The *Stata Journal* publishes reviewed papers together with shorter notes or comments, regular columns, book reviews, and other material of interest to Stata users. Examples of the types of papers include 1) expository papers that link the use of Stata commands or programs to associated principles, such as those that will serve as tutorials for users first encountering a new field of statistics or a major new technique; 2) papers that go “beyond the Stata manual” in explaining key features or uses of Stata that are of interest to intermediate or advanced users of Stata; 3) papers that discuss new commands or Stata programs of interest either to a wide spectrum of users (e.g., in data management or graphics) or to some large segment of Stata users (e.g., in survey statistics, survival analysis, panel analysis, or limited dependent variable modeling); 4) papers analyzing the statistical properties of new or existing estimators and tests in Stata; 5) papers that could be of interest or usefulness to researchers, especially in fields that are of practical importance but are not often included in texts or other journals, such as the use of Stata in managing datasets, especially large datasets, with advice from hard-won experience; and 6) papers of interest to those who teach, including Stata with topics such as extended examples of techniques and interpretation of results, simulations of statistical concepts, and overviews of subject areas.

For more information on the *Stata Journal*, including information for authors, see the webpage

http://www.stata-journal.com

The *Stata Journal* is indexed and abstracted in the following:

- CompuMath Citation Index®
- Current Contents/Social and Behavioral Sciences®
- RePEc: Research Papers in Economics
- Science Citation Index Expanded (also known as SciSearch®)
- Scopus™
- Social Sciences Citation Index®

**Copyright Statement:** The *Stata Journal* and the contents of the supporting files (programs, datasets, and help files) are copyright © by StataCorp LP. The contents of the supporting files (programs, datasets, and help files) may be copied or reproduced by any means whatsoever, in whole or in part, as long as any copy or reproduction includes attribution to both (1) the author and (2) the *Stata Journal*.

The articles appearing in the *Stata Journal* may be copied or reproduced as printed copies, in whole or in part, as long as any copy or reproduction includes attribution to both (1) the author and (2) the *Stata Journal*.

Written permission must be obtained from StataCorp if you wish to make electronic copies of the insertions. This precludes placing electronic copies of the *Stata Journal*, in whole or in part, on publicly accessible websites, file servers, or other locations where the copy may be accessed by anyone other than the subscriber.

Users of any of the software, ideas, data, or other materials published in the *Stata Journal* or the supporting files understand that such use is made without warranty of any kind, by either the *Stata Journal*, the author, or StataCorp. In particular, there is no warranty of fitness of purpose or merchantability, nor for special, incidental, or consequential damages such as loss of profits. The purpose of the *Stata Journal* is to promote free communication among Stata users.

The *Stata Journal*, electronic version (ISSN 1536-8734) is a publication of Stata Press. Stata, *Stata*, Stata Press, Mata, *mata*, and NetCourse are registered trademarks of StataCorp LP.
The k-means cluster algorithm is a well-known partitional clustering method but is also widely used as an iterative or exploratory clustering method within unsupervised learning procedures (Hastie, Tibshirani, and Friedman 2009, chap. 14). When the number of clusters is unknown, several k-means solutions with different numbers of groups \( k = 1, \ldots, K \) are computed and compared. To detect the clustering with the optimal number of groups \( k^* \) from the set of \( K \) solutions, we typically use a scree plot and search for a kink in the curve generated from the within sum of squares (WSS) or its logarithm \( \log(\text{WSS}) \) for all cluster solutions. Another criterion for detecting the optimal number of clusters is the \( \eta^2 \) coefficient, which is quite similar to the \( R^2 \), or the proportional reduction of error (PRE) coefficient (Schwarz 2008, 72):

\[
\eta^2_k = 1 - \frac{\text{WSS}(k)}{\text{WSS}(1)} = 1 - \frac{\text{WSS}(k)}{\text{TSS}} \quad \forall k \in K
\]

\[
\text{PRE}_k = \frac{\text{WSS}(k - 1) - \text{WSS}(k)}{\text{WSS}(k - 1)} \quad \forall k \geq 2
\]

Here \( \text{WSS}(k) \) \( [\text{WSS}(k - 1)] \) is the WSS for cluster solution \( k \) \( (k - 1) \), and \( \text{WSS}(1) \) is the WSS for cluster solution \( k = 1 \), that is, for the nonclustered data. \( \eta^2_k \) measures the proportional reduction of the WSS for each cluster solution \( k \) compared with the total sum of squares (TSS). In contrast, \( \text{PRE}_k \) illustrates the proportional reduction of the WSS for cluster solution \( k \) compared with the previous solution with \( k - 1 \) clusters.

Because the \texttt{cluster kmeans} command does not store any results in \( \texttt{e()} \), we must use the same trick as in the \texttt{cluster stop} ado-file for hierarchical clustering to gather the information on the WSS for different cluster solutions. The following example uses 20 different cluster solutions, \( k = 1, \ldots, 20 \), and \texttt{phyled.dta}, which measures different characteristics of 80 students and is discussed in [MV] \texttt{cluster kmeans and kmedians}. The dataset is available at

\[
\text{use http://www.stata-press.com/data/r12/phyled}
\]

After the variables \texttt{flexibility}, \texttt{speed}, and \texttt{strength} are standardized by typing

\[
\text{local list1 "flex speed strength"}
\text{foreach v of varlist `list1´ {}
2. egen z_`v´ = std(`v´)
3. }}
\]

© 2012 StataCorp LP  st0262
we calculate 20 cluster solutions with random starting points and store the results in \texttt{name(clname)}:

\begin{verbatim}
    . local list2 "z_flex z_speed z_strength"
    . forvalues k = 1(1)20 {
        2. cluster kmeans `list2´, k(`k´) start(random(123)) name(cs`k´)
    3. }
\end{verbatim}

To gather the WSS of each cluster solution \texttt{cs`k'}, we calculate an ANOVA using the \texttt{anova} command, where \texttt{cs`k'} is the cluster variable. \texttt{anova} stores the residual sum of squares for the chosen variable within the defined groups in \texttt{cs`k'} in \texttt{e(rss)}, which is exactly the same as the variable’s sum of squares within the clusters. To collect the information on all cluster solutions, we generate a $20 \times 5$ matrix to store the WSS, its logarithm, and both coefficients for every cluster solution $k$.

\begin{verbatim}
    . * WSS matrix
    . matrix WSS = J(20,5,.)
    . matrix colnames WSS = k WSS log(WSS) eta-squared PRE
    . * WSS for each clustering
    . forvalues k = 1(1)20 {
        2. scalar ws`k´ = 0
        3. foreach v of varlist `list2´ {
            4. quietly anova `v´ cs`k´
            5. scalar ws`k´ = ws`k´ + e(rss)
        6. }
        7. matrix WSS[`k´, 1] = `k´
        8. matrix WSS[`k´, 2] = ws`k´
        9. matrix WSS[`k´, 3] = log(ws`k´)
        10. matrix WSS[`k´, 4] = 1 - ws`k´/WSS[1,2]
        11. matrix WSS[`k´, 5] = (WSS[`k´-1,2] - ws`k´)/WSS[`k´-1,2]
        12. }
\end{verbatim}

Finally, we use the columns of the output matrix \texttt{WSS} and the \texttt{matplot} command to produce plots of the calculated statistics.

\begin{verbatim}
    . matrix list WSS
    WSS[20,5] k WSS log(WSS) eta-squared PRE
    r1 1 237 5.4680601 0 .
    r2 2 89.351871 4.4925871 .62298789 .62298789
    r3 3 56.208349 4.0290653 .76283397 .3709326
    r4 4 16.471059 2.8016049 .93050186 .70696419
    r5 5 13.823239 2.6263512 .9416741 .16075591
    r6 6 12.737676 2.5445642 .94625453 .07853172
    (output omitted)
    . local squared = char(178)
    . _matplot WSS, columns(2 1) connect(l) xlabel(#10) name(plot1, replace) nodraw>
    . noname
    . _matplot WSS, columns(3 1) connect(l) xlabel(#10) name(plot2, replace) nodraw>
    . noname
    . _matplot WSS, columns(4 1) connect(l) xlabel(#10) name(plot3, replace) nodraw>
    . noname ytitle({&eta}squared')
\end{verbatim}
A. Makles

. _matplot WSS, columns(5 1) connect(1) xlabel(#10) name(plot4, replace) nodraw
> noname
(1 points have missing coordinates)
. graph combine plot1 plot2 plot3 plot4, name(plot1to4, replace)

The results indicate clustering with \( k = 4 \) to be the optimal solution. At \( k = 4 \), there is a kink in the \( WSS \) and \( \log(WSS) \), respectively. \( \eta_4^2 \) points to a reduction of the \( WSS \) by 93\% and \( \text{PRE}_4 \) to a reduction of about 71\% compared with the \( k = 3 \) solution. However, the reduction in \( WSS \) is negligible for \( k > 4 \).

![Figure 1. WSS, log(WSS), \( \eta^2 \), and \( \text{PRE} \) for all \( K \) cluster solutions](image)

In figure 2, we see a scatterplot matrix of the standardized variables for the four-cluster solution, which indicates the four distinct groups of students.

. graph matrix z_flex z_speed z_strength, msym(i) mlab(cs4) mlabpos(0)
> name(matrixplot, replace)
Although the results seem quite clear, this is not always the case. The results of a traditional $k$-means algorithm always depend on the chosen initialization (that is, the initial cluster centers) and, of course, the data.

Figure 3 again shows results for *physed.dta* but for 50 different starting points. Here our optimal solution with four clusters occurs 37 times (75%). Ten (20%) results point to the five-cluster solution to be the optimal number of groups. Hence, depending on the initialization, natural clusters may be divided into subgroups, or sometimes no kink is even visible. The best way to evaluate the chosen solution is therefore to repeat the clustering several times with different starting points and then compare the different solutions as done here.

Figure 2. Scatterplot matrix of the standardized variables for the four-cluster solution
Figure 3. Fifty different WSS, log(WSS), $\eta^2$, and PRE curves for $K = 20$

References
