Clustering Smoothed Dissimilarities in Tertiary Data

Bridget Manning and David B. Hitchcock

Department of Statistics

University of South Carolina

October 11, 2022

Abstract

Cluster analysis of categorical data classifies data objects into homogeneous groups based on the categorical outcomes recorded on each object. However, such methods may not produce clustering partitions that accurately reflect the underlying process from which the data has been generated, especially in cases of noisy observations and high variability within the latent variables underlying the measurement process. In this paper a shrinkage-based statistical smoother is used in dissimilarity-based cluster analysis to combat these problems, specifically for tertiary data objects. Smoothing the dissimilarities in a shrinkage-based manner produces cluster partitions that more accurately reflect the underlying process from which the tertiary data has arisen by shrinking the dissimilarities toward model-based estimates supported by the data set as a whole. The results, shown via simulation and a real diabetes data application, indicate that the implementation of such statistical smoothing produces cluster partitions more reflective of the underlying structure of noisy tertiary data than traditionally used clustering methods.

Keywords: Cluster analysis, Shrinkage-based Smoothing, Pima Indian Diabetes Data, Adjusted Rand Index, Contingency Tables

1. Introduction

Suppose there exists a set of grade-school students for whom performance in subjects like Mathematics, Reading, and History is recorded as being "below", "meeting," or "exceeding" the expectations set by a local school district. Suppose further that the interest is to determine whether there exist groups of students such that students in the same group are similar to each other, based only on placement of students into these three categories. In this paper we introduce a method of creating such groups by combining a shrinkage-based statistical estimator from categorical data analysis with cluster analysis. We show the combination of the two methodologies results in more accurate groupings than standard clustering algorithms under certain conditions.

Cluster analysis is a method of separating a set of images, patterns, or data objects into homogeneous groups such that objects placed in the same group share some property that makes them more similar to each other than they are to any other objects within different groups. Pertinent to many clustering algorithms is a mathematically defined measure of dissimilarity (or, alternatively, similarity). How this dissimilarity is defined depends on the type of variables recorded on each observation.

When variables are completely quantitative, dissimilarity is usually based on a distance measure, e.g., the Euclidean or Manhattan distance between observations, such that a smaller distance between observations implies the observations are more similar, or other distance metrics found in Everitt et al. (2011) or Friedman et al. (2017). On the other hand, when variable measurements are completely qualitative, the notion of distance is not as natural, and thus dissimilarity may be defined by looking at the proportion of attribute measurements upon which objects disagree. Other methods of defining similarity and dissimilarity for qualitative data can be found in Everitt et al. (2011) or Boriah et al. (2008). It is also possible for variable types to be mixed, including both qualitative and quantitative variables in the same dataset,

in which clustering algorithms like K-Prototypes (Huang, 1997) may be used.

Let us consider the case where all variables are qualitative. The simplest case of this is when each attribute measurement is binary, taking one of two outcomes. The clustering of binary data has been studied and used extensively in recent years (see e.g., Cornell et al. (2009); Dolnicar and Leisch (2004); and Hitchcock and Chen (2008)). In this paper, however, we focus on tertiary data. We propose a dissimilarity-based method of creating clusters when the attribute measurements are tertiary: qualitative with 3 classes. For example, a variable may measure level of autism with the responses being "requiring support", "requiring substantial support", or "requiring very substantial support." Though clustering algorithms exist within the field of cluster analysis for use on such data, many of the algorithms for qualitative attribute measurements tend to neglect variability and noise, which may affect the accuracy of clustering solutions produced.

Consider the aforementioned example of clustering a set of grade-school children based on their performance in various subjects. Merely observing the category (say, "below expectations") a student is in gives no indication how close the student was to meeting the standard. Similarly, one has no idea how close a student who met the standard was to exceeding those standards in the subject. Furthermore, when working with qualitative data, it is possible for some information to be obscured within the categories, such as variability of the latent variables underlying the structure of the data. Situations of high latent variability may complicate the clustering task. Therefore, we propose to use clustering methods that compensate for the imperfections that may plague qualitative data, e.g., by the use of statistical smoothing.

Statistical smoothing is a technique used commonly to help find a signal or uncover the true structure of the data that may be buried by noise. We propose smoothing via shrinkage, which allows us to smooth the observed data towards a particular model, should such a model be supported by the data. Our approach guards against misspecification errors that may occur when assuming a particular model (see e.g., Agresti (2012) or Simonoff (1998)) and allows us to supplement the information in a pair of observations with that of the entire data set (Hitchcock and Chen, 2008). We propose a dissimilarity-based method for the clustering of tertiary

Table 1: Cell probabilities for a pair of objects $\mathbf{Y}_k$ and $\mathbf{Y}_{k'}$.

	$\mathbf{Y}_{k'}$		
$\mathbf{Y}_k$	1	2	3
1	π_{11}	π_{12}	π_{13}
2	π_{21}	π_{22}	π_{23}
3	π_{31}	π_{32}	π_{33}

The true probability objects $\mathbf{Y}_k$ and $\mathbf{Y}_{k'}$ both have a particular variable outcome of category 1 is denoted as π_{11}; the true probability object $\mathbf{Y}_k$ has a variable outcome in category 1 while object $\mathbf{Y}_{k'}$ has a variable outcome in category 3 is denoted as π_{13}, and so forth.

observations that uses a shrinkage-based smoother to combat variability and underlying noise that may exist in the data. The ideas presented are an extension of the work of Hitchcock and Chen (2008), who showed that pre-smoothing dissimilarities helped improve partitioning accuracy in the case of binary data that had a noisy underlying structure.

The outline of the paper is as follows: In Section 2, we provide some background information relevant to our method, formally define pairwise dissimilarities for a set of tertiary data objects, and introduce a clustering algorithm based on a smoothed version of the dissimilarity matrix. Section 3 describes a simulation study illustrating the effect of the proposed method of smoothing dissimilarities on the accuracy of cluster partitions. In Section 5, we apply the proposed algorithm to the Pima Indian Diabetes dataset (National Institute of Diabetes and Digestive and Kidney Diseases (1990)) to assess the method's performance on a real data application . We conclude the paper in Section 6 with a brief discussion of the methodology and its possible ramifications.

2. Method

In this section we discuss in detail our proposed method of pre-smoothing tertiary dissimilarities as a preliminary step to clustering. When cross-classifying tertiary data objects, data on each pair of observations can be summarized within a 3×3 contingency table as shown in Table 1 where the entries within the table, $\{\pi_{kk'}\}$, denote the true cell probabilities. Table 1 can be thought of as probabilities of a multinomial random variable, $\mathbf{X} = (X_1, X_2, \ldots, X_n)$, where $\boldsymbol{\pi} = (\pi_{11}, \pi_{12}, \pi_{13}, \pi_{21}, \pi_{22}, \pi_{23}, \pi_{31}, \pi_{32}, \pi_{33})$ represents the true cell probabilities, with the con-

ditions $0 \leq \pi_{ij} \leq 1, i = 1, 2, 3, j = 1, 2, 3$ and $\sum_i \sum_j \pi_{ij} = 1$. Let $\hat{\pi}_{ij}, i = 1, 2, 3, j = 1, 2, 3$, denote an estimate for the true cell probability π_{ij}. When the goal is to simultaneously estimate multiple cell probabilities, (in the 3×3 case, we estimate 9, of which 8 are free parameters), Fienberg and Holland (1973) presented a shrinkage estimator $\boldsymbol{\pi}^*$, which we will use to smooth our dissimilarity matrix, $\mathbf{D}$.

In statistics, smoothing is used to detect the underlying signal or latent structure that may be hidden by noisy data with a smoothed estimator in the multinomial setting in many cases defined as shown in equation (1)

$$\pi^*_{ij} = (1 - \lambda)\hat{\pi}_{ij} + \lambda(\widetilde{\pi}_{ij}). \tag{1}$$

(see e.g., Albert (1987) or Hitchcock and Chen (2008)). In equation (1), $\hat{\pi}_{ij}$ represents the observed cell proportions, $\widetilde{\pi}_{ij}$ represents the estimated cell probabilities under an assumed model, and λ denotes the degree of smoothing. The $\boldsymbol{\pi}^*$ used in this paper will be a "data-dependent" smoothed estimator; Simonoff (1995) discussed other methods of smoothing categorical data that could be used as alternatives to this approach. For small λ, more emphasis would be placed on the observed cell probabilities $\hat{\pi}_{ij}$ and for larger values, more emphasis places on the model-based cell probability estimates $\widetilde{\pi}_{ij}$. Though π^*_{ij} is a biased estimator of π_{ij}, it may be more robust than $\hat{\pi}_{ij}$ under sparse multinomial tables, and it allows us to use information from neighboring cells to garner better estimates for cell probabilities (see Simonoff (1995) or Simonoff (1998)). This method of smoothing is closely related to Stein estimation (see Efron and Morris (1977)).

2.1 *Dissimilarities for a tertiary data set*

Dissimilarities for tertiary data observations can be calculated as the proportion of mismatches among P tertiary attribute measurements for each pair of observations. Consider Table 2 which depicts the cross-tabulation of two multivariate objects, $\mathbf{Y}_k$ and $\mathbf{Y}_{k'}$, from which we define *pairwise dissimilarity*. In this table, each attribute has an outcome of 1, 2,

Table 2: Summary of matches and mismatches for a pair of objects $\mathbf{Y}_k$ and $\mathbf{Y}_{k'}$.

	$\mathbf{Y}_{k'}$			
$\mathbf{Y}_k$	1	2	3	Totals
1	a	b	c	$a+b+c$
2	d	e	f	$d+e+f$
3	g	h	i	$g+h+i$
Totals	$a+d+g$	$b+e+h$	$c+f+i$	P

In the table, a denotes the total number (among the P variables) of variables for which objects $\mathbf{Y}_k$ and $\mathbf{Y}_{k'}$ both have outcomes in category 1, b denotes the total number of variables upon which object $\mathbf{Y}_k$ has an outcome in category 1 and object $\mathbf{Y}_{k'}$ has an outcome in category 2, and so forth.

or 3 denoting its membership. Letters a, e, and i represent the total number of variables for which both objects have an outcome classified as 1, 2, and 3, respectively. The other letters denote the number of attributes on which objects $\mathbf{Y}_k$ and $\mathbf{Y}_{k'}$ have different values. For example, b denotes the number of variables for which object $\mathbf{Y}_k$ has a value of 1 and object $\mathbf{Y}_{k'}$ has a value of 2; similarly, c denotes the number of variables for which object $\mathbf{Y}_k$ has a value of 1 and $\mathbf{Y}_{k'}$ has a value of 3. Based on this 3×3 table, we define similarity for the kth and k'th object as $S_{kk'} = \frac{1}{P}(a+e+i)$ and dissimilarity as $D_{kk'} = 1 - S_{kk'}$. This particular method of defining dissimilarity is as an extension to the matching coefficient method often used in the binary case or analogous to the overlap method used in computer science with matches and mismatches weighted equally. Several alternatives to the matching coefficient method exists (see, e.g., Everitt et al. (2011) or Boriah et al. (2008)); however, regardless of the choice of definition, the smoothing procedure presented in Sections 2.2 and 2.3 still applies.

In several traditional methods of clustering, once the pairwise dissimilarities for each pair of observations have been calculated, they serve as the input to a clustering algorithm of choice. We propose, instead, pre-smoothing the dissimilarities before implementing the clustering process.

2.2 *Choice for Model-Based Estimators*

In this section we discuss possible choices for the model used to estimate $\boldsymbol{\pi}$ along with rational underlying each model. In our notation, we use i and j to denote the ith row and jth column of the 3×3 table formed for a particular pair of objects k and k'. However, this procedure would be repeated for each pair of objects.

Previously we discussed how counts in a 3×3 contingency table like Table 2 could be thought of as data arising from a multinomial distribution. In this context, we view the problem from the perspective of estimating the cell probabilities of the multinomial distribution. Let

$$\boldsymbol{\pi} = (\pi_{11}, \pi_{12}, \pi_{13}, \pi_{21}, \pi_{22}, \pi_{23}, \pi_{31}, \pi_{32}, \pi_{33})$$

denote the set of true probabilities for each cell in the 3×3 table shown in Table 1, and let

$$\widetilde{\boldsymbol{\pi}} = (\widetilde{\pi}_{11}, \widetilde{\pi}_{12}, \widetilde{\pi}_{13}, \widetilde{\pi}_{21}, \widetilde{\pi}_{22}, \widetilde{\pi}_{23}, \widetilde{\pi}_{31}, \widetilde{\pi}_{32}, \widetilde{\pi}_{33})$$

denote an estimate of the probabilities of observations falling in each cell of the 3×3 table under the assumed model.

If the researcher has no prior information about the relationship between a pair of observations then a uniform or non-informative model may be used. In this case, one possibility is an equal-probability model. Under this model, $\widetilde{\pi}_{ij} = \frac{1}{9}, i = 1, 2, 3, j = 1, 2, 3$. This suggests observations are just as likely to fall into any cell in the 3×3 contingency table. If, instead, the researcher has the belief that a pair of observations are independent in their tertiary variable measurements, then an independence model may be appropriate. In this setting, the rows and columns of Table 1 are independent; therefore, $\widetilde{\pi}_{ij} = \hat{\pi}_{i+}\hat{\pi}_{+j}, i = 1, 2, 3, j = 1, 2, 3$. Lastly, if the researcher feels that a pair of observations are more likely to match in a particular category and less likely to match on another, then a model of dependence could be chosen to reflect those prior beliefs. For example, $\widetilde{\pi}_{ij} = \begin{cases} \frac{1}{10}, & \text{for } i = j \\ \frac{1}{60} & \text{for } i \neq j \end{cases}$ would mean the pair of observations are more likely to have the same tertiary attribute value for a particular attribute than to

have any other combination of category values.

This method of choosing a model is similar to the Bayesian framework of choosing a prior distribution that reflects the researchers' prior belief about the structure of the data. Just as it is important to choose an appropriate prior in Bayesian inference, it is also important to choose a smoothing model carefully, being mindful that an appropriate model should be supported by the data. Once the appropriate smoothing model is chosen, then shrinkage-based smoothing is implemented.

2.3 *Shrinkage-type smoother for the* 3×3 *table*

We now discuss our proposed method of smoothing the dissimilarity matrix using the Fienberg-Holland (1973) estimator in the notation of Hitchcock and Chen (2008). We begin with a discussion of the estimator and then show how each of the cells in the 3×3 table of pairwise dissimilarities can be smoothed.

2.3.1. Fienberg-Holland Estimator

The Fienberg-Holland estimator (Fienberg and Holland, 1973) has been shown to be a better estimator of $\boldsymbol{\pi}$ than the multivariate sample mean $\hat{\boldsymbol{\pi}}$ in terms of minimizing the total mean squared error loss, and can be used to reflect prior information about the latent structure of the data. This is significant because when we perform clustering using the original observed dissimilarities, this is akin to using the cell proportions in $\hat{\boldsymbol{\pi}}$ to estimate the cell probabilities while neglecting knowledge about the latent structure of the data.

Consider placing a Dirichlet prior with mean vector $\boldsymbol{\gamma}$ on $\boldsymbol{\pi}$. The posterior mean associated with the $\{i, j\}$ cell probability is given by

$$(1 - \lambda)\hat{\pi}_{ij} + \lambda(\gamma_{ij}) \tag{2}$$

with

$$\lambda = \frac{\kappa}{P + \kappa},$$

where P denotes the number of attributes recorded on each observation. Fienberg and Holland (1973) denote the value that minimizes the expected squared error loss between $\boldsymbol{\pi}$ and the estimate given in equation (2) by κ. Thus,

$$\kappa = \frac{1 - \sum \pi_{ij}^2}{\sum (\gamma_{ij} - \pi_{ij})^2}.$$

Then a pseudo-Bayes estimator can be written as shown in equation (3)

$$\frac{P}{P+\kappa}(\hat{\pi}_{ij}) + \frac{\kappa}{P+\kappa}(\gamma_{ij}). \tag{3}$$

Model-based estimates of $\boldsymbol{\pi}$ can be used as is traditionally done in an empirical Bayesian approach (see Fienberg and Holland (1973)). Lastly, $\boldsymbol{\pi}$ and κ must also be estimated as their true values are not known. Therefore, their maximum likelihood estimators are used, and equation (3) can be rewritten with

$$\hat{\kappa} = \frac{1 - \sum \hat{\pi}_{ij}^2}{\sum (\widetilde{\pi}_{ij} - \hat{\pi}_{ij})^2}.$$

In the case of tertiary data, the Fienberg-Holland estimate of κ can be written specifically as

$$\hat{\kappa} = \frac{1 - (\hat{\pi}_{11}^2 + \hat{\pi}_{12}^2 + \cdots + \hat{\pi}_{33}^2)}{(\widetilde{\pi}_{11} - \hat{\pi}_{11})^2 + (\widetilde{\pi}_{12} - \hat{\pi}_{12})^2 + \cdots + (\widetilde{\pi}_{33} - \hat{\pi}_{33})^2}.$$

Therefore, the Fienberg-Holland estimate for π_{ij} is given by

$$\pi_{ij}^* = \frac{P}{P+\hat{\kappa}}(\hat{\pi}_{ij}) + \frac{\hat{\kappa}}{P+\hat{\kappa}}(\widetilde{\pi}_{ij}). \tag{4}$$

Equation (4) can be used to smooth each of the cells of the table for a particular pair of observations.

2.3.2. Smoothing the 3×3 *table*

To smooth the 3×3 table, we multiply the James-Stein-type estimators shown in equation (4) by P, as shown in Equation (5):

$$\{i,j\}^{(smooth)} = \widetilde{\pi}^{*}_{ij} P = [\frac{P}{P+\hat{\kappa}}(\hat{\pi}_{ij}) + \frac{\hat{\kappa}}{P+\hat{\kappa}}(\widetilde{\pi}_{ij})]P. \tag{5}$$

The multiplication by P rescales the estimate to reflect the expected number of attribute outcomes falling in each cell. We use these to obtain our inputs for the smoothed dissimilarity matrix.

For cell $\{i,j\}, i = 1,2,3, j = 1,2,3$, we use equation (5) to obtain smoothed cell counts that correspond to each cell location as shown in Table 1. Then

$$S^{smooth}_{kk'} = \frac{1}{P}(a^{smooth} + e^{smooth} + i^{smooth})$$

$$D^{smooth}_{kk'} = 1 - S^{smooth}_{kk'}$$

Once these smoothed dissimilarities are formed for each pair of observations,they are collected into a $n \times n$ smoothed dissimilarity matrix, $\mathbf{D_{smooth}}$, that is used as the input for the clustering algorithm of choice.

2.4 *Clustering algorithms used*

To investigate the situations in which smoothing via shrinkage may be useful for cluster analysis, we examine two different algorithms: Average Linkage and K-Medoids.

The Average Linkage algorithm (Sokal and Michener, 1958) is an agglomerative hierarchical method of clustering that merges observations based on the average pairwise dissimilarity or distance between cluster members. Since the output of hierarchical clustering can be shown with a dendrogram that can be cut to obtain the desired number of clusters, using such an algorithm eliminates the need to know the number of clusters a priori (see e.g., Friedman et al. (2017) or Albalate and Minker (2011)).

The K-Medoids algorithm (Kaufman and Rousseeuw, 1987) is a partitioning-based method of clustering, which partitions observations into groups based on the distance to a central, "most representative," cluster member (see Kaufman and Rousseeuw (1987)). This algorithm

Table 3: Cross-Tabulation of Two Partitions

	Partition One	
Partition Two	Same Group	Different Group
Same Group	A	B
Different Group	C	D

does not require Euclidean distance to be used as the dissimilarity measure, and has been shown to be a more robust method of clustering than K-Means clustering (see e.g., Friedman et al. (2017) or Albalate and Minker (2011)).

Each algorithm is implemented using R (R Core Team, 2019). The Average Linkage algorithm is implemented using the `hclust` function in the `stats` package, while the K-Medoids algorithm is implemented using the `pam` (Kaufman and Rousseeuw, 1987) function in the `cluster` package.

3. Simulations

In this section we discuss a simulation study undertaken to assess the performance of the proposed method of pre-smoothing tertiary dissimilarities using the Fienberg-Holland estimator. We assess the method's effect on the accuracy of clustering partitions produced, when using the Average Linkage and K-Medoids algorithms. Our simulations assume the data has arisen from a mixture of multinomial distributions. We measure this "accuracy" of the clustering solution in terms of the Adjusted Rand Index (ARI) as proposed by Hubert and Arabie (1985).

3.1 *Adjusted Rand Index*

Consider Table 3 (see, e.g., McNicholas (2017) for this type of table) which shows the cross-tabulation of the results of two different clustering methods. In this table, the columns refer to the partition created from one method and the rows denote the partition created by another method. Here, A denotes the number of pairs of objects that both partitioning methods put in the same groups; B the number of pairs of objects put in the same group by

the first method but placed in different groups by the second method; C the number of pairs of objects put in the same group by the second method but in different groups by the first method; and D the number of pairs of objects that both partitioning methods put in different groups. These values, then, can be used to assess cluster partition accuracy when one of the partitions is assumed to represent the ground-truth partition.

Using the notation of McNicholas (2017), the ARI can be computed as

$$ARI = \frac{N(A+D) - [(A+B)(A+C)+(C+D)(B+D)]}{N^2 - [(A+B)(A+C)+(C+D)(B+D)]}$$

where N denotes the total number of possible pairs of objects.

The ARI is a correction to the Rand Index (Rand, 1971) and can take values as large as 1, with higher values denoting more agreement between two clustering solutions and values closer to 0 denoting chance agreement. For our simulation study, one of the partitions will denote the true clustering structure and the other, the proposed clustering partition produced by a clustering algorithm. A method which results in higher ARI values is then considered to be more reflective of the true latent structure of the data and hence a better method of clustering the observations in the context of our simulation study.

3.2 *Simulation setup*

In this section we discuss simulated data generation and conclude with a discussion of the clustering scenarios and separation settings used to assess cluster goodness.

3.2.1. Data generation

In this simulation, we assume there are n data objects, each with P tertiary features recorded on them, that have arisen from C clusters. We further assume that each of these P measurements is independent of the other $P-1$ measurements (mutually independent). In this case, the tertiary observation, Y_{lkp}, refers to the specific categorical outcome of the kth object in cluster l on feature p where $l = 1, 2, \ldots, C, k = 1, 2, \ldots, n_l, \sum_{l=1}^{C} n_l = n, p =$

Table 4: Parameter Settings for Multinomial Simulations

	Separation Setting				
Vector	I	II	III	IV	V
$\boldsymbol{\tau}_1$	(0.40,0.30,0.30)	(0.50,0.25,0.25)	(0.60,0.20,0.20)	(0.70,0.15,0.15)	(0.80,0.10,0.10)
$\boldsymbol{\tau}_2$	(0.30,0.40,0.30)	(0.25,0.50,0.25)	(0.20,0.60,0.20)	(0.15,0.70,0.15)	(0.10,0.80,0.10)
$\boldsymbol{\tau}_3$	(0.30,0.30,0.40)	(0.25,0.25,0.50)	(0.20,0.20,0.60)	(0.15,0.15,0.70)	(0.10,0.10,0.80)

Note: Row 1 defines the parameter vectors used to generate the features for objects from cluster 1 in each separation setting, row two defines the parameter vectors used to generate the features for objects from cluster 2 in each separation setting, and row 3 gives the parameter vectors to generate the features for objects from cluster 3 in each separation setting.

$1, 2, \ldots, P$. Thus $Y_{lkp} \in \{1, 2, 3\}$. Lastly, the probability of an attribute measurement being in either category remains constant for each of the P measurements. Therefore, in this setting, $\mathbf{Y}_{lk} \sim Multi(P, \boldsymbol{\tau}_l)$ with $\boldsymbol{\tau}_l = (a_l, b_l, c_l), 0 \leq a_l \leq 1, 0 \leq b_l \leq 1, 0 \leq c_l \leq 1$. $\boldsymbol{\tau}_l$ denotes the parameter vector for the lth cluster, and $a_l, b_l,$ and c_l refer to the probability of obtaining an attribute measurement that falls in category 1, 2, or 3, respectively, for data objects in the lth cluster.

3.2.2. Parameter settings

For the simulations, we generate 5000 data sets each with a total of $n = 600$ objects each with $P = 10$ tertiary features. We assume these objects have arisen from $C = 3$ clusters with a varying number of observations from each cluster ($n_1 = 100, n_2 = 200, n_3 = 300$). We generate the kth tertiary object in cluster 1, 2, and 3, respectively, as follows: $\mathbf{Y}_{1k} \sim Multi(10, \boldsymbol{\tau}_1)$, $\mathbf{Y}_{2k} \sim Multi(10, \boldsymbol{\tau}_2)$, and $\mathbf{Y}_{3k} \sim Multi(10, \boldsymbol{\tau}_3)$. (Note we utilize the `sample` function in base `R` (R Core Team, 2019) to generate each sample.) The value of $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2,$ and $\boldsymbol{\tau}_3$ are specified as shown in Table 4.

The cluster separation settings denoted I, II, III, IV, and V, represent the distance between clusters. As we increase the settings, the discrepancy between clusters increases. Therefore, Setting I denotes the smallest distance between clusters and setting V denotes the largest. Table 4 shows the parameter vector $\boldsymbol{\tau}_l$ used for clusters $l = 1, 2, 3$ for each separation setting in our simulations. To better understand these settings, consider separation setting V. Under this

last setting, observing an object with several attributes with measures of "1" would suggest it is more likely that that particular object arose from sub-population 1 as opposed to either of the other two sub-populations. On the other hand, if this same outcome was observed in cluster separation setting I, it would be tougher to determine from which cluster it had arisen. Thus, the overlap between clusters decreases, in general, as the separation settings increase. This suggests the clustering problem gets easier as we increase from setting I to V.

After generating the tertiary data objects, each object is stored in a $n \times P$ data matrix as shown below where object m's measurements are stored in row m.

$$\boldsymbol{Y} = \begin{bmatrix} Y_{11} & Y_{12} & \dots & Y_{1P} \\ \vdots & \vdots & \vdots & \vdots \\ Y_{m1} & Y_{m2} & \dots & Y_{mP} \\ \vdots & \vdots & \vdots & \vdots \\ Y_{n1} & Y_{n2} & \dots & Y_{nP} \end{bmatrix}.$$

Once the tertiary objects are created, we perform clustering using the Average Linkage and K-Medoids algorithms with the observed dissimilarities and the dissimilarities smoothed under three smoothing models: independence, equal probability, and high probability of match. For the independence model, we set our smoothed cell estimates to $\widetilde{\pi}_{ij} = \hat{\pi}_{i+}\hat{\pi}_{+j}, i = 1, 2, 3, j = 1, 2, 3$. For the equal-probability model, we set $\widetilde{\pi}_{ij} = \frac{1}{9}, i = 1, 2, 3, j = 1, 2, 3$. Finally, for the high probability of match model, we set $\widetilde{\pi}_{ij} = \begin{cases} \frac{1}{10}, & \text{for } i = j \\ \frac{1}{60} & \text{for } i \neq j \end{cases}$. We then compare the outcomes of each method using the average ARI and show the results in Section 3.2.3.

3.2.3. Results

In this simulation, the Average Linkage algorithm produced the most accurate clustering partitions. This suggests that a hierarchical method of clustering may be a better method to use in this setting rather than a partitioning-based method like K-Medoids. Furthermore, in most cases smoothing under the assumption of independence produced more accurate clus-

Table 5: Table gives average ARI for the Average Linkage clustering of the simulated data, based on different smoothing methods. There are 100 observations from cluster one, 200 from cluster two, and 300 from cluster three.

Average Linkage Algorithm

I	II	III	IV	V
0.541 (o)	0.579 (o)	0.985 (o)	0.990 (o)	0.997 (o)
0.551 (s/i)	0.877 (s/i)	0.987 (s/i)	0.990 (s/i)	0.997 (s/i)
0.011 (s/E)	0.084 (s/E)	0.551 (s/E)	0.868 (s/E)	0.980 (s/E)
0.014 (s/H)	0.235 (s/H)	0.601 (s/H)	0.872 (s/H)	0.979 (s/H)

NOTE: Each value is the average (across 5000 data sets) ARI for the clustering produced from an Average Linkage algorithm based on (top within each cell) the observed dissimilarities (o); (second within cell) the smoothed dissimilarities based on the independence model (s/i); (third within cell) the smoothed dissimilarities based on the equal-probability model (s/E); (last within cell) the smoothed dissimilarities based on the high probability of match model (s/H).

tering solutions, as measured by the average ARI, compared to not smoothing, and always performed better than any other smoothing method considered. This suggests clustering via smoothing with an independence assumption may be the better method to use when clustering tertiary data arising from a multinomial setting.

Table 5 shows the specific accuracies obtained when the clustering solutions were produced using the Average Linkage algorithm. Since the average ARI values in Table 5 are averages of 5000 shift-adjusted proportions, we can say the Monte Carlo standard error associated with each entry in Table 5 is, at most, approximately 0.007. (The same upper bound on the Monte Carlo standard error can be stated for Table 6, which also contains averages of 5000 ARI values). In the Average Linkage case, the accuracy of the clustering solutions produced using the unsmoothed dissimilarities and for those resulting from pre-smoothing the dissimilarities under a model of independence are highest of all the smoothing methods in each separation setting. In the case of more variability in the latent structure of the data, separation settings I-III, the average ARI values produced utilizing the dissimilarities pre-smoothed under independence are notably higher than those obtained through the use of the unsmoothed dissimilarities. In separation settings IV and V, the accuracy of both methods is the same.

The opposite, however, is true when the dissimilarities are pre-smoothed towards a model

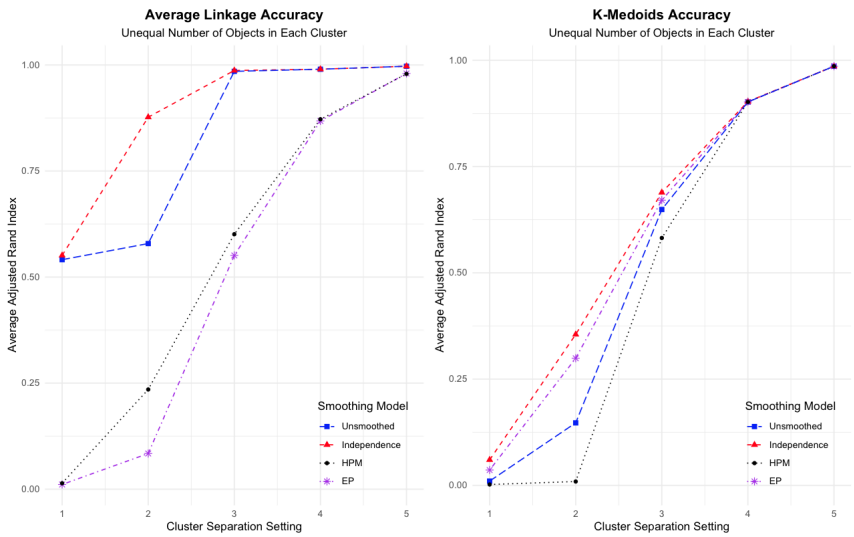

Figure 1: Average ARI value assuming 100 observations from cluster one, 200 from cluster two, and 300 from cluster three. The left plot corresponds to the Average Linkage clustering results and the right corresponds to the K-medoids clustering results.

of equal probability or high probability of match. Under these two assumptions the resulting clustering accuracy is substantially lower than those obtained from the use of the unsmoothed dissimilarities. This suggests it is of importance to choose a smoothing model that is supported by the structure of the data, especially when using the Average Linkage algorithm. The results can be seen visually in the left plot of Figure 1.

In Table 6, the resulting clustering accuracies are shown when observations are clustered using the K-Medoids algorithm. In this case, the most accurate solutions are still obtained when using the pre-smoothed dissimilarities under a model of independence versus any other method; however, the improvement seen in separation settings I-III are much more noticeable than in Table 5. We also see that while most of the average ARI values see a decline from the aforementioned table, those obtained by pre-smoothing under a model of equal probability see an increase. In separation settings I-III, the accuracy obtained under the assumption of equal probability, too, outperforms that from the use of the unsmoothed dissimilarities. This suggests the K-Medoids algorithm may be more robust to smoothing choice. In the final settings, separation settings IV and V, the accuracy of each method is the same. This is to be expected as the clustering problem is easiest in these two cases.

Table 6: Table gives average ARI for the K-Medoids clustering of the simulated data, based on different smoothing methods. There are 100 observations from cluster one, 200 from cluster two, and 300 from cluster three.

K-Medoids Algorithm

I	II	III	IV	V
0.010 (o)	0.147 (o)	0.649 (o)	0.902 (o)	0.986 (o)
0.060 (s/i)	0.355 (s/i)	0.689 (s/i)	0.903 (s/i)	0.986 (s/i)
0.036 (s/E)	0.299 (s/E)	0.671 (s/E)	0.902 (s/E)	0.986 (s/E)
0.002 (s/H)	0.009 (s/H)	0.582 (s/H)	0.902 (s/H)	0.986 (s/H)

NOTE: Each value is the average (across 5000 data sets) ARI for the clustering produced from the K-Medoids algorithm based on (top within each cell) the observed dissimilarities (o); (second within cell) the smoothed dissimilarities based on the independence model (s/i); (third within cell) the smoothed dissimilarities based on the equal-probability model (s/E); (last within cell) the smoothed dissimilarities based on the high probability of match model (s/H).

Overall the simulation results suggest that if smoothing will be performed, it is better to smooth the dissimilarities towards a model of independence when the tertiary data is believed to have arisen from a multinomial setting. In each setting explored, the accuracy obtained from using the pre-smoothed dissimilarities based on an independence model were as high or higher than that resulting from the observed dissimilarities. This finding agrees with Hitchcock and Chen (2008), who found pre-smoothing may not be necessary in cases of larger separation between clusters but may result in better performance with smaller separation between clusters. The results also suggest that the Average Linkage algorithm may be the better algorithm to use under such settings in general as the accuracy obtained from this algorithm is typically higher than what is seen with K-Medoids. Thus smoothing (while helpful when using the Average Linkage algorithm) may be even more beneficial if a practitioner will be using the K-Medoids algorithm. A practitioner should also be mindful of the limitations of this simulation study. It only considers the framework in which the data has arisen from a discrete underlying process. Results may be different had a continuous process been used to generate the data. In the data application of Section 4 we consider the framework in which tertiary objects arise from the discretization of a mixture of discrete and continuous processes. This should not be seen as problematic as Simonoff (1998) discusses smoothing is more natural in cases such as those where categorical outcomes are ordinal or have resulted

from the discretization of a continuous process.

4. An application to Diabetes

We here apply the pre-smoothing method proposed in Section 2 of this paper to the Pima Indian Diabetes data (National Institute of Diabetes and Digestive and Kidney Diseases, 1990), obtained from the UCI Repository for Machine Learning (Dua and Graff, 2020). The original dataset consists of $n = 768$ Pima women with recordings for $P = 9$ variables: Number of pregnancies (Preg), plasma glucose (Glucose), blood pressure (BP), tricep skinfold thickness (Tricep), serum insulin level (Insulin), body mass index (BMI), diabetes pedigree function (Ped), age (Age), and diabetes status at the time of the study. We omit the diabetes status variable from the cluster analysis, as we are treating this as an unsupervised problem. We will use this variable, instead, as a type of standard against which to compare our clustering results. Note that this may not be a perfect gold standard as a representation of the "true" clustering structure, but it does provide some sort of standard partition to which we can compare our clustering results. We consider the existence of $C = 2, 3$, or 4 clusters in the dataset and choose the value of C that results in the highest average silhouette width (Rousseeuw, 1987) to identify the best number of clusters. Once the optimum C has been identified, clustering results obtained using the Average Linkage and K-Medoids algorithm will be discussed. These results will be shown in Section 4.2.

4.1 *Data preprocessing and variable transformations*

The original Pima Indian Diabetes dataset included several observations of 0 for variable measurements, such as plasma glucose level, where such a value is nonsensical. Therefore, such observations were removed from the dataset. The final dataset used in this section thus consists of $n = 391$ observations with $P = 8$ variables.

Since the 8 variables recorded for each subject are either discrete or continuous variables, at the next stage of preprocessing, each variable was converted to a tertiary variable. Mea-

surement values of each variable that fall in the first category are denoted as 0. Measurement values of each variable that fall in the second category are denoted as 1. Measurement values for each variable that fall in the last category are denoted as 2.

The variables *Preg*, *Tricep*, and *Age* were transformed into the categories given based on careful examination of each variables' distribution. The remaining variables were transformed into their respective categories based on practical cutoffs described in readily available literature. For example, research suggests a glucose tolerance test outcome below 140 mg/dL is normal, while a measure between 140 mg/dL and 199 mg/dL is considered pre-diabetic, and a level above 199 mg/dL is considered diabetic (Mayo Clinic, 2019). The remaining 4 variables were treated similarly.

4.2 *Results*

The average silhouette widths were highest when $C = 2$. Therefore, the results presented in this section assume there are two subpopulations in the Pima Indian Diabetes Dataset.

Figure 2 shows the Average Linkage clustering of the subjects using the unsmoothed dissimilarities and the three proposed models for pre-smoothing. Note, cluster 1 when using either the unsmoothed dissimilarities or the dissimilarities pre-smoothed towards an independence model seemingly corresponds to cluster 2 when using the equal probability or high probability of match model. (Since the numerical labeling of the clusters in the output is arbitrary, it is irrelevant whether a cluster is labeled 1 or 2. What matters is how the individuals are partitioned into the two clusters). In this figure, the dissimilarities pre-smoothed towards a model of equal probability or high probability of match appear to result in a clustering structure with more inter-cluster separation and less overlap than that obtained using the unsmoothed dissimilarities or those pre-smoothed towards an independence model.

In Figure 3, the clustering results produced through the use of the K-Medoids algorithm are shown. Here, the use of any of the options for the dissimilarities (except those pre-smoothed towards a high probability of match model) results in a similar amount of overlap between

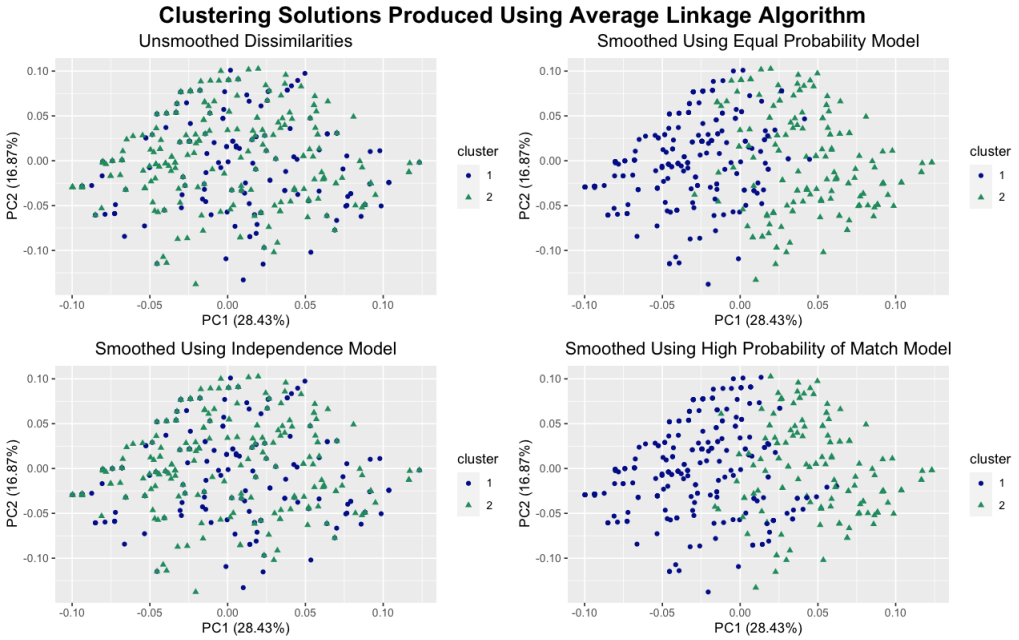

Figure 2: The plots above show the clustering of the Pima Indian subjects produced using the unsmoothed dissimilarities (top-left), equal probability pre-smoothed dissimilarities (top-right), independence pre-smoothed dissimiliarities (bottom-left), and high probability of match pre-smoothed dissimilarities (bottom-right) within the Average Linkage clustering algorithm.

each cluster. To assess the clustering solutions objectively the ARI is used.

Table 7 shows the classification accuracy of each algorithm resulting from the use of smoothed and non-smoothed dissimilarities. The highest accuracy, as indicated by a higher ARI, is denoted in bold for each algorithm. The table suggests the clustering accuracy is highest when the dissimilarities are pre-smoothed. This comes through the use of the equal probability model when using the Average Linkage algorithm and the independence model when using the K-Medoids algorithm. Furthermore, in both cases, the highest ARI is no-

Table 7: ARI values obtained from the clustering of 391 Pima Indian Women using K-medoids and Average Linkage Algorithms using the unsmoothed dissimilarities and three different smoothing methods.

Cluster Goodness

Smoothing Method	**Average Linkage**	**K-medoids**
Unsmoothed	0.637	0.640
Equal Probability	**0.725**	0.703
Independence	0.637	**0.736**
High Probability of Match	0.688	0.638

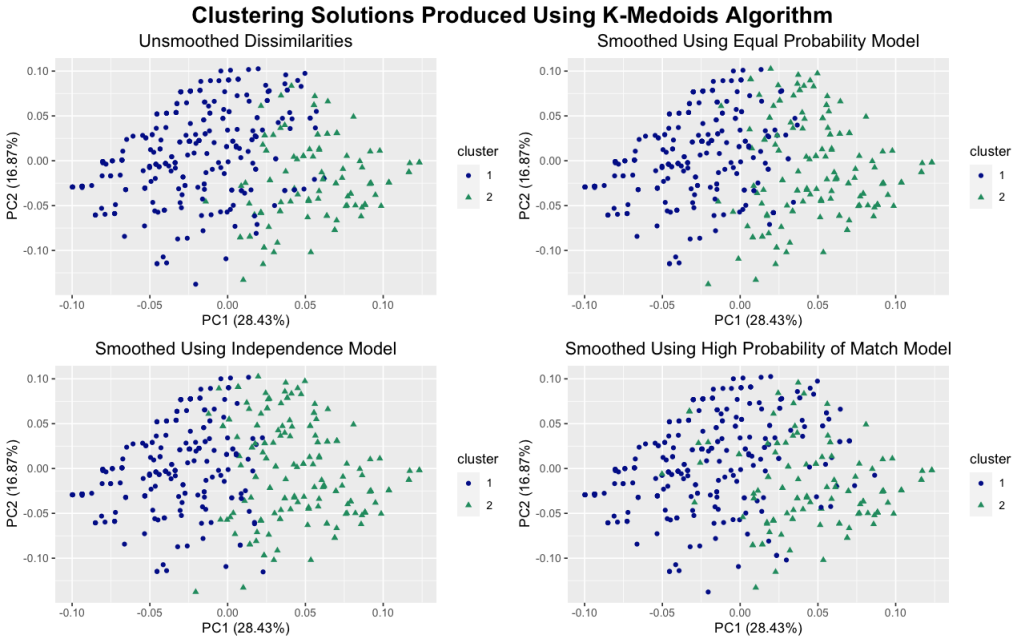

Figure 3: The plots above show the clustering of the Pima Indian subjects produced using the unsmoothed dissimilarities (top-left), equal probability pre-smoothed dissimilarities (top-right), independence pre-smoothed dissimiliarities (bottom-left), and high probability of match pre-smoothed dissimilarities (bottom-right) within the K-Medoids clustering algorithm.

ticeably higher than that for the clustering resulting through the use of the unsmoothed dissimilarities. This suggests clustering the pre-smoothed dissimilarities (via equal probability or independence) may better reflect the true underlying structure of the data than does the clustering of the unsmoothed dissimilarities.

To further examine the actual differences between the clustering results, Tables 8 and 9 are provided. The ARI was highest for pre-smoothing the dissimilarities towards an equal probability model for the Average Linkage algorithm, thus Table 8 compares the cross-tabulation of the partition resulting from the use of the unsmoothed dissimilarities and that produced by this smoothing model. There are 19 subjects who tested negative for diabetes that are placed in different clusters between the smoothing and non-smoothing methods (about 5% of the observations). Similarly, there are 30 subjects who tested positive for diabetes that are placed in different clusters between the smoothing and non-smoothing method (about 8% of the observations).

For the K-Medoids algorithm, cluster accuracy was highest when the dissimilarities were

Table 8: Cross-tabulation of the segmentation of subjects using the unsmoothed dissimilarities and dissimilarities smoothed under equal probability in the Average Linkage algorithm compared to the actual diabetic test results.

	Unsmoothed		**Equal Probability**	
Test Result	**Cluster 1**	**Cluster 2**	**Cluster 2**	**Cluster 1**
Positive	64	66	94	36
Negative	96	165	77	184

Table 9: Cross-tabulation of the segmentation of subjects using the unsmoothed dissimilarities and dissimilarities smoothed under independence in the K-medoids algorithm compared to the actual diabetic test results.

	Unsmoothed		**Independence**	
Test Result	**Cluster 1**	**Cluster 2**	**Cluster 1**	**Cluster 2**
Negative	218	43	187	74
Positive	60	70	34	96

smoothed towards a model of independence. Consequently, Table 9 shows the cross-tabulation of the actual diabetes test results for each subject compared to the clustering results produced by the use of the unsmoothed dissimilarities and the dissimilarities smoothed under a model of independence within the K-Medoids algorithm. The two methods account for a difference of 31 and 26 subjects who tested negative and positive, respectively—a difference in clustering output for about 15% of the subjects.

5. Discussion

In this paper we proposed a dissimilarity-based method for the clustering of tertiary observations. The proposed method utilizes statistical smoothing to help recover the true latent structure from which observations have arisen.

The results from the simulation study suggest when the tertiary observations have arisen from a multinomial setting, more accurate clusters are formed in most cases by using pre-smoothed dissimilarities. Within the Average Linkage algorithm, it appears to be best to pre-smooth the dissimilarities towards a model of independence. With the K-Medoids algorithm, the benefit of smoothing toward independence is apparent, and the equal-probability model also appears to be effective. The main findings suggest pre-smoothing is most influential, in

this setting, when there is more overlap between clusters. In the cases when there is much more distance between cluster centers, the accuracy obtained using the pre-smoothed dissimilarities is comparable to using the observed dissimilarities. This may suggest that pre-smoothing may be a good idea to implement regardless of the believed distance between the cluster centers or within-cluster variability in many cases, if a good smoothing model that is supported by the data can be applied.

In our diabetes application, results suggest the obtained cluster partitions more accurately reflected the underlying structure of the data and were more comparable to the blood diabetes test results when the pre-smoothed dissimilarities were used rather than when the traditional (non-smoothed) dissimilarities are used.

Overall, the hypothesis that pre-smoothing the observed dissimilarities may result in the formation of clusters that more accurately reflect the true underlying structure seems to be supported in many cases. A natural next step would be to explore other methods of smoothing and methods to generalize to the case of an arbitrary number of categories. Such future approaches might consist of putting a Bayesian prior on the smoothing parameter or even exploring other estimators of $\boldsymbol{\pi}$ that could be used in place of the Fienberg-Holland estimator. It is also worth noting that the increase in accuracy resulting from pre-smoothing the dissimilarities within the K-Medoids algorithm suggests pre-smoothing may be more influential when using partitioning-based methods of clustering rather than a hierarchical algorithm. This is promising as it has been noted in many papers that such partitioning-based methods tend to be more computationally efficient than hierarchical clustering methods (see, e.g., Huang (2008)). Consequently, a generalized method could have the ability to impact a variety of fields and applications. Such tasks may include those of clustering large datasets based on the Likert scale or text, the clustering of microarray data in genomics, or clustering images and documents in information retrieval.

References

Agresti, A. (2012). *Categorical Data Analysis*. John Wiley & Sons, Hoboken, New Jersey.

Albalate, A. and Minker, W. (2011). *Semi-Supervised and Unsupervised Machine Learning: Novel Strategies*. John Wiley & Sons, Hoboken, New Jersey.

Albert, J. H. (1987). Empirical Bayes estimation in contingency tables. *Communications in Statistics-Theory and Methods*, 16(8):2459–2485. doi:10.1080/03610928708829518.

Boriah, S., Chandola, V., and Kumar, V. (2008). Similarity measures for categorical data: A comparative evaluation. In *Proceedings of the 2008 SIAM International Conference on Data Mining*, pages 243–254. SIAM.

Cornell, J. E., Pugh, J. A., Williams Jr, J. W., Kazis, L., Lee, A. F., Parchman, M. L., Zeber, J., Pederson, T., Montgomery, K. A., and Noël, P. H. (2009). Multimorbidity clusters: Clustering binary data from multimorbidity clusters: Clustering binary data from a large administrative medical database. *Applied Multivariate Research*, 12(3):163–182. doi:10.1080/03610928708829518.

Dolnicar, S. and Leisch, F. (2004). Segmenting markets by bagged clustering. *Australasian Marketing Journal (AMJ)*, 12(1):51–65. doi:10.1016/S1441-3582(04)70088-9.

Dua, D. and Graff, C. (2020). UCI Machine Learning Repository. `http://archive.ics.uci.edu/ml`.

Efron, B. and Morris, C. (1977). Stein's paradox in statistics. *Scientific American*, 236(5):119–127. doi:10.1038/scientificamerican0577-119.

Everitt, B. S., Landau, S., Leese, M., and Stahl, D. (2011). *Cluster Analysis*. John Wiley & Sons, West Sussex, United Kingdom.

Fienberg, S. E. and Holland, P. W. (1973). Simultaneous estimation of multinomial cell probabilities. *Journal of the American Statistical Association*, 68(343):683–691. doi:10.1080/01621459.1973.10481405.

Friedman, J., Hastie, T., and Tibshirani, R. (2017). *The Elements of Statistical Learning: Data Mining, Inference,and Prediction*. Springer Series in Statistics, New York, New York. doi:10.1007/b94608.

Hitchcock, D. B. and Chen, Z. (2008). Smoothing dissimilarities to cluster binary data. *Computational Statistics and Data Analysis*, 52(10):4699–4711. doi:10.1016/j.csda.2008.03.012.

Huang, A. (2008). Similarity Measures for Text Document Clustering. In *Proceedings of the Sixth New Zealand Computer Science Research Student Conference(NZCSRSC2008), Christchurch, New Zealand*, volume 4, pages 9–56.

Huang, Z. (1997). A fast clustering algorithm to cluster very large categorical data sets in data mining. *DMKD*, 3(8):34–39.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1):193–218. doi:10.1007/bf01908075.

Kaufman, L. and Rousseeuw, P. J. (1987). Clustering by means of medoids. *Statistical Data Analysis based on the L1 Norm*, pages 405–416.

Mayo Clinic (2019). Glucose tolerance test. `http://www.mayoclinic.org/tests-procedures/glucose-tolerance-test/about/pac-20394296`.

McNicholas, P. D. (2017). *Mixture Model-Based Classification*. Taylor & Francis Group, Boca Raton, Florida.

National Institute of Diabetes and Digestive and Kidney Diseases (1990). Pima Indians Diabetes Dataset. `http://archive.ics.uci.edu/ml/machine-learning-databases/pima-indians-diabetes`. Accessed June 2017.

R Core Team (2019). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. `https://www.R-project.org`.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850. doi:10.1080/01621459.1971.10482356.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65. doi:10.1016/0377-0427(87)90125-7.

Simonoff, J. S. (1995). Smoothing categorical data. *Journal of Statistical Planning and Inference*, 47(1-2):41–69. doi:10.1016/0378-3758(94)00121-b.

Simonoff, J. S. (1998). *Smoothing Methods in Statistics*. Springer-Verlag, New York, New York.

Sokal, R. R. and Michener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38:1409–1438.