Documentation for the R-code to implement the Jump methodology in “Finding the Number of Clusters in a Data Set : An Information Theoretic Approach”

CATHERINE SUGAR[*] AND GARETH M. JAMES[*]

The R directory contains three functions. The main function that calls the others is “jump”.

Description

This function takes a set of data and attempts to estimate the number of clusters using the Jump methodology outlined in the paper.

Installing

To install the software download the file jump, open R and type

```
> source(‘‘filename’’)
```

where filename is the name you saved the file under (don’t forget to give the directory structure in addition if needed.) Now if you type ls() you should see

```
> ls()
[1] "compute.jump"    "jump"            "kmeans.rndstart"
```

jump is the main procedure. Try the following commands one at a time.

```
> testdata <- matrix(c(rnorm(2000),3+rnorm(2000)),byrow=T,ncol=4)
> temp <- jump(testdata,y=c(1.5,2,2.5),rand=10,trace=F)
```

You should then see 9 plots and the following output.

```
[1] "The maximum jump occurred at  2 clusters with Y= 1.5"
[1] "The maximum jump occurred at  2 clusters with Y= 2"
[1] "The maximum jump occurred at  2 clusters with Y= 2.5"
```

This tells us that with the transformations $Y = 1.5, Y = 2$ and $Y = 2.5$ the jump method selected 2 clusters as the optimal number. testdata consists of 2 well separated clusters so this is as we would hope. Try reducing the separation by changing the 3 in the testdata line above. Eventually, the jump method will select 1 cluster at $Y = 1.5$ and many clusters at $Y = 2.5$. However, you should still see the correct answer at $Y = 2$ until there is little separation between the clusters. The 9 plots correspond to the distortion, transformed distortion and jumps for each value of Y. See Value below for further details.

[*]Marshall School of Business, University of Southern California

Arguments

jump allows for up to nine inputs. They are:

data : This should be a n by p matrix with each row corresponding to a p-dimensional observation. This is the data we wish to cluster.

K : This corresponds to the maximum number of clusters to test i.e. we will cluster the data into 1 cluster, 2 clusters etc. up to K clusters. By default $K = 10$.

y : This is the transformation power i.e. the distortion will be transformed to the power $-y/2$. This can be a vector containing more than one value for y. The theory suggests $y = p/2$ as the optimal value. However, this assumes that the covariance within each cluster is a multiple of the identity. In practice this will not be the case so values of $y < p/2$ should be tried. It is recommended that multiple different values be tested (there is almost no additional computational burden) to assess the sensitivity of the final answer to y. By default $y = p/2$.

plotjumps : This is a true/false variable indicating whether the distortions, transformed distortions and jumps should be plotted. The jump plot also contains a dashed line and a red dot indicating the maximum jump point. These plots are often very useful because they can identify patterns that one may miss if only the maximum jump is considered. For example, a plot where the jump seems to keep increasing with the number of clusters is probably an indication that y is too large. One can also potentially identify substructures in the data. For example, a plot showing large jumps at 3 and 6 clusters may indicate 3 main clusters each consisting of 2 subclusters. By default plotjumps=T.

rand : This indicates the number of random restarts to use in the kmeans fitting algorithm. Each restart chooses a new random set of cluster centers. The fit corresponding to the lowest distortion (for each possible number of clusters) is chosen. By default rand=10.

fits : This is an optional argument for supplying the kmeans fits from a previous run of jump. Supplying this argument saves time because the kmeans algorithm is not required. One may use this, for example, to test different values of y. By default fits=NULL.

B : This indicates the number of bootstrap iterations to run. If $B = 0$ then the bootstrap procedure is skipped. If $B > 0$ then the data is resampled B different times. For each bootstrapped data set, kmeans is run and the jump method is used to compute the maximum jump on the new data. The proportion of bootstrap data sets for which the maximum jump corresponds to the maximum jump on the original data is printed (for each value of y).

dist : The jump function can also be run using presupplied distortions. For example, one may wish to use a different clustering procedure than kmeans. Simply, compute the distortions for $k = 1, \ldots, K$ clusters, supply them as dist= and leave data empty. This will cause jump to skip the kmeans step and simply calculate the jumps and plots etc. on the distortion that has been supplied. Note that the bootstrap procedure can not be implemented in this case. By default dist=NULL.

trace : This is a true/false variable indicating whether to print each bootstrap iteration as it is performed. This should only be used for very large data sets where the procedure may run slowly. The default is trace=F.

Value

jump returns a list containing up to six components. In addition jump prints the maximum jump point for each value of y supplied and the proportion of bootstrap data sets with maximum jump equal to that for the original data. Plots of the raw distortion, transformed distortion and the jumps are plotted for each value of y. The jump plots also contain a dashed line with a red dot at the end to indicate the maximum jump point. The six components in the list are:

maxjump : This is a vector of length q where q is the length of the vector y. Each element corresponds to the number of clusters where the maximum jump occurs.

dist : This is a vector of length K with the distortions (either supplied by the user or calculated using kmeans) for $k = 1,\dots,K$ clusters.

transdist : This is a q by K matrix. The ith row corresponds to the transformed distortion using the ith element of y.

jumps : This is a q by K matrix. The ith row corresponds to the jumps, i.e. difference in transformed distortions, using the ith element of y. Hence the first row gives the jump from 0 to 1 cluster, the second row the jump from 1 to 2 clusters etc.

fits : A list of length $K-1$ with the kmeans output for $k = 2,\dots,K$. This can be fed back into jump to avoid having to recompute the kmeans fits.

boot.result : This is a q by K matrix. The i, jth element gives the proportion of bootstrap data sets for which the jump method selected j as the optimal number of clusters while using the ith component of y. This gives an idea of the certainty in the predicted number of clusters. Ideally, a high percentage of the bootstrap data sets should pick the same number of clusters as with the original data.