE(6,1001) (YCLS(K),K=1,NCLL)
1001 FORMAT(1HO,' QUANTILES 10,10*10.2,/
) NLINE = NLINE + 3
DO 10 I = 1, NNG
DO 10 J = 1, NCL
10 DUMY(I,J) = 100.0 * POP(I,J)
NUMLIN = 2 * (NLINE - 2)
CALL PAGER(NUMLIN)
IF (NLINE .GT. 2) WRITE(6,1004)
1004 FORMAT(1HO)
WRITE(6,1002)
1002 FORMAT(1H ,5X,'PERCENT SHARES OF GROUPS IN QUANTILE RANGES',/ ,6X,
1 'QUANTILE INCOMES ARE GIVEN IN FINAL RICW',/
) NLINE = NLINE + 3
CALL MATOUT(DUMY,XNAME,NG,NCL,NRD,NCD,INAME,2,1)
WRITE(6,1001) (YCLS(K),K=1,NCLL)
NLINE = NLINE + 3
DO 25 J = 1, NCL
SUM = 0.0
DO 15 I = 1, NG
DUMY(I,J) = Wf(I) * POP(I,J)
15 SUM = SUM + DUMY(I,J)
DO 20 I = 1, NG
20 DUMY(I,J) = 100.0 * DUMY(I,J) / SUM
25 CONTINUE
NUMLIN = NLINE + 11 + NG
CALL PAGER (NUMLIN)
IF (NLINE .GT. 2) WRITE(6,1004)
WRITE(6,1003)
1003 FORMAT(1H ,5X,'PERCENT COMPOSITION OF QUANTILES BY GROUPS',/ ,6X,
1 'QUANTILE INCOMES ARE GIVEN IN FINAL RICW',/
) NLINE = NLINE + 3
CALL ADDER(DUMY,NG,NCL,NRD,NCD)
CALL MATOUT(DUMY,XNAME,NG,NCL,NRD,NCD,INAME,2,1)
WRITE(6,1001) (YCLS(K),K=1,NCLL)
NLINE = NLINE + 3
RETURN
900 WRITE(6,1010) NG, NCL
1010 FORMAT(1HO,'*** ERROR IN CLASS. NG = ',1S,' NCL = ',1S)
NLINE = NLINE + 2
RETURN
910 WRITE(6,1010) NG, NCL
WRITE(6,1001) (YCLS(K),K=1,NCLL)
NLINE = NLINE + 5
RETURN
END
SUBROUTINE ADDER(Z,NROW,NCOL,KROW,KCOL)
DOUBLE PRECISION ROWSUM,COLSUM,SUM
C THE ONLY DOUBLE PRECISION STATEMENT
DIMENSION Z(KROW,KCOL)
IP(NCOL .EQ. 1) GO TO 5
DO 60 I=1,NROW
ROWSUM = 0.
DO 50 J=1,NCOL
50 ROWSUM = ROWSUM + Z(I,J)
60 Z(I,NCOL+1) = ROWSUM
5 CONTINUE
   IF(NROW.EQ.1) GO TO 15
   DO 20 I=1,NCOL
   COLSUM = 0.0
   DO 10 J=1,NROW
   10 COLSUM = COLSUM + Z(J,I)
   20 Z(NROW+1,I) = COLSUM
   15 IF(NROW.EQ.1 .OR. NCOL.EQ.1) RETURN
   
   CALCULATE SUM OF COLUMN SUMS
   SUM = 0.
   DO 30 I=1,NROW
30 SUM = SUM + Z(I,NCOL+1)
   Z(NROW+1,NCOL+1) = SUM
   RETURN
   END

SUBROUTINE CUMDF(F,X,XMU,SIGMA,W,NF,INDEX)
DIMENSION XMU(NF),SIGMA(NF),W(NF)

PROGRAM TO CALCULATE CUMULATIVE FREQUENCY OF SUM OF NF LOGNORMAL DISTRIBUTIONS GIVEN INCOME X. THE FREQUENCIES ARE WEIGHTED BY W.
IF INDEX=0, PROGRAM TAKES F AND DELIVERS GEOMETRIC MEANS OF THE QUANTILES FROM THE NF DISTRIBUTIONS.
IF INDEX=1, PROGRAM TAKES X AND DELIVERS F.

IF(INDEX.EQ.0) GO TO 110
   F = 0.0
   IF(X.GT.1.0E-8) GO TO 10
   WRITE(6,1001)X
1001 FORMAT(1HO,** ERROR IN CUMDF, X = ','E13.6')
   RETURN
10 Y = ALOG(X)
   DO 100 I=1,NF
      YY = (X-XMU(I))/SIGMA(I)
      CALL NORMAL(YY1,YY,IER)
   100 F = F + W(I)*YY1
   RETURN
110 CONTINUE
X = 0.0
   DO 200 I=1,NF
      CALL MORTINV(F,YY,IER)
      IF (IER.EQ.1) WRITE(6,1000) F, YY
   1000 FORMAT(1HO,** ERROR IN CUMDF, F = ','E12.6, 'YY = ','F12.6,'****')
      YY = YY * SIGMA(I) + XMU(I)
   200 X = X + W(I)*YY
   X = EXP(X)
   RETURN
END

SUBROUTINE GINI (X,XMU,SIGMA,N,K)
DIMENSION X(N)
COMMON /WCRK/NRD,NCD,DUMY(35,20)

SUBROUTINE TAKES N MEAN INCOMES IN QUANTILES (IN X) AND CALCULATES SHARES OF TOTAL INCOME BY QUANTILES AND STORES THE RESULTS IN ROW K OF DUMY. THE GINI COEFFICIENT IS ALSO CALCULATED AND STORED IN COLUMN N+1 OF DUMY. SIGMA**2 IS STORED IN COLUMN N+2.
XMU IS STORED IN COLUMN N+3

SUM = 0.0
DO 10 I=1,N
10 SUM = SUM + X(I)
DO 20 I=1,N
20 DUMY(K,I) = 100.0 * X(I) / SUM

CALCULATE GINI COEFFICIENT ASSUMING LOGNORMAL

Y = SIGMA / SQRT(2.0)
CALL NORMAL (FY,Y,IER)
DUMY(K,N+1) = 2.0 * FY - 1.0
DUMY(K,N+3) = SIGMA * SIGMA
DUMY(K,N+4) = EXP(XMU)

CALCULATE GINI COEFFICIENT
INTEGRATE LORENZ CURVE NUMERICALLY USING EQUAL INTERPOLATION POLYNOMIAL.
NOTE CORRECTION FOR ODD NUMBER OF INTERVALS.

DX = 1.0 / FLOAT(N)
NN = 2 * INT(FLOAT(N) / 2.0)
SUM = 0.0
CUM = 0.0
DO 70 I=1,N
CUM = CUM + 0.01 * DUMY(K,I)
ADD = 0.0
IF (NN .NE. N) GO TO 60
II = I
55 CONTINUE
IF (MOD(II,2) .EQ. 0) ADD = 2.0 * CUM
IF (MOD(II,2) .NE. 0) ADD = 4.0 * CUM
IF (I .EQ. N) ADD = CUM
GO TO 70

60 II = I + 1
IF (I .GT. 1) GO TO 55
ADD = 0.5 * CUM * DX + CUM
70 SUM = SUM + ADD
SUM = SUM * DX / 3.0
DUMY(K,N+2) = 1.0 - 2.0 * SUM
RETURN

SUBROUTINE LNORMAL(XMU2, SIGMA2, XMU, SIGMA, W, NF, RATIO)
DIMENSION X(1), XMU(NF), SIGMA(NF), W(NF)

SUBROUTINE TO ESTIMATE MEAN AND VARIANCE OF A POOL OF NF DISTRIBUTIONS WITH MEANS XMU AND VARIANCES SIGMA**2

XMU2 = 0.0
SIGMA3 = 0.0
SUMW = 0.0
XMUSQ = 0.0
DO 10 I=1,NF
SUMW = SUMW + W(I)
XMU2 = XMU2 + W(I) * XMU(I)
XMUSQ = XMUSQ + W(I) * XMU2 * XMU(I)
10 SIGMA3 = SIGMA3 + W(I) * SIGMA(I) * SIGMA(I)
XMU2 = XMU2 / SUMW
XMUSQ = XMUSQ / SUMW
SIGMA2 = SIGMA3 / SUMW + XMUSQ - XMU2 * XMU2
RATIO = 100.*SIGMA3/(SIGMA2*SUMW)
IF(SIGMA2 .LT. 0.0) GO TO 25
SIGMA2 = SQRT(SIGMA2)
RETURN

ENTRY INORM1 (XMU2, SIGMA2, XMU, SIGMA, W, NF, RATIO)

PROGRAM TO ESTIMATE PARAMETERS OF A LOGNORMAL DIST, XMU AND SIGMA, 
BY METHOD OF QUANTILES.  SEE AITCHISON AND BROWN, PAGES 40-42 
TWO SETS OF TWO QUANTILES ARE NEEDED.  FOR XMU, QUANTILES OF 
THE OVERALL DIST.  ARE CALCULATED AT P=.27 AND .73.  FOR SIGMA, 
THEY ARE P=.93 AND .07.

CALL QUANT (X, 1, XMU, SIGMA, W, NF, .73)
X1 = X(1)
CALL QUANT (X, 1, XMU, SIGMA, W, NF, .27)
X2 = X(1)
XMU2 = 0.5*(ALOG(X2) + ALOG(X1))

CALL NORINV (.93, ETA, IER)
IF (IER .EQ. 1) GO TO 25
ETA = 2.0 * ETA
CALL QUANT (X, 1, XMU, SIGMA, W, NF, .07)
X1 = X(1)
CALL QUANT (X, 1, XMU, SIGMA, W, NF, .93)
X2 = X(1)
SIGMA2 = [ALOG(X2) - ALOG(X1)]/ETA
RETURN

25 WRITE(6,1000)
1000 FORMAT(15H, '***FAILURE TO FIND ETA IN INORM****')
XMU2 = 0.0
SIGMA2 = 0.0
RETURN
END

SUBROUTINE MATOUT(Z, SNAME, NROW, NCOL, KROW, KCOL, KS, KK, LS)
DIMENSION Z(KROW, KCOL), SNAME(KS, 3)
COMMON/PAGE/ NPAGE, NLINE, HDG(20)

Z = SINGLE PRECISION MATRIX TO BE PRINTED
SNAME(I, J) = MATRIX OF ROW NAMES, I FOR ROW, J=1,3 FOR 3 WORDS TO FORM 
A 12 CHARACTER STRING
NROW, NCOL = NO. OF ROWS AND COLS TO BE PRINTED
KROW, KCOL = DIMENSIONS OF Z IN MAIN PROGRAM
KS = DIMENSION OF SNAME IN MAIN PROGRAM
KK = NO. OF DIGITS TO RIGHT OF DECIMAL POINT
LS = OPTION FOR SKIPPING LINE AND PRINTING COL SUMS
IF LS .LT. 1, (NROW*1) TH ROW IS PRINTED AFTER SKIPPING A LINE.
NPCOL = NO. OF COLUMNS ACROSS PRINTED PAGE.  SET TO 10 FOR BATCH
OUTPUT.  SET TO 5 FOR TERMINAL OUTPUT.
NPCOL = 10
LW = 6
NADD = 5
KF = KK + 1
IF(KF .GT. 7) KF=7
M= 1
MM = NPCOL
I? (MM .GT. NCOL) MM = NCOL
LSTART = 1
LEND = NROW
LSKIP = LS
C COL. NOS. ACROSS PAGE
WRITE(IW, 1011) (I, I=M, MM)

1011 FORMAT (1H, '1X, 'COLUMN',7X, 10(I5, 5X))
WRITE(IW, 1012)

1012 FORMAT (1H, 'ROW ')
NLNE = NLNE + 2
8 DO 30 I = LSTART, LEND
NLNE = NLNE + 1
GO TO (10, 11, 12, 13, 14, 15, 16), KP

10 WRITE(IW, 1000) I, (SNAME(I,J), J=1, 3), (Z(I,J), J=M, MM)

1000 FORMAT (1H, 12, 1X, 12A1, 12A3, 10(F10.0))
GO TO 30

11 WRITE(IW, 1001) I, (SNAME(I,J), J=1, 3), (Z(I,J), J=M, MM)

1001 FORMAT (1H, 12, 1X, 2A4, 2A2, 10(F10.4))
GO TO 30

12 WRITE(IW, 1002) I, (SNAME(I,J), J=1, 3), (Z(I,J), J=M, MM)

1002 FORMAT (1H, 12, 1X, 2A4, 2A2, 10(F10.2))
GO TO 30

13 WRITE(IW, 1003) I, (SNAME(I,J), J=1, 3), (Z(I,J), J=M, MM)

1003 FORMAT (1H, 12, 1X, 2A4, 2A2, 10(F10.1))
GO TO 30

14 WRITE(IW, 1004) I, (SNAME(I,J), J=1, 3), (Z(I,J), J=M, MM)

1004 FORMAT (1H, 12, 1X, 2A4, 2A2, 10(F10.4))
GO TO 30

15 WRITE(IW, 1005) I, (SNAME(I,J), J=1, 3), (Z(I,J), J=M, MM)

1005 FORMAT (1H, 12, 1X, 2A4, 2A2, 10(F10.5))
GO TO 30

16 WRITE(IW, 1006) I, (SNAME(I,J), J=1, 3), (Z(I,J), J=M, MM)

1006 FORMAT (1H, 12, 1X, 2A4, 2A2, 10(E10.4))
30 CONTINUE
IF(LSKIP.LE.0) GO TO 35
WRITE(IW, 2000)

2000 FORMAT (1H )
NLNE = NLNE + 1
LSKIP = 0
LSTART = NROW + 1
LEND = LSTART
GO TO 8

CCC DO WE HAVE MORE THAN NPCOL COLUMNS TO PRINT?
35 IF (MM .GE. NCOL) GO TO 75

CCC WILL THEY FIT ON SAME OUTPUT PAGE?
NUMLIN = NLNE + NADD + NROW
CALL PAGE(NUMLIN)
WRITE(IW, 2001)

2001 FORMAT (1H0)
NLNE = NLNE + 2
73 M = MM + 1
MM = MM + NPCOL
GO TO 5

75 RETURN
END

SUBROUTINE MEANQ(X, Q1, Q2, XMU, SIGMA, XMEAN, PROB)

C SUBROUTINE TO CALCULATE MEAN INCOME OF PEOPLE BETWEEN QUANTILES
C Q1 AND Q2 FOR A LOGNORMAL DISTRIBUTION.
IF (Q2 .GT. 1.E-9 .AND. Q2 .LE. Q1) GO TO 900
IF (ABS(XMEAN) .LE. 1.E-8) XMEAN=EXP(XMU+0.5*SIGMA*SIGMA)
XMU2 = XMU + SIGMA * SIGMA
X1 = 0.0
X2 = 0.0
IF (Q1 .GT. 1.E-8) X1 = (ALOG(Q1) - XMU) / SIGMA
IP (Q2 .GT. 1.E-8) X2 = (ALOG(Q2) - XMU) / SIGMA
F1 = 0.0
F2 = 1.0
IF (Q1 .GT. 1.E-8) CALL NORMAL(F1,X1,IER)
IF (Q2 .GT. 1.E-8) CALL NORMAL (F2,X2,IER)
PROB = F2 - F1
X1 = 0.0
X2 = 0.0
IF (Q1 .GT. 1.E-8) X1 = (ALOG(Q1) - XMU2) / SIGMA
IF (Q2 .GT. 1.E-8) X2 = (ALOG(Q2) - XMU2) / SIGMA
F1 = 0.0
F2 = 1.0
IF (Q1 .GT. 1.E-8) CALL NORMAL (F1,X1,IER)
IF (Q2 .GT. 1.E-8) CALL NORMAL (F2,X2,IER)
PROB2 = F2 - F1
X = 0.0
IF (ABS(PROB) .GT. 1.E-8) X = XMEAN * PROB2 / PROB
RETURN
900 WRITE(6,1000) Q1,Q2
1000 FORMAT(1HC, '***** ERROR IN MEANQ. Q1 = ', E13.6, ' Q2 = ', E13.6, '*')
RETURN
END
SUBROUTINE MEANQ(X,Q,XMU,SIGMA,XMEAN,N)
DIMENSION X(N),Q(N),P(20)
C
SUBROUTINE TO CALCULATE MEAN INCOMES OF QUANTILES OF
LOGNORMAL DIST. GIVEN IN Q
C
IF (N .GT. 20) GO TO 910
Q1 = 0.0
Q(N) = 0.0
SUMP = 0.0
SUMX = 0.0
DO 10 I = 1, N
Q2 = Q(I)
IF (Q2 .LE. Q1 .AND. Q2 .GT. 1.E-8) GO TO 900
CALL MEANQ(X(I),Q1,Q2,XMU,SIGMA,XMEAN,P(I))
Q1 = Q2
SUMP = SUMP + P(I)
10 SUMX = SUMX + X(I)*P(I)
IF (ABS(SUMP-1.0) .LT. 0.001 .AND. ABS(SUMX-XMEAN) .LT. 0.001*XMEAN)
1 RETURN
WRITE(6,2000) SUMP, SUMX, XMEAN
2000 FORMAT(1HC, '*** IN MEAN, SUMP = ', E13.6, ' SUMX = ', E13.6, 
1 ' XMEAN = ', E13.6)
RETURN
900 WRITE(6,1000) (Q(K),K=1,N)
1000 FORMAT(1HC, '***** ERROR IN MEANQ. Q(I) = ', F13.6))
RETURN
910 WRITE(6,1001) N
1001 FORMAT(1HC, '***** ERROR IN MEANQ, N = ', I5))
RETURN
END
SUBROUTINE MEANQ2(X,Q1,Q2,XMU,SIGMA,W,NF,XX,PR)
DIMENSION XMU(NF),SIGMA(NF),W(NF),XX(NF),PR(NF)
C
SUBROUTINE TO CALCULATE MEANS OF QUANTILES OF OVERALL DISTRIBUTION
C Q1 AND Q2 ARE TWO QUANTILES. QUANTILE MEAN OF DIST BETWEEN THE
C TWO QUANTILES IS CALCULATED BY SUMMING MEANS OF INDIVIDUAL DISTRI-
SUBROUTINE BETWEEN THE SAME TWO INCOMES. IF Q1=0.0, RANGE IS FROM MINUS INFINITY. IF Q2=0.0, RANGE IS TO PLUS INFINITY.

IF (Q2 .GT. 1.0E-8 .AND. Q2 .LT. Q1) GO TO 900
SUM = 0.0
DO 10 I = 1, NF
XMEAN = 0.0
CALL MEANO (XX(I), Q1, Q2, XMU(I), SIGMA(I), XMEAN, PR(I))
10 SUM = SUM + W(I) * PR(I)
X = 0.0
IF (SUM .LT. 1.0E-8) RETURN
DO 20 I = 1, NF
X = X + W(I) * PR(I) * XX(I) / SUM
RETURN
900 WRITE(6,1000) Q1, Q2
1000 FORMAT (1X, '**** ERROR IN MEANY2, Q1 = ','E13.6,' Q2 = ','E13.6')
RETURN
END

SUBROUTINE NORMAL (PY, X, IER)

SUBROUTINE TO CALCULATE NORMAL INTEGRAL FROM MINUS INFINITY TO X AND RETURN THE VALUE IN PY
ENTRY NORINV TAKES PY AND RETURNS X, IER = 1 IF THERE IS AN ERROR.

THE ROUTINE CAN USE IN-HOUSE LIBRARY ROUTINES FOR THE STANDARD NORMAL AND INVERSE NORMAL INTEGRALS. THEY ARE ERFC AND MDNRIS FOR IBM 360
RHNORM AND TINORM FOR UNIVAC 1108

IER = 0
IF (X .GT. 10.0) GO TO 100
IF (X .LT. -10.0) GO TO 110
PY = SNORM(X)
PY = 0.5 * ERP(-SQRT(0.5) * X)
CALL NORM1 (X, PY)
RETURN
100 PY = 1.0
RETURN
110 PY = 0.0
RETURN
ENTRY NORINV(PY, X, IER)
IER = 0
IF (PY .GT. 1.0 .OR. PY .LT. 0.0) GO TO 900
X = TINORM(PY, IER)
CALL MDNRIS (PI, X, IEB)
IF (IER .GT. 123) GO TO 900
CALL NORM2 (X, PY, IER)
IF (IER .EQ. 1) GO TO 900
RETURN
900 WRITE(6,1000) PY, X
1000 FORMAT (1X, '**** ERROR IN NORINV PY = ','E12.6,' X = ','E12.6',
1 ' ****')
IER = 1
X = 0.0
RETURN
END
SUBROUTINE NORM1 (X, P)

FUNCTION TO CALCULATE STANDARD NORMAL INTEGRAL FROM MINUS INFINITY TO X
X = INPUT VALUE
P = VALUE OF INTEGRAL
ACCURACY IS 7. E - 7
D = PROBABILITY DENSITY AT X
DATA A1,A2,A3,A4,A5,A6,A7 / 0.2316419,0.3193815,-0.3565638,
1.1784178,-1.021256,1.330274,0.3989423/
AX = ABS (X)
T = 1.0 / ( 1.0 + A1 * AX)
D = A7 * EXP ( -X * X / 2.0 )
P = 1.0-D*T*(((A6*T+A5)*T+A4)*T+A3)*T+A2)
IF (X) 5, 12, 10
5 P = 1.0 - ?
10 RETURN
END
SUBROUTINE NORM2 (X, P, IER)
SUBROUTINE TO CALCULATE INVERSE NORMAL INTEGRAL
P = VALUE OF INTEGRAL 0. LE. P. LE. 1
X = OUTPUT VALUE OF X
IER = ERROR CONDITION = 0 IF NO ERRCR
= 1 IF ERROR
MAX. ERROR IS .00045
D = PROBABILITY DENSITY AT X
DATA B1,B2,B3,B4,B5,B6,B7 / 2.515517,0.802853,0.10328,1.432788,
1.0.189269,0.001308,0.3989423 /
IER = 0
X = 1.0E70
D = X
IF (P) 11,14,12
11 IER = 1
GO TO 50
12 IF (P - 1.0) 17,15,11
14 X = -1.0E70
15 D = 0.0
GO TO 50
17 D = P
IF (D - 0.5) 19,19,18
18 D = 1.0 - D
19 T2 = ALOG (1.0 / (D * D))
T = SQRT (T2)
X = T - (B1*B2*T+B3*T2)/(1.0+B4*T+B5*T2+B6*T*T2)
IF (P - 0.5) 20,20,21
20 X = - X
21 D = B7 * EXP ( -X*X / 2.0 )
50 RETURN
END
SUBROUTINE PAGE(N)
COMMON/PAGE/, NPAGe, NLINe, HDG(2C)
IF (N .LT. 55) RETURN
WRITE(6,1000) (HDG(J),J=1,2C),NPAGe
1000 FORMAT(1H1,5X,20A4,5X,'PAGE ',I3)
WRITE(6,1001)
1001 FORMAT(1H )
NLINe = 2
NPAGe = NPAGe + 1
RETURN
END
SUBROUTINE QUANT(X,N,XMU,SIGMA,W, NF,PECE)
DIMENSION X(N),XMU(NF),SIGMA(NF),W(NF)

C PROGRAM TO ESTIMATE N-1 QUANTILES OF X DEFINING N RANGES EACH
CONTAINING 1/N OF THE FREQUENCY. THIS IS DONE FOR THE OVERALL DISTRIBUTION CALCULATED BY CUMDF.

FOR EACH QUANTILE, GUESS AN INITIAL VALUE BY TAKING THE GEOMETRIC MEAN OF THE SAME QUANTILE FOR ALL THE GROUPS.

USE NEWTON ITERATIONS WITH NUMERICAL ESTIMATE OF THE DERIVATIVE

MAXTRY = 25
TEST = 0.0001
NN = N-1
IF (N.EQ.1) NN=1
P = 0.0
DO 200 K=1,NN
P = P + 1.0/FLOAT(N)
IF (N.EQ.1) P=PROB

MAKE INITIAL GUESS
CALL CUMDF(P,XX,XMU,SIGMA,W,NF,0)
X1 = 0.0
P1 = 0.0
ITRY = 1

10 CONTINUE

TRY OUT THE VALUE OF XX
CALL CUMDF(PP,XX,XMU,SIGMA,W,NP,1)
IF (ABS (P-PP).LT.TEST) GO TO 200
IF (ITRY.GT.MAXTRY) GO TO 190

CALCULATE THE NUMERICAL DERIVATIVE AND ITERATE

ITRY = ITRY + 1
IF (ITRY.EQ.2) DERIV=XX/EP
IF (ABS (PP-PP1).GT.1.0.E-6) DERIV=(XX-X1)/(PP-PP1)
X1 = XX
P1 = PP
DDX = DERIV*(P-PP)
IF (ABS (DDX).GT.0.25*XX) DDX=0.25*XX*SIGN(1.0,DDX)
XX = XX + DDX
GO TO 10

190 WRITE(6,1000) K,ITRY
1000 FORMAT(1H,***QUANTILE NO ',I2,' DID NOT CONVERGE AFTER ',I2,
1 ' ITERATIONS IN QUANT***')

200 X(K) = XX
RETURN
END

SUBROUTINE QUANT2(X,N,XMU,SIGMA,PROB)
DIMENSION X(N)

PROGRAM TO CALCULATE N-1 INCCME QUANTILES FOR LOGNORMAL DISTRIBUTION EACH QUANTILE CONTAINS 1/N OF THE PROBABILITY. IF N=1, ONLY ONE QUANTILE IS CALCULATED, USING PROB.
SEE ATCHISON AND BROWN, PAGES 8-9

NN = N-1
IF (N.EQ.1) NN=1
P=0.0
DO 100 K=1,NN
P = P + 1.0/FLOAT(N)
IF (N.EQ.1) P=PROB
CALL NORINV(P,YY,IER)
IF (IER .NE. 1) GO TO 95
Y = YY*SIGMA + XMU
X(K) = EXP(Y)
GO TO 100
95 WRITE(6,1000) K
1000 FORMAT(1H **IN QUANT2. QUANTILE 'i2,' FAILED TO SOLVE**) 
    X(K) = 0.0 
100 CONTINUE 
RETURN 
END 
SUBROUTINE QUANT3(X, XMU, SIGMA, W, NF, N) 
DIMENSION X(1), XMU(1), SIGMA(1), W(1) 
COMMON /WORK/ NR, NCD, DUMY(35,20) 
SUBROUTINE TO CALCULATE SHARES OF TOTAL DISTRIBUTION QUANTILES 
GIVEN IN X THAT COME FROM VARIOUS GROUPS. RESULTS ARE 
STORED IN DUMY 
    NN = N - 1 
    DO 10 I=1,NF 
    II = I 
    F1 = 0.0 
    DO 5 J=1,N 
    JJ = J 
    IF (ABS(X(J)).LT.1.E-8) GC TC 200 
    Y = (ACOS(X(J)) - XMU(I)) / SIGMA(I) 
    CALL NORMAL(P2,Y,IER) 
    DUMY(I,J) = P2 - F1 
      5 F1 = F2 
    DUMY(I,N) = 1.0 - F1 
10 CONTINUE 
USING SHARES IN GROUPS, EEDC CELLS IN DUMY AS FREQUENCIES, 
THEN AS SHARES 
    DO 15 I=1,NF 
    DO 15 J=1,N 
15 DUMY(I,J) = DUMY(I,J) * W(I) 
    CALL ADDEF (DUMY, NF, N, NR, NCD) 
    DO 20 I=1,NF 
    DO 20 J=1,N 
20 DUMY(I,J) = 100. * DUMY(I,J) / DUMY(NF+1,J) 
PUT QUANTILES IN NF+1 ROW OF DUMY 
    UNIT= 1.0 
    DO 25 J=1,N 
25 DUMY(NF+1,J) = UNIT * X(J) 
    DUMY(NF+1,N) = 0.0 
200 WRITE(6,1000) II, JJ 
1000 FORMAT(1H **ERROR IN QUANT3. GROUP 'i2,' QUANTILE 'i2, 
1 ' **' ) 
RETURN 
END 
SUBROUTINE QUANT4(XNAME, XMEAN, XMU, SIGMA, W, NF, INAME) 
DIMENSION XNAME(INAME,3), XMEAN(1), XMU(1), SIGMA(1), W(1), Q(1), 
1 WORK1(35), WORK2(35) 
COMMON /PAGE/ NPAGE, NLINE, HWND(20) 
COMMON /WORK/ NR, NCD, DUMY(35,20) 
EQUIVALENCE (DUMY(1,19), WORK1(1)), (DUMY(1,20), WORK2(1)) 
SUBROUTINE TO CALCULATE AND PRINT DISTRIBUTION STATISTICS FOR 
TOP 5 PERCENT AND TOP 1 PERCENT OF DISTRIBUTION
UNIT = 1.0
Q2 = 0.0
DO 100 IJ=1,2
IF (IJ.EQ.1) P=.95
IF (IJ.EQ.2) P=.99
PP = 1.0 - P
PPP = PP*100.
DO 10 I=1,NF
II = I
CALL QUANT2(Q,1,XMU(I),SIGMA(I),P)
Q1 = Q(1)
IF (Q1 .LT. 1.E-6) GO TO 900
CALL MEANQ(XX,Q1,Q2,XMU(I),SIGMA(I),XMEAN(I),P3)
IF (ABS(PP-P3).GT.0.005) WRITE(5,1010) I,P,P3,PP
1010 FORMAT(1H7,'*** IN QUANT4, I = ',I4,' P = ',E13.6,' P3 = ',E13.6,' PP = ',E13.6)
IF (ABS(PP-P3).GT.0.005) NLINE = NLINE + 2
DUMY(I,1) = XX*UNIT
DUMY(I,2) = 100.*XX*PP/XMEAN(I)
C
CALL QUANT(Q,1,XMU,SIGMA,W,NF,P)
Q1 = Q(1)
CALL MEAN2(XX,Q1,Q2,XMU,SIGMA,W,NF,WORK1,WORK2)
DUMY(NF+1,2) = 100.*XX*PP/XMEAN(NF+1)
DUMY(NF+1,1) = UNIT*XX
SUM = 0.0
DO 20 I=1,NF
X1 = (ALCG(Q(1)) - XMU(I))/SIGMA(I)
CALL NORMAL (XX1,X1,IER)
DUMY(I,3) = 100.**(1.0-XX1)
DUMY(I,4) = DUMY(I,3)*W(I)
20 SUM = SUM + DUMY(I,4)
DO 25 I=1,NF
DUMY(I,4) = 100.*DUMY(I,4)/SUM
C
DUMY(NF+1,3) = PPP
DUMY(NF+1,4) = 100.
NUMLIN = NLINE + 8 + NF
IF (IJ .EQ. 1) NUMLIN = 100
CALL PAGER(NUMLIN)
IF (NLINE .LE. 2) GO TO 50
WRITE(6,1001)
1001. FORMAT(1H )
NLINE = NLINE + 1
50 WRITE(6,1002) PPP,Q1
1002 FORMAT(1H5X,'ANALYSIS OF TOP ',F5.2,' PERCENT OF INCOME ',F12.4,1x,'QUANITILE FOR OVERALL DISTRIBUTION IS ',F4.2,1x,'DISTRIBUTION',//,1x,'OVERALL DISTRIBUTION IS ',F4.2,1x,'OVERALL DISTRIBUTION')
NLINE = NLINE + 3
IF (IJ .EQ. 2) GO TO 55
WRITE(6,1003) PPP,PPP,PPF,PPP
1003 FORMAT(1H',COL 1 IS MEAN INCOMES OF TOP ',1F4.2,' PERCENT OF EACH GROUP',//,1x,
2 'COL 2 IS SHARES OF TOP ',F4.2,' PERCENT IN GROUP INCOME',//,1x,
3 'COL 3 IS PERCENT OF GROUP POPULATION IN TCP ',F4.2,4x,'PERCENT OF OVERALL INCOME DISTRIBUTION',//,1x,
5 'COL 4 IS PERCENT COMPOSITION OF TOP ',F4.2,' PERCENT OF ',//,1x,
6 'OVERALL DISTRIBUTION')
NLINE = NLINE + 4
55 WRITE(6,1001)
CALL MATOUT (DUMMY, XNAME, NF, 4, NR, NCD, INAME, 2, 1)

100 CONTINUE
RETURN

900 WRITE (6, 1004) Q1
1004 FORMAT (1HO, '*** ERROR IN QUANT4. Q1 =', E11.6)
NLINE = NLINE + 2
RETURN
SUBROUTINE PLOT (LENGTH, NWIDTH, X, NPTS, NY, Y1, Y2, Y3, Y4, KEY, NOFPMT,
1 XMAX, XMIN, YMAX, YMIN)
COMMON /PAGE/ NPAGE, NLINE, HDG(20)
SUBROUTINE PLOT SCALARS AND DRAW PRINTED FLATS, A MAXIMUM 2 PAGE GRAPH.
WRITTEN SPRING TERM 1975 BY ALICE ANNE NAVIN AND SHERMAN ROBINSON,
PRINCETON UNIVERSITY. REVISED SUMMER 1976 TO INCLUDE SUB. PAGE.
THE ROUTINE RANKS AND SOFTS UP TO 4 Y VECTORS, THE DEPENDENT VARIABLES.
ASSIGNS THEM TO A 2 DIMENSIONAL ARRAY, SAVING ORIGINAL SUBSCRIPTS.

INTEGER BLANK

YLABEL DIMENSION NOW 21, IF LENGTH MAX. GREATER THAN 101 BE SURE TO INCREASE
DIMENSION X(1), Y1(1), Y2(1), Y3(1), Y4(1),
DIMENSION X(NPTS), Y1(NPTS), Y2(NPTS), Y3(NPTS), Y4(NPTS),
1 LOG(4), NMARK(4), LCOLA(3), LROW(2),
2 NWATCH(10), YLABEL(21), LINE(101), JL(111), XLABEL(11)
MEMORY SAVINGS CAN BE ACHIEVED BY PASSING THE FOLLOWING ARRAYS
AS ARGUMENTS AND GIVING THEM VARIABLE DIMENSIONS.
IF MAX. NPTS CHANGED TO GREATER THAN 100, SUBROUTINE RANKHI
HAS TWO DIMENSIONS TO CHANGE--IRK(100), ITY(100)----2/19/75
DIMENSION DUMY(100), RANKY(4, 100), IPTY(4, 100), IPT(100)

UNIVAC NEEDS UPPER CASE I IN LIEU OF FIG. CCC
DATA NMARK,'*', '#', '@', '6', '/', ' ', BLANK,'/','/',
1 NWATCH,'2', '3', '4', '5', '6', '7', '8', '9', '0', '/
2 LCOLA/,----', '----', '-----', '-----', '-----', '----', '----', '----', '/

FORTRAN UNIT LINE PRINTER
LW = 6
MULTIPLE NUMBER FOR ROWS DOWN, YLABELS
NR = 5
MAX. ROWS PER PAGE
NRP = 51
LIMIT FOR TIE NOTATIONS WHERE FOR TIES MORE THAN 9, WE INSERT 0
NTINE = 9
MULTIPLE FOR COLUMNS ACROSS, X LABELS
NTEN = 10
A MAX. LENGTH OF ROWS IS 101, A 2 PAGE GRAPH, 2/19/75
MAXLEN = 101
IF (LENGTH .LE. 1) GO TO 199
A MAX. WIDTH OF COLS. ALSO IS 101, TO FIT 132 PRINTER SPACES.
MAXWD = 101
IF (NWIDTH .LE. 1) GO TO 199
IF (NY .LE. 0 OR. NY .GT. 4) GO TO 199
IF (NPTS .LE. 0) GO TO 199
IF (KEY .LE. 0 OR. KEY .GT. 4) GO TO 199

IF ALL CK, PROCEED WITH FLCI
CONTINUE
ROUND THE WIDTH UP TO A MULTIPLE OF 10 COLUMNS + 1 FOR XLABEL(NW).
CCC KEY = 3 MAY LOWER THESE VALUES
NWID = MINO(NWIDTH, MAXWID)
NWID = (NWIDTH - 1) / NTEN
IF (MOD ((NWIDTH - 1), NTEN) .NE. 0) NWID = NWID + 1
NWIDTH = NWID * NTEN + 1
NW = NWID + 1

C ROUND THE LENGTH UP TO A MULTIPLE OF 5 ROWS, PLUS 1 FOR BOTTOM LINE.

CCC KEY = 3 MAY LOWER THESE VALUES
LEN = MINO(LENGTH, MAXLEN)
LEN = (LENGTH - 1) / NR
IF (MOD(LENGTH - 1, NR) .NE. 0) LEN = LEN + 1
LENGTH = (LEN * NR) + 1

C WILL USE FOR YAXIS LABELS, I.E. EACH + LINE.
NL = LEN + 1

C RANK Y'S HIGH TO LOW
DO 17 J = 1, NY
DO 15 I = 1, NPTS
GO TO (11, 12, 13, 14), J
11 DUMY(I) = Y1(I)
GO TO 15
12 DUMY(I) = Y2(I)
GO TO 15
13 DUMY(I) = Y3(I)
GO TO 15
14 DUMY(I) = Y4(I)
15 CONTINUE
C
IF (NOPRTN .GT. 0) GO TO 10
WRITE(LW, 3)
3 FORMAT(180, ' J ', 'IX ', 'I ', 'IX ', 'DUMY(I) ', '3X',
1 'IPTY(J,I)', 'IX ', 'RANKY(J,I)', '1X ', 'I (IPTY(J,I)) ')
10 CALL RANKHY (DUMY, NPTS, IPT)
C IPTY(J,I) IS ITEM WHOSE RANK IS I.
C RANKY(J,I) IS EACH Y IN DESCENDING ORDER OF RANK (AND VALUE).
DO 16 I = 1, NPTS
RANKY(J,I) = DUMY (IPTY(I))
IPTY(J,I) = IPT(I)
IF (NOPRTN .GT. 0) GO TO 16
WRITE(LW, 4) J, I, DUMY(I), IPTY(J,I), RANKY(J,I), X(IPTY(J,I))
4 FORMAT(1H, 'I2, IX, IX, FX, .6F, .6F, .6F, .6F, .6F, .6F')
16 CONTINUE
17 CONTINUE
C
IF (KEY .NE. 1) GO TO 20
C USER-SUPPLIED MIN/MAXES, BUT SAVE AND TEST DATA TO BE SURE.
SAVEMAX = XMAX
SAVEMN = XMN
SAVEYMAX = YMAX
SAVEYMN = YMIN
CCC SCAN THROUGH THE DATA GETTING MIN, MAX VALUES.
C PURPOSELY RIDICULOUS FIGURES
20 XMAX = 1.E-37
XMIN = 1.E+37
YMAX = 1.E-37
YMIN = 1.E+37
C X VECTOR IS IN ORIGINAL ORDER, NOT RANKED
DO 22 I = 1, NPTS
XMAX = AMAX1(XMAX, X(I))
22 XMN = AMIN1(XMIN, X(I))
C Y VALUES IN RANK ORDER IN MATRIX, IN DESCENDING ORDER, YHI TO YLO
DO 24 J = 1, NY

YMAX = AMAX1(RANKY(J, 1), YMAX)
YMIN = AMIN1(RANKY(J, NPTS), YMIN)

IF (KEY .NE. 1) GO TO 110

C WE DO SUBSTITUTE OUR MIN/MAX FOR USER'S WHEN AN OBSERVED VALUE IS CLOSER
YMIN = AMIN1(YMIN, SAVMIN)
YMAX = AMAX1(YMAX, SAVMAX)
YMIN = AMIN1(YMIN, SAVMIN)
XMAX = AMAX1(XMAX, SAVMAX)

C ALL OPTIONS REQUIRE RANGES, BUT TSCALE MAY CHANGE MIN/MAX VALUES.
C THESE FOR COMPARISONS AFTER TSCALE

110     SAVMIN = XMIN
         SAVMAX = XMAX

C THESE FOR ARGUMENTS TO/FROM TSCALE
SAVMIN = YMIN
SAVMAX = YMAX

C ALL OPTIONS REQUIRE RANGES
C RANGE FOR X IS A PLUS INCREMENT ACROSS THE ROW COL. BY COL., XLO TO XHI
DIFFX = XMAX - XMIN
C RANGE FOR Y IS A MINUS INCREMENT DOWN EACH ROW, FROM YHI TO YLO. NOTE M:
C SIGN IN THE FOLLOWING STATEMENT.
DIFFY = -(YMAX - YMIN)
IF (NOPRNT .GT. 0) GO TO 123
WRITE(LW, 1050) XMAX, YMAX, XMIN, YMIN

1050 FORMAT(1HO, 10X, 'XMAX = ', 'G15.4', ' , YMAX = ', 'G15.4/1H, 10X, 'XMIN = ',
1 'G15.4, 'YMIN = ', 'G15.4')
WRITE(LW, 1051) DIFFX, DIFFY, KEY

1051 FORMAT(1HO, 10X, 'XSPAN = ', 'G15.4, ' YSPAN = ', 'G15.4, ' KEY = ', 'I1')
WRITE(LW, 1049) NWID, NW, LEN, NL

1049 FORMAT(1HO, 10X, 'NWID, NW = ', '2I5, ' LEN, NL = ', '2I5)

C IF (KEY .EQ. 1 .OR. KEY .EQ. 2) GO TO 140
C
C CALL TSSCALE (XMIN, XMAX, XLABEL, NW, NWID)
C
C NOTE THAT XMIN, XMAX MAY BE ADJUSTED BY TSSCALE.
C SPAN = XMAX - XMIN
C IF (SPAN .LE. .1E-10) GO TO 499

140     SPAN = XMAX - XMIN
      IF (SPAN .LE. .1E-10) GO TO 499

      IF (NOPRNT .GT. 0) GO TO 141
      IF (XMAX .NE. SAVMAX .OR. XMIN .NE. SAVMIN)
      1 WRITE(LW, 1052) XMAX, XMIN

1052 FORMAT(1HO, 1CX, 'SCALED XMAX AND SCALED XMIN = ', '2G15.4)

141     RANDEX = SPAN/ PLCAT(NWIDTH - 1)
      TESTX = 0.5 * RANDEX
      IF (KEY .EQ. 1 .OR. KEY .EQ. 2) GO TO 150

C TSSCALE STARTS WITH MIN. VALUES, AND 'EFFETTIES UP' LABELS FOR X AND Y.
C OPTIONAL CODE HERE TO SHRINK TSSCALE GRAPH AT RIGHT SIDE OF X AXIS(XLABEL(N)
C SEE STMT. 146 FF. BELOW FOR Y AXIS.
C SET KEY = 3 IN CALLING PROGRAM IF WANT TO SHRINK GRAPH, ELSE
C IF (KEY .EQ. 4) GO TO 146
C
C IF LARGEST OBSERVED GREATER THAN OR EQUAL TO NEXT-TO-LAST XLABEL PLUS TS
C CANNOT SHRINK.
C IF (SAVMAX .GE. (XLABEL(NWID) + TESTX)) GO TO 146
C
C ELSE WE CAN SHRINK NWIDTH OF GRAPH, AT RIGHT SIDE OF X AXIS
C CHANGES IN XMAX DO NOT AFFECT GRAPH, BUT MUST USE REVISI ED XMIN FOR XLO
C FOR STMT. 529 BELOW
NWID = NWID - 1
NW = NWID + 1
WIDTH = NWID * NTEN + 1

ABOVE CODE TO ELIM. LAST AXIS BLOCK WHERE XMAX > SAVMAX+FLOAT(NTEN)*RANGEX

CALL TSCALE(SAVMIY, SAVMAY, YLABEL, NL, LEN)

SPAN = (SAVMAY - SAVMIY)
IF (SPAN .LE. .1E-10) GO TO 498
IF (NORPT .GT. 0) GO TO 151
IF (YMIN .NE. SAVMIY .OR. YMAX .NE. SAVMAY)
1 WRITE(LW, 1053) SAVMAY, SAVMIY
1053 FORMAT(1H0, 10X, 'SCALED YMAX AND SCALED YMIN = ', 2G15.4)

RANGEY = SPAN / FLOAT(LENGTH - 1)
TESTY = .5 * RANGEY

GO TO (251, 251, 245, 250), KEY

CODE TO SHRINK TSCALE GRAPH AT TOP OF Y AXIS (YLABEL(NL))
WHERE LEN = 10 PER PAGE, NL = LEN + 1 (ECR YLABELS)
RECALL, TSCALE YLABELS STILL IN LOW TO HIGH ORDER
COMPARSE WITH TSCALE'S NEXT TO LAST LABEL, THE OBSERVED YMAX.

IF (YMAX .GE. (YLABEL(LEN) + TESTY)) GO TO 250
SAVMAY = YLABEL(LEN)
LEN = LEN - 1
LENGTH = LEN * NR + 1

IF TOP OF GRAPH SHRUNK, WOULD DO THIS BEFORE CALL REVERSE, HOWEVER.
NL = LEN + 1

CALL REVERSE(YLABEL, NL)

NOTE IF TOP OF GRAPH SHRUNK MUST DO THIS BEFORE CALL REVERSE
IF (KEY .EQ. 3) NL = LEN + 1
ABOVE WILL ONLY CHANGE IN VALUE IF LEN HAS BEEN DIMINISHED

IF (NORPT .GT. 0) GO TO 300
IF (KEY .EQ. 3)
1 WRITE(LW, 1049) NWID, NW, LEN, NL
1049 FORMAT(1H0, 10X, 'RANGEX, RANGEX, KEY
GETTING READY FOR Y AND X LABELS, TSCALE'S AND CUES.

JYLB = 0
FOR KEY = 1 OR = 2
NEED ORIGINAL, OBSERVED YMAX FOR YHI = BELOW.
SEE ALSO, STMT. 110 ABOVE
IF KEY = 3 OR = 4 XMIN = XMIN POSSIBLY REVISED BY TSCALE

YHI = SAVMAY
IF (KEY .EQ. 3 .OR. KEY .EQ. 4) GO TO 306

FOR KEY = 1 OR 2, SET XLABELS HERE
XLABEL(1) = XMIN
DO 305 I = 2, NW
IM1 = I - 1
LAST XLABEL WILL CONTAIN XMAX VALUE

XLABEL(I) = XLABEL(I-1) + (FLOAT(NLEN) * FANGEY)

CONTINUE

IF (NOPRTN.EQ.1) GO TO 310

THIS IS PAGE PRINTED WHEN NOPRTN = 2 (CR = 0), BUT NOT WHEN NOPRTN=1

FORCE HEADER
CALL PAGER(100)
WRITE(LW,6001)

FORMAT(1H,10X,'THIS PROGRAM WAS WRITTEN IN FORTRAN IV FOR THE
1 IBM 360/31 AT'/'1H,6X,'PRINCETON UNIVERSITY,SPRING TERM 1975,BY AL
2 INE ANNE NAVIN AND'/'1H,6X,'SHERMAN EISINGER,REVISING TSCALE/PSCA
3 NLE ROUTINES WRITTEN 11/70'/'1H,6X,'BY B. K. REID,UNIVERSITY OF M
4 ARYLAND.'/
51H0,10X,'IT REQUIRES REGION=18K AND T=.16 MINUTES FOR THE MAXIMUM'
61H,6X,'NPTS(100),NY(4),NWIDTH(101),AND LENGTH (UPTO 101 ROWS)
7 FOR A '1H,6X,'TWO PAGE GRAPH.'/
)

WRITE(LW,6003) NWIDTH,LENGTH

FORMAT(1H,10X,'THE FOLLOWING GRAPH IS','I3,'COLUMNS WIDE ON THE
1 HORIZONTAL, X'/1H,6X,'AXIS, AND ','I4,'ROWS LENGTH ON THE VERTICAL.
2 Y AXIS.')

IPL = NY * NPTS

WRITE(LW,6004) NPTS,NY,IPL,KEY

FORMAT(1H0,10X,'THE NUMBER OF OBSERVATIONS, NPTS, ARE ','I5,' AND T
1 HE NUMBER '/H,6X,'OF Y'S NY, ARE','I1.' NY*NPTS PLOTTED FOR
2 NTS) SHOULD BE ','I3,<'/H,6X,'THE OPTION KEY WAS SET TO ','I3,
3.'/
)

CALL PAGER(100)

CONTINUE

IS MAX. NW FOR + FOR XLABELS, PER PAGE .IE. 132 CCLS., PCB NWIDTH =
101 INCLUSIVE.
WRITE(LW,6010) (COLA(K),K=1,3),N=1,NWID)

FORMAT(1H,17X,1H*,11(2A4,A2))

THE BIG LOOP IS DO 1000 I.E. FLOW BY FLOW START TC END
TOP ROW BY ALL COLUMNS, XLC TO XHT, 2ND FLOW BY ALL COLS., ETC....

DO 1000 JRCW = 1, LENGTH
COUNT LINES, ETC. RE PAGES, LABELS, ETC.
IF (MOD(JROW,5) .EQ. 0) CALL PAGER (100)
JR = JRCW + NR - 1
ONLY INCREMENT EVERY FIFTH ROW FOR YLABELS
IF (MOD(JR,NE) .EQ. 0) JJR = 1
IF (MOD(JR,NE) .NE. 0) JJR = 2
SEE STMT. 30C ABOVE, JYLB = 0
JYLB = JYLB - JJR + 2

DO 512 NCOL = 1,NWIDTH
LINE(NCOL) = BLANK
JL(NCOL) = 0

CONTINUE
IF (JROW .GT. 1) YHI = YHI + RANGEY

DO 700 J = 1, NY
  NWATCH(1) = NMARK(J)
  IF (JROW .EQ. 1) LOI(J) = 0
  NUM1 = LOI(J) + 1
  IF (NUM1 .GT. NPTS) NUM1 = NPTS

DO 600 I = NUM1, NPTS
  RECALL, FANKY'S ALREADY IN DESCENDING ORDER, HIGH TO LOW.
  RECALL, DIFY = -(YMAX - YMIN)
  IP (FANKY(J,I)) .GT. (YHI - TESTY) .CF.
  1 FANKY(J,I) .LE. (YHI + TESTY)) GO TO 700

  LOI(J) = I

AT THE START OF EACH NEW ROW, FIRST COLUMNS
  XLO = XMIN

DO 550 NCOL = 1, NWIDTH

CCOLUMNS BY COLUMN POSITIVE INCREMENT FOR X.
  IF (NCOL .GT. 1) XLO = XLO + RANGEX

  IF (X(IPTY(J,I)) .LT. (XLO - TESTX) .CF.
  1 X(IPTY(J,I)) .GE. (XLO + TESTX)) GO TO 550

WE HAVE A POINT

CCC MUST FIND ANY AND ALL TIES FOR THIS (JROW, NCOL)
  JL(NCOL) = JL(NCOL) + 1

IF MORE THAN NINE TIES, WE MARK THE GRAPH WITH A ZERC(0)
  IP (JL(NCOL) .GT. NINE) JL(NCOL) = 10
  LINE(NCOL) = NWATCH(JL(NCOL))

550 CONTINUE

600 CONTINUE

700 CONTINUE

NOW, TO PRINT THIS LINE OF BLANKS AND/OR NMARKS, WITH APPROPRIATE YLABELS
  AND/OR Y AXIS NOTATION.
  JJR IS SET ABOVE STMT. 512 ABOVE FOR + OR | AT START OF EACH ROW.
  IF (JJR .NE. 1) GO TO 982
  JYLB IS ALSO SET ABOVE STMT. 512 ABOVE.
  IF (KEY.EQ.1 .OR. KEY.EQ.2) YLABEL(JYLB) = YHI
  WRITE(LW,985) YLABEL(JYLB), LRCWA(JJR), (LINE(N), N=1, NWIDTH),
  1 LRCWA(JJR)

985 FORMAT(1H, 1X, G15.4, A1,101A1,A1)

GO TO 1000

982 WRITE(LW,983) LROWA(JJR), (LINE(N), N=1, NWIDTH), LROWA(JJR)

983 FORMAT(1H, 16X, A1,101A1, A1)

CCC END OF A ROW IN BIG LOOP, DO 1000 JROW = 1, LENGTH

1000 CONTINUE
ALL ROWS PLOTTED, NOW XAXIS AND XLABELS PRINTED AT BOTTOM OF GRAPH.
WRITE(LW,6010) ((ICOLA(K),K=1,3),N=1,NWID)

WILL THE XLABELS FIT IN 10 MAX. SPACES ??
CANNOT KEEP LABELS IN G11.4 FORMAT IF ICHK = 0
ICHK = 0
ICHK2 = 0
DO 950 I = 1, NW
IF (ABS(XLABEL(I)) .LT. 0.1) ICHK2 = ICHK2 + 1
IF (ABS(XLABEL(I)) .GE. 10000.0) ICHK = 1
950 CONTINUE
IF (ICHK2 .GE. 2) ICHK = 1
IF (ICHK .EQ. 0) GO TO 953

NOTE WE PRINT THE XLABELS ON TWO LINES IN A STAGGERED WAY.
WRITE(LW,601') (XLABEL(I),I=1,NW,2)
6011 FORMAT(1H,4X,6(9X,G11.4))
WRITE(LW,6014) (XLABEL(I),I=2,NW,2)
6014 FORMAT(1H,4X,6(9X,G11.4))
GO TO 954
953 WRITE(LW,6013) (XLABEL(I),I=1,NW)
6013 FORMAT(1H,11X,11F10.4)

CONTINUE
OUR GRAPH IS COMPLETED

WE SHOW GRAPH NOTATIONS FOR EACH OF 4 POSSIBLE Y'S (WHEN NO TIES).
WRITE(LW,6005) (J,NMARK(J),J=1,4)
6005 FORMAT(1H0, 6X,4('Y(',I1',') IS ',A1,3X))
GO TO 999

*** ERROR MESSAGE RETURNS ***

CONTINUE
WRITE(LW,198)
198 FORMAT(1H0,9H**********,/)
WRITE(LW,195) LENGTH,NWIDTH,NY,NPTS,KEY
195 FORMAT(1H0,'CALLING SEQUENCE ERROR IN CALL TO PLCT'/
1 1H ,'SPECIFIED LENGTH = ',I11/
2 1H ,'SPECIFIED NWIDTH = ',I11/
3 1H ,'NO. OF ORDINATES = ',I11/
4 1H ,'NO. OF DATA POINTS = ',I11/
5 1H ,'OPTION CODE , KEY = ',I11)
WRITE(LW,198)
GO TO 999

SAVMAX,SAVMIN,XMAX,XMIN,DIFPX
495 FORMAT(1H0,'ERROR IN SPAN',SAVMAX = ',1PE10.3,' SAVMIN = ',1PE10.
495 13,' XMAX = ',1PE10.3,' XMIN = ',1PE10.3,' DIFPX = ',1PE10.3/)
WRITE(LW,198)
GO TO 999

YMAX,YMIN,SAVMAX,SAVMIN,DIFPY
496 FORMAT(1H0,'ERROR IN SPAN',YMAX = ',G15.4,' YMIN = ',G15.4,
496 1 ' SAVMAX = ',G15.4,' SAVMIN = ',G15.4,' DIFPY = ',G15.4)
WRITE(LW,198)
GO TO 999

RETURN
END
SUBROUTINE RANKH(Y,NPTS,IPT)
C
A REVISION OF A RANK LOW TO HIGH ROUTINE BY LARRY E. WESTPHAL.
C
SUB: RANKH YIELDS RANKING FROM HIGH TO LOW OF ITEMS BASED ON VECTOR Y VA:
C
IRK(I) IS RANK OF I'TH ITEM
C
IPT(I) IS THE ITEM WHOSE RANK IS I
C
2/19/75, SUB. PLOT DOES NOT UTILIZE IRK
REAL IRK

C
SINCE MAX. NPTS = 100, DIMENSION ITY(100), IRK(100) 2/19/75
C

DIMENSION Y(NPTS),IPT(NPTS),IRK(100),ITY(100)

DO 2 I = 1,NPTS
IRK(I) = 0.0
2 IPT(I) = 0
I = 1
14 R = -1000000.0
LT = 0
ITY = 0
DO 20 L = 1,NPTS
20 ITY(L) = 0
DO 10 L = 1,NPTS
IF (IRK(L) .NE. 0.0) GO TO 10
11 IF (R - Y(L)) 11, 12, 10
11 LT = L
R = Y(L)
ITY = 0
DO 21 L1 = 1,NPTS
21 ITY(L1) = 0
GO TO 10
12 IY = IY + 1
ITY(IY) = I
CONTINUE
IF (LT .EQ. 0) GO TO 998
RT = FLOAT(I) + (FLOAT(IY) / 2.0)
IRK(LT) = RT
IPT(I) = LT
IF (ITY .EQ. 0) GO TO 22
DO 23 L1 = 1, IY
23 IRK(ITY(L1)) = RT
I = I + 1

22 I = I + 1
IF (I .LE. NPTS) GO TO 14
GO TO 999
998 WRITE(6,997)
997 FORMAT(1HO,9H**********/1H,'ERROR IN RANKH'/,
1H9H**********/)
999 RETURN
END
SUBROUTINE REVERSE(YLABEL,NL)
C
WHERE YLABELS FROM TSSCALE ARE IN ASCENDING, RATHER THAN DESCENDING, ORDER.
C
OF VALUE, CALL REVERSE(YLABEL,NL).
C
WORKS FOR UNMATCHED PAIRS AS WELL AS MATCHED PAIRS, SINCE CENTER WOULD
C
STAY PUT.
C
WRITTEN SPRING 1975, AAN/SR.
DIMENSION YLABEL(NL)
NTEST = NL / 2
DO 349 J = 1, NTEST
I = J - 1
K = NL - I
SAVE = YLABEL(J)
YLABEL(J) = YLABEL(K)
YLABEL(K) = SAVE

349 CONTINUE
RETURN
END

SUBROUTINE TSCALE(VMIN, VMAX, VSCA, NDM, NAXIS)
C SUBROUTINE TO SCALE A PLOT.
C WRITTEN BY B. K. REID, UNIV. OF MARYLAND, 1970. REVISED 2/75 SF/AAN
C THE INPUT VARIABLES
C VMIN AND VMAX CONTAIN THE OBSERVED VALUES OF MINIMUM/MAIMUM
C VALUES OF THE VARIABLE TO BE SCALED. ON OUTPUT, THEY WILL
C CONTAIN THE ADJUSTED VALUES TO BE USED FOR 'NICE' NUMBERS
C ALONG THE AXIS OF THE PLOT.
C VSCA IS THE ARRAY WHICH WILL CONTAIN AXIS LABELS. NAXIS IS
C THE NUMBER OF AXIS BLOCKS, SO THAT THERE WILL BE 'NAXIS+1'
C ENTRIES MADE IN VSCA.
C
NDIM = NAXIS + 1
DIMENSION VSCA(NDIM), XSCF(12)
DATA XSCF/0., 1., 1.25, 1.5, 2., 3., 4., 5., 6., 7.5, 8., 10.0/
NLOOP = 0
SPAN = 0.0
NSCF = 12
NSM1 = NSCF - 1
ZSEC=VMIN

650 SCF = (VMAX - VMIN) / FICAT(NAXIS)
MINSIG=ISIGN;1,1;FIX(VMIN))
DX1=0.
IF (SCF .LE. 0.0) GO TO 120
XNC=ALOG10(SCF)
INC=XNC
IF(XNC.LE.0.) INC=INC-1
IF(INC.GT.0) XNC10=10.***INC
IF(INC.LE.0) XNC10=-10.***ABS(FLOAT(INC))
DX=PSCALE(SCF,-XNC10)
DO 651 ISC = 1, NSM1
651 IF(DX.GT.XSCF(ISC)) DX1=XSCF(ISC+1)

C A PROBABLE SCALE FACTOR HAS BEEN COMPUTED. LET US
C NOW TRY TO FIND A MINIMUM. (THIS CODE IS TAKE 3 OF
C MINIMUM-FINDING.)

DX1=PSCALE(DX1,XNC10)
NEARBY=1+MINSIG;FIPX(ABS(VMIN)/DX1)
VMIN=FLOAT(NEARBY)*DX1

128 CONTINUE
IF(VMIN.LE.ZERO) GO TO 129
VMIN=VMIN-DX1
GO TO 128

129 CONTINUE
SPAN = VMIN + FLOAT(NAXIS) * DX1
IF(SPAE.GE.VMAX) GO TO 649

C WE HAVE MOVED THE MINIMUM DOWN SO FAR THAT THE INCREMENT
C DOESN'T WORK ANY MORE
NLOOP=NLOOP+1
IF(NLOOP.LE.2) GO TO 650

120 WRITE(6,124)
124 FORMAT(1H0, 9H**********/1H , 'ERROR IN TSCALE ROUTINE' ,
1 1H, 9H**********/ )

649 CONTINUE
DX=DX1
* V MAX = SPAN
* DO 655 IH = 1, NDIM
* VS CA (IH) = VMIN + FLOAT (IH - 1) * DX
* 655 CONTINUE
* RETURN
* END

FUNCTION PS O L E ( MANTIS, CHAR AC )
C FUNCTION TO SCALE BY EXPONENTIAL SCALE FACTOR
* REAL M ANTIS
* IF ( CHAR AC .GT. 0 ) PS O L E = MANTIS * CHAR AC
* IF ( CHAR AC .LT. 0 ) PS O L E = MANTIS / AES ( CHAR AC )
* RETURN
* END
References


