The Stata Journal (2006)
6, Number 3, pp. 425–427

Stata tip 34: Tabulation by listing

David A. Harrison
Intensive Care National Audit and Research Centre
London, UK
david.harrison@icnarc.org

The command list is often regarded as simply a data management tool for listing observations, but it has several little-used options that make it a useful tool for producing customized tables.

The dataset auto.dta contains the 1978 repair records (rated 1–5) for various makes of car. We can use list in its typical manner to look at some of the data:

```
use make rep78 using http://www.stata-press.com/data/r9/auto
(1978 Automobile Data)
drop if missing(rep78)
(5 observations deleted)
list in 1/5
```

<table>
<thead>
<tr>
<th>make</th>
<th>rep78</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. AMC Concord</td>
<td>3</td>
</tr>
<tr>
<td>2. AMC Pacer</td>
<td>3</td>
</tr>
<tr>
<td>3. Buick Century</td>
<td>3</td>
</tr>
<tr>
<td>4. Buick Electra</td>
<td>4</td>
</tr>
<tr>
<td>5. Buick LeSabre</td>
<td>3</td>
</tr>
</tbody>
</table>

Suppose that we wish to tabulate the makes of car according to repair record. There is no simple approach to produce tables containing strings; however, if we can modify the data so that the variables in the new dataset represent the columns of our desired table and the observations represent the rows, then we can produce the table with a list command. The reshape command (see [D] reshape) provides the ideal tool to do this:

```
by rep78 (make), sort: gen row = _n
reshape wide make, i(row) j(rep78)
(note: j = 1 2 3 4 5)
```

The first command above generated a variable defining the rows of our table. Sorting on make will ensure that the makes of car appear in alphabetical order within the table. After this reshape, the results of a simple list will still not be ideal, as the columns
will be headed \texttt{make1}, \texttt{make2}, etc. We can change this by using the \texttt{subvarname} option, which substitutes the characteristic \texttt{varname} for each variable as the column heading. Characteristics (see [P] \texttt{char}) are named items of text that can be attached to any variable or to the entire dataset. We use \texttt{forvalues} to loop through the values of \texttt{rep78} creating these characteristics. A previous article in the \textit{Stata Journal} has discussed more complex looping, including using the \texttt{levelsof} (previously \texttt{levels}) and \texttt{foreach} commands to cycle through all values of a variable (Cox 2003). Other options for \texttt{list} remove observation numbers, remove the default horizontal lines every five rows, insert dividers between the columns, and make the columns of equal width:

\begin{verbatim}
.forvalues i = 1/5 {
  2. char make'i'[varname] "Repair record 'i'"
  3. }
.list make1-make3, noobs sep(0) divider nocompress subvarname
\end{verbatim}

\begin{verbatim}
Olds Starfire
Pont. Firebird
\end{verbatim}

\begin{verbatim}
<table>
<thead>
<tr>
<th>Repair record 1</th>
<th>Repair record 2</th>
<th>Repair record 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cad. Eldorado</td>
<td>AMC Concord</td>
<td>Audy Fox</td>
</tr>
<tr>
<td>Chev. Monte Carlo</td>
<td>AMC Pacer</td>
<td></td>
</tr>
<tr>
<td>Chev. Monza</td>
<td>Buick Century</td>
<td></td>
</tr>
<tr>
<td>Dodge Diplomat</td>
<td>Buick LeSabre</td>
<td></td>
</tr>
<tr>
<td>Dodge Magnum</td>
<td>Buick Regal</td>
<td></td>
</tr>
<tr>
<td>Dodge St. Regis</td>
<td>Buick Riviera</td>
<td></td>
</tr>
<tr>
<td>Plym. Volare</td>
<td>Buick Skylark</td>
<td></td>
</tr>
<tr>
<td>Pont. Sunbird</td>
<td>Cad. Deville</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cad. Seville</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Chev. Chevette</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Chev. Malibu</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Chev. Nova</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fiat Strada</td>
<td></td>
</tr>
</tbody>
</table>
\end{verbatim}

(output omitted)

Only the first three columns are displayed here because of the width of the page, but if you require more columns than can be displayed in your results window (and are logging your output), you can use the \texttt{linesize(\#)} option to increase the available width.

Two-way tables can be achieved in a similar manner:

\begin{verbatim}
.use make rep78 foreign using http://www.stata-press.com/data/r9/auto
(1978 Automobile Data)
.drop if missing(rep78)
(5 observations deleted)
.by rep78 foreign (make), sort: gen row = _n
.qui reshape wide make, i(rep78 row) j(foreign)
gen str1 rep78txt = string(rep78) if row == 1
(50 missing values generated)
.format rep78txt %1s
.char rep78txt[varname] "Repair record"
.char make0[varname] "Domestic"
.char make1[varname] "Foreign"
\end{verbatim}
Do not underestimate what can be achieved with a simple `list`!

**References**