METHODS FOR DATA DIRECTED MICROAGGREGATION IN ONE OR MORE DIMENSIONS

Gordon Sande
Sande & Associates, Inc.
600 Sanderling Court
Secaucus, NJ 07094

Abstract: Microaggregation is a technique used for the protection of the confdentiality of respondents in microdata releases. It is typically used for economic data where respondent identifability is quite high. Rather than releasing a perturbed version of the data, microaggregation releases the averages of small groups in which no single respondent is dominant.

The original form of microaggregation was for univariate data. It was implemented by sorting the data and then reporting the averages of adjacent groups of fx ed size. Any partial group at the end would be pooled with the fnal complete group to ensure that the desired minimum group size was obtained. The typical group size was small, with fv e a common choice. An immediate improvement would be to allow some number of internal groups, perhaps near the center of the data, to be larger to compensate for the incomplete group.

As a further improvement the groups can be allowed to have varying sizes so that no group will include a large gap in the sorted data. Each of the resulting groups can be more homogeneous when the group boundaries are allowed to be sensitive to the distribution of the data. This can be described as a clustering problem with a variable number of clusters and a minimum cluster size. The number of clusters is chosen to be as large as possible consistent with homogeneous clusters and the minimum cluster size.

Techniques for determining such data directed microaggregations have been proposed which use randomized searching methods. These methods are typically terminated early as they are quite expensive to operate. They seek to minimize the total within cluster sum of squares as suggested by some clustering methods. They have two disadvantages of not leading to readily solved optimization problems and of not being the most suitable criterion for highly skewed data typical of economic applications.

For highly skewed data the width of the clusters may be a more suitable measure. The total within cluster width may be obtained by summing the gaps between adjacent members of the clusters. Cluster size may be controlled by requiring a minimum number of adjacent gaps be included in any cluster. The result is an optimization problem for a linear objective function over the indicator variables for the gap inclusions. Each data point and its potential cluster neighbors would appear in a constraint which enforces the minimum cluster size. The resulting system can be readily solved.

For bivariate, or higher dimensional, data a the notion of adjacency is defned even though sorting is no longer well defned. The size of a cluster can be measured by the length of its minimal spanning tree. The problem of fnding groups of size exactly two is the well known perfect matching problem. One form of clustering is minimal spanning tree partitioning which resembles the univariate method above. Suitable constraints for minimum cluster size, which are more elaborate than in the univariate case, can be constructed and the resulting systems solved. For larger problems, or higher dimensions, we may choose to use only a Delaunay triangulation rather than all adjacencies.

Keywords: Statistical Disclosure Control, Statistical Confdentiality, Microdata Release, Microaggregation

1. Introduction

The demand for public use samples of the fles collected by statistical agencies is very strong. For demographic data, this demand has been met by standard practices for some time. These practices do not carry over to fles of establishments. A specialized technique has been developed to address the needs for establishment data for secondary analysis. The released data is the average of a small number of similar records. This release technique is called microaggregation [4], [1].

The number of records in the groups to be averaged is as small as the disclosure requirements will permit. Often this means fv e records in a released group average. We will use fv e as our fx ed example for simplicity although other values are possible. The original microaggregation proposal was for a single data variable. The single variable would be sorted and fv e adjacent records would be assigned to a group. If the fle for release had one thousand records, there would be two hundred groups of size fv e and the released public use fle would have two hundred records. For other sizes there could be a partial group left over. In the initial proposal, this partial group would be combined with the last complete group so that the fnal group would have a size of fv e to nine members. A modifcation of this is to have some number of groups of size six and for the enlarged groups to be internal groups rather than the fnal groups [3]. To deal with higher dimensional data the technique of dimensional reduction was used so that the original proposals could be used. The dimensional reduction was a projection, often that suggested by a principal component analysis. More direct techniques are possible, although more complex technically. The diffculty is that sorting is not well defned in two or more dimensions although the notion of adjacent can be effectively defned.

The microaggregation technique is typically applied to establishment data. Like most economic data, it is highly skewed. When the effects of microaggregation on secondary analysis are examined, an immediate question is the effect of the technique on the distribution of the data. One measure of the effect is the spread within the groups. For skewed data, the fnal groups will have the highest internal spread. Having the fnal group be of varying size will further increase its variability so the modifcation of varying internal group size is very natural. For skewed data, variance may not be the preferred measure of within group spread. Often we would prefer to use the range of the group. Variance is associated with the Gaussian distribution. Skewed distributions are more often like the exponential distribution. The Laplace, or double exponential, distribution is a symmetric distribution with the same long tails as the exponential distribution. The Laplace distribution leads to medians and mean absolute deviations in the same way that the Gaussian distribution leads to means and variances. We will measure within group spread by the group range for one dimensional data. For higher dimensional data we would use a measure of cluster size. We shall fnd that the length of the minimal spanning tree is a convenient measure.

The next modifcation of the microaggregation technique is to deliberately have some groups be larger than fv e, or even six, in order to reduce the total within group spread. The count of groups might decrease but allow for lower total within group spread. When there is a large gap in the data, we would like it to be between groups rather than within some group, if this is consistent with our overall objectives. This has been suggested. Methods to achieve it have been proposed and experiments have been done to demonstrate that it is a sensible suggestion. The proposed methods, based on genetic algorithm minimization of total within group variance, require much computer time to achieve their results [2]. We will demonstrate

direct methods requiring smaller amounts of computer time. The corresponding techniques in higher dimensions lead to problems well known to be computationally diffcult. Our frst interest will be in whether the higher dimensional results are useful. Only if they prove to be useful would it be worth pursuing the question of how to reduce their computational cost. The problems may also be of independent interest to those studying algorithm complexity issues.

When we seek groups that have small within group spread there are elementary observations that are obviously true for univariate data. Any two groups will not be interleaved. If they were interleaved, then groups with smaller spreads could be obtained by exchanging members to remove the interleaving. A group of size greater than or equal to ten can be broken into two groups so that we will only observe groups with sizes from fv e to less than ten. These observations are problematic in higher dimensions. One might defne two groups to be disjoint if their convex hulls do not overlap but it is easy to construct examples where this defnition is not compatible with keeping the group size small.

2. Clustering Approach

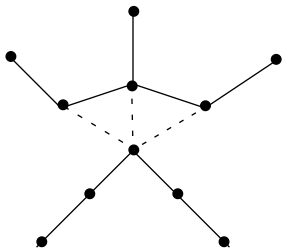
Viewing forming microaggregations as a clustering problem is very natural. However, it does not readily lead to an optimal solution although it provides useful insights. As an approximation technique it is quite useful. For one dimension we may readily construct a cluster tree in which the sorted data values are the external leaves of the tree. The frst cluster would be of the two data items with the smallest gap between them. The node joining these two data points would be labeled with the midpoint of the gap separating them, which is also their average in this simple case. The following clusters would be of the two data points, the data point and the cluster or the two clusters with the smallest gap between them. In each case the new node would be labeled with the midpoint of the gap. Eventually there will be a single large cluster containing all the data points. This is a bottom up procedure. A top down procedure would start with the sorted data and the gaps to form two clusters by using the midpoint of the largest gap to separate the data into two subclusters. We would repeat this within each cluster until all the clusters are of size one. However we construct it, the cluster tree will represent the data. When we seek to form the microaggregations we will discover a difference between clustering and microaggregation. We may fnd a cluster of an adequate size to form two micro aggregations but the subclusters violate the microaggregation size requirements. One of the subclusters may be too small while the other is of an acceptable, if slightly large, size. This problem occurs when the largest gap in the cluster is too close to one of the cluster ends and we must reorganize the internal structure of the cluster to match the microaggregation requirements. The top down procedures can be redefned to ignore large gaps which are too close to the end points of the current cluster. With this redefnition we will have a procedure which avoids large gaps and permits variable size microaggregates. We will later see that it is an effective approximate procedure.

The top down procedure could be organized to follow the cluster groupings in the form of a recursive partitioning process or to follow the gap sizes in the form of a greedy algorithm. Recursive partitioning is the basis of the common quicksort algorithm where an internal value is used to separate the current partition into smaller and larger values by moving the entries. For microaggregation, the initial data would already be sorted so that the gaps can be easily determined. The largest gap within a partition can be identifed and the subpartitions determined with no need for any data moving. Gaps near the endpoints of a partition can be ignored so the partitions are always suitable

for forming microaggregations. The process of fnding the largest gap in the current partition is essentially that of quicksort if it were to be used to sort the gaps. There is no data moving involved as we are trying to determine which gaps are boundaries between the microaggregation groups. If the gaps are sorted before starting, a processing sequence of the partitions can be based on using the largest gap not yet processed. Here, the next gap to be processed can be anywhere in the data rather than just within the active partition of the recursive procedure. If a gap is too close to a partition endpoint it is ignored. This sequence of processing of the partitions follows the structure of a greedy algorithm. The recursive partitioning algorithm would examine the unsorted gaps repeatedly while the greedy algorithm would examine the sorted gaps only once.

We may also learn about the limitations of the clustering notions by examining alternate data sets that are equivalent under the clustering procedures. All of the clustering decisions are based on the comparison of gaps. A new set of gaps, with a reference data point, will defne a new data set. For example, we could defne the new gaps to be 1 plus a small positive multiple of the given gaps with a reference point of 1 as the smallest value. This new data set would be the integers with a small perturbation but with unchanged gap comparisons. For small perturbations the microaggregation groups would all be of size fv e. However the clustering based groupings could be quite different.

For two or more dimensions the sorting based procedure can not be applied. The ability to judge adjacency by sorting and examining gaps is lost. In one dimension there are two adjacent points, except at the ends. In more dimensions there are many adjacent points and even the number of adjacent points may vary considerably. To form clusters in a bottom up fashion we would join the two points which are closest, as we did in one dimension. We would then join two points, a point and a cluster or two clusters as our next step. If they were already in the same cluster then we would not join them again. This is a complication that does not arise in one dimension. This process would be repeated until all the points have been joined into a single cluster. This process is the well known Kruskal algorithm for fnd - ing the minimal spanning tree. There is a corresponding top down procedure for fnding the minimal spanning tree in which we repeatedly remove the longest connection. We do not remove connections which would lead to two components. This alternate algorithm requires more steps than does the Kruskal algorithm so is rarely used or even described. Both require a sorted list of all the connections as input data. Tarjan[5] provides a generalized proof for greedy algorithms that covers both these algorithms and many other variants. Partitioning the minimal spanning tree is one of many clustering algorithms. It may not satisfy the size requirements of the microaggregation problem as we have seen in the one dimensional case. Rather we would use the top down procedure with the requirement that the removal of a connection should not create a subcluster which is too small to be a microaggregate. We would start this procedure with all the connections as, in general, the fnal set of connections is not contained in the minimal spanning tree. This would usually be more data than we would like so we would choose a triangulation which includes the minimal spanning tree, such as the Delaunay triangulation. For two dimensional data the total number of connections in the Delaunay triangulation is a small multiple of the number of data points. A Delaunay triangulation in two dimensions can be determined at a low cost comparable to sorting the data on one of its coordinates, which is the frst step in the standard algorithms. Some of the fnal groupings may be larger than twice the microaggregation size. There will be a single central point with several groups, each too small to be a microaggregate, surrounding it. The number of



Large group postprocessed into 2 groups

surrounding groups is limited by the geometry of crowding, so that the regular hexagon is a boundary case in two dimensions. An illustration of this is would be fv e pairs of points around a central point. After a postprocessing stage this would become two microaggregates, one of fv e points consisting of two pairs and the central point and the other of six points consisting of three pairs which have been joined. A more awkward example would be three groups of size three around a central point. The postprocessing can be done with the optimization technique discussed below.

3. Optimization Approach

To treat the microaggregation problem as an optimization problem we will need both an objective function, to choose between various sets of microaggregations, and a model, with constraints, to defne possible sets of microaggregations. We can model the problem by having two data points be in the same microaggregation if the connection between them is chosen as shown by a value of 1 for its indicator function. The objective function would be the total length of all selected connections. In one dimension this is relatively simple as each point has two connections, except the two end points with one connection. For groups of size fv e or greater we would require that at least four consecutive connections be made before a connection could be absent. Such a condition would be $x_{i+1} + x_{i+2} + x_{i+3} + x_{i+4} + x_{i+5} \geq 4$. The number of such conditions is limited by the number of data points. We would need minor modifcations to this at the end points. If we avoid technical issues such as connections of zero or the same length, we will have solutions in which the indicator variables assume values of zero or one when they are only assumed to be continuous on the interval from zero or one. The optimal solutions can be obtained with linear programming with reasonable cost.

For two or more dimensions the optimization problem is more diffcult to solve. The connection structure of the groups is more complex than is the structure in one dimension. The structure has many similarities to the structure of the traveling salesman problem, which has been used to develop many techniques in operations research. A comparable development is beyond the scope of the current work. Finding a solution for smaller problems will allow us to judge the quality of the approximate method that was developed above. The extreme case of exactly one group may be expressed as an optimization problem. This is the optimization formulation for the minimal spanning tree. This formulation requires that the total number of connections selected be the number of links in the spanning tree, which is one less than the number of data points, and that there be no loops in the selected connections. This last condition must be true for all subsets of the data. For n data points, there are 2n subsets. This formulation is impractical to use if all the conditions must be expressed before starting the computation. Very few of the conditions are required in any particular example so an initial trial solution is found. Conditions are added to eliminate any loops which are found. The newly added conditions may allow some of the earlier conditions to be dropped. The process is repeated until the trial solution is free of loops. A different example would require a different set of conditions. However in practice this iterative formulation is not used because the minimal spanning tree problem has very effcient solutions which directly use its special structure.

We are seeking multiple spanning trees for disjoint subsets of the data. We could use discrete optimization with our conditions of a lower limit on the size of the subsets and the conditions that the spanning trees have no loops. Discrete optimization is typically slow as it often is based on very generalized methods used to guide an underlying continuous optimization. Much of discrete methods research is directed at exploiting the properties of the problem under study to guide a continuous optimization method to fnd the discrete solution. The frst diffculty we notice about the microaggregation problem is that we do know how many spanning trees we are trying to construct. If we knew this we could ask what happens as we modify our objective function to successively merge the group spanning trees, perhaps with some reorganization, until we arrive at the minimal spanning tree for all the data. When we try to apply continuous optimization to fnd the spanning trees of many groups we encounter fractional values. This is not surprising as the same phenomena arises in the traveling salesman problem and is addressed by the so called comb inequalities. The fractional values are not an issue for the full minimal spanning tree problem. We observe that if we decrease the number of groups, by increasing the numbers of connections that are to be selected, we will have no fractional values at some point even though we have only used conditions to ensure no loops and minimal group size. We will obtain some number of unmerged groups and of merged groupings. The merged groupings defne smaller subproblems that can be addressed by the same methods. The calculation of the inequality systems for specifc small examples suggests that this reduction will always work although general proofs are not available.

We would like to have stronger conditions which will allow us to fnd both more and smaller subproblems at each stage of out processing. The smallest example of fractional values would be three points joined with connections of weight 1/2. The most elementary condition is that every point should be connected to some other point. The equation for this would be that the sum of all connections to a point should sum to 1 or more. Two connections of weight 1/2 satisfes this condition. The condition that the three points should not form a loop requires that the sum of the three indicators should be 2 or less, which is met in this case by the fractional weights. We want the internal connections to sum to 2, unless there are also connections from three points to other points. If the value we choose for the limit is 2 then we are permitting a group of size three. In fact we do not want such small groups so the test value must be 3 to keep the group size up. We can add up all the connections from the three points, being careful to avoid using the connections between the points twice. The condition of avoiding the double counting of the internal connections makes these conditions stronger than just adding all the connections to the three points. (In practice we will have variables representing the sum of all connections to a point so we can sum these and subtract the double counted connections to construct more compact equations for the optimization software.) We would certainly apply this condition to any isolated group of size three that was observed. We could also search for triples of points which violate the condition. We have constructed a new set of conditions directed at removing fractional values from the continuous optimization, or a cut in the operations research usage. This cut has two uses of either extending a group which is too small or of helping eliminate fractional values. Such a cut could be used to extend an isolated group of size two. It could also be used for groups of size four which could either be extended or help have fractional values eliminated. It could also be applied to larger groups except it would no longer have its test value increased to extend the group size above the minimum group size.

A working search procedure would be to apply several sets of conditions until a new fraction free group has been identifed and the set of conditions is not changing. The conditions would be those for no loops, for no small isolated groups and for no fractional values in cuts of size two and three. If a new group has been found we separate it out and start over on the smaller problem. If the condition set stops changing with no new group found then we would increase the order of the cut being used. There are many higher order cuts which we would prefer not to have to use. This reserves the additional power of the higher order cuts for the smaller problems which can be isolated with the lower order cuts.

4. Reference Approach

Microaggregation was originally defned by sorting and grouping. In one dimension the defnition is both pragmatic and effective. We have provided two enhanced methods; one an approximate or heuristic method and the other an exact method. The extension of the original defnition to two or more dimensions is somewhat problematic.

The problem of fnding groups of size two, or exact matching, is a very well studied problem in operations research. The data is the distance between points or the cost of a connection in some graph. The optimal exact matching problem is now a classical problem which was used to develop many methods and has been subject to many improvements with the best algorithms being very effcient, low order polynomial, but somewhat elaborate. The extension to groups of size three is mostly notable for its discovery that the problem is qualitatively harder. It is called X3C (Exact 3 Cover) in the list of well known *NP Complete* problems. The extensions to larger groups will not lead to easier problems. The operations research methods will tend to avoid long connections as the infuence of any connection extends to all matching through the objective function.

To follow the style of the one dimensional sorting method, we would like a method which is based on comparisons without the global balancing of the numerical procedures. In the sorting method we take an extreme point and collect the points which are closest to it into a group and repeat until all points have been assigned to a group. The extreme points would be on the convex hull of the data points. In two dimensions the chosen extreme point could be defned by the point on the convex hull which subtends the most acute angle along the convex hull. For higher dimensions we would use solid angle or its extensions. The points to be grouped with the extreme point could be its nearest neighbors. This is a procedure which is based on comparisons and effected by the local points only. It can be readily implemented as convex hull and nearest neighbor computations can be implemented at low cost. The procedure will tend to *squeeze around* empty regions rather than just reach across them as is required in one dimension. It may also leave isolated points so the groups may not well separated.

5. Examples

We have three sets of procedures that can be applied to data. The one dimensional procedures have corresponding procedures for two or more dimensions. A simple indicative example serves to illustrate the differences within the one dimensional procedures. The example data are 1000 values from a random number generator supplied with a Fortran compiler with its default starting value. The idealized version of this data would have each of the 1000 values centered in its own equal sized panel for all gaps of size 0.001. For a group size of fv e, the total width of all groups would be 0.8. The reference technique produces results much as the idealized data would indicate. The approximate technique has been successful in avoiding the larger gaps with a reduced group count. The optimal technique has improved the grouping with a slightly

decreased total width and a slightly increased group count. The observed data are:

	Count	Total Width
Reference	200	0.79
Approximate	164	0.63
Optimal	170	0.62

The methods for two or more dimensions are analogs of the one dimensional procedures. The methods are illustrated for two dimensions and readily extended to more dimensions. The test data in 1000 points distributed uniformly in the unit square. The data are displayed in Figure 1 below. (1000 data points in a small display may exceed the reproduction process capabilities used for this note. Multiple generation copies are unlikely to be successful.) If the points were placed on a uniform grid and connected the total connection length would be 1000 * (1/) = 31.62 when the approximation is made that the grid fts exactly. A minimal spanning tree of the data can be constructed and has a length of 20.55. The minimal spanning tree is shown in Figure 2 below. The poor approximation illustrates the extent to which it is possible with the extra freedom to move in two dimensions to fnd paths around the gaps in the data. A natural comparison value for microaggregations would be 4/5 * 20.55 = 16.44. The reference groups are given in Figure 3 below, the approximate groups in Figure 5 below and the optimal groups in Figure 6 below. A hybrid process to use the approximation methods to fnd gross groupings with the optimization methods used for the fnal details was also tried. The gross groupings were approximate microaggregations of minimal size 25. Each of these gross groups were then reduced to multiple optimal microaggregations of minimal size fv e. The results for this hybrid calculation were 163 groups with a total length of 16.03 as shown in Figure 4 below. We see that the approximate, hybrid and optimal solutions have sizes that are consistent with avoiding the longer connections in the minimal spanning tree as the groups are formed. The reference groups are larger than the minimal spanning tree approximation would suggest. The observed data are:

	Count	Total Length
0.8 * MST		16.44
Reference	200	18.28
Approximate	155	16.18
Hybrid	163	16.03
Optimal	168	15.90

6. Conclusion

The univariate microaggregation technique can be extended to allow for varying group size. This permits the groups to be chosen for greater within group homogeneity. An approximation algorithm, which is a modifcation of the usual quicksort algorithm, will produce data dependent microaggregations at a cost comparable to sorting the data. The quality of the grouping found is less than that obtained by use of optimization techniques. The difference in quality between the approximation and the optimization result is pleasantly small. The optimization based solution is not diffcult to achieve but may be awkward for some organizations. The robustness of the observation on the quality of the approximation should be tested by more extensive experimentation with both artifcial and real data.

The univariate methods have natural extensions to two or more dimensions. The underlying notion of adjacency is both simple and natural in one dimension and easily implemented by sorting. For two or more dimensions the notion of adjacency is natural but simplicity and ease of implementation are lost as sorting is not well defned. Standard techniques from computational geometry can be adapted to the microaggregation problem. An approximation method, which is a modifca - tion of a minimal spanning tree algorithm, is quite effective. Unfortunately it requires two stages of processing as it can generate over sized groups. The two stages can be used to advantage to obtain a better approximation by using the optimization based second stage to process small local problems. The approxima-

tion method is effective but provides less quality in grouping than does the optimization based method. The optimization problem is not one of the standard problems which has been addressed by computational geometry. Further elaboration of the techniques for fnding the optimal microaggregates both in the complete problem and in the post processing phase for the oversize groups from the approximate method is indicated. The robustness of the observation on the quality of the approximation should be tested by more extensive experimentation with both artifcial and real data.

The use of computational geometry techniques to determine variable size data directed microaggregations is useful. The univariate techniques can be easily implemented. The techniques for two dimensions are more diff - cult to implement with the approximate technique readily implemented but requiring a second phase of the more diffcult optimization technique. The optimization technique is suitable for small groups but is not, without further development, for direct use on larger fles. The extension to higher dimensions pose no additional diffculties beyond those of triangulation in higher dimensions.

7. References

[1] Defays, D., Nanopoulos, P., (1993), "Panels of enterprises and confdentiality: the small aggregates method", in Proceedings of 92 Symposium on Design and Analysis of Longitudinal Surveys, Ottawa: Statistics Canada, 195-204.

[2] Mateo-Sanz, J. M., Domingo-Ferrer, J. (1999), "A Method for Data-Oriented Multivariate Microaggregation", Proceedings of Statistical Data Protection '98, Luxembourg, Offce for Offcial Publications of the European Communities, pp. 89-99.

[3] Sande, G. (1996), "Putting Blurred Data in an SOI Context", lecture notes, Confdentiality Workshop for Internal Revenue Service / Statistics Of Income, Washington, DC.

[4] Strudler, M., Oh, H. L. and Scheuren, F. (1986), "Protection of Taxpayer Confdentiality with Respect to the Tax Model," Proceedings of the Section on Survey Research Methods, American Statistical Association, Alexandria, VA, pp. 375-381.

[5] Tarjan, R. E. (1983), "Data Structures and Network Algorithms," CBMS-NSF Regional Conference Series in Applied Math, 44, SIAM, Philadelphia.

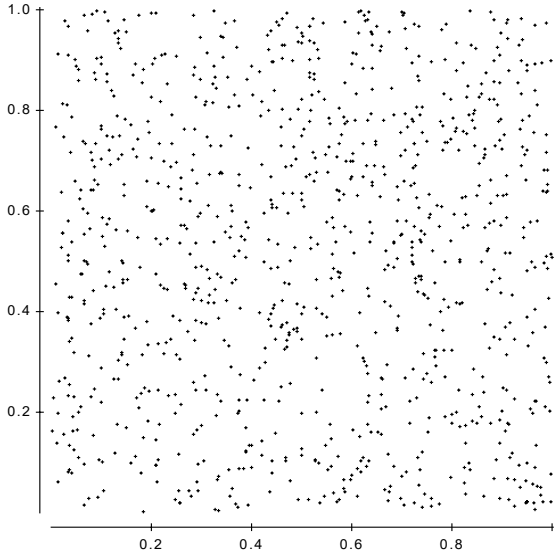


Figure 1: Data Points

1000 points

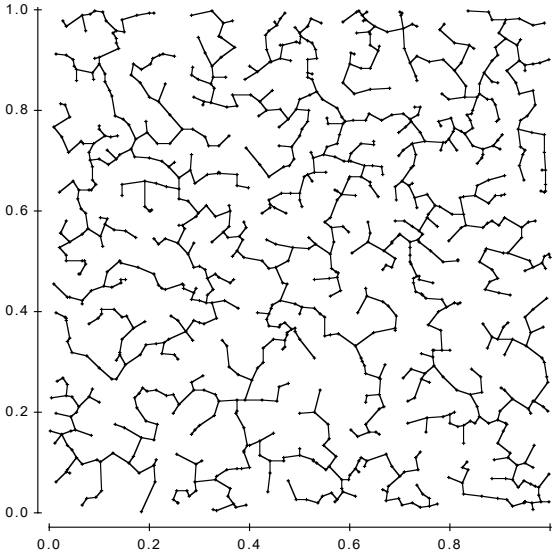


Figure 2: Minimal Spanning Tree

20.55 length

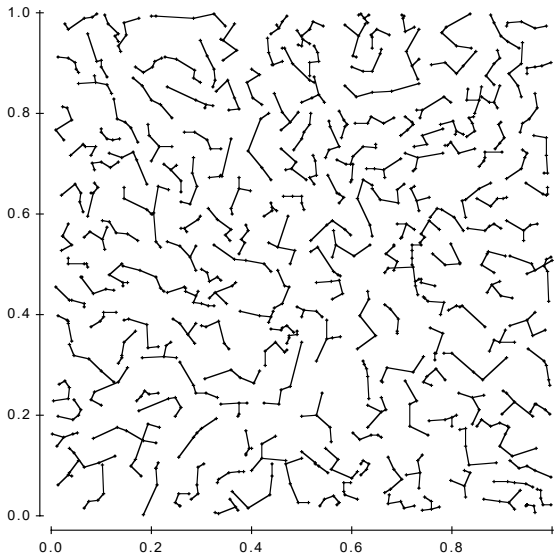


Figure 3: Reference Microaggregates

200 groups - 18.28 length

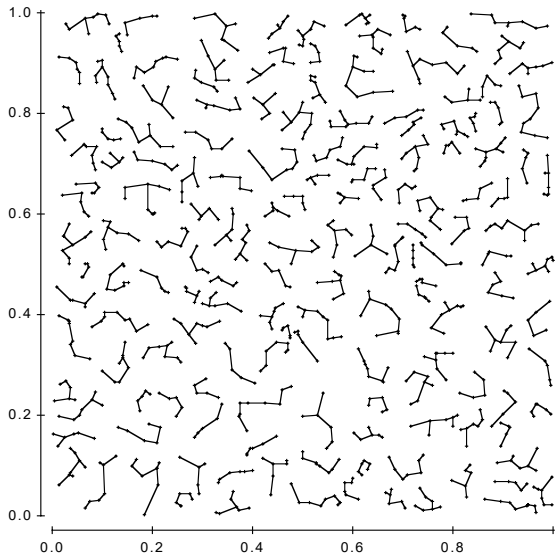


Figure 4: Approximate Microaggregates

155 groups - 16.18 length

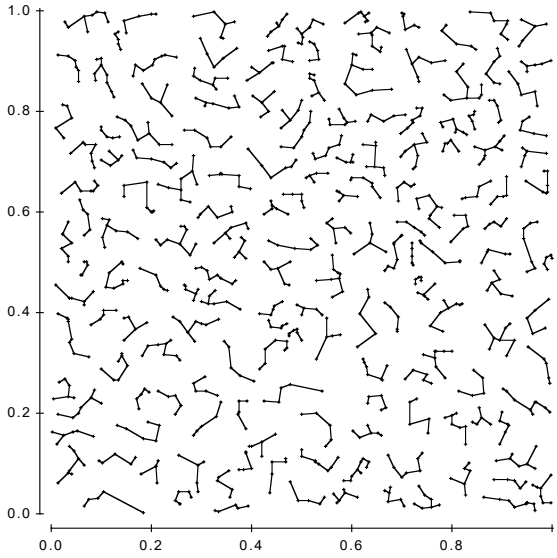


Figure 5: Hybrid Microaggregates

163 groups - 16.03 length

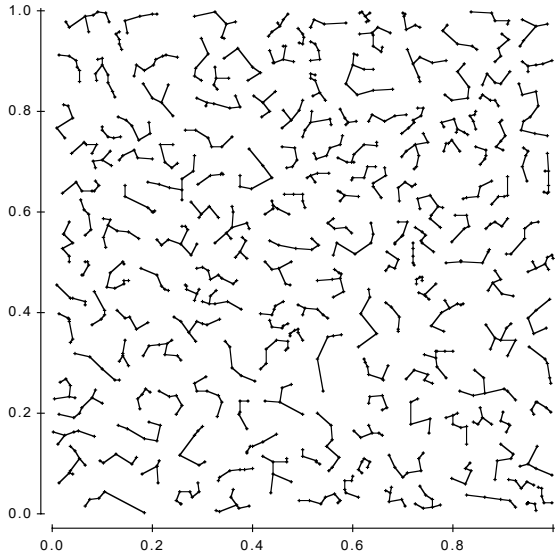


Figure 6: Optimal Microaggregates

168 groups - 15.90 length