Gifi Goes Logistic

Jan de Leeuw

Gifi goes Logistic

Abstract

The techniques in the Gifi system for descriptive multivariate analysis are based on least squares loss functions, alternating least squares algorithms, and star plots.

We develop an alternative system here using logistic likelihood functions, majorization algorithms, and Voronoi plots.

The Data

Suppose $H=\{h_{ij}\}$ is a *data frame*, i.e. a set of n observations on m categorical variables. Variable j has k_j categories (values). There is no restriction that $n \geq m$.

The variables can have *numerical*, *ordered*, or *nominal* categories, and they can be grouped into *sets of variables* that have different roles in the analysis (such as input-output, background, predictors, confounders, outcomes, and so on).

Coding

Data are coded as *m indicator matrices* G_j, where G_j is an $n \times k_j$ binary matrix whose rows add up to one.

There are extensions possible to *missing data*, in which some rows add up to zero, and to *fuzzy indicators* in which rows are non-negative and add up to one but are not necessarily binary.

Star Plots

Suppose we have a *representation* (*map*) of the n objects as n points in low-dimensional space (usually the plane).

Make m copies of the map, one for each variable. For each variable j connect all points in category one to the centroid of category one, ... , all points in category k_j to the centroid of category k_j. Thus each map gets n lines, connecting the objects to the centroid of the category they are in.

This plot is the *star plot* for variable j. We want the k_j stars in the plot to be small, relative to the total size of the plot. Or: we want the *within-category variance* to be small relative to the *total variance*.

Thus the basic problem of homogeneity analysis: make a map of the n objects such that the stars are small (for each variable, if possible). In the usual case we actually measure size by using *squared* line length.

Homogeneity

A solution X for the objects is *normalized* if $X'X=I$.

Heterogeneity of a normalized solution is measured by

$$\sigma(X) = \frac{1}{m}\sum_{j=1}^{m}\mathbf{tr}\,(X - G_jY_j)'(X - G_jY_j),$$

where

$$Y_j = (G_j'G_j)^{-1}G_j'X = D_j^{-1}G_j'X.$$

Alternating Least Squares

To minimize heterogeneity over normalized maps we use *alternating least squares* or *reciprocal averaging*.

$$Y_j^{(k)} = D_j^{-1}G_j'X^{(k)},$$

$$Z_j^{(k)} = \frac{1}{m}\sum_{j=1}^{m}G_jY_j^{(k)},$$

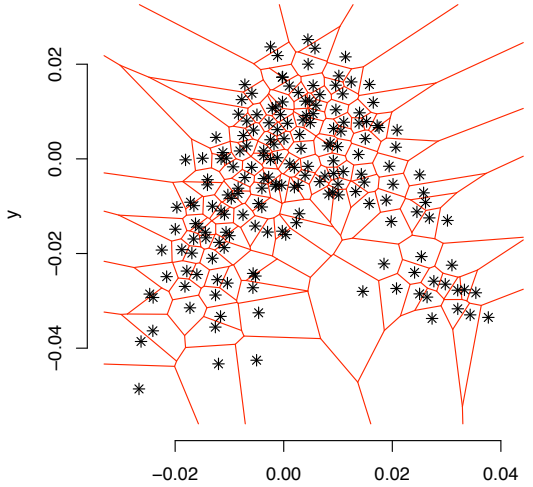
$$X^{(k+1)} = \mathbf{orth}(Z_j^{(k)}).$$

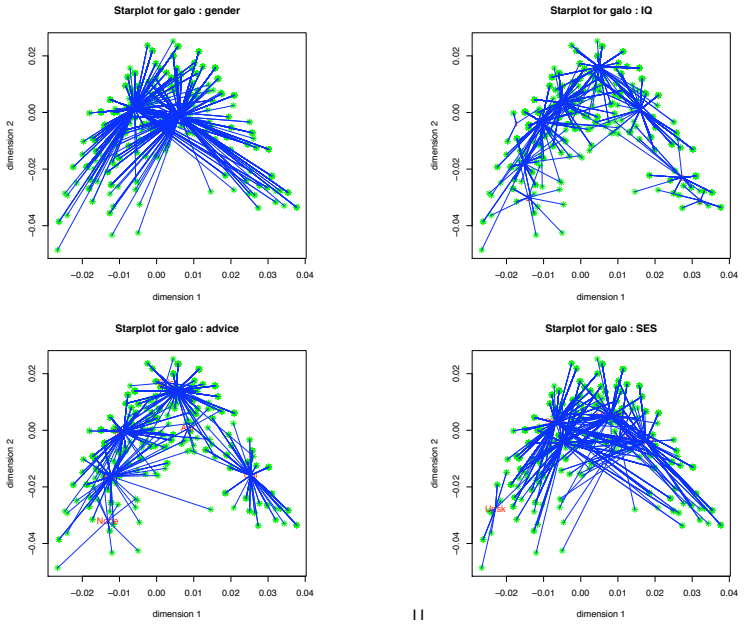
This is the Bauer-Rutishauser *simultaneous iteration method* to compute the *eigenvectors* corresponding to the p dominant *eigenvalues* of

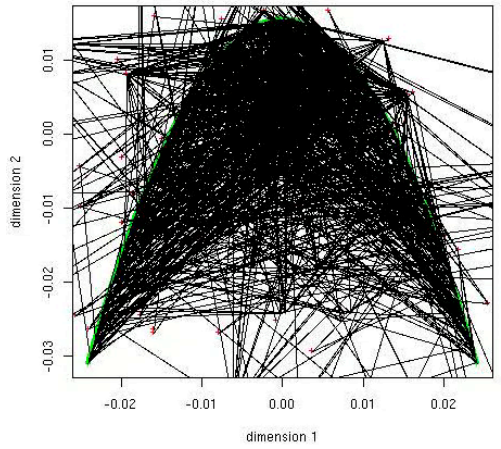
$$P_\star = \frac{1}{m}\sum_{j=1}^{m} P_j \text{ with } P_j = G_j D_j^{-1} G_j'.$$

The method capitalizes nicely on the sparseness of the indicators and converges quickly and reliable to the global minimum of the loss function.

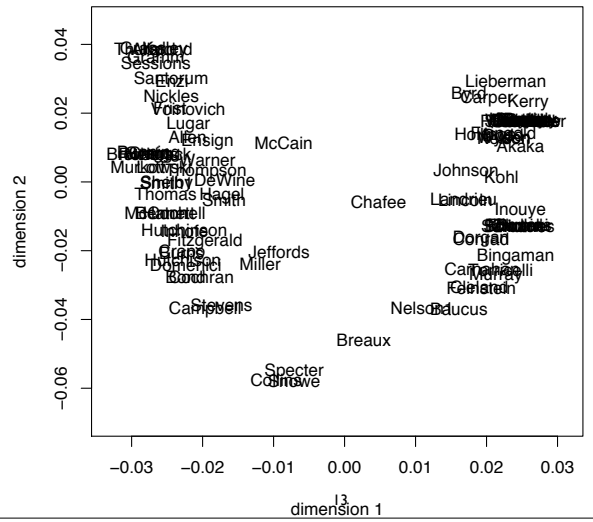
GALO Example

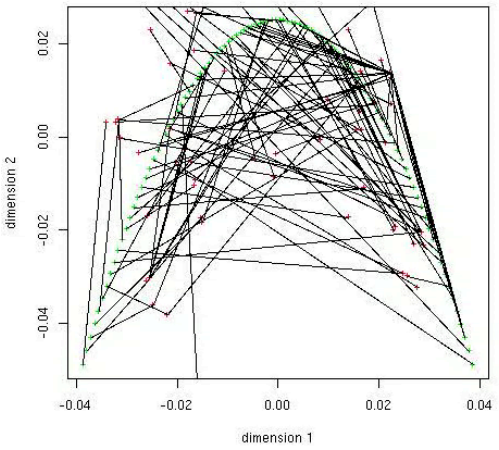




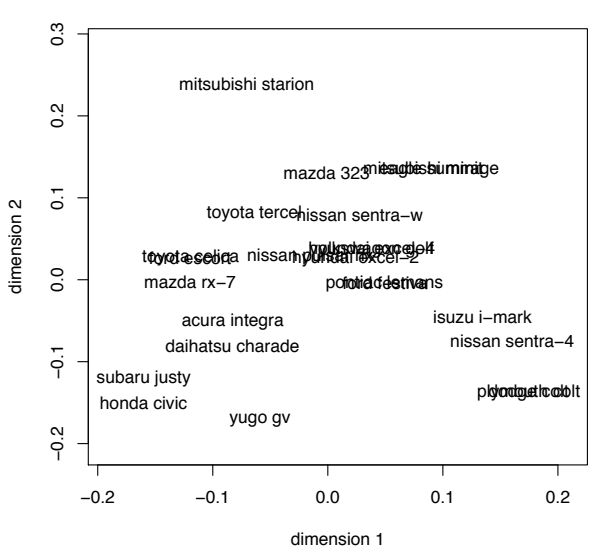


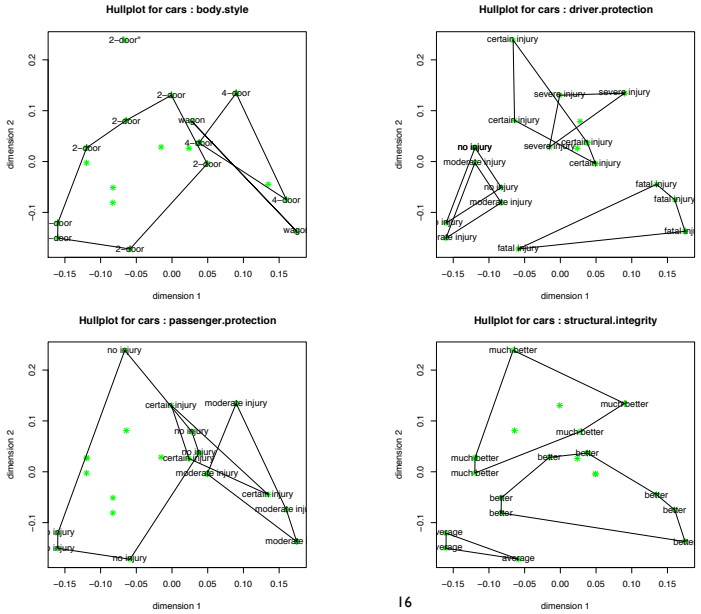
Senate Example





Cars Data





Two variables

If there are only two variables, then homogeneity analysis becomes correspondence analysis. One way of thinking about CA is making the approximation

$$f_{ij} \approx \alpha_i \beta_j (1 + \sum_{s=1}^{p} x_{is} y_{js})$$

using a least squares loss function. Homogeneity analysis with $m > 2$ makes a similar approximation to the *Burt table*.

Rank and Level Restrictions

We see from the examples that *order relations* between categories are not always respected, and that in some cases maps are bend into somewhat redundant *horseshoes*.

In the Gifi system these problems are resolved by restricting the Yj in the loss function by the rank restrictions

$$Y_j = q_j a_j',$$

which requires the Y_j to be on a line through the origin.

Now for *computation* we use

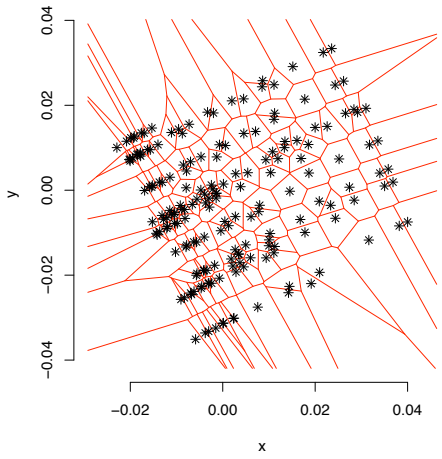
$$\sigma(X, Y) = \frac{1}{m}\mathbf{tr}\,(X - G_j\hat{Y}_j)'(X - G_j\hat{Y}_j) + \\ + \frac{1}{m}\mathbf{tr}\,(\hat{Y}_j - q_j a_j')D_j(Y_j - q_j a_j').$$

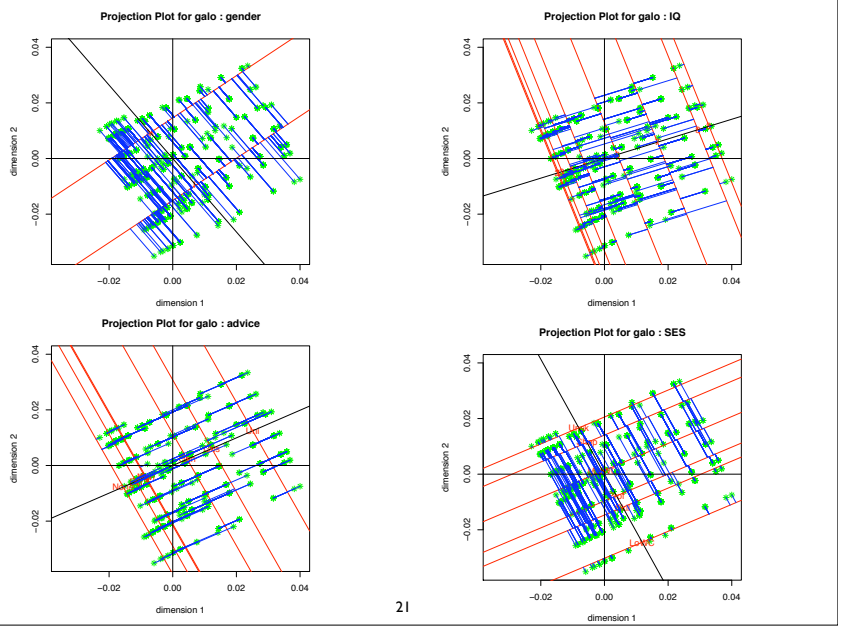
and for *interpretation* we use

$$\sigma(X, Y) = \frac{1}{m}\mathbf{tr}\,(X - G_j q_j a_j')'(X - G_j q_j a_j') = \\ = \frac{1}{m}\mathbf{tr}\,(R - XA')'(R - XA') + (p - 1).$$

where R has the m columns $G_j q_j$.

GALO Again





Interactive Coding and Additivity

If we have r variables with $k_1,\ldots,k_r$ categories these can be coded as one variable with $k_1 \times \ldots \times k_r$ categories. We can then restrict the Y to be on a grid

$$y_{j_1,\cdots,j_r;s} = y^1_{j_1 s} + \cdots + y^r_{j_r s}$$

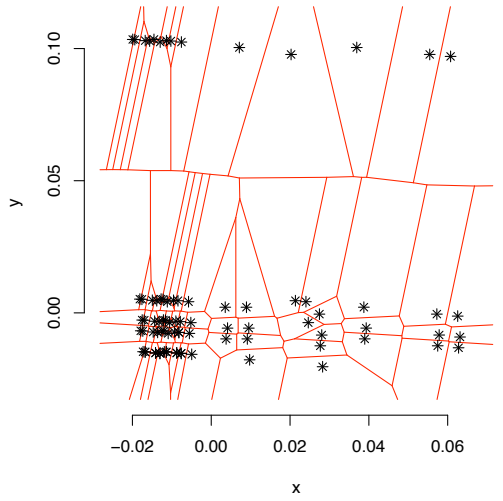
or on a rank-one grid

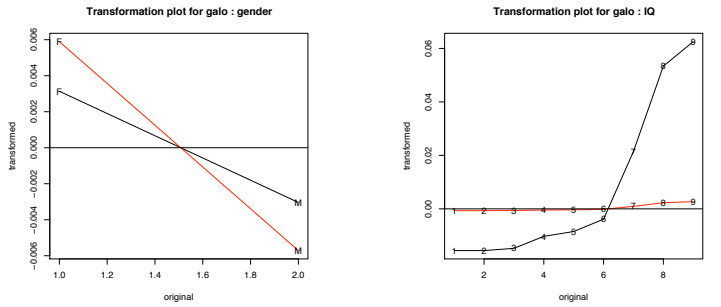
$$y_{j_1,\cdots,j_r;s} = q^1_{j_1} a^1_s + \cdots + q^r_{j_r} a^r_s.$$

These additivity restrictions allow one to efficiently incorporate *sets of variables*. If we combine the predictors into a single set, for instance, we have *regression analysis*; in the same way we can have *canonical analysis* in its various forms.

This allows one to incorporate much of classical descriptive multivariate analysis, coupled with the notion of optimal scaling or transformation.

GALO Again





Summary

We have: flexible system with fast least squares algorithms.

We have but may not want: we have to impose orthogonality constraints to get multidimensional solutions and they bring us *horseshoes*.

We do not have: interpretation on a natural probability scale, notion of fitting a model, notion of separation.

Gifi Goes Logistic

Instead of starting from

$$f_{ij} \approx \alpha_i \beta_j (1 + \sum_{s=1}^{p} x_{is} y_{js})$$

start from

$$f_{ij} \approx \alpha_i \beta_j \exp(\eta(x_i, y_j))$$

with choices

$$\eta(x_i, y_j) = x_i' y_j,$$
$$\eta(x_i, y_j) = -\|x_i - y_j\|,$$
$$\eta(x_i, y_j) = -\|x_i - y_j\|^2.$$

Instead of starting from least squares loss, start from the *negative Poisson log-likelihood*

$$\Delta(\alpha, \beta, X, Y) = \sum_{i=1}^{n} \sum_{j=1}^{m} \{\lambda_{ij} - f_{ij} \log \lambda_{ij}\},$$

with

$$\lambda_{ij} = \alpha_i \beta_j \exp(\eta(x_i, y_j)).$$

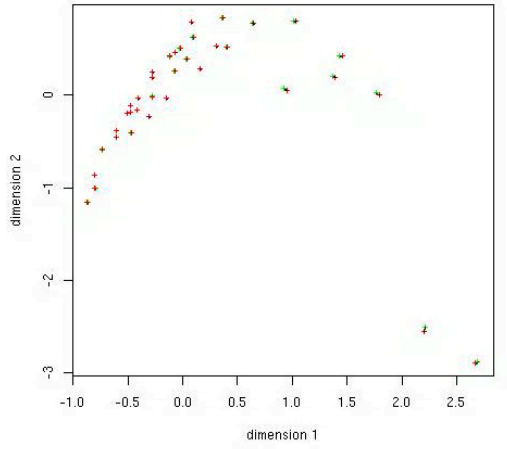
And instead of alternating least squares uses *majorization*.

In this context we use *Uniform Quadratic Majorization*, which amounts to (lots of technical detail omitted) minimizing (or continuously decreasing) loss functions of the form

$$\sigma(\alpha,\beta,X,Y)=\sum_{i=1}^n\sum_{j=1}^m(\eta(x_i,y_j)-z_{ij}^{(k)})^2.$$

Here the *target Z* changes from one iteration to the next, but all subproblems are standard least squares multidimensional scaling problems.

Rothkopf Morse Code Data



Now this just seems to generalize ordinary CA, not homogeneity analysis. But now make the step of applying the same loss function to indicator matrices

$$\Delta(\alpha,\beta,X,Y)=\sum_{i=1}^n\sum_{j=1}^m\sum_{\ell=1}^{k_j}\{\lambda_{ij\ell}-g_{ij\ell}\log\lambda_{ij\ell}\},$$

where

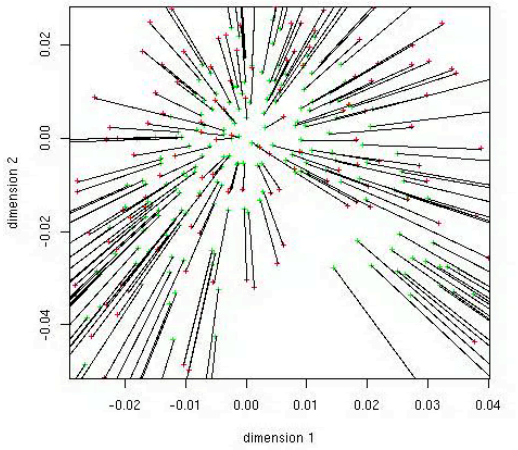
$$\lambda_{ij\ell}=\alpha_{ij}\beta_{j\ell}\exp(\eta(x_i,y_{j\ell})).$$

This brings us close to the basic Gifi setup, but we need one final step.

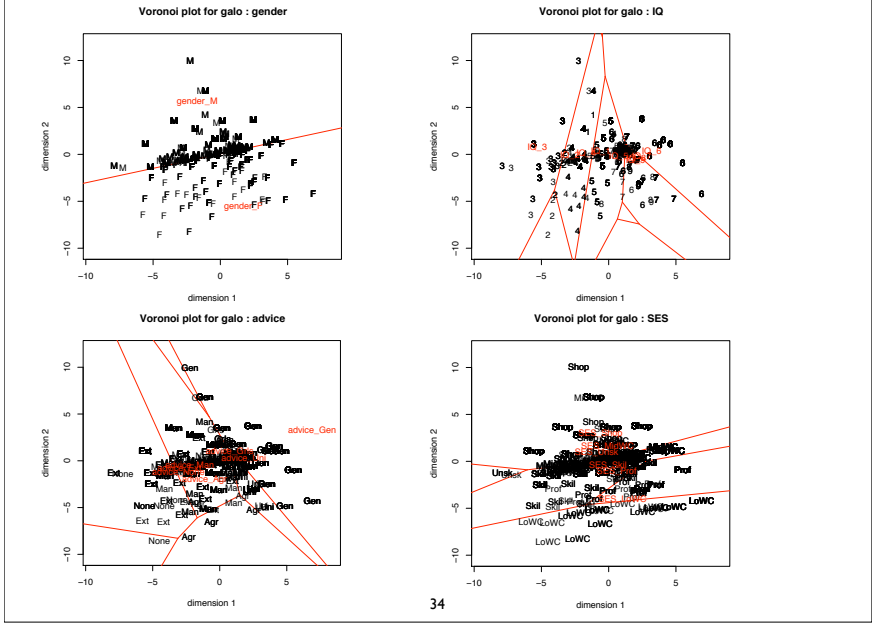
$$\min_\alpha\Delta(\alpha,\beta,X,Y)=$$
$$=\sum_{i=1}^n\sum_{j=1}^m\sum_{\ell=1}^{k_j}g_{ij\ell}\log\frac{\beta_{j\ell}\exp(\eta(x_i,y_{j\ell}))}{\sum_{\nu=1}^{k_j}\beta_{j\nu}\exp(\eta(x_i,y_{j\nu}))},$$

except for some constants. And this is what we hit with majorization, potentially using all the Gifi restrictions on the Y_j.

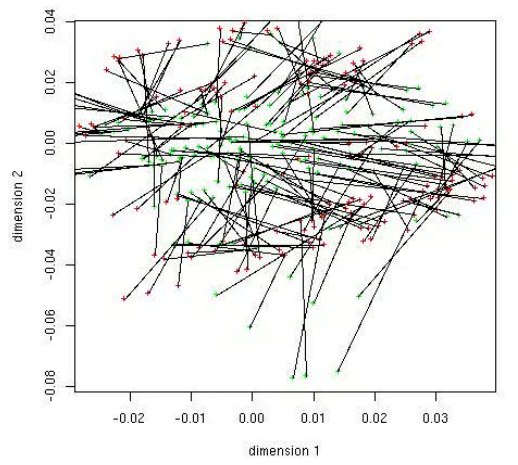
GALO Again



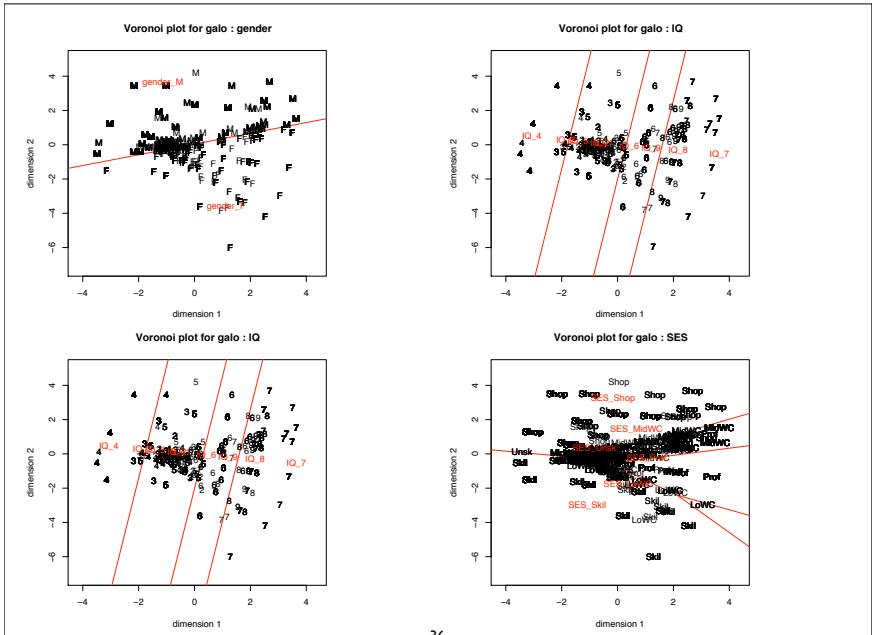
33



34

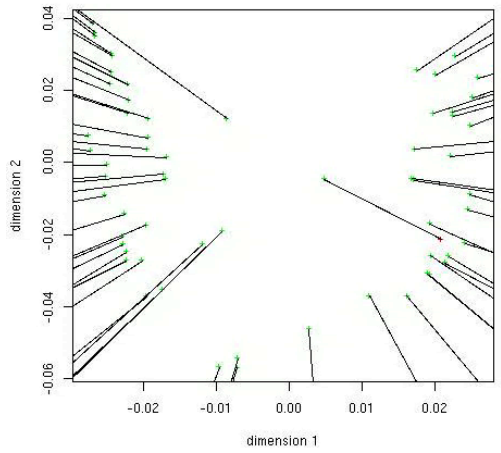


35

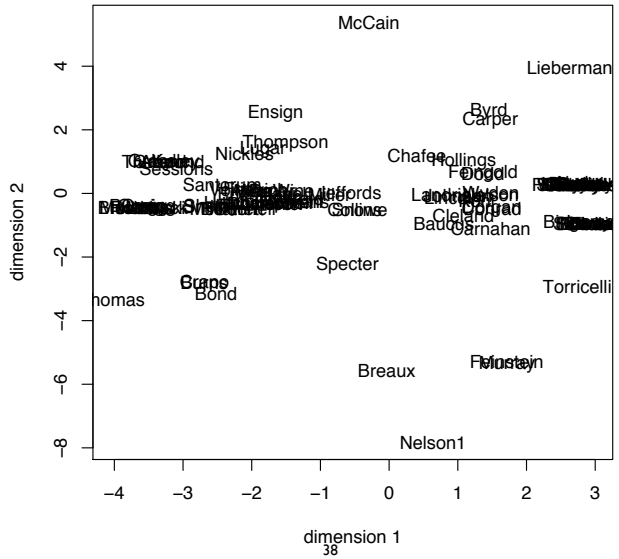


36

Senate Again



Row Objects senate



I did say earlier that the star plots are replaced by *Voronoi plots*. This can be made more clear. Suppose there are *no weights* and we use one of the *distance combination rules*.

Loss can be made equal to zero if and only if there is a solution to the system of strict inequalities

$$\|x_i - y_{j\ell}\| < \|x_i - y_{jv}\| \qquad \forall i, j, \ell \ni g_{ij\ell} = 1.$$

Or: if and only if each object can be placed closest to the category point the object is in if and only if each object is in the correct category Voronoi cell.

Summary

We have: an alternative system, which inherits (and extends) the flexibility of the Gifi system. A new geometrical interpretation, new convergent algorithms, a likelihood interpretation.

We have but may not want: very heavy computation, slow convergence, more clumping.

We do not have: experience with the new system.