clusterfly

exploring cluster analysis using *R* and *GGobi*

http://had.co.nz/clusterfly

Hadley Wickham, Iowa State University

Typically, there is somewhat of a divide between statistics and visualisation software. Statistics software, particularly R, provides implementation of cutting edge research methods, but limited graphics. Visualisation software will provide sophisticated visual interfaces, but few statistical algorithms. This poster presents some early experimentation aimed at overcoming this deficiency by linking R and GGobi. Cluster analysis was chosen as it is an exploratory method that needs sophisticated visualisation and statistical algorithms

Components

R: a statistical environment used by most researchers in statistics. Add on packages provide implementation of cutting-edge research methods. Available from http://www.r-project.org

GGobi: Interactive and dynamic visualisation for high-dimensional (especially continuous) data. Available from http://www.ggobi.org

rggobi: An R package for controlling GGobi from R. Currently allows transfer of data and visual attributes between the two programmes. Available form http://www.ggobi.org/rggobi

clusterfly: An R package that implements the methods presented in this poster. Available from http://had.co.nz/clusterfly

Clustering methods

Since we are using R, we immediately get a rich list of clustering techniques (package name in parentheses):

Divisive hierarchical: diana (cluster)

Agglomerative hierarchical: hclust

Convex clustering: cclust (flexclust)

K-means, k-mediods, k-centroids: kmeans, kcca (flexclust), clara (cluster)

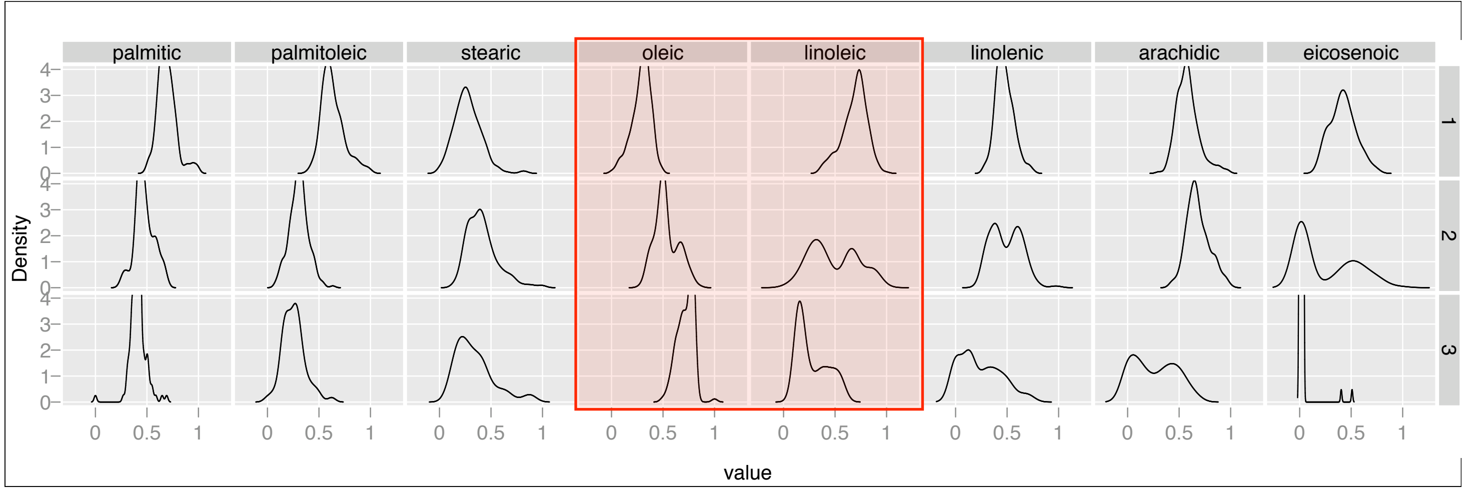
Model based clustering: Mclust (mclust)

Fuzzy clustering: fanny (cluster)

Self organising maps: som (som)

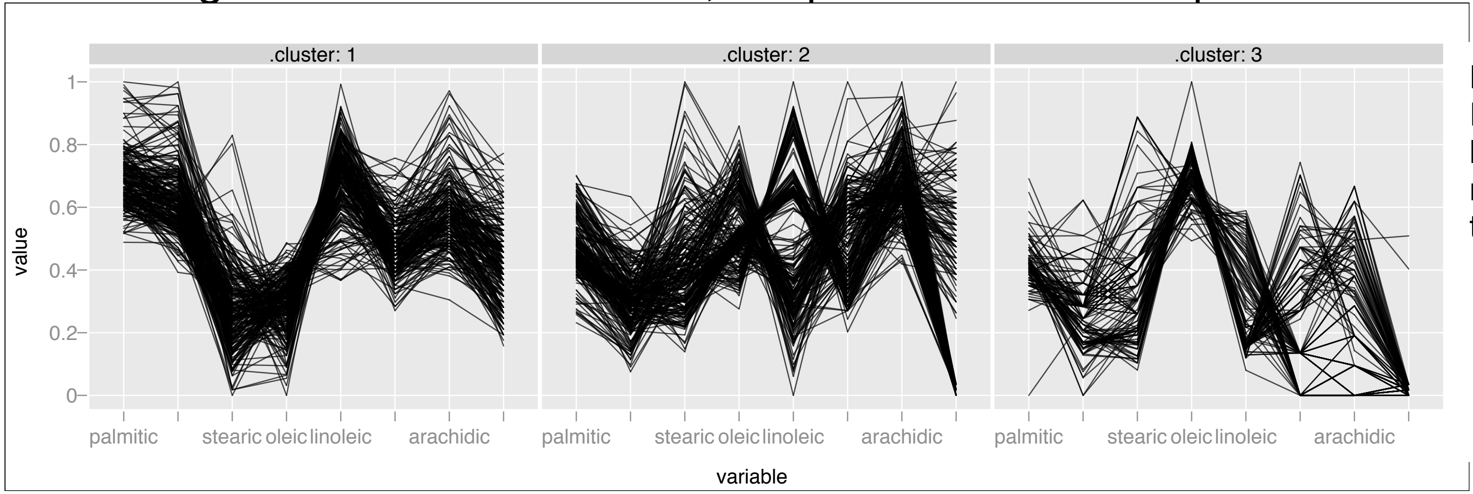
Cluster structure

After clustering, we are typically interested in how the clusters differ (and are similar). The first thing to look at is univariate distributions:



Oleic and linoleic variables seem to differ considerably between clusters.

For finding multivariate differences, the parallel coordinate plot can be useful:

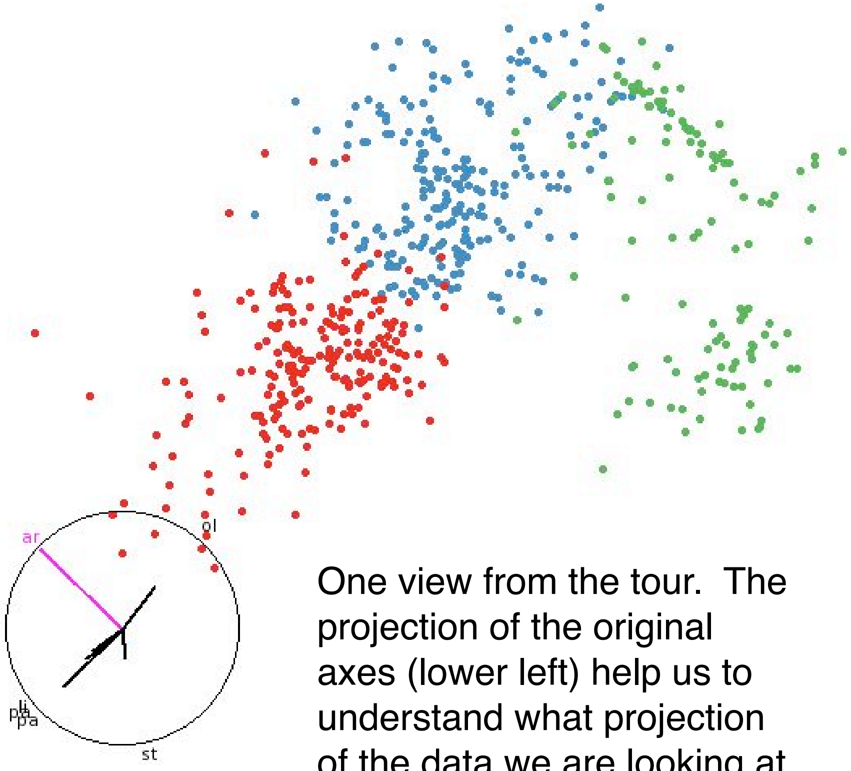


Looking at the PCP, cluster two looks a bit like a mixture of clusters 1 and 3.

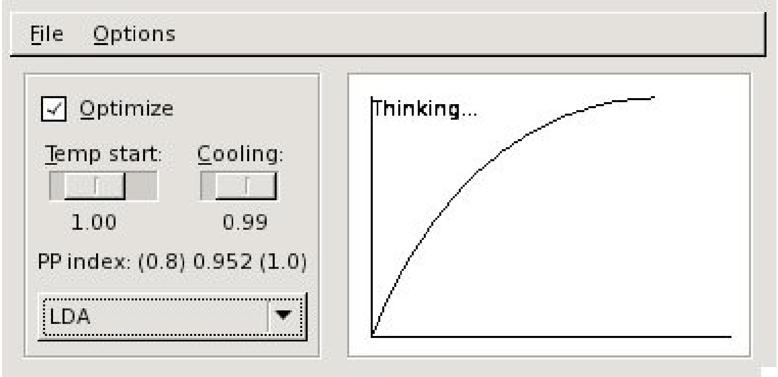
These are static plots and only proofs of concept: interactivity is essential to make them useful.

Grand and guided tour

With a high-dimensional data set, it is useful to get a sense of the clustering in the original dimensions. One way to do this is using the tour in GGobi. The **grand** tour interpolates between randomly chosen projections to (eventually) show every possible view of the dataset. The **guided** tour (= projection pursuit + tour) interpolates between projections which optimise some criterion. One useful index is LDA, which is high when then between groups variance is large relative to the within group variance.



One view from the tour. The projection of the original axes (lower left) help us to understand what projection of the data we are looking at



☑ Optimize

Temp start: 1.00

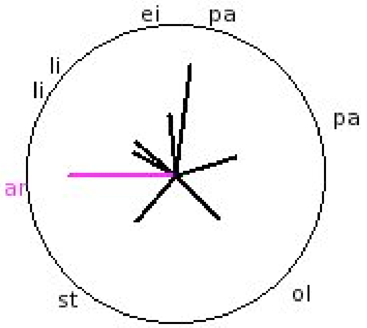
Cooling: 0.99

PP index: (0.8) 0.952 (1.0)

LDA

Thinking...

Projection pursuit tour dialog box control options for the guided tour. Plot on the right shows how the objection function changes with time.



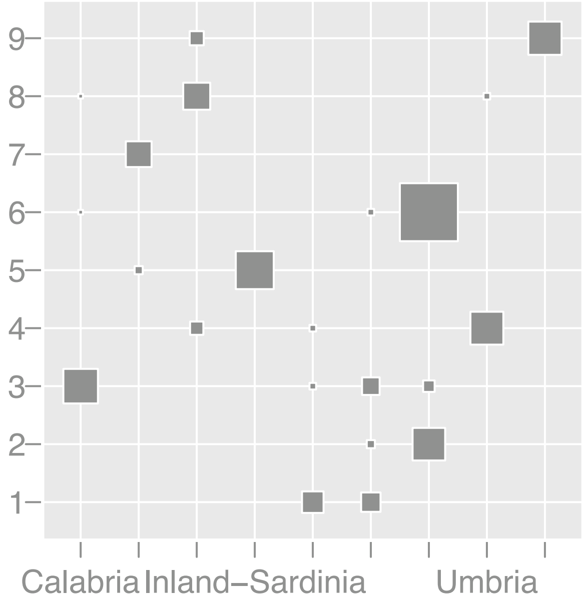
The axes can be interpreted like the axes of a biplot: Each line represents a variable. The closer to the circle, the better represented the variable is. The edge of the line points towards high values of this variable.

Fluctuation diagram

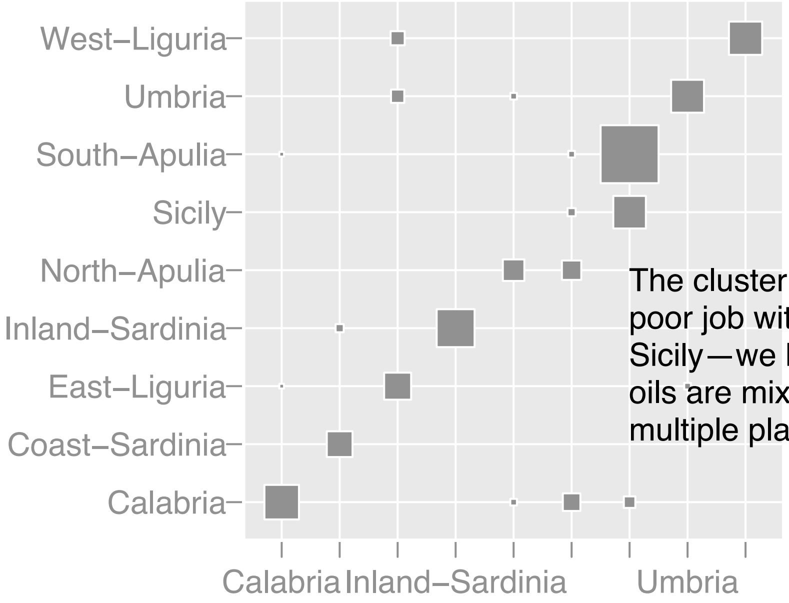
The fluctuation diagram is a type of *mosaic* plot, and can be used to visualise tabulated data. It is related to Bertin’s *reorderable matrices* and *heatmaps*.

One problem we encounter when comparing clusterings is that cluster labelling is arbitrary, and we need some way of matching labels between groups.

There are linear programming techniques available, but we use simple row maximum heuristic provided by the e1071 package.



Uninformative and arbitrary labelling of clusters found with kmeans clustering (y-axis). True regions are labeled on the x-axis.



The clustering does a poor job with oils from Sicily—we know these oils are mixture from multiple places

After relabelling, the pattern is much more obvious. Although typcially we won’t have a true labelling, this method is still useful when comparing results from two methods

Example code

```
library(clusterfly)
o <- clusterfly(olives[, -(1:2)])
o[["Region"]] <- olives$Region
o <- cfly_cluster(o, hierachical, 3)
o <- cfly_cluster(o, kmeans, 3)
cfly_show(o, “Region”)
cfly_show(o, “hierarchical”)
clfy_pcp(o, “kmeans”)
cfly_animate(o,
   c(“Region”, “kmeans”)
)
```

Try the live demonstration!

Future plans

Provide custom visualisations for different clustering methods: eg. show interactive hierarchy for hierarchical methods, underlying grid for SOM

Provide a graphical user interface to make it easy to experiment with changing clustering parameters.

Make static graphics interactive and linked to all other views (part of my thesis work)

Figure out how to deal with much larger numbers of clusters (small tour multiples?)