CIScd: Search Software for the *CIS/ED* Version 2.2 User Guide[*]

Ronald Thisted

March, 2003

Contents

1 CIScd, Version 2.2

CIScd is a program for searching the *Current Index to Statistics* Extended Database (*CIS/ED*) for entries that contain combinations of keywords specified by the user. The program is free software that is made available under the GNU General Public License. See the section on “Fine Print” for additional information.

CIScd is not fancy, but it can be used to conduct quite intricate searches. Although it does not currently have a menu-driven interface, much of its power can be realized by learning a handful of commands. If you really want a menu-driven version, feel free to add it.

1.1 New Features

Version 2.0 was the first publicly released version to accomodate the four-digit year format of the *CIS/ED*. In addition, abstracts now appear (when present) in SHORT, TEX, and REFER display formats by default. This was also the first version to be released as free software.

Version 2.1 supported search and refinement restrictions by year of publication, introduced a Unix version, and corrected minor bugs in diagnostic output.

Version 2.2 expands the online documentation, adds support for output logging, and (slightly) updates this *User Guide*.

Give the command `help commands` to see additional commands added since version 2.0 (described in the Tutorial below), and consult the online help for each new command.

1.2 Installing the binary programs (Macintosh and Windows)

Two executable programs are provided, each in Macintosh and Windows formats. The first, `Make Configuration`, is run just once to construct a configuration file that CIScd uses to locate the *CIS/ED* files on the CD-ROM, on your hard disk, or on a network (where you may have installed them). The second program is CIScd itself.

1.2.1 Installing on Macintosh computers

To install the programs on a Macintosh

1. Copy the programs `CIScd` and `Make Config` from the MACCISCD folder to a folder on your computer. (The Applications folder is a good choice in OS X.)

2. Run the program `Make Config` from the folder on your hard drive. When prompted by the program, give the name of the *CIS/ED* CD-ROM that appears on your desktop unless the default name provided is correct. (For Release 11 of *CIS/ED*, it is likely that the default included in the program is already correct). This will create a file named `cis.cfg` which CIScd needs to locate the files to search.

   If you have installed the *CIS/ED* files on your hard disk or on a network, you need to provide the full path (with folders separated by colons) to the folder containing the `MAC` *CIS/ED* folder at the first prompt of `Make Configuration`.

3. There is no step 3.

1.2.2 Installing on Windows computers

To install the program on Windows 95 (or later)

- Copy the programs `CIScd` and `Make Config` from the WINCISCD folder to a folder on your computer.
- Run the program `Make Config` from the folder on your hard drive. When prompted by the program, give the drive letter of the *CIS/ED* CD-ROM that appears on your desktop unless the default name provided is correct. This will create a file named `cis.cfg` which CIScd needs to locate the files to search.

  If you have installed the *CIS/ED* files on your hard disk or on a network, you need to provide the full path (with folders separated by backslashes, and preceded by the appropriate drive letter) to the folder containing the `DOS` *CIS/ED* folder at the first prompt of `Make Configuration`.

1.2.3 Installing on Unix computers

To install the program on Unix, Linux, or related operating systems, consult the `readme.txt` document in the UNXCISCD folder. You will have to compile the `CIScd` and `Make Config` programs using your `c` compiler. Each Unix system is configured slightly differently, so some adjustment to the compilations instructions on the CD may be necessary.

1.3 How the Program Works (Important)

To use the search program to its best advantage, it is essential to learn how the program works. The Current Index Database on CD-ROM actually consists of a set of files containing the *CIS* entries themselves, as well as a set of "index files" which are nearly as large. These index files are tables of keywords together with the location of records in the *CIS* file that contain those keywords. The keywords used to construct the index files on the CD-ROM were all words that

1. begin with an alphabetic character,
2. are followed by a numeral, punctuation mark, or blank space,
3. contain at least two characters, and

4. are no more than six characters in length.

Words in the database that exceed six characters are, for purposes of indexing, truncated to six characters. Thus, the words “regress”, “regression”, and “regressor” are all considered instances of the index word “regres”’. For purposes of indexing, upper and lower case letters are considered equivalent. Certain very common words (such as “of”, “by”, “and”, etc) are also not indexed.

The author, title, keyword, and alternative spelling fields of the records are indexed.

A *search* looks up one or more keywords in the index files, and then uses the list of *CIS/ED* entries found there to construct a new *search result* containing all of the entries that match the keyword criteria.

Once a search result is obtained, the search can then be *refined* by adding additional constraints, or by combining the current search result with one obtained earlier.

CIScd operates on a *stack* of search results (in a fashion similar to calculators that use reverse-Polish notation). Each new `search` command creates a new search result, which becomes the top-most search result on the stack. All previous search results are “pushed down” onto the stack, maintaining the order in which they were created. Refinement commands operate either on the most recent search (top of the stack), or the two most recent searches. The contents of the stack after executing a command depend on the particular operation being carried out, but generally, the top-most search result on the stack is the output from the last operation.

Always remember: **Search** first, then **refine.**

2 Tutorial: Getting Started

To start a search, open (or double-click on) the CIScd icon — the one on your hard disk, not the CD-ROM. Something like the following will appear on the screen. The examples in this chapter use *Release 8* of the database. Output using later versions of either the database or of CIScd will of course be slightly different.

```
CIScd -- Search software for the Current Index to Statistics
      -- from Socratic Media, Inc.

CIScd Version 2.0 Copyright  2001 Socratic Media, Inc.
```

```
            CIScd comes with ABSOLUTELY NO WARRANTY; for details type ‘help
            warranty’.  This is free software, and you are welcome to redistribute
            it under certain conditions; type ‘help conditions’ for details.


        == Type  'help' for general help.
}

To search for all papers by William Krasker, enter ‘‘search Krasker’’ at the
prompt.  This will produce

{\small \begin{verbatim}
=> search Krasker
...........
[16 records found]
[16 main records/0 backup records]
```

The dots are produced sequentially as the CD-ROM is searched. The time required for this query to find these 16 of the 200,000+ records on a Power Macintosh was about one second.

An alternative would be to enter “`search William Krasker`” at the prompt. In that case, all records that contain both `William` and `Krasker` will be produced. Note that the software automatically converts “William” to “willia” and “Krasker” to “kraske” before doing the search in the index. This search takes a bit longer (about 4 seconds), since every record that contains “William” must be checked to see if it also contains “Krasker”.

2.1 An Initial Refinement

Having done the search above we can *refine* the results using any of a collection of powerful commands. These commands can do arbitrary Boolean searches on terms (or even on strings of consecutive characters). The `16 main records` found in the original search are stored in a temporary file on your disk.

The search refinement commands do not operate on the database as a whole, but only on this subset of records contained in the “main records” file. Each command that operates on the “main records” file leaves its result in the “main records”, and the previous result in the “backup records”.

In fact, CIScd creates a stack of search results. This stack has two visible elements, the “main results” and “backup results” file. This stack can be manipulated with commands such as `push`, `pop`, and `swap`. The file operations can be thought of commands given to a reverse-Polish calculator. The `stack` command gives a status report of all files in the stack. The `push` and `pop` commands are used to manipulate the stack.

These commands tend to be quite fast. For instance, starting with the 16 “Krasker” records, we can issue the request “and William”, which produces

```
=> and William
[13 (of 16) records meet criteria]
[13 main records/16 backup records]
```

The 13 remaining records now contain both “William” and “Krasker” somewhere in them. This search was virtually instantaneous.

What were the characteristics of the 3 records omitted in the refinement? Two of them were by others commenting on a procedure of Krasker and Welsch. The other was by W. S. Krasker and J. W. Pratt — an article that we would like to have found! The difficulty is that names of authors are rarely consistent across journals or other publications, and often authors' names are misspelled. (The author index on the *CIS* CD-ROM can be browsed to find similar names represented on the database.)

2.2 Restricting the Search

Here is an alternative strategy for refining the initial search—use a search restricted to the `author` field of the records. After the initial search on “Krasker”, we could proceed with

```
=> search krasker
=> author krasker
[14 (of 16) records meet criteria]
```

which finds the appropriate entries.

2.3 A More Difficult Example

How should we find articles written by Al Best? Here is a transcript that illustrates several features of CIScd.

```
=> #
=> # First, do an initial SEARCH (which uses the inverted index)
=> #
=> search best
....................................
[954 records found]
[954 main records/14 backup records]
=> #
=> # That took over a minute, due to the number of records
=> # that had to be retrieved.  A moment’s reflection suggests
=> # that many of these records will deal with ‘‘Best invariant
=> # tests’’, or ‘‘Best linear’’.  So we refine the search by
=> # looking only in the author field.
=> #
=> author best
[82 (of 954) records meet criteria]
[82 main records/954 backup records]
=> list

        1971Biomtrcs 27     895- 901 J
        Descriptive functions in disease
        Best, William R.;Becktel, Jack M.;Johnson, Arthur F.
        Regression;Discriminant analysis;Public health;Multivariate analysis

        1973Biomtrka 60     429- 430 J
        Extended tables for Kendall’s tau
        Best, D. J.
        Rank correlation coefficient;Nonparametric regression;Ties in ranks

        1974ApplStat 23      98- 100 J
        [Algorithm AS 71] The upper tail probabilities of Kendall’s tau
        Best, D. J.;Gipps, P. G.
        Edgeworth approximation

        1974Biomtrka 61     385- 386 J
        The variance of the inverse binomial estimator
        Best, D. J.
-- MORE? [yn] n
=> #
=> # Although Best is in the author field, many of them are not
=> # the individual we are after.  We can search for records
=> # whose author fields contain ‘‘A.’’ and ‘‘Best’’
=> allauthor A. Best
[7 (of 82) records meet criteria]
[7 main records/82 backup records]
=> list

        1983Biomtrka 70     447- 453 J
        A test for comparing large sets of tau values
        Best, D. J.;Cameron, M. A.;Eagleson, G. K.
        Kendall’s tau;Multiple comparisons;Poisson-limit theorem

        1983STATCOMP83       41-  49 P
        Power approximations for Pearson’s chi-squared test
        Best, D. J.;Rayner, J. C. W.;Turnbull, A. P.
        Goodness-of-fit

        1989AstrlJSt 31     491- 492 J
        Review of ‘‘Goodness-of-fit statistics for discrete multivariate data’’
        Read, Timothy R. C.;Cressie, Noel A. C. (Auth);Best, D. J. (Rev)

        1991Biomtrcs 47     788- 789 J
        Review of ‘‘Smooth tests of goodness of fit’’
        Rayner, J. C. W.;Best, D. J. (Auth);Kemp, A. W. (Rev)

[... some output omitted ...]


        1996JRSS-A  159     323- 342 J
        Bayesian analysis of realistically complex models
        Best, N. G.;Spiegelhalter, D. J.;Thomas, A.;Brayne, C. E. G.
        Markov chain Monte Carlo;Categorical data;Random effects model

        1996StatMed  15    2123-2124 J
        Review of ‘‘Bayesian data analysis’’
        Gelman, A.;Carlin, J. B.;Stern, H. S.;Rubin, D. B. (Auth);Best, Nicky (Rev)
=> #
=> # Well, that was positively unhelpful -- none were the targets
=> # of our search!  We can back out of the last result by
=> # interchanging (SWAPping) the main and backup files:
=> swap
[82 main records/7 backup records]
=> #
=> # The ALLAUTHOR command finds records that contain ALL of the
=> # listed search terms in the author field.  We can also request
=> # records that meet ANY of a set of criteria, using the AUTHOR
=> # command -- Think of AUTHOR = AUTH + OR:
=> author Al Alvin Theodore Simon
[13 (of 82) records meet criteria]
[13 main records/82 backup records]
=> list

        1979ASAProStCp      153- 153 P
        Discussion of LINMOD: A flexible system for multivariate linear models
        computations
        Best, Alvin M., III

        1979Psymtrka 44     395- 408 J
        On the precision of a Euclidean structure
        Best, Alvin M., III;Young, Forrest W.;Hall, Robert G.
        Multidimensional scaling;Data analysis

[... some output omitted ...]

        1997JASA     92    1403-1412 j
        Dynamic conditional independence models and Markov chain Monte Carlo methods
        Berzuini, Carlo;Best, Nicola G.;Gilks, Walter R.;Larizza, Cristiana
        [Theory and Methods];Bayesian inference;Graphical model;Importance sampling;
        Metropolis algorithm;Forecasting;[Metropolis-Hastings algorithm];[Real-time
        forecasting];[Sequential updating]
          In dynamic statistical modeling situations, observations arise
          sequentially, causing the model to expand by progressive incorporation of new
          data items and new unknown parameters. For example, in clinical monitoring,
          patients and data arrive sequentially, and new patient-specific parameters
          are introduced with each new patient. Markov chain Monte Carlo (MCMC) might
          be used for continuous updating of the evolving posterior distribution, but
          would need to be restarted from scratch at each expansion stage. Thus MCMC
          methods are often too slow for real-time inference in dynamic contexts. By
          combining MCMC with importance resampling, we show how real-time sequential
          updating of posterior distributions can be effected. The proposed {\it
          dynamic sampling} algorithms use posterior samples from previous updating
          stages and exploit conditional independence between groups of parameters to
          allow samples of parameters no longer of interest to be discarded, such as
          when a patient dies or is discharged. We apply the methods to monitoring of
          heart transplant recipients during infection with cytomegalovirus.
```

```
[... some output omitted ...]

        1998GraphMod        575- 598 P
        Hepatitis B: A case study in MCMC
        Spiegelhalter, D. J.;Best, N. G.;Gilks, W. R.;Inskip, H.
        Graphical model;Markov chain Monte Carlo
=> #
=> # Near Success!
=> # A better way to accomplish the last search is to conduct a
=> # search on the string of characters ‘‘Best, A’’ in the
=> # author field:
=> swap
[82 main records/13 backup records]
=> author "Best, A"
[9 (of 82) records meet criteria]
[9 main records/82 backup records]
=> #
=> # Real Success!!
=> #
=> # We can save these results to a file, in exactly the format above:
=> #
=> save best.cis
[9 records written to file best.cis in SHORT format]
=> quit
== Would you like me to delete temporary files [ny]? n
```

Of special note is the 1997 *JASA* article, whose abstract is automatically displayed. This is the default behavior for displaying records in this format. The presence of an abstract is denoted in the *CIS/ED* record by a lower case ‘`j`’ at the end of the first line rather than an upper case ‘`J`’. (The J stands for “journal article”.)

If we suddenly were to discover that we forgot something, we can start CIScd again, and the main and backup results will be exactly as we left them, since the temporary files are not erased at the end of a session—unless we specifically request them to be deleted. If you would like to erase them (if there are some big search results, for instance, that you won’t need again), you can add the word “clear” to the end of the quit request.

2.4 Commands

Detailed information about many commands can be obtained by typing the command name followed by a question mark (for example, `SAVE ?`) A list of commands can be obtained at any time with the request “`help commands`”, which produces a list like this:

```
help     Prints this text
swap     Interchange main and backup searches
count    Show current contents of main and backup searches
config   Initialize and restart, optionally specifying configuration file
and      Finds records in current search that have ALL search terms
andor    Finds records in current search that have ANY search terms
not      Finds records in current search that match NO search terms
notboth  Omits records having ALL search terms, keeps all others
merge    Merges current search and backup search, REPLACING current search
save     Saves results of current search to named file
retrieve Retrieves a named file of CIS database records
pop      Remove main search from results stack, move others up
push     Move all search results down in results stack
display  Show results of main search in current format
quit     Quit current program
set      Set options.  For more info, type 'HELP SET'
show     Show options.  For more info, type 'HELP SET'
batch    Read search commands from named file
search   Extract records from CISED using inverted index
version  Display version number of software and database
more     Set pager options
#        Comment
author   Find records with ANY of keys, in author field only
allauth  Find records with ALL of keys, in author field only
noauth   Omits records with ANY of keys, in author field only
type     Display the contents of a disk file to the screen
next     Display the next k records.  'NEXT k' is used to set k
diff     Find all records on backup but not on main search
whatis   Looks up an abbreviation in the abbreviation file
year     Restrict future searches to a year or range of years
pub      Restrict future searches to one or more entry types
andyear  Apply current year filter to working database
print    DOS/WINDOWS command to print current search results
wrap     Sets column at which line wrapping takes place
```

2.5 Search Commands

2.5.1 The Preliminary `search`

Every *CIS* search will start with a broad search using the inverted index to the database. The `search` command looks for all entries in the database that match the first six (case insensitive) characters of each of the keywords listed on the request. Records that match all of the specified keywords are retained as the main search result. If at least one record was found, the previous main search result becomes the current backup search result.

2.5.2 Refinements

The syntax for the requests that refine searches is more flexible. Each request has the form

```
REQUEST  Token1 Token2 ... TokenK
```

where `REQUEST` represents a command such as `not`, `and`, `andor`, `author`, or `andyEAR`. If the first four characters of `REQUEST` match a valid request, the corresponding request will be honored. Thus, `auth` is an abreviation for `author`.

Each token consists either of a string of non-blank characters, or a string of characters that include blanks, enclosed in double quotation marks ("). Neither the double quotation mark (") nor the tilde (~) is used in the *CIS*, and these characters are not valid in search tokens. The number of tokens on a single request is not limited (for practical purposes).

After each successful refinement request, the previous master search result becomes the current backup, and the results of the request become the current master.

Here are the common refinement commands, and a brief description of their function:

`allauth`	`Find records with ALL of keys, in author field only`
`and`	`Finds records in current search that have ALL search terms`
`andor`	`Finds records in current search that have ANY search terms`
`andyear`	`Apply current year filter to working database`
`author`	`Find records with ANY of keys, in author field only`
`diff`	`Find all records on backup but not on main search`
`merge`	`Merges current and backup search, REPLACING current search`
`noauth`	`Omits records with ANY of keys, in author field only`

```
not      Finds records in current search that match NO search terms
notboth  Omits records having ALL search terms, keeps all others
```

2.5.3 Example

Papers which cite, refer to, or are written by (any) Efron, and which cite, refer to, or are written by (any) Stein:

```
=> search efron
................................
[158 records found]
[158 main records/9 backup records]
=> and stein
[18 (of 158) records meet criteria]
[18 main records/158 backup records]
```

Of these papers, some are written by Efron and Stein; ALLAUTH searches for records whose author fields contain all of the listed tokens.

```
=> allauth efron stein
[1 (of 18) records meet criteria]
[1 main records/18 backup records]
=> list

        1981AnlsStat  9     586- 596 J
        The jackknife estimate of variance
        Efron, B.;Stein, C.
        Analysis of variance;Bootstrap;$U$-statistic
        94x;U-statistic
```

Of course, some of the papers might have been written by Efron about Stein or his work; after returning to the original search results (using SWAP) we have

```
=> swap
[18 main records/1 backup records]
=> author Efron
[12 (of 18) records meet criteria]
[12 main records/18 backup records]
=> list

        1971JASA     66     807- 815 J
        Limiting the risk of Bayes and empirical Bayes estimators -- Part I: The Bayes
        case (Ref: V67 p130-139)
        Efron, Bradley;Morris, Carl
        Stein estimator;Shrinkage estimator
...
```

These 12 references also contain the Efron/Stein paper cited earlier. To find the papers written by Efron and Morris regarding Stein shrinkage estimation, we use:

```
=> allauth efron morris
[9 (of 12) records meet criteria]
[9 main records/11 backup records]
```

Of the original 18 papers (with two `pop`'s from this point), how many dealt with the "Efron-Stein inequality"?

```
=> pop
[12 main records/18 backup records]
=> pop
[18 main records/1 backup records]
=> count
[18 main records/0 backup records]
=> and Efron-Stein
[4 (of 18) records meet criteria]
[4 main records/18 backup records]
=> list

        1984IneqalStPr      112- 114 P
        An expansion for symmetric statistics and the Efron-Stein inequality
        Vitale, Richard A.
        $U$-statistic
        94x;U-statistic

        1986AnlsStat 14     753- 758 J
        An Efron-Stein inequality for nonsymmetric statistics
        Steele, J. Michael
        Symmetric function;Variance bound

        1987AnlsStat 15    1317-1320 J
        An application of the Efron-Stein inequality in density estimation
        Devroye, Luc
        Kernel estimator
```

```
     1988StPrLet   7     105- 112 J
     A differential version of the Efron-Stein inequality: Bounding the variance of
     a function of an infinitely divisible variable
     Vitale, Richard A.
```

Note that the SEARCH command would not accept “Efron-Stein” as a search token — or more accurately, it would take the hyphen as a word-boundary, and search for the token “efron”, which we know produces more than 100 matches. Similarly, we could have refined the search in the following way (note the use of quotation marks):

```
=> and "Efron-Stein inequality"
[4 (of 18) records meet criteria]
```

2.5.4 Another Example

Here is an example of commands that find all indexed works by Paul Switzer written without a co-author (and excluding reviews):

```
=> search switzer
.....................
[42 records found]
[42 main records/4 backup records]
=> author Switzer
[35 (of 42) records meet criteria]
[35 main records/42 backup records]
=> #
=> # That gets rid of 'Switzerland', for example
=> noauth (Rev)
[29 (of 35) records meet criteria]
[29 main records/35 backup records]
=> # Now the reviews written by Switzer are gone
=> noauth ;
[14 (of 29) records meet criteria]
=> # That deleted all multi-authored works (!)
=> author "Switzer, P"
[14 (of 14) records meet criteria]
[14 main records/14 backup records]
=> # The last request verifies that the author field of all of
=> # the selected records contain both 'Switzer' as a last name,
=> # with a given name that starts with P (such as 'P.' or 'Paul')
=> pop
```

```
[14 main records/29 backup records]
=> # This command "pops" back up one step
=> diff
[15 records meet criteria]
[15 main records/14 backup records]
=> # These should contain only jointly-authored papers
=> author "Switzer, P"
[9 (of 15) records meet criteria]
[9 main records/15 backup records]
=> # These papers contain an author with one of the variants above:
=> list

        1982JIAMaGeo 14     433- 444 J
        A prior probability method for smoothing discriminant analysis classification
        maps
        Switzer, Paul;Kowalik, William S.;Lyon, Ronald J. P.
[etc]
=> whatis JIAMaGeo
   Abbreviation 'JIAMaGeo' corresponds to
   Journal of the International Association for Mathematical Geology
```

3 Refinements

Although the basic functionality of CIScd is described above, a number of powerful tools are available to conduct very focused searches.

3.1 Output Formats

CIScd can produce output (either displayed on the screen or saved to a file) in several different formats.

Short format is the default format for displaying and saving. The records in the Tutorial section above are all displayed using the **short** format.

The search results (main and backup) are stored using the same cryptic format that the Current Index Database itself uses; we call this format the **raw** format. This format makes it possible to create a mini-database that can hold the results of a complex search. It can then be refined or combined with the results of other searches at a later time (see `retrieve`, below).

Long) format lists the contents of each fields on a separate line. It is principally of use to the program developers.

Refer format outputs each record in a form that can be used by the UNIX bibliographic utility *Refer*. Files written in this format can be imported as well as by some other bibliographic management software (such as *EndNote* on the Macintosh, for example). This format is also called “Bib/Refer format” in some contexts.

Tex format produces a file that can be run through the TEX typesetting language to produce reasonably readable bibliography lists.

HTML format produces a numbered HTML list for display by a Web browser. These may require some hand editing.

Abstracts format is a special output format that lists nothing for a record unless an abstract for that entry is also present, in which case, the abstract is displayed in LATEX format. For instance, the result of the last search in the tutorial resulted in one article with an abstract. Here is what the abstract format produces:

```
=> set abstract
=> set nomore
=> list
\B{1992JCmpGrSt  1       3-  20 j}
\T{A Strategy for Binary Description and Classification}
\A{Robert Tibshirani and Michael LeBlanc}
\X
  We describe a procedure for the classification and description of
binary response data. The model is a special case of the multivariate
adaptive regression splines (MARS) model; the emphasis is on piecewise
constant basis functions, as in the classification and regression
trees (CART) approach of Breiman, Friedman, Olshen, and Stone. The
procedure is based on the logistic model for binary data. A binary
logistic model is built up as the sum of products of indicator
functions, as in MARS. The model is then pruned in a backward stepwise
manner, using cross-validation as a guide. The pruning is strictly
hierarchical (as in CART) to preserve interpretability of the final
model. Through simulated and real examples, the procedure is shown to
be more effective than CART in uncovering ``main effects'' and it can
lead to a simpler description of the data. On the other hand, it is
not as effective as CART in terms of classification error.
\Y
\K{Logistic regression; MARS; Splines.}
\E
```

3.2 Setting Output Format (`set`, `list`)

There are two ways to change the output format from the default value of **short**. One option is to permanently change the default using the `set` command, as in

```
=> set refer
```

All subsequent requests for displaying records will be in the newly requested format.

If you would like to display the current results in an alternative format without changing the default format, add the format name at the end of the list request. The following command will display the current results in `long` format, for instance, but will continue to use the default format for any subsequent requests.

```
=> list long
```

3.3 Restricting to a Range of Years (`year`, `andyear`)

If you are interested only in a range of years (for instance, if you are writing a review of materials published between 1992 and 1997), you can restrict *all* of your searches to these years using the `YEAR` command, for example:

```
=> year 1992 1997
   Year range set to 1992 through 1997
   Future searches will be restricted to these years.
```

If, however, you wish to *refine* the results of a previously conducted search, you can use the refinement command `ANDYEAR`. This method is generally more useful:

```
=> andyear 1992 1997
[5 (of 9) records meet criteria]
[5 main records/9 backup records]
```

The command `YEAR ALL` restores searches to the unrestricted range of years.

3.4 Saving to a File (save)

If it is desired to save the results of a search on a disk, the save command will prompt you for a file name, and then CIScd will copy the contents of the main search results to the file with that name. If the file name contains spaces or special characters, or if it specifies a full path name, then the file name should be enclosed in double quotation marks ("). This makes it possible to save results to a floppy disk, for instance. Here are two examples:

```
=> save "ridge regression biblio"
=> save "a:\papers\searches\ridge.cis"
```

3.4.1 Saving a file using a specified format

The save file is written using the current default format. You can specify a different format for saving the file by adding the the format name to the end of the save request, as in this example:

```
=> save best.cis refer
```

3.5 Loading a Saved File (retrieve)

It may be particularly useful to save the results of a long search, for further refinement (or perhaps for using as a fresh starting point later on). To do this, save the current search in raw format. Any file that is saved in raw format can be loaded into the main search results by using the retrieve command. Thus, the contents of the main results will be the same both before and after the following sequence of commands is given.

```
=> save best.raw raw
=> .....
=> retrieve best.raw
```

Loading files that are saved in formats other than raw format will produce unpredictable results.

3.6 More (more)

On some platforms, records that scroll off the top of the screen can no longer be viewed without doing another search. For this reason, the more command can be used to cause screen output to stop after roughly every 24 lines of output. The next screen of output is produced by typing the Enter or Return keys on the keyboard. This paging feature can be toggled on and off by repeated uses of the `more` command.

Alternatively, one can use the following commands to control this feature.

```
=> set more             (enables paging)
=> set nomore           (disables paging)
=> more 60              (stops after every 60 lines or so)
```

3.7 Screen Display (`list`, `display`)

The `list` and `display` commands show the results of the current main search at the screen. These names are synonyms. The output format can be selected as described above under "Output format".

3.8 Showing the Next *k* Records (`next`)

A convenient way of looking at a set of search results is to display from two to four records at a time, depending on the display format currently in effect. The command

```
=> next 4
```

will cause the next four records to be displayed each time the command is issued. If the last request issued was a `next` command (either with or without a following integer) an empty command line will be treated as equivalent to the `next` command. Thus, setting `next 3` followed by an empty line will display the next three items in the current search. Every time the Return key is pressed another three records will be displayed. Any command other than `next` or Return will cause the Return key to revert to its usual function.

3.9 Manipulating the Search-Result Stack (`push`, `pop`, `stack`)

The `stack` command gives a somewhat cryptic status report of the contents of the stack of search results. The `push` command and `pop` command move the stack contents by one position.

3.10 Displaying the contents of a disk file (`type`)

The `type` command will copy a named disk file to the screen. This command is intended only to examine files containing ASCII text; if it is used with binary files you deserve what you get!

3.11 Displaying Settings (`show`)

The current values of all options that can be `set` are displayed with the `show` command.

3.12 Version (`version`)

The `version` command displays both the current version of the software, and a brief description of the database being examined (as told by the configuration file).

3.13 Comment (#)

It is sometimes useful to put comments between commands. Any command line beginning with "`#`" is merely printed and ignored. Blank lines are also ignored (except immediately after a `next` command).

3.14 Batch searches (`batch`)

You can create a text file that contains CIScd commands that can be done at a later date. For instance, you may have a carefully constructed search strategy to identify articles dealing with Bayesian frequency theory. This strategy may involve the combination and refinement of many CIScd searches. The commands that perform these searches can be placed in a file which can then be run as a "batch" of commands, for instance, whenn the next release of the database becomes available. The `batch` command accomplishes this task. Any CIScd command *except* `batch` can appear in the batch file.

```
=> batch bayesfreq.txt
   ... [output would follow]
```

3.15 Looking up an Abbreviation (`whatis`)

Not every abbreviation used in the *CIS/ED* is self-explanatory. The `whatis` command will look up the abbreviation in the abbreviation master list (provided that CIScd has been configured with an abbreviation file):

```
=> whatis CmpBiomd
   Abbreviation 'CmpBiomd' corresponds to
   Computer Methods and Programs in Biomedicine
```

The command may be abbreviated to its first four letters. The lookup is *not* case-sensitive.

3.16 Printing Search Results

The availability of printing varies by computing platform. For all platforms, however, records are printed using the format currently in effect for displaying records on the screen (typically, `SHORT` format).

Better formatting can be achieved by using the `SAVE` command to write the current results to a file, then using a word processing program to open the saved file, format it, and print it.

Note. Some printers may not begin printing a physical page until the page's entire contents have been sent to it by the program (Hewlett-Packard inkjet printers are an example). In this case, pressing the reset button on the printer once you have finished sending it the material that you wish to print will initiate actual printing.

3.16.1 Macintosh Computers

Macintosh users should use the print features of the File menu.

3.16.2 Windows 95, 98, and NT

The `PRINT` command will print the current search results to `LPT1:` on DOS and Windows configurations. To have the records printed using a different format, follow the `PRINT` command with the format name, for example, `PRINT LONG`.

3.17 Restarting, or Changing Configurations (config)

Although this is rarely needed or desirable, it is possible to start the program again from the beginning, optionally reading a specified configuration file. The commands (in these two variants) are:

```
=> config            # to restart using the default configuration
=> config alt.cfg    # to restart using alt.cfg
```

3.18 Quit (quit, exit)

This command will exit the program. On the Macintosh, the window from the search session will remain active, so that it can be printed and/or saved to a file. Cut and paste are also active, and after the `quit` command is issued, the contents of the window can be edited or formatted (to delete dead ends or errors in the search, for instance). To completely quit on the Macintosh, the **Quit** option must be selected from the **File** menu (or COMMAND-Q must be typed from the keyboard).

The `exit` command is a synonym for `quit`, provided as a courtesy to *Stata* users.

CIScd creates temporary files to hold the results of searches and refinement requests. These temporary files are not deleted on quitting the program unless you specifically ask for them to be removed. (This makes it possible to exit from the program, and then to start the program again later, taking up exactly where you left a search project. If you are certain that you will not need the current intermediate results, you can give the command

```
=> quit clear
```

which will delete the work files. If you do not specify `clear`, you will be asked whether you wish to delete the files. In this case, the files will be deleted only if you explicitly request them to be.

4 Source and binary versions

CIScd is distributed in a single package that contains both binary and source versions of the program. The source code is provided so that, subject to the license agreement, users can modify, correct, improve the program for themselves and others. Executable versions are provided as a courtesy to those who are able to use them.

Executable versions of CIScd that run under Windows and MacOS operating systems are included with the program distribution. The programs are written in the C programming language, and these programs are also part of the distribution package. The source code in the Unix directories has been successfully compiled and tested on a Sun Sparc station and BSD Unix on a G4 Macintosh.

4.1 Compiling the programs

The program was developed using Metrowerks CodeWarrior version 7, a development system for the Macintosh that permits cross compilation for Windows-based platforms. The CodeWarrior projects are included in the distribution, and they may prove helpful to anyone who has CodeWarrior on the Macintosh. Others will have to construct their own `make` files or equivalent. However, the projects have a very simple structure. All `.c` and `.h` files contained in the directory for one of the programs is necessary for that program. The `main()` routine for CIScd is in the file `cislook.c`.

5 CIScd Fine Print

Just as this document is made available to you under under the terms of the GNU Free Documentation License, the software described in this section is also copyrighted material that is made available to you without charge under the GNU General Public License (GPL), which you should consult for details. A copy of the GNU GPL license must be included with the software for you to inspect.

6 GNU Free Documentation License

GNU Free Documentation License
Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified

Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this

License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
      Copyright (c)  YEAR  YOUR NAME.
      Permission is granted to copy, distribute and/or modify this document
      under the terms of the GNU Free Documentation License, Version 1.1
      or any later version published by the Free Software Foundation;
      with the Invariant Sections being LIST THEIR TITLES, with the
      Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
      A copy of the license is included in the section entitled "GNU
      Free Documentation License".
```

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no

Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.