

## VOTEORD: A revision of VOTEPOW which includes an algorithm for analyzing sets of power orderings

ARNOLD B. URKEN

Department of Humanities, Stevens Institute of Technology  
Hoboken, New Jersey 07030

and

ANTJE VAN DER VEEN-HUIJSMAN

Computer Center, Stevens Institute of Technology  
Hoboken, New Jersey 07030

VOTEPOW used combinatorial mathematics to provide a posteriori measures of actors' power in voting bodies (Kushner & Urken, 1974). One of the limitations of that analysis is that the power orderings generated by the PDIFF (pairwise probabilistic power) and E(PDIFF) (expected pairwise power differential) are not interpreted (Kushner & Urken, 1973). VOTEORD treats this problem by finding the average power of an actor in each voter ordering and ranking order averages.

**Description.** Like VOTEPOW, VOTEORD uses roll-call data to develop conditional measures of power in voting bodies. The major difference between the programs is that VOTEORD includes subprograms which analyze the implications of conditional power orderings.

To illustrate, in a voting body of N voters, VOTEORD sorts the information about power relationships into sets for each actor. In a three-person voting body, for instance, the program would list the decreasing advantage of combining with other actors from the viewpoint of each actor as follows:

Actor 1	Actor 2	Actor 3
3 (0.44)	1 (0.30)	1 (0.30)
2 (0.00)	3 (-.22)	2 (0.05)

If we assume that the orderings reflect a normalized distribution of interactions, it is possible to use the average power of each ordering to create a weak rank ordering for the entire voting body. In cases in which N is large, VOTEORD provides a means of arriving at non-obvious evaluations of power relationships in voting bodies.

**Computer and language.** VOTEORD is written in basic FORTRAN IV for a PDP-10 computer. Running time depends on the size of the voting body and the number of roll calls analyzed. Input is the same used in VOTEPOW.

**Availability.** Additional information about the program may be obtained free of charge from the Computer Center, Stevens Institute of Technology, Hoboken, New Jersey 02030. Any questions concerning the interpretation of VOTEORD may be addressed to the authors.

## REFERENCES

- Kushner, H. W., & Urken, A. B. Measuring power in voting bodies. *Public Choice*, XV, Summer, 1973, 77-85.  
Kushner, H. W., & Urken, A. B. VOTEPOW: A computer program for the measurement of voting data using roll call data. *Behavior Research Methods & Instrumentation*, 1974, 3, 219-220.

*Behavior Research Methods & Instrumentation*  
1974, Vol. 6 (6), 571

## LEXSTAT: A PL/I program for computation of lexical statistics

COLIN MARTINDALE

University of Maine, Orono, Maine 04473

LEXSTAT is a program for computation of a number of statistics commonly used to describe the lexical diversity of natural language texts. The following statistics are calculated for word frequency, word length, and sentence length: mean, standard deviation, coefficient of variation,  $S_0$ ,  $S_1$ ,  $S_2$ , and Yule's characteristic K. The program also computes the type-token ratio, log type-token ratio, hapax legomena, and several measures of information (first-order entropy,  $H_{max}$ , relative information, and redundancy) for the word-frequency distribution. Complete frequency distributions may also be printed out for word frequency, word length, and sentence length. Input text may be divided into as many documents or sections as desired. Statistics are calculated separately for each document.

**Input.** A control card specifying user-selected output options and a tape containing the input text are necessary. Text may be keypunched in free format but must be transferred to tape using a simple preprocessing utility program, STRIPT.

**Output.** The statistics described above are automatically printed. A control card option allows them to be punched onto cards also. Complete frequency distributions are printed unless suppressed by the control card. Also optional is the production of a tape containing each word type in a document, its frequency of occurrence, its length in characters, and the title of the document in which it occurred.

**Computer and programming language.** LEXSTAT is written in PL/I, Level D, for an IBM 360 or 370 with at least 200K bytes of core.

**Program capacity.** The program can process a maximum of 4,000 word types per document. In general, documents containing up to 10,000-12,000 word tokens will not exceed this limitation.

**Availability.** User's manual, program listings, and program decks may be obtained from the author at no cost.