Robust Objective Function Cluster Analysis

Michael P. Windham

University of South Alabama

Abstract: Cluster analytic methods include those which identify and describe clusters by optimizing an objective function that measures goodness of fit between a cluster description and data describing the objects being studied. The most common are the partitioning methods, such as k-means, mixture analysis, and fuzzy clustering. These procedures are not always robust in that results can misrepresent essential structure due to the presence of noisy data, particularly outliers. On the other hand there are many approaches to robust estimation, for example M-estimation. This paper will show how to systematically incorporate M-estimators into objective function based cluster analysis. I will also discuss how concave functions play a fundamental role in the structure and minimization of these objective functions.

Keywords: Partitioning, k-means, mixture analysis, fuzzy clustering, M-estimators

Author's address: Michael P. Windham, Department of Mathematics and Statistics, University of South Alabama, Mobile, AL 36699-0002, USA

1. Introduction

Understanding and using cluster analysis is now more important than ever, because of the increasing complexity of situations for which data analysis is needed. Cluster analysis attempts to identify substructure in a data set or population by clustering objects based on their being more similar to each other than to other objects in the study. Objective function methods attempt to determine descriptions of clusters that minimize a measure of incompatibility between the descriptions and data describing the objects being studied.

A difficulty in the use of these methods is illustrated by an example. Figure 1.(a) shows a 2-dimensional data set with three concentrations of points that might represent "clusters". There are also points that appear to be outliers or noise. The goal is to identify and describe the three clusters that represent the bulk of the structure of the data using ellipses that give the location and shape of the concentrations as illustrated in Figure 1.(b). Unfortunately, the result obtained from a classic objective function method produces the ellipses shown in Figure 1.(c). The confusion caused by the noise in the data is apparent. This paper will describe a general procedure for building objective functions for clustering that are not as easily confused by noise, that is, they are robust to the presence of outliers in data.

Section 2. describes the classical objective functions for partitioning, mixture and fuzzy clustering methods. Section 3. discusses the M-estimator approach to robust statistics. Section 4. shows how to combine M-estimation with clustering objective functions.

This paper assembles twenty years of my work into a coherent whole. Numerical experiments in abundance are not to be found since they have been done

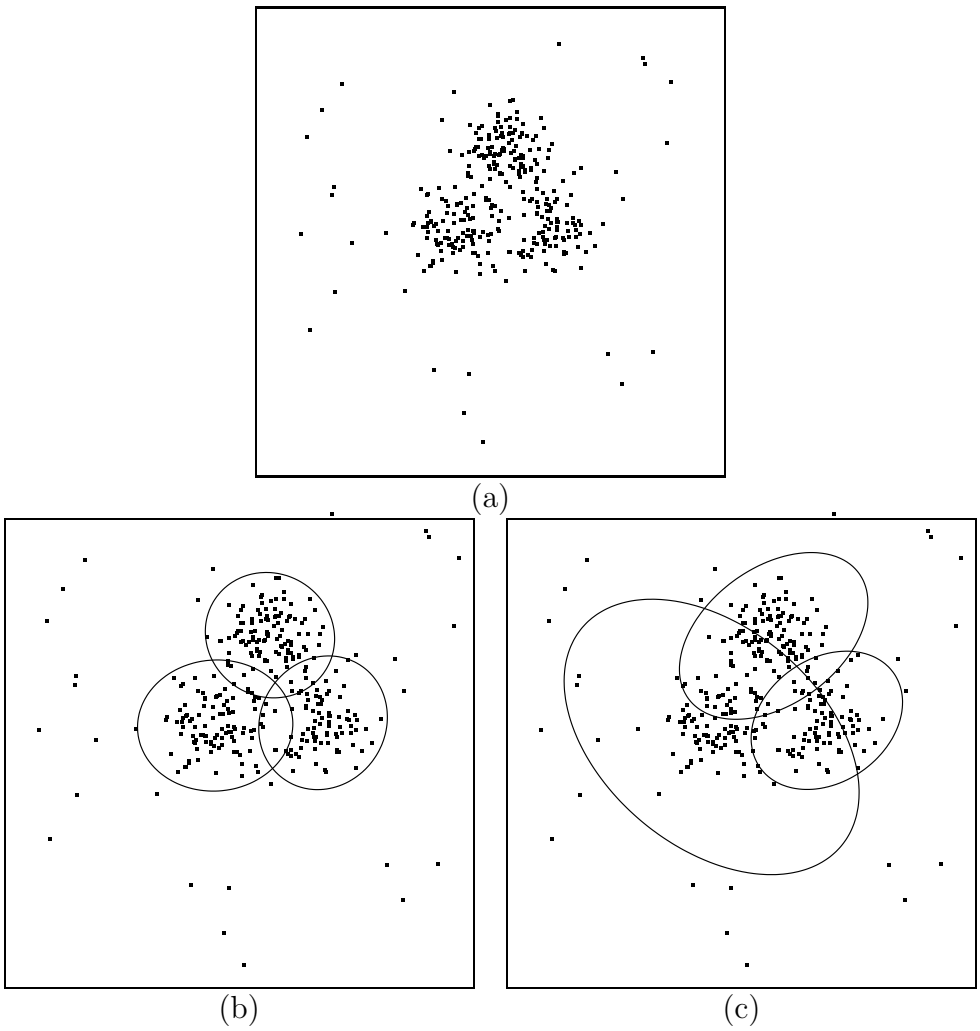

Figure 1: Cluster analysis with outliers.

in special cases elsewhere. The purpose of the paper is to provide a mathematical foundation for robust objective function clustering. In the final section I will discuss some interesting threads that somehow keep arising in objective function based data analysis.

2. Objective Function Clustering Methods

2.1 Objective functions for describing data

Objective function clustering methods are usually adaptations to the clustering problem of objective function methods for describing the structure of data sets as a whole. The methods begin with a choice of a type of data descriptor and a measure of the incompatibility of a data point with a given descriptor. We will focus on the situation where the objects under study are described by data vectors x in a euclidean space R^d. In this context, incompatibility is measured by assigning to a data point and a descriptor a number so that the more incompatible the two are the larger the number is.

Perhaps, the simplest descriptor of a data set is a vector m in R^d that is to be in some sense the "center" of the data, that is, describing its *location*. A simple measure of incompatibility between a data point x and a data center m is $|x-m|^2$ the square of the euclidean distance between them. The further x is from m, the less compatible they are with each other. A best choice for a center of a data set $X = \{x_1, \ldots, x_n\}$ can be obtained by minimizing the total incompatibility given simply by

$$\sum_{x \in X} |x - m|^2,$$

which leads, of course, to using $m = \bar{x} = \sum x/n$

A next step is to add to the description of location a symmetric, positive

definite matrix, S, that describes the *shape* of the data set in terms of ellipsoids determined by the squared distance $(x-m)^T S^{-1}(x-m)$. A "natural" choice for total incompatibility is $J(m,S) = \sum_x (x-m)^T S^{-1}(x-m)$, which is minimized as a function of m by $\bar{x}$, but has no minimum as a function of S. On the other hand, a reasonable choice for the S is the data covariance matrix $\hat{S} = \sum (x-\bar{x})(x-\bar{x})^T/n$, which can be obtained as a minimizer of modifications of J. These modifications are obtained as follows.

If S is a symmetric matrix, then there is an orthonormal basis for R^d of eigenvectors of S, so that $S = B\Lambda B^T$, where B is the orthogonal matrix whose columns are the basis vectors and Λ is a diagonal matrix with corresponding eigenvalues of S along the main diagonal. For a function $g : R \to R$ we extend g to a function g_M taking symmetric matrices to symmetric matrices by $g_M(L) = \text{diag}(g(l_1), \ldots, g(l_d))$ for a diagonal matrix L, and for any symmetric matrix, S, $g_M(S) = Bg_M(\Lambda)B^T$. Using this notion and $\text{tr}(A)$ to denote the trace of a matrix A, it was shown in Windham (2000) that if g is concave and increasing, G satisfies $g(s') \leq G(s)(s'-s) + g(s)$ for all s and s', $t = (m,S)$, and

$$r(x,t) = (x-m)^T G_M(S)(x-m) + \text{tr}(g_M(S) - G_M(S)S), \tag{1}$$

then $L(t) = \sum_x r(X,t)$ is minimized by $(\bar{x}, \hat{S})$, the sample mean and covariance matrix. Moreover, if $g(0) \geq 0$ then $r(x,t) \geq 0$ for all x, m, and positive semidefinite S. The function G will typically be the derivative of g.

For example, for $g(s) = \log(s)$, then $G(s) = 1/s$ and $r(x,t) = (x-m)^T S^{-1}(x-m) + \log\det S +$ a constant. This example does not satisfy the condition that $g(0) = 0$, but $g(s) = \log(1+s)$ does and yields $r(x,t) = (x-m)^T(I+S)^{-1}(x-m) + \text{tr}(\log(I+S) - (I+S)^{-1}S)$, which has been used in many of the examples in this paper. The functions h in Table 1 also satisfy the necessary criteria, so that

$r(x,t)$ is non-negative.

Therefore, for a population descriptor t, a measure of incompatibility, $r(x,t)$, between a data point x and t, and a data set X, for our purposes the "best" population descriptor is the value of t that minimizes the objective function

$$L(t) = \sum_{x \in X} r(x,t).$$

2.2 Adapting objective functions to clustering

An objective function can be constructed for cluster analysis by introducing another variable that describes in some sense membership in clusters. It is assumed that there are k clusters or subpopulations represented by the data and that k is fixed.

The three famous objective function clustering procedures are partitioning methods, mixture analysis, and fuzzy clustering.

- **Partitioning:** For $C_1, \ldots, C_k$ a partition of the data into clusters and descriptors $t_1, \ldots, t_k$, one for each cluster,

  $$\sum_i \sum_{x \in C_i} r(x,t_i)$$

  is the total incompatibility between the data and the given cluster structure, so find the $C_1, \ldots, C_k$ and $t_1, \ldots, t_k$ that minimize it.

- **Mixture analysis:** This approach assumes that the data are a sample from a population whose probability density is a finite mixture of densities of the form $e^{-r(x,t_i)}$ and the structure is determined by finding t and $p = (p_1, \ldots, p_k)$ to maximize

  $$\sum_x \log \left(\sum_i p_i e^{-r(x,t_i)} \right)$$

where p_i is the probability of belonging to the i-th subpopulation. In this case, t and p are maximum likelihood estimators.

- **Fuzzy clustering:** This method is based on the notion that some concepts that appear to define sets really do not. For example, who is in the "set of tall people"? There is no clear cutoff in height that would universally be accepted as the dividing line between tall and not tall. A fuzzy set is a function that assigns to each object a number between zero and one that measures the degree to which the object has the property that the set represents. A fuzzy partition of R^d is a vector of k such functions defined on R^d, $u = (u_1, \ldots, u_k)$, with u_i the membership function for the i-th cluster and restricted so that $\sum u_i(x) = 1$ for each x. A fuzzy clustering is obtained by finding u and t to minimize

$$\sum_x \sum_i u_i^m(x)\ r(x, t_i)$$

where $m > 1$ is a fixed parameter used to adjust "fuzziness" of the clusters. The larger m is the fuzzier the clusters will be and the closer m is to one, the closer the results are to the partitioning results. Bezdek and Pal (1991) survey the literature on the subject.

These methods can be described in a manner that somewhat unifies the objective function philosophy. Define a cluster structure, a, by $a = \{a_1, \ldots, a_k\}$, where a_i is a function that assigns to each point x in R^d a number $0 \leq a_i(x) \leq 1$ with $\sum_i a_i(x) = 1$.

The variable a is incorporated into the objective functions as follows.

- **Partitioning methods:** Choose a and t to minimize

$$L(a,t) = \sum_i \sum_x a_i(x) r(x,t_i). \tag{2}$$

- **Mixture analysis:** Choose a, t,and p to minimize

$$L(a,t,p) = \sum_i \sum_x a_i(x)(r(x,t_i) - \log p_i + \log a_i(x)). \tag{3}$$

  where $\sum_i p_i = 1$.

- **Fuzzy clustering:** Choose a and t to minimize

$$L(a,t) = \sum_i \sum_x a_i^m(x) r(x,t_i). \tag{4}$$

The solution is obtained in all cases by an alternating optimization that at least decreases the objective function and converges, though the solution produces usually only a local minimum and in rare circumstances a saddle point. Details of the properties of the algorithm and the equivalence to the earlier formulation of the objective functions can be found in Windham (1987) and Bezdek, et al. (1987). In general, the procedure iterates from current values of the variables, a^c, t^c and for mixtures p^c to the next iterate, a^+, t^+, p^+. Choose an initial value for a.

Repeat the following until the difference between successive parameter values, is sufficiently small.

1. Choose t^+ so that for each i, t_i^+ minimizes

$$\sum_x a_i^c(x) r(x,t_i)$$

   This step will be discussed in more detail shortly.

   For mixtures: Choose p^+ by $p_i^+ = \sum_x a_i^c(x)/n$

2. Choose a^+

   - For partitioning:

$$a_i^+(x) = \begin{cases} 1, & \text{if } r(x, t_i^+) \leq r(x, t_j^+) \text{ for all } j \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

     The function a_i identifies a cluster $C_i = \{x : a_i(x) = 1\}$, that contains the data least incompatible with t_i.

   - For mixture analysis:

$$a_i^+(x) = \frac{p_i^+ e^{-r(x,t_i^+)}}{\sum_j p_j^+ e^{-r(x,t_j^+)}} \tag{6}$$

     The value $a_i(x)$ is the probability of belonging to the i-th subpopulation, knowing x.

   - For fuzzy clustering:

$$a_i^+(x) = \frac{r(x, t_i^+)^{\frac{1}{1-m}}}{\sum_j r(x, t_j^+)^{\frac{1}{1-m}}} \tag{7}$$

     The value $a_i(x)$ is the degree of membership of x in the i-th fuzzy cluster.

Step 1 requires that $\sum_x a_i(x) r(x, t_i)$ be minimized in t_i. How this is done, naturally depends on r. But, the objective function is just a weighted version a non-clustering objective function, so the minimization might be straightforward. For example, if $r(x, t)$ is the function described in (1), then the minimizing values m and S are

$$\begin{aligned} \hat{m}_i &= \sum_x a_i(x) x / \sum_x a_i(x) \\ \hat{S}_i &= \sum_x a_i(x)(x - \hat{m})(x - \hat{m})^T / \sum_x a_i(x) \end{aligned}$$

Step 2 is also an optimization step, in that, a^+ is precisely the value of a that minimize the objective function in a for a fixed $t = t^+$.

The fact that mixture analysis is based on a probability density function model warms the hearts of statisticians everywhere. In fact, all three clustering methods can be viewed as based on density function models.

Substituting the values of a that minimize the objective function for a fixed t leads to equivalent objective functions in only t

Partitioning: $\sum_x \min_i(r(x,t_i))$

Mixture: $-\sum_x \log \sum_i p_i e^{-r(x,t_i)}$

Fuzzy: $\sum_x (\sum_i r(x,t_i)^{1/1-m})^{1-m}$

And doing a little arithmetic in the partitioning and fuzzy case, leads to the fact that minimizing the usual objective function is equivalent to maximizing the appropriate one of the following "log-likelihood" functions.

Partitioning: $\sum_x \log \max_i e^{-r(x,t_i)}$

Mixture: $\sum_x \log \sum_i p_i e^{-r(x,t_i)}$

Fuzzy: $\sum_x \log e^{-(\sum_i r(x,t_i)^{1/1-m})^{1-m}}$

So, all three methods can be viewed as based on density functions $\max_i e^{-r(x,t_i)}$ for partitioning, $\sum_i p_i e^{-r(x,t_i)}$ for mixtures, and $e^{-(\sum_i r(x,t_i)^{1/1-m})^{1-m}}$ for fuzzy clustering. This observation, if nothing else, allows one to see how similar these three methods are to each other. The similarity is illustrated in Figure 2, which shows each of the three density functions for a simple one-dimensional, two cluster situation.

The next step will be to consider the problem of making the estimation process more robust.

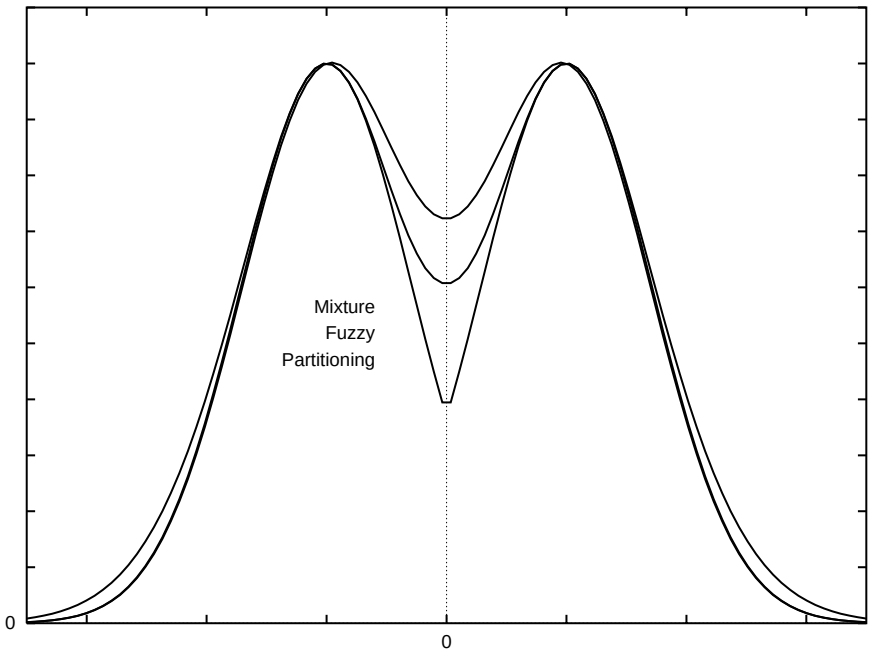

Figure 2: Density functions for clustering methods

3. Robust M-Estimation

The solutions to optimizing objective functions are called M-estimators in the context of robust statistical analysis. For the moment we will forget the clustering problem and return to estimating parameters to minimize objective functions $L(t) = \sum_x r(x,t)$. Hampel, et al. (1986) and Huber (1981) discuss M-estimators in detail and introduce a variety of procedures for estimating location and shape.

As it happens many of the most popular robust M-estimators can be obtained in a systematic way from familiar objective functions. The procedure is to replace $r(x,t)$ in the objective function with $h(\lambda\ r(x,t))/\lambda$, where h is a concave, increasing function and λ is a positive constant.

The function h will be called a *robustizer*. The concavity of h reduces the

impact of larger values of incompatibility relative to smaller ones. The effect is illustrated in Figure 3 which shows the robustizers for the Huber, Cauchy, and Welsch robustizers. Table 1 shows the robustizer, $h.$ for some of the most common robust estimation procedures. For those with two possible values, the change occurs at $r = 1$.

The constant λ is a "tuning" parameter. Larger values of λ increase the effect of the robustizer, since smaller values of $r(x,t)$ are magnified. On the other hand, when h is differentiable at zero, $\rho(x,t)$ approaches $h'(0)r(x,t)$ as λ goes to zero, so that less robustizing occurs for λ's close to zero. How λ should be chosen is a problem, in that the same value will have different effect, depending on the scale of the data. Table II of Holland and Welsch (1977) gives values used for achieving 95% efficiency in estimating location under a standard normal model. The constant $\lambda = 1/c^2$, where c is the value in the table. These choices give, at least, a ball park value to use and are listed in Table 1.

There is a circumstance in which it is possible to compensate for λ in a reasonable way. If $r(x,m,S) = (x-m)^T G_M(S)(x-m) + \mathrm{tr}(g_M(S) - G_M(S)S)$ as in (1), then in effect one is assuming that the clusters are essentially shaped like ellipsoids. Therefore, a reasonable model for the situation would be that data are from a normal distribution $\phi(x,\mu,\Sigma)$ and that the parameters μ and Σ are the real descriptors of data. If one uses the Welsch robustizer, and minimizes the theoretical analog of the objective function for the model $\int_{R^d}(1-e^{-\lambda r(x,m,S)})\phi(x,\mu,\Sigma)\,dx$ to obtain m^* and S^* then the following hold.

$$\begin{aligned} \mu &= m^* \\ \Sigma &= S^*(I - 2\lambda G(S^*)S^*)^{-1} \end{aligned} \tag{8}$$

Therefore, one can adjust the solutions to compensate for the value of λ used with

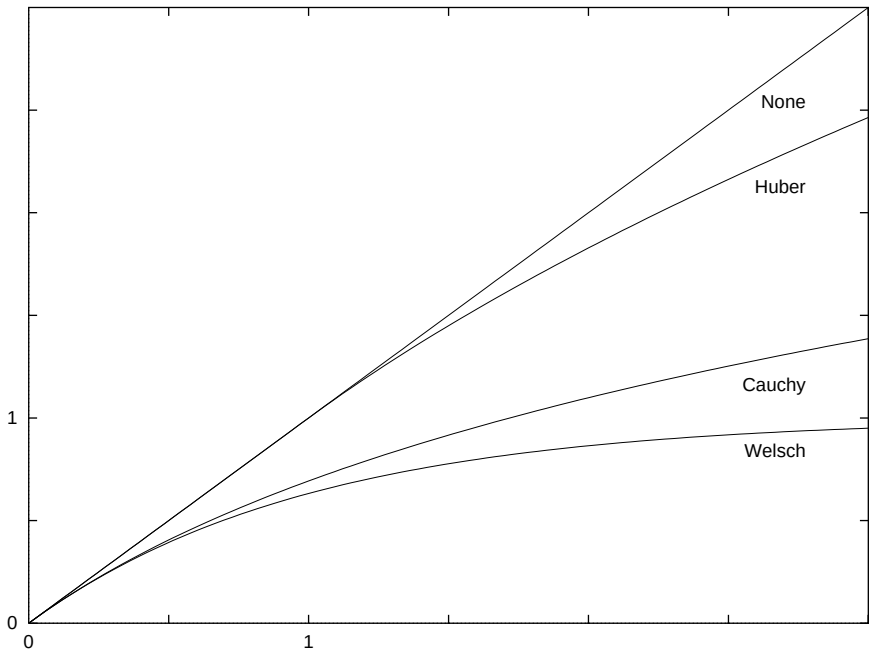

Figure 3: Huber, Cauchy, and Welsch robustizers.

data by using these formulae with $\bar{x}$ for m^* and $\hat{S}$ for S^*. A similar approach was used successfully in Windham (1995).

Estimates for t are obtained from the robustized objective function with an iterative procedure that at least decreases the objective function as the iterations proceed. The procedure uses a function H which satisfies $h(r) \leq H(r_0)(r - r_0) + h(r_0)$ for all r and r_0. This function exists since h is concave and is usually the derivative of h, see Table 1.

1. Choose an initial t^c to be the value of t that minimizes $\sum_x r(x,t)$.

2. Repeat until the change in successive iterates is sufficiently small:

   Find t^+ to minimize

$$\sum_x H(\lambda\ r(x,t^c))r(x,t)$$

	$h(r)$	$H(r)$	λ
None	r	1	0
Median	$2r^{\frac{1}{2}}$	$r^{-\frac{1}{2}}$	1
Huber	$\begin{cases} r \\ 2r^{\frac{1}{2}}-1 \end{cases}$	$\begin{cases} 1 \\ r^{-\frac{1}{2}} \end{cases}$	0.553
Biweight	$\begin{cases} \frac{1}{3}(1-(1-r)^3) \\ \frac{1}{3} \end{cases}$	$\begin{cases} (1-r)^2 \\ 0 \end{cases}$	0.046
Cauchy for $\beta \geq 0$	$\begin{cases} \frac{1}{1-\beta}((1+r)^{1-\beta}-1), \beta \neq 1 \\ \log(1+r), \quad \beta = 1 \end{cases}$	$(1+r)^{-\beta}$	0.176
Fair	$2(r^{\frac{1}{2}}-\log(1+r^{\frac{1}{2}}))$	$(1+r^{\frac{1}{2}})^{-1}$	0.510
Logistic	$2\log(\cosh(r^{\frac{1}{2}}))$	$r^{-\frac{1}{2}}\tanh(r^{\frac{1}{2}})$	0.689
Talwar	$\begin{cases} r \\ 1 \end{cases}$	$\begin{cases} 1 \\ 0 \end{cases}$	0.128
Welsch	$1-e^{-r}$	e^{-r}	0.112
Andrews	$\begin{cases} \frac{2}{\pi^2}(1-\cos(\pi r^{\frac{1}{2}})) \\ \frac{4}{\pi^2} \end{cases}$	$\begin{cases} \sin(\pi r^{\frac{1}{2}})/(\pi r^{\frac{1}{2}}) \\ 0 \end{cases}$	0.558

Table 1: Robustizers

That this procedure reduces the objective function follows from $h(r^+) \leq H(r^c)(r^+ - r^c) + h(r^c)$ and the fact that the minimization ensures that $H(r^c)r^+ \leq H(r^c)r^c$. Therefore, $h(r^+) \leq h(r^c)$.

It is also apparent that the minimization problem to be solved at each iteration again is a weighted version of the unrobustized problem, so it is likely to be easily solved, as was the case with the extension to clustering objective functions. The role function H in forming the weight also provides insight into how the robustizing works. The weighting characteristics are clearly indicated in Figure 4.

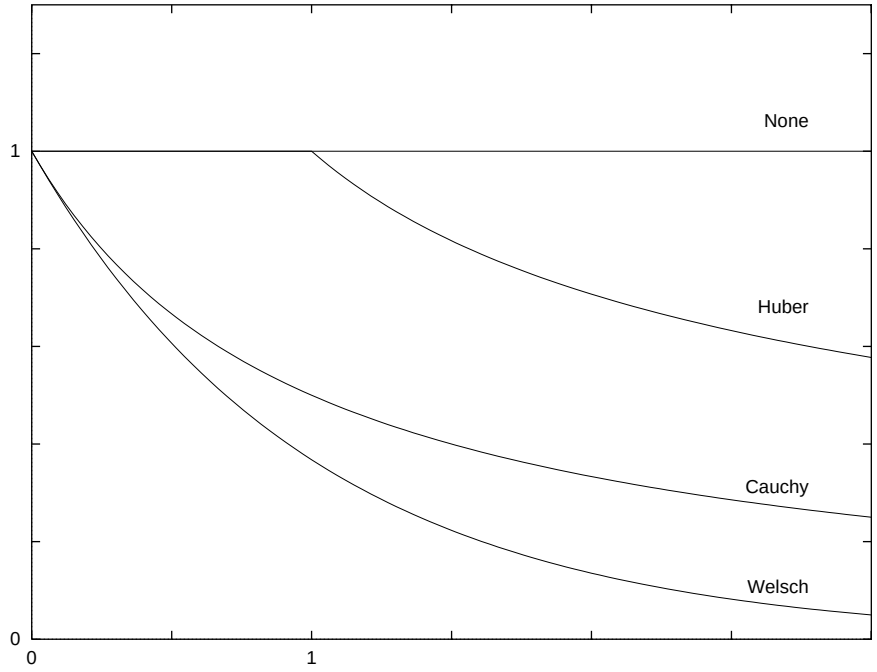

Figure 4: Weighting function in robust estimation

4. Robustizing Objective Function Clustering

Robustizing a clustering objective function is now quite straightforward. Merely replace $r(x, t_i)$ with $\rho(x, t_i) = h(\lambda r(x, t_i))/\lambda$ in (2), (3) and (4).

The minimization procedure is obtained by incorporating one iteration of the robustizing procedure into the clustering procedure to obtain

Robust Objective Function Clustering Algorithm

1. Choose an initial a^c. Choose t^c so that t_i^c minimizes $\sum_x a_i^c(x) r(x, t_i)$.

2. Repeat the following until the difference between successive parameter values, is sufficiently small.

(a) Choose t^+ so that for each i, t_i^+ minimizes

$$\sum_x a_i^c(x) H(\lambda r(x, t_i^c)) r(x, t_i)$$

For mixtures: Choose p^+ by $p_i^+ = \sum_x a_i^c(x)/n$

(b) Choose a^+ as in (5), (6) or (7) with ρ in place of r.

The procedure is an alternating optimization algorithm in the sense described in Windham (1987), so each iteration decreases the objective function.

The ellipses in Figure 1.(b) were obtained by procedure using the Cauchy metric, $g(s) = \log(1 + s)$, and the Welsch robustizer, with $\lambda = 0.112$.

When the Welsch robustizer is used and $r(x, t)$ is of the form (1), the procedure can be modified to compensate for λ. In Step 2(a) t_i^+ is adjusted as in (8) for each i before proceeding to 2(b). Then, in 2(b) use $r(x, t)$ instead of $\rho(x, t)$ in obtaining a^+. The latter adjustment compensates for the shrinking caused by λ in the membership functions. With these adjustments the procedure is no longer and alternating optimization and so the convergence properties are not clear. It has been my experience that the modified procedure behaves well provided λ is not so large as to make the adjusted shape matrix negative semidefinite. A similar, but more specialized approach is described in Windham (1996).

5. Conclusion

This papers assembles several pieces of work into a coherent whole. The whole itself has two pieces, the robust objective functions and the algorithm for optimizing them.

The common thread in robust and clustering objective function methodology is the appearance of concavity and its effect on successfully numerically solving the

problem. The latter is based on the fact that for a concave, increasing function h, its derivative h, r^c is any value of r, and r^+ is any value of r satisfying $H(r^c)r^+ \leq H(r^c)r^c$.

$$h(r^+) \leq H(r^c)(r^+ - r^c) + h(r^c) \leq h(r^c).$$

This inequality changes the problem of minimizing $h(r)$ to minimizing $H(r^c)r$, a weighted version of the problem without the added concavity.

The concavity appears in three places.

1. Metric functions, $(x-m)^T G(S)(x-m) + \mathrm{tr}(g(S) - G(S)S) = \mathrm{tr}(G(S)((x-m)(x-m)^T - S) + g(S))$, which ensures that $S = \hat{S}$ minimizes $\sum_x (x - m^T G(S)(x-m) + \mathrm{tr}(g(S) - G(S)S) = \mathrm{tr}(G(S)(\hat{S} - S) + g(S))$

2. Robustizers h for M-estimation.

3. Cluster analysis objective functions are obtained by putting $r(x,t)$ or $\rho(x,t)$ into

   - for partitioning, $h(r) = \min_i(r_i)$
   - for mixtures, $h(r) = -\log(\sum_i p_i e^{-r_i})$
   - for fuzzy clustering, $h(r) = (\sum_i r_i^{1/(1-m)})^{(1-m)}$.

   which are concave functions, increasing in each variable and $H(r)$ is nothing more than corresponding a^+, namely, $H_i(r) = \partial h / \partial r_i$ is given by

   - for partitioning, $H_i(r) = \begin{cases} 1, & \text{if } r_i \leq r_j \text{ for all } j \\ 0, & \text{otherwise} \end{cases}$
   - for mixtures. $H_i(r) = p_i e^{-r_i} / \sum_j p_j e^{-r_j}$
   - for fuzzy. $H_i(r) = r_i^{\frac{1}{1-m}} / \sum_j r_j^{\frac{1}{1-m}}$.

Many methods for modeling the real world using data are based on goodness-of-fit objective functions. People have produced algorithms that appear to find solutions, but the seemingly good behavior of the procedure is not always understood at first. In some cases, the behavior has been understood at a later date, and in some cases, not yet. Or perhaps for a reweighting procedure, they can be understood, by asking if the weight is the built from the derivative of some concave function in the background. If so, the the procedure is an alternating optimization algorithm that decreases the objective function and most likely will be well-behaved.

References

BEZDEK, J.C., HATHAWAY,R.J., HOWARD, R.E., WILSON, C.A. and WINDHAM, M.P. (1987), "Local convergence analysis of a grouped variable version of coordinate descent," *Journal of Optimization Theory and Applications, 54,* 471–477.

BEZDEK, J.C. and PAL, S.K. (1991), *Fuzzy Models for Pattern Recognition,* New York: IEEE Press.

HAMPEL, F.R., RONCHETTI, E.M., ROUSSEEUW, P.J., and STAHEL, W.A. (1986), *Robust Statistics: the Approach Based on Influence Functions,* New York: Wiley.

HOLLAND, P.W. and WELSCH, R.E. (Robust regression using iteratively reweighted least-squares), "Communications in Statistics Theory and Methods, A6(9)," *813-827,*

.HUBER, P.J. (1981), *Robust Statistics,* New York: Wiley.

WINDHAM, M.P. (1987), "Parameter modification for clustering criteria," *Journal of Classification, 4,* 191–214.

WINDHAM, M.P. (1995), "Robustifying model fitting," *Journal of the Royal Statistical Society, B, 57, No. 3,* 599–609.

WINDHAM, M.P. (1996), "Robustizing mixture analysis using model weighting," *From Data to Knowledge,* Eds. W. Gaul and D. Pfeifer.Heidelberg: Springer116–123

WINDHAM, M.P. (2000), "Robust clustering," *Data Analysis: Scientific Modeling and Practical Application,* Eds. W. Gaul, O. Opitz, and M. Schader, Berlin: Springer, 385–392.