Visualizing New Testament Language Structure Using SAS/Graph®

Abstract
The enclosed graphs attempt to illuminate the underlying grammatical and syntactic structure of New Testament Greek. In particular, displaying these language elements of the text helps to highlight usage patterns both of the language and the author.

History
The data and general techniques are based on work done by Edward A. Robson for a dissertation in linguistics at Syracuse University during the 1970s. He has continued the work as a personal research project; I became involved in 1990 to help with SAS programming and output design.

Data
The data used to create the graphs has one observation for each word in the text. Each observation currently has variables indicating chapter, word number (consecutively numbered from chapter beginning), part of speech, verb tense (if applicable), preposition information (if applicable: preposition, case, and prepositional phrase duration), and relative clause information (similar to prepositional phrase). The data structure has evolved and grown over time; other items may be added in the future. For simplicity, numbers are used to represent the information, although the value assigned has little or no meaning. Frequencies are generated to find data item distributions, although they are not the primary output of the analysis.

Display Techniques
A linear graph is generated, with each word occupying an equal amount of space. Early versions (done by Ed), plotted the data on an ordinal scale. Although this identified patterns, it implied nonexistent quantitative information (verbs represented by a "7" plotted "higher" than nouns represented by a "3"). Currently, shapes and colors are used to represent data values, and different data items are graphed on parallel "lines".

Coding Techniques
Early versions used PROC GPLOT. The current technique uses the data step to generate input for PROC GANNO. Although the Data Step Graphics Interface could have been used, the chosen technique allows examination of the data for debugging purposes prior to plotting. The program begins by setting up a dataset describing the graphic elements used to represent individual data items and values. A separate data step reads the data and generates the annotate dataset. This data step utilizes several macros to create the data points for the graph. Macros were chosen over LINK statements as they allow the use of replaceable parameters. The final step is to use PROC GANNO to generate the graph.

The program accommodates a variety of output devices, including Postscript (B&W and Color), Paintjet, HP7550, and TEK4010. This flexibility facilitates testing, but increases program complexity. Because device characteristics affect (among other things) item placement on the output, it became necessary to have scaling and placement adjustments that depend on the output device. In addition, the program has many adjustments controlling output size.

Nelson R. Pardee
c/o Syracuse University
Computing Services
120 Hinds Hall
Syracuse, New York 13244
(315) 443-1079
ACDNRP@SUVM ACDNRNP@SUVM.ACS.SYR.EDU
Language Structures in the New Testament: Revelation 1

Part of Speech
- Article
- Pronoun
- Noun
- Particle
- Participle
- Negative
- Preposition
- Adversative
- Infinitive
- Kal
- Perfect

Tense
- Active
- Passive
- Present
- Imperfect
- Future
- Perfect

Preposition
- Genitive
- Dative
- Accusative
- Flat end means continued
- 5 Apo
- 10 Ek
- 20 En
- 35 Eis
- 40 Dia
- 5 Apo
- 10 Ek
- 20 En
- 35 Eis
- 40 Dia


LET DATA

INPUT CI $1. .;

FUTURE

p.ra bes f de

hypokato under

Inl up

e15 Into 40 dh through

an in

UNUSED BLACK. SPECIAL . • • • (HAMOND

PRESENT BLACK J SPECIAL •• 2 BULLET cap J

UNUSED BLACK. SPECIAL

BLACK. SPECIAL

BLACK. SPECIAL

SOLID DBR BIO

SOLID OTl GRAY

SOLID SOLIO
** Macro MBLOCK

** Macro MYBLOCK

** Macro TENSE

** Extent TENSE

** Extent MBLOCK

** Extent MBLOCK

** Extent MBLOCK

** Extent MBLOCK

** Extent MBLOCK

** Extent MBLOCK

** Extent MBLOCK
PROC GPLOT; RUN;
*PROC GPLOT; RUN;