CrossMark

Clustering with missing features: a penalized dissimilarity measure based approach

**Shounak Datta[1] · Supritam Bhattacharjee[2] ·
Swagatam Das[1]**

Abstract Many real-world clustering problems are plagued by incomplete data character-
ized by missing or absent features for some or all of the data instances. Traditional clustering
methods cannot be directly applied to such data without preprocessing by imputation or
marginalization techniques. In this article, we overcome this drawback by utilizing a penalized
dissimilarity measure which we refer to as the feature weighted penalty based dissimilarity
(FWPD). Using the FWPD measure, we modify the traditional k-means clustering algo-
rithm and the standard hierarchical agglomerative clustering algorithms so as to make them
directly applicable to datasets with missing features. We present time complexity analyses
for these new techniques and also undertake a detailed theoretical analysis showing that the
new FWPD based k-means algorithm converges to a local optimum within a finite number
of iterations. We also present a detailed method for simulating random as well as feature
dependent missingness. We report extensive experiments on various benchmark datasets for
different types of missingness showing that the proposed clustering techniques have gen-
erally better results compared to some of the most well-known imputation methods which
are commonly used to handle such incomplete data. We append a possible extension of the
proposed dissimilarity measure to the case of absent features (where the unobserved features
are known to be undefined).

Keywords Missing features · Penalized dissimilarity measure · k-means · Hierarchical
agglomerative clustering · Absent features

Editor: Joao Gama.

✉ Swagatam Das
   swagatam.das@isical.ac.in

[1] Electronics and Communication Sciences Unit, Indian Statistical Institute, 203, B. T. Road,
   Kolkata 700 108, India

[2] Instrumentation and Electronics Engineering Department, Jadavpur University, Salt Lake Campus,
   Salt Lake City, Block-LB, Plot No. 8, Sector - III, Kolkata 700 098, India

1 Introduction

In data analytics, clustering is a fundamental technique concerned with partitioning a given dataset into useful groups (called *clusters*) according to the relative similarity among the data instances. Clustering algorithms attempt to partition a set of data instances (characterized by some *features*), into different clusters such that the member instances of any given cluster are akin to each other and are different from the members of the other clusters. Greater the similarity within a group and the dissimilarity between groups, better is the clustering obtained by a suitable algorithm.

Clustering techniques are of extensive use and are hence being constantly investigated in statistics, machine learning, and pattern recognition. Clustering algorithms find applications in various fields such as economics, marketing, electronic design, space research, etc. For example, clustering has been used to group related documents for web browsing (Broder et al. 1997; Haveliwala et al. 2000), by banks to cluster the previous transactions of clients to identify suspicious (possibly fraudulent) behaviour (Sabau 2012), for formulating effective marketing strategies by clustering customers with similar behaviour (Chaturvedi et al. 1997), in earthquake studies for identifying dangerous zones based on previous epicentre locations (Weatherill and Burton 2009; Shelly et al. 2009; Lei 2010), and so on. However, when we analyze such real-world data, we may encounter incomplete data where some features of some of the data instances are missing. For example, web documents may have some expired hyper-links. Such missingness may be due to a variety of reasons such as data input errors, inaccurate measurement, equipment malfunction or limitations, and measurement noise or data corruption, etc. This is known as unstructured missingness (Chan and Dunn 1972; Rubin 1976). Alternatively, not all the features may be defined for all the data instances in the dataset. This is termed as structural missingness or absence of features (Chechik et al. 2008). For example, credit-card details may not be defined for non-credit card clients of a bank.

Missing features have always been a challenge for researchers because traditional learning methods (which assume all data instances to be *fully observed*, i.e. all the features are observed) cannot be directly applied to such incomplete data, without suitable preprocessing. When the rate of missingness is low, the data instances with missing values may be ignored. This approach is known as *marginalization*. Marginalization cannot be applied to data having a sizable number of missing values, as it may lead to the loss of a sizable amount of information. Therefore, sophisticated methods are required to fill in the vacancies in the data, so that traditional learning methods can be applied subsequently. This approach of filling in the missing values is called *imputation*. However, inferences drawn from data having a large fraction of missing values may be severely warped, despite the use of such sophisticated imputation methods (Acuña and Rodriguez 2004).

1.1 Literature

The initial models for feature missingness are due to Rubin (1976); Little and Rubin (1987). They proposed a three-fold classification of missing data mechanisms, viz. Missing Completely At Random (MCAR), Missing At Random (MAR), and Missing Not At Random (MNAR). MCAR refers to the case where missingness is entirely haphazard, i.e. the likelihood of a feature being unobserved for a certain data instance depends neither on the observed nor on the unobserved characteristics of the instance. For example, in an annual income survey, a citizen is unable to participate, due to unrelated reasons such as traffic or schedule problems. MAR eludes to the cases where the missingness is conditional to the observed features of an instance, but is independent of the unobserved features. Suppose,

college-goers are less likely to report their income than office-goers. But, whether a college-goer will report his or her income is independent of the actual income. MNAR is characterized by the dependence of the missingness on the unobserved features. For example, people who earn less are less likely to report their incomes in the annual income survey. Datta et al. (2016b) further classified MNAR into two sub-types, namely MNAR-I when the missingness only depends on the unobserved features and MNAR-II when the missingness is governed by both observed as well as unobserved features. Schafer and Graham (2002) and Zhang et al. (2012) have observed that MCAR is a special case of MAR and that MNAR can also be converted to MAR by appending a sufficient number of additional features. Therefore, most learning techniques are based on the validity of the MAR assumption.

A lot of research on the problem of learning with missing or absent features has been conducted over the past few decades, mostly focussing on imputation methods. Several works such as Little and Rubin (1987) and Schafer (1997) provide elaborate theories and analyses of missing data. Common imputation methods (Donders et al. 2006) involve filling the missing features of data instances with zeros [Zero Imputation (ZI)], or the means of the corresponding features over the entire dataset [Mean Imputation (MI)]. Class Mean Imputation or Concept Mean Imputation (CMI) is a slight modification of MI that involves filling the missing features with the average of all observations having the same label as the instance being filled. Yet another common imputation method is k-Nearest Neighbor Imputation (kNNI) (Dixon 1979), where the missing features of a data instance are filled in by the means of corresponding features over its k-Nearest Neighbors (kNN), on the observed subspace. Grzymala-Busse and Hu (2001) suggested various novel imputation schemes such as treating missing attribute values as special values. Rubin (1987) proposed a technique called Multiple Imputation (MtI) to model the uncertainty inherent in imputation. In MtI, the missing values are imputed by a typically small (e.g. 5–10) number of simulated versions, depending on the percentage of missing data (Chen 2013; Horton and Lipsitz 2001). Some more sophisticated imputation techniques have been developed, especially by the bioinformatics community, to impute the missing values by exploiting the correlations between data. A prominent example is the Singular Value Decomposition based Imputation (SVDI) technique (Troyanskaya et al. 2001) which performs regression based estimation of the missing values using the k most significant eigenvectors of the dataset. Other examples inlcude Least Squares Imputation (LSI) (Bo et al. 2004), Non-Negative LSI (NNLSI) and Collateral Missing Value Estimation (CMVE) (Sehgal et al. 2005). Model-based methods are related to yet distinct from imputation techniques. These methods attempt to model the distributions for the missing values instead of filling them in Dempster and Rubin (1983); Ahmad and Tresp (1993); Wang and Rao (2002a, b).

However, most of these techniques assume the pattern of missingness to be MCAR or MAR because this allows the use of simpler models of missingness (Heitjan and Basu 1996). Such simple models are not likely to perform well in case of MNAR as the pattern of missingness also holds information. Hence, other methods have to be developed to tackle incomplete data due to MNAR (Marlin 2008). Moreover, imputation may often lead to the introduction of noise and uncertainty in the data (Dempster and Rubin 1983; Little and Rubin 1987; Barceló 2008; Myrtveit et al. 2001).

In light of the observations made in the preceding paragraph, some learning methods avoid the inexact methods of imputation (as well as marginalization) altogether, while dealing with missingness. A common paradigm is random subspace learning where an ensemble of learners is trained on projections of the data in random subspaces and an inference is drawn based on the concensus among the ensemble (Krause and Polikar 2003; Juszczak and Duin 2004; Nanni et al. 2012). Chechik et al. (2008) used the geometrical insight of max-margin

classification to formulate an objective function which was optimized to directly classify the incomplete data. This was extended to the max-margin regression case for software effort prediction with absent features in Zhang et al. (2012). Wagstaff (2004); Wagstaff and Laidler (2005) suggested a k-means algorithm with Soft Constraints (KSC) where soft constraints determined by fully observed objects are introduced to facilitate the grouping of instances with missing features. Himmelspach and Conrad (2010) provided a good review of partitional clustering techniques for incomplete datasets, which mentions some other techniques that do not make use of imputation.

The idea to modify the distance between the data instances to directly tackle missingness (without having to resort to imputation) was first put forth by Dixon (1979). The Partial Distance Strategy (PDS) proposed in Dixon (1979) scales up the *observed distance*, i.e. the distance between two data instances in their *common observed subspace* (the subspace consisting of the observed features common to both data instances) by the ratio of the total number of features (observed as well as unobserved) and the number of common observed features between them to obtain an estimate of their distance in the fully observed space. Hathaway and Bezdek (2001) used the PDS to extend the Fuzzy C-Means (FCM) clustering algorithm to cases with missing features. Furthermore, Millán-Giraldo et al. (2010) and Porro-Muñoz et al. (2013) generalized the idea of the PDS by proposing to scale the observed distance by factors other than the fraction of observed features. However, neither the PDS nor its extensions can always provide a good estimate of the actual distance as the observed distance between two instances may be unrelated to the distance between them in the unobserved subspace.

1.2 Motivation

As observed earlier, one possible way to adapt supervised as well as unsupervised learning methods to problems with missingness is to modify the *distance* or *dissimilarity measure* underlying the learning method. The idea is that the modified dissimilarity measure should use the common observed features to provide approximations of the distances between the data instances if they were to be fully observed. PDS is one such measure. Such approaches neither require marginalization nor imputation and are likely to yield better results than either of the two. For example, let $X_{full} = \{\mathbf{x}_1 = (1, 2), \mathbf{x}_2 = (1.8, 1), \mathbf{x}_3 = (2, 2.5)\}$ be a dataset consisting of three points in $\mathbb{R}^2$. Then, we have $d_E(\mathbf{x}_1, \mathbf{x}_2) = 1.28$ and $d_E(\mathbf{x}_1, \mathbf{x}_3) = 1.12$, $d_E(\mathbf{x}_i, \mathbf{x}_j)$ being the Euclidean distance between any two fully observed points $\mathbf{x}_i$ and $\mathbf{x}_j$ in X_{full}. Suppose that the first coordinate of the point $(1, 2)$ be unobserved, resulting in the incomplete dataset $X = \{\widetilde{\mathbf{x}}_1 = (*, 2), \mathbf{x}_2 = (1.8, 1), \mathbf{x}_3 = (2, 2.5)\}$ ('*' denotes a missing value), on which learning must be carried out. Notice that this is a case of unstructured missingness (because the unobserved value is known to exist), as opposed to the structural missingness of Chechik et al. (2008) 0. Using ZI, MI and 1NNI respectively, we obtain the following filled in datasets:

$$X_{ZI} = \{\widehat{\mathbf{x}}_1 = (0, 2), \mathbf{x}_2 = (1.8, 1), \mathbf{x}_3 = (2, 2.5)\},$$
$$X_{MI} = \{\widehat{\mathbf{x}}_1 = (1.9, 2), \mathbf{x}_2 = (1.8, 1), \mathbf{x}_3 = (2, 2.5)\},$$
$$\text{and } X_{1NNI} = \{\widehat{\mathbf{x}}_1 = (2, 2), \mathbf{x}_2 = (1.8, 1), \mathbf{x}_3 = (2, 2.5)\},$$

where $\widehat{\mathbf{x}}_1$ denotes an estimate of $\mathbf{x}_1$. If PDS is used to estimate the corresponding distances in X, then the distance $d_{PDS}(\mathbf{x}_1, \mathbf{x}_i)$ between the implicit estimate of $\mathbf{x}_1$ and some other

instance $\mathbf{x}_i \in X$ is obtained by

$$d_{PDS}(\mathbf{x}_1, \mathbf{x}_i) = \sqrt{\frac{2}{1}(x_{1,2} - x_{i,2})^2},$$

where $x_{1,2}$ and $x_{i,2}$ respectively denote the 2nd features of $\mathbf{x}_1$ and $\mathbf{x}_i$, and 2 is the numerator of the multiplying factor due to the fact that $\mathbf{x}_i \in \mathbb{R}^2$ and 1 is the denominator owing to the fact that only the 2nd feature is observed for both $\mathbf{x}_1$ and $\mathbf{x}_i$. Then, we get

$$d_{PDS}(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{\frac{2}{1}(2-1)^2} = 1.41,$$

$$\text{and } d_{PDS}(\mathbf{x}_1, \mathbf{x}_3) = \sqrt{\frac{2}{1}(2-2.5)^2} = 0.71.$$

The improper estimates obtained by PDS are due to the fact that the distance in the common observed subspace does not reflect the distance in the unobserved subspace. This is the principal drawback of the PDS method, as discussed earlier. Since the observed distance between two data instances is essentially a lower bound on the Euclidean distance between them (if they were to be fully observed), adding a suitable penalty to this lower bound can result in a reasonable approximation of the actual distance. This approach (Datta et al. 2016b) called the Penalized Dissimilarity Measure (PDM) may be able to overcome the drawback which plagues PDS. Let the penalty between $\mathbf{x}_1$ and $\mathbf{x}_i$ be given by the ratio of the number of features which are unobserved for at least one of the two data instances and the total number of features in the entire dataset. Then, the dissimilarity $\delta_{PDM}(\mathbf{x}_1, \mathbf{x}_i)$ between the implicit estimate of $\mathbf{x}_1$ and some other $\mathbf{x}_i \in X$ is

$$\delta_{PDM}(\mathbf{x}_1, \mathbf{x}_i) = \sqrt{(x_{1,2} - x_{i,2})^2 + \frac{1}{2}},$$

where the 1 in the numerator of the penalty term is due to the fact that the 1st feature of $\mathbf{x}_1$ is unobserved. Therefore, the dissimilarities $\delta_{PDM}(\mathbf{x}_1, \mathbf{x}_2)$ and $\delta_{PDM}(\mathbf{x}_1, \mathbf{x}_3)$ are

$$\delta_{PDM}(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{(2-1)^2 + \frac{1}{2}} = 1.5,$$

$$\text{and } \delta_{PDM}(\mathbf{x}_1, \mathbf{x}_3) = \sqrt{(2-2.5)^2 + \frac{1}{2}} = 1.$$

The situation is illustrated in Fig. 1a. The reader should note that the points estimated using ZI, MI and 1NNI exist in the same 2-D Cartesian space to which X_{full} is native. On the other hand, the points estimated by both PDS and PDM exist in their individual abstract spaces (likely distinct from the native 2-D space). Therefore, for the sake of easy comparison, we illustrate all the estimates together by superimposing both these abstract spaces on the native 2-D space so as to coincide at the points $\mathbf{x}_2$ and $\mathbf{x}_3$. It can be seen that the approach based on the PDM does not suffer from the drawback of PDS and is better able to preserve the relationship between the points. Moreover, it should be noted that there are two possible images for each of the estimates obtained by both PDS and PDM. Therefore, had the partially observed point instead been $\mathbf{x}'_1 = (3, 2)$ with the first feature missing (giving rise to the same incomplete dataset X; $\widetilde{\mathbf{x}}'_1$ replacing the identical incomplete point $\widetilde{\mathbf{x}}_1$), PDS and PDM would still find reasonably good estimates (PDM still being better than PDS). This situation is also illustrated in Fig. 1b. In general,
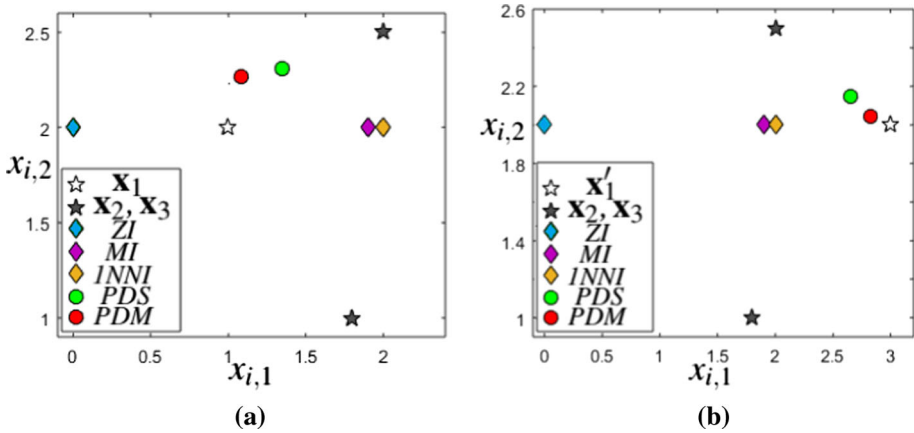
Fig. 1 Comparison of various techniques for handling missing features. **a** Comparison for $\mathbf{x}_1$. **b** Comparison for $\mathbf{x}'_1$

1. ZI works well only for missing values in the vicinity of the origin and is also origin dependent;
2. MI works well only when the missing value is near the observed mean of the missing feature;
3. kNNI is reliant on the assumption that neighbors have similar features, but suffers from the drawbacks that missingness may give rise to erroneous neighbor selection and that the estimates are restricted to the range of observed values of the feature in question;
4. PDS suffers from the assumption that the common observed distances reflect the unobserved distances; and
5. None of these methods differentiate between identical incomplete points, i.e. $\widetilde{\mathbf{x}}_1$ and $\widetilde{\mathbf{x}}'_1$ are not differentiated between.

However, a PDM successfully steers clear of all these drawbacks (notice that $\delta(\mathbf{x}_1, \mathbf{x}'_1) = \frac{1}{2}$). Furthermore, such a PDM can also be easily applied to the case of absent features, by slightly modifying the penalty term (see "Appendix A"). This knowledge motivates us to use a PDM to adapt traditional clustering methods to problems with missing features.

1.3 Contribution

The FWPD measure is a PDM used in Datta et al. (2016b) for kNN classification of datasets with missing features.[1] The FWPD between two data instances is a weighted sum of two terms; the first term being the observed distance between the instances and the second being a penalty term. The penalty term is a sum of the penalties corresponding to each of the features which are missing from at least one of the data instances; each penalty being directly proportional to the probability of its corresponding feature being observed. Such a weighting scheme imposes greater penalty if a feature which is observed for a large fraction of the data is missing for a particular instance. On the other hand, if the missing feature is unobserved for a large fraction of the data, then a smaller penalty is imposed.

The contributions of the current article are in order:

---

[1] The work of Datta et al. (2016b) is based on the FWPD measure originally proposed in the archived version of the current article (Datta et al. 2016a).

1. In the current article, we formulate the k-means clustering problem for datasets with missing features based on the proposed FWPD and develop an algorithm to solve the new formulation.
2. We prove that the proposed algorithm is guaranteed to converge to a locally optimal solution of the modified k-means optimization problem formulated with the FWPD measure.
3. We also propose Single Linkage, Average Linkage, and Complete Linkage based HAC methods for datasets plagued by missingness, based on the proposed FWPD.
4. We provide an extensive discussion on the properties of the FWPD measure. The said discussion is more thorough compared to that of Datta et al. (2016b).
5. We further provide a detailed algorithm for simulating the four types of missingness enumerated in Datta et al. (2016b), namely MCAR, MAR, MNAR-I (missingness only depends on the unobserved features) and MNAR-II (missingness depends on both observed as well as unobserved features).
6. Moreover, since this work presents an alternative to imputation and can be useful in scenarios where imputation is not practical (such as structural missingness), we append an extension of the proposed FWPD to the case of absent features (where the absent features are known to be undefined or non-existent). We also show that the FWPD becomes a semi-metric in the case of structural missingness.

Experiments are reported on diverse datasets and covers all four types of missingness. The results are compared with the popularly used imputation techniques. The comparative results indicate that our approach generally achieves better performance than the common imputation approaches used to handle incomplete data.

1.4 Organization

The rest of this paper is organized in the following way. In Sect. 2, we elaborate on the properties of the FWPD measure. The next section (Sect. 3) presents a formulation of the k-means clustering problem which is directly applicable to datasets with missing features, based on the FWPD discussed in Sect. 2. This section also puts forth an algorithm to solve the optimization problem posed by this new formulation. The subsequent section (Sect. 4) covers the HAC algorithm formulated using FWPD to be directly applicable to incomplete datasets. Experimental results (based on the missingness simulating mechanism discussed in the same section) are presented in Sect. 5. Relevant conclusions are drawn in Sect. 6. Subsequently, "Appendix A" deals with the extension of the proposed FWPD to the case of absent features (structural missingness).

2 Feature weighted penalty based dissimilarity measure for datasets with missing features

Let the dataset $X \subset \mathbb{R}^m$, i.e. the data instances in X are each characterized by m feature values in $\mathbb{R}$. Further, let X consist of n instances $\mathbf{x}_i$ ($i \in \{1, 2, \cdots, n\}$), some of which have missing features. Let $\gamma_{\mathbf{x}_i}$ denote the set of observed features for the data point $\mathbf{x}_i$. Then, the set of all features $S = \bigcup_{i=1}^n \gamma_{\mathbf{x}_i}$ and $|S| = m$. The set of features which are observed for all data instances in X is defined as $\gamma_{obs} = \bigcap_{i=1}^n \gamma_{\mathbf{x}_i}$. $|\gamma_{obs}|$ may or may not be non-zero. $\gamma_{miss} = S \backslash \gamma_{obs}$ is the set of features which are unobserved for at least one data point in X. The important notations used in this section (and beyond) are summarized in Table 1.

Table 1 Some important notations used in Sect. 2 and beyond

Notation	Meaning
X	Dataset with incomplete data points
n	Number of data points in X
$\mathbf{x}_i$	A data point in X
$x_{i,l}$	l-th feature of $\mathbf{x}_i$
S	Set of all features in X
m	Number of features in S, i.e. $|S|$
γ	General notation for a set of features in S
$\gamma_{\mathbf{x}_i}$	Set of features observed for point $\mathbf{x}_i$
γ_{obs}	Set of features observed for all instances in X
γ_{miss}	Set of features which are unobserved for some point in X
$d_\gamma(\mathbf{x}_i, \mathbf{x}_j)$	Distance between poinst $\mathbf{x}_i$ and $\mathbf{x}_j$ in the subspace defined by the features in γ
$d(\mathbf{x}_i, \mathbf{x}_j)$	Observed distance between points $\mathbf{x}_i$ and $\mathbf{x}_j$
$d_E(\mathbf{x}_i, \mathbf{x}_j)$	Euclidean distance between fully observed points $\mathbf{x}_i$ and $\mathbf{x}_j$
w_l	Number of instances in X having observed values for the l-th feature
$p(\mathbf{x}_i, \mathbf{x}_j)$	Feature Weighted Penalty (FWP) between $\mathbf{x}_i$ and $\mathbf{x}_j$
p_γ	FWP corresponding to the subspace defined by γ
$\delta(\mathbf{x}_i, \mathbf{x}_j)$	Feature Weighted Penalty based Dissimilarity (FWPD) between $\mathbf{x}_i$ and $\mathbf{x}_j$
d_{max}	Maximum observed distance between any two data points in X
α	Coefficient of relative importance between observed distance and FWP for FWPD
$\rho_{i,j,k}$	$p(\mathbf{x}_i, \mathbf{x}_j) + p(\mathbf{x}_j, \mathbf{x}_k) - p(\mathbf{x}_k, \mathbf{x}_i)$ for some $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in X$
ϕ	An empty set

Definition 1 Let the distance between any two data instances $\mathbf{x}_i, \mathbf{x}_j \in X$ in a subspace defined by γ be denoted as $d_\gamma(\mathbf{x}_i, \mathbf{x}_j)$. Then the observed distance (distance in the common observed subspace) between these two points can be defined as

$$d_{\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_j}}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{l \in \gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_j}} (x_{i,l} - x_{j,l})^2}, \tag{1}$$

where $x_{i,l}$ denotes the l-th feature of the data instance $\mathbf{x}_i$. For the sake of convenience, $d_{\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_j}}(\mathbf{x}_i, \mathbf{x}_j)$ is simplified to $d(\mathbf{x}_i, \mathbf{x}_j)$ in the rest of this paper.

Definition 2 If both $\mathbf{x}_i$ and $\mathbf{x}_j$ were to be fully observed, the Euclidean distance $d_E(\mathbf{x}_i, \mathbf{x}_j)$ between $\mathbf{x}_i$ and $\mathbf{x}_j$ would be defined as

$$d_E(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{l \in S} (x_{i,l} - x_{j,l})^2}.$$

Now, since $(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_j}) \subseteq S$, and $(x_{i,l} - x_{j,l})^2 \geq 0 \ \forall \ l \in S$, it follows that

$$d(\mathbf{x}_i, \mathbf{x}_j) \leq d_E(\mathbf{x}_i, \mathbf{x}_j) \ \forall \ \mathbf{x}_i, \mathbf{x}_j \in X.$$

Therefore, to compensate for the distance in the unobserved subspace, we add a Feature Weighted Penalty (FWP) $p(\mathbf{x}_i, \mathbf{x}_j)$ (defined below) to $d(\mathbf{x}_i, \mathbf{x}_j)$.

Definition 3 The FWP between $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as

$$p(\mathbf{x}_i, \mathbf{x}_j) = \frac{\sum_{l \in S \setminus (\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_j})} w_l}{\sum_{l' \in S} w_{l'}}, \tag{2}$$

where $w_l \in (0, n]$ is the number of instances in X having observed values of the feature l. It should be noted that FWP exacts greater penalty for unobserved occurrences of those features which are observed for a large fraction of the data instances. Moreover, since the value of the FWP solely depends on the taxable subspace $S \setminus (\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_j})$, we define an alternative notation for the FWP, viz. $p_\gamma = \sum_{l \in \gamma} w_l / \sum_{l' \in S} w_{l'}$. Hence, $p(\mathbf{x}_i, \mathbf{x}_j)$ can also be written as $p_{S \setminus (\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_j})}$.

Then, the definition of the proposed FWPD follows.

Definition 4 The FWPD between $\mathbf{x}_i$ and $\mathbf{x}_j$ is

$$\delta(\mathbf{x}_i, \mathbf{x}_j) = (1 - \alpha) \times \frac{d(\mathbf{x}_i, \mathbf{x}_j)}{d_{max}} + \alpha \times p(\mathbf{x}_i, \mathbf{x}_j), \tag{3}$$

where $\alpha \in (0, 1]$ is a parameter which determines the relative importance between the two terms and d_{max} is the maximum observed distance between any two points in X in their respective common observed subspaces.

2.1 Properties of the proposed FWPD

In this subsection, we discuss some of the important properties of the proposed FWPD measure. The following theorem discusses some of the important properties of the proposed FWPD measure and the subsequent discussion is concerned with the triangle inequality in the context of FWPD.

Theorem 1 *The FWPD measure satisfies the following important properties:*

1. *$\delta(\mathbf{x}_i, \mathbf{x}_i) \leq \delta(\mathbf{x}_i, \mathbf{x}_j) \, \forall \, \mathbf{x}_i, \mathbf{x}_j \in X$,*
2. *$\delta(\mathbf{x}_i, \mathbf{x}_i) \geq 0 \, \forall \, \mathbf{x}_i \in X$,*
3. *$\delta(\mathbf{x}_i, \mathbf{x}_i) = 0$ iff $\gamma_{\mathbf{x}_i} = S$, and*
4. *$\delta(\mathbf{x}_i, \mathbf{x}_j) = \delta(\mathbf{x}_j, \mathbf{x}_i) \, \forall \, \mathbf{x}_i, \mathbf{x}_j \in X$.*

Proof 1. From Eqs. (1) and (3), it follows that

$$\delta(\mathbf{x}_i, \mathbf{x}_i) = \alpha \times p(\mathbf{x}_i, \mathbf{x}_i). \tag{4}$$

It also follows from Eq. (2) that $p(\mathbf{x}_i, \mathbf{x}_i) \leq p(\mathbf{x}_i, \mathbf{x}_j) \, \forall \, \mathbf{x}_i, \mathbf{x}_j \in X$. Therefore, $\delta(\mathbf{x}_i, \mathbf{x}_i) \leq \alpha \times p(\mathbf{x}_i, \mathbf{x}_j)$. Since $\alpha \leq 1$, we have $\alpha \times p(\mathbf{x}_i, \mathbf{x}_j) \leq p(\mathbf{x}_i, \mathbf{x}_j)$. Now, it follows from Eq. (3) that $p(\mathbf{x}_i, \mathbf{x}_j) \leq \delta(\mathbf{x}_i, \mathbf{x}_j)$. Hence, we get $\delta(\mathbf{x}_i, \mathbf{x}_i) \leq \delta(\mathbf{x}_i, \mathbf{x}_j) \, \forall \, \mathbf{x}_i, \mathbf{x}_j \in X$.
2. It can be seen from Eq. (3) that $\delta(\mathbf{x}_i, \mathbf{x}_i) = \alpha \times p(\mathbf{x}_i, \mathbf{x}_i)$. Moreover, it follows from Eq. (2) that $p(\mathbf{x}_i, \mathbf{x}_i) \geq 0$. Hence, $\delta(\mathbf{x}_i, \mathbf{x}_i) \geq 0 \, \forall \, \mathbf{x}_i \in X$.
3. It is easy to see from Eq. (2) that $p(\mathbf{x}_i, \mathbf{x}_i) = 0$ iff $\gamma_{\mathbf{x}_i} = S$. Hence, it directly follows from Eq. (4) that $\delta(\mathbf{x}_i, \mathbf{x}_i) = 0$ iff $\gamma_{\mathbf{x}_i} = S$.
4. From Eq. (3) we have

$$\delta(\mathbf{x}_i, \mathbf{x}_j) = (1 - \alpha) \times \frac{d(\mathbf{x}_i, \mathbf{x}_j)}{d_{max}} + \alpha \times p(\mathbf{x}_i, \mathbf{x}_j),$$

$$\text{and } \delta(\mathbf{x}_j, \mathbf{x}_i) = (1 - \alpha) \times \frac{d(\mathbf{x}_j, \mathbf{x}_i)}{d_{max}} + \alpha \times p(\mathbf{x}_j, \mathbf{x}_i).$$

However, $d(\mathbf{x}_i, \mathbf{x}_j) = d(\mathbf{x}_j, \mathbf{x}_i)$ and $p(\mathbf{x}_i, \mathbf{x}_j) = p(\mathbf{x}_j, \mathbf{x}_i) \; \forall \; \mathbf{x}_i, \mathbf{x}_j \in X$ (by definition). Therefore, it can be easily seen that $\delta(\mathbf{x}_i, \mathbf{x}_j) = \delta(\mathbf{x}_j, \mathbf{x}_i) \; \forall \; \mathbf{x}_i, \mathbf{x}_j \in X$.

□

The triangle inequality is an important criterion which lends some useful properties to the space induced by a dissimilarity measure. Therefore, the conditions under which FWPD satisfies the said criterion are investigated below. However, it should be stressed that the satisfaction of the said criterion is not essential for the functioning of the clustering techniques proposed in the subsequent text.

Definition 5 For any three data instances $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in X$, the triangle inequality with respect to (w.r.t.) the FWPD measure is defined as

$$\delta(\mathbf{x}_i, \mathbf{x}_j) + \delta(\mathbf{x}_j, \mathbf{x}_k) \geq \delta(\mathbf{x}_k, \mathbf{x}_i). \tag{5}$$

The three following lemmas deal with the conditions under which Inequality (5) will hold.

Lemma 1 *For any three data instances* $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in X$ *let* $\rho_{i,j,k} = p(\mathbf{x}_i, \mathbf{x}_j) + p(\mathbf{x}_j, \mathbf{x}_k) - p(\mathbf{x}_k, \mathbf{x}_i)$. *Then* $\rho_{i,j,k} \geq 0 \; \forall \; \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in X$.

Proof Let us rewrite the penalty term $p(\mathbf{x}_i, \mathbf{x}_j)$ in terms of the spanned subspaces as $p(\mathbf{x}_i, \mathbf{x}_j) = p_{S \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j})} + p_{\gamma_{\mathbf{x}_i} \setminus \gamma_{\mathbf{x}_j}} + p_{\gamma_{\mathbf{x}_j} \setminus \gamma_{\mathbf{x}_i}}$. Now, accounting for the subspaces overlapping with the observed subspace of $\mathbf{x}_k$, we get

$$p(\mathbf{x}_i, \mathbf{x}_j) = p_{S \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})} + p_{\gamma_{\mathbf{x}_k} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j})} + p_{\gamma_{\mathbf{x}_i} \setminus (\gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})}$$
$$+ p_{(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_j}} + p_{\gamma_{\mathbf{x}_j} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_k})} + p_{(\gamma_{\mathbf{x}_j} \cap \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_i}}.$$

Similarly, $p(\mathbf{x}_j, \mathbf{x}_k) = p_{S \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})} + p_{\gamma_{\mathbf{x}_i} \setminus (\gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})} + p_{\gamma_{\mathbf{x}_j} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_k})}$
$$+ p_{(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_j}) \setminus \gamma_{\mathbf{x}_k}} + p_{\gamma_{\mathbf{x}_k} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j})} + p_{(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_j}},$$

and $p(\mathbf{x}_k, \mathbf{x}_i) = p_{S \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})} + p_{\gamma_{\mathbf{x}_j} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_k})} + p_{\gamma_{\mathbf{x}_k} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j})}$
$$+ p_{(\gamma_{\mathbf{x}_j} \cap \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_i}} + p_{\gamma_{\mathbf{x}_i} \setminus (\gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})} + p_{(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_j}) \setminus \gamma_{\mathbf{x}_k}}.$$

Hence, after canceling out appropriate terms, we get

$$\rho_{i,j,k} = p_{S \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})} + p_{\gamma_{\mathbf{x}_k} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j})} + 2 p_{(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_j}}$$
$$+ p_{\gamma_{\mathbf{x}_i} \setminus (\gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})} + p_{\gamma_{\mathbf{x}_j} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_k})}.$$

Now, since

$$p_{\gamma_{\mathbf{x}_i} \setminus (\gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})} + p_{\gamma_{\mathbf{x}_k} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j})} + p_{(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_j}} = p_{(\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_j}},$$

we can further simplify to

$$\rho_{i,j,k} = p_{(\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_j}} + p_{(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_k}) \setminus \gamma_{\mathbf{x}_j}} + p_{\gamma_{\mathbf{x}_j} \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_k})} + p_{S \setminus (\gamma_{\mathbf{x}_i} \cup \gamma_{\mathbf{x}_j} \cup \gamma_{\mathbf{x}_k})}. \tag{6}$$

Since all the terms in Expression (6) must be either zero or positive, this proves that $\rho_{i,j,k} \geq 0 \; \forall \; \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in X$.

□

Lemma 2 *For any three data points* $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in X$, *Inequality (5) is satisfied when* $(\gamma_{\mathbf{x}_i} \cap \gamma_{\mathbf{x}_j}) = (\gamma_{\mathbf{x}_j} \cap \gamma_{\mathbf{x}_k}) = (\gamma_{\mathbf{x}_k} \cap \gamma_{\mathbf{x}_i})$.

Proof From Eq. (3) the Inequality (5) can be rewritten as

$$
\begin{aligned}
(1-\alpha) &\times \frac{d(\mathbf{x}_i, \mathbf{x}_j)}{d_{max}} + \alpha \times p(\mathbf{x}_i, \mathbf{x}_j) \\
&+ (1-\alpha) \times \frac{d(\mathbf{x}_j, \mathbf{x}_k)}{d_{max}} + \alpha \times p(\mathbf{x}_j, \mathbf{x}_k) \\
&\geq (1-\alpha) \times \frac{d(\mathbf{x}_k, \mathbf{x}_i)}{d_{max}} + \alpha \times p(\mathbf{x}_k, \mathbf{x}_i).
\end{aligned}
\tag{7}
$$

Further simplifying (7) by moving the penalty terms to the Left Hand Side (LHS) and the observed distance terms to the Right Hand Side (RHS), we get

$$
\alpha \times \rho_{i,j,k} \geq \frac{(1-\alpha)}{d_{max}} \times (d(\mathbf{x}_k, \mathbf{x}_i) - (d(\mathbf{x}_i, \mathbf{x}_j) + d(\mathbf{x}_j, \mathbf{x}_k))).
\tag{8}
$$

When $(\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_j}) = (\gamma_{\mathbf{x}_j} \bigcap \gamma_{\mathbf{x}_k}) = (\gamma_{\mathbf{x}_k} \bigcap \gamma_{\mathbf{x}_i})$, as $d(\mathbf{x}_i, \mathbf{x}_j) + d(\mathbf{x}_j, \mathbf{x}_k) \geq d(\mathbf{x}_k, \mathbf{x}_i)$, the RHS of Inequality (8) is less than or equal to zero. Now, it follows from Lemma 1 that the LHS of Inequality (8) is always greater than or equal to zero as $\rho_{i,j,k} \geq 0$ and $\alpha \in (0, 1]$. Hence, LHS ≥ RHS, which completes the proof. □

Lemma 3 *If* $|\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_j}| \to 0$, $|\gamma_{\mathbf{x}_j} \bigcap \gamma_{\mathbf{x}_k}| \to 0$ *and* $|\gamma_{\mathbf{x}_k} \bigcap \gamma_{\mathbf{x}_i}| \to 0$*, then Inequality (8) tends to be satisfied.*

Proof When $|\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_j}| \to 0$, $|\gamma_{\mathbf{x}_j} \bigcap \gamma_{\mathbf{x}_k}| \to 0$ and $|\gamma_{\mathbf{x}_k} \bigcap \gamma_{\mathbf{x}_i}| \to 0$, then LHS $\to \alpha^+$ and RHS $\to 0$ for the Inequality (8). As $\alpha \in (0, 1]$, Inequality (8) tends to be satisfied. □

The following lemma deals with the value of the parameter $\alpha \in (0, 1]$ for which a relaxed form of the triangle inequality is satisfied for any three data instances in a dataset X.

Lemma 4 *Let* $\mathscr{P} = min\{\rho_{i,j,k} : \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in X, \rho_{i,j,k} > 0\}$*. Then, for any arbitrary constant* ϵ *satisfying* $0 \leq \epsilon \leq \mathscr{P}$*, if* $\alpha \geq (1 - \epsilon)$*, then the following relaxed form of the triangle inequality*

$$
\delta(\mathbf{x}_i, \mathbf{x}_j) + \delta(\mathbf{x}_j, \mathbf{x}_k) \geq \delta(\mathbf{x}_k, \mathbf{x}_i) - \epsilon^2,
\tag{9}
$$

is satisfied for any $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in X$*.*

Proof 1. If $\mathbf{x}_i, \mathbf{x}_j$, and $\mathbf{x}_k$ are all fully observed, then Inequality (5) holds. Now, since $\epsilon \geq 0$, therefore $\delta(\mathbf{x}_k, \mathbf{x}_i) \geq \delta(\mathbf{x}_k, \mathbf{x}_i) - \epsilon^2$. This implies $\delta(\mathbf{x}_i, \mathbf{x}_j) + \delta(\mathbf{x}_j, \mathbf{x}_k) \geq \delta(\mathbf{x}_k, \mathbf{x}_i) \geq \delta(\mathbf{x}_k, \mathbf{x}_i) - \epsilon^2$. Hence, Inequality (9) must hold.

2. If $(\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_j} \bigcap \gamma_{\mathbf{x}_k}) \neq S$ i.e. at least one of the data instances is not fully observed, and $\rho_{i,j,k} = 0$, then $(\gamma_{\mathbf{x}_i} \bigcup \gamma_{\mathbf{x}_k}) \backslash \gamma_{\mathbf{x}_j} = \phi$, $(\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}) \backslash \gamma_{\mathbf{x}_j} = \phi$, $S \backslash (\gamma_{\mathbf{x}_i} \bigcup \gamma_{\mathbf{x}_j} \bigcup \gamma_{\mathbf{x}_k}) = \phi$, and $\gamma_{\mathbf{x}_j} \backslash (\gamma_{\mathbf{x}_i} \bigcup \gamma_{\mathbf{x}_k}) = \phi$. This implies that $\gamma_{\mathbf{x}_j} = S$, and $\gamma_{\mathbf{x}_k} \bigcup \gamma_{\mathbf{x}_i} = \gamma_{\mathbf{x}_j}$. Moreover, since $\rho_{i,j,k} = 0$, we have $\delta(\mathbf{x}_i, \mathbf{x}_j) + \delta(\mathbf{x}_j, \mathbf{x}_k) - \delta(\mathbf{x}_k, \mathbf{x}_i) = d(\mathbf{x}_i, \mathbf{x}_j) + d(\mathbf{x}_j, \mathbf{x}_k) - d(\mathbf{x}_k, \mathbf{x}_i)$. Now, $\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k} \subseteq \gamma_{\mathbf{x}_i}$, $\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k} \subseteq \gamma_{\mathbf{x}_k}$ and $\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k} \subseteq \gamma_{\mathbf{x}_j}$ as $\gamma_{\mathbf{x}_k} \bigcup \gamma_{\mathbf{x}_i} = \gamma_{\mathbf{x}_j} = S$. Therefore, $d(\mathbf{x}_i, \mathbf{x}_j) + d(\mathbf{x}_j, \mathbf{x}_k) - d(\mathbf{x}_k, \mathbf{x}_i) \geq d_{\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}}(\mathbf{x}_i, \mathbf{x}_j) + d_{\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}}(\mathbf{x}_j, \mathbf{x}_k) - d_{\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}}(\mathbf{x}_k, \mathbf{x}_i)$. Now, by the triangle inequality in subspace $\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}$, $d_{\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}}(\mathbf{x}_i, \mathbf{x}_j) + d_{\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}}(\mathbf{x}_j, \mathbf{x}_k) - d_{\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}}(\mathbf{x}_k, \mathbf{x}_i) \geq 0$. Hence, $\delta(\mathbf{x}_i, \mathbf{x}_j) + \delta(\mathbf{x}_j, \mathbf{x}_k) - \delta(\mathbf{x}_k, \mathbf{x}_i) \geq 0$, i.e. Inequalities (5) and (9) are satisfied.

3. If $(\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_j} \bigcap \gamma_{\mathbf{x}_k}) \neq S$ and $\rho_{i,j,k} \neq 0$, as $\alpha \geq (1-\epsilon)$, LHS of Inequality (8) $\geq (1-\epsilon) \times (p_{(\gamma_{\mathbf{x}_i} \bigcup \gamma_{\mathbf{x}_k}) \backslash \gamma_{\mathbf{x}_j}} + p_{(\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{x}_k}) \backslash \gamma_{\mathbf{x}_j}} + p_{\gamma_{\mathbf{x}_j} \backslash (\gamma_{\mathbf{x}_i} \bigcup \gamma_{\mathbf{x}_k})} + p_{S \backslash (\gamma_{\mathbf{x}_i} \bigcup \gamma_{\mathbf{x}_j} \bigcup \gamma_{\mathbf{x}_k})})$. Since $\epsilon \leq \mathscr{P}$, we further get that LHS $\geq (1-\epsilon)\epsilon$. Moreover, as $\frac{1}{d_{max}}(d(\mathbf{x}_k, \mathbf{x}_i) - (d(\mathbf{x}_i, \mathbf{x}_j) + d(\mathbf{x}_j, \mathbf{x}_k))) \leq 1$, we get RHS of Inequality (8) $\leq \epsilon$. Therefore, LHS - RHS $\geq (1-\epsilon)\epsilon - \epsilon = -\epsilon^2$. Now,

as Inequality (8) is obtained from Inequality (5) after some algebraic manipulation, it must hold that (LHS - RHS) of Inequality (8) = (LHS - RHS) of Inequality (5). Hence, we get $\delta(\mathbf{x}_i, \mathbf{x}_j) + \delta(\mathbf{x}_j, \mathbf{x}_k) - \delta(\mathbf{x}_k, \mathbf{x}_i) \geq -\epsilon^2$ which can be simplified to obtain Inequality (9). This completes the proof.

$\square$

Let us now elucidate the proposed FWP (and consequently the proposed FWPD measure) by using the following example.

Example 1 Let $X \subset \mathbb{R}^3$ be a dataset consisting of $n = 5$ data points, each having three features ($S = \{1, 2, 3\}$), some of which (marked by '*') are unobserved. The dataset is presented below (along with the feature observation counts and the observed feature sets for each of the instances).

Data Point	$x_{i,1}$	$x_{i,2}$	$x_{i,3}$	$\gamma_{\mathbf{x}_i}$
$\mathbf{x}_1$	*	3	2	$\{2, 3\}$
$\mathbf{x}_2$	1.2	*	4	$\{1, 3\}$
$\mathbf{x}_3$	*	0	0.5	$\{2, 3\}$
$\mathbf{x}_4$	2.1	3	1	$\{1, 2, 3\}$
$\mathbf{x}_5$	-2	*	*	$\{1\}$
Obs. Count	$w_1 = 3$	$w_2 = 3$	$w_3 = 4$	–

The pairwise observed distance matrix A_d and the pairwise penalty matrix A_p, are as follows:

$$A_d = \begin{bmatrix} 0, & 2, & 3.35, & 1, & 0 \\ 2, & 0, & 3.5, & 3.13, & 3.2 \\ 3.35, & 3.5, & 0, & 3.04, & 0 \\ 1, & 3.13, & 3.04, & 0, & 4.1 \\ 0, & 3.2, & 0, & 4.1, & 0 \end{bmatrix} \text{ and } A_p = \begin{bmatrix} 0.3, & 0.6, & 0.3, & 0.3, & 1 \\ 0.6, & 0.3, & 0.6, & 0.3, & 0.7 \\ 0.3, & 0.6, & 0.3, & 0.3, & 1 \\ 0.3, & 0.3, & 0.3, & 0, & 0.7 \\ 1, & 0.7, & 1, & 0.7, & 0.7 \end{bmatrix}.$$

From A_d it is observed that the maximum pairwise observed distance $d_{max} = 4.1$. Then, the normalized observed distance matrix $A_{\bar{d}}$ is

$$A_{\bar{d}} = \begin{bmatrix} 0, & 0.49, & 0.82, & 0.24, & 0 \\ 0.49, & 0, & 0.85, & 0.76, & 0.78 \\ 0.82, & 0.85, & 0, & 0.74, & 0 \\ 0.24, & 0.76, & 0.74, & 0, & 1 \\ 0, & 0.78, & 0, & 1, & 0 \end{bmatrix}.$$

$\mathscr{P} = 0.3$. While it is not necessary, let us choose $\alpha = 0.7$. Using Eq. (3) to calculate the FWPD matrix A_δ, we get:

$$A_\delta = 0.3 \times A_{\bar{d}} + 0.7 \times A_p = \begin{bmatrix} 0.21, & 1.02, & 1.22, & 0.51, & 0.7 \\ 1.02, & 0.21, & 1.47, & 1.15, & 1.45 \\ 1.22, & 1.47, & 0.21, & 1.12, & 0.7 \\ 0.51, & 1.15, & 1.12, & 0, & 1.72 \\ 0.7, & 1.45, & 0.7, & 1.72, & 0.49 \end{bmatrix}.$$

It should be noted that in keeping with the properties of the FWPD described in Sect. 2.1, A_δ is a symmetric matrix with the diagonal elements being the smallest entries in their corresponding rows (and columns) and the diagonal element corresponding to the fully observed point $\mathbf{x}_4$ being the only zero element. Moreover, it can be easily checked that the relaxed form of the triangle inequality, as given in Inequality (9), is always satisfied.

3 k-means clustering for datasets with missing features using the proposed FWPD

This section presents a reformulation of the k-means clustering problem for datasets with missing features, using the FWPD measure proposed in Sect. 2. The important notations used in this section (and beyond) are summarized in Table 2. The k-means problem (a term coined by MacQueen (1967)) deals with the partitioning of a set of n data instances into $k(< n)$ clusters so as to minimize the sum of within-cluster dissimilarities. The standard heuristic algorithm to solve the k-means problem, referred to as the *k-means algorithm*, was first proposed by Lloyd in 1957 (Lloyd 1982), and rediscovered by Forgy (1965). Starting with random assignments of each of the data instances to one of the k clusters, the k-means

Table 2 Some important notations used in Sect. 3 and beyond

Notation	Counter-part in k-means-FWPD iteration t	Meaning
k	–	Number of clusters for k-means
C_j	C_j^t	j-th cluster for k-means
$u_{i,j}$	$u_{i,j}^t$	Membership of the data point $\mathbf{x}_i$ in the cluster C_j
U	U^t	$n \times k$ matrix of cluster memberships
$\mathscr{U}$	–	Set of all possible U values
$\mathbf{z}_j$	$\mathbf{z}_j^t$	Centroid of cluster C_j
$z_{j,l}$	$z_{j,l}^t$	l-th feature of the cluster centroid $\mathbf{z}_j$
Z	Z^t	Set of cluster centroids
$\mathscr{Z}$	–	Set of all possible Z values
$f(U, Z)$	$f(U^t, Z^t)$	k-means objective function defined on $\mathscr{U} \times \mathscr{Z}$
X_l	–	Set of all $\mathbf{x}_i \in X$ having observed values for feature l
U^*	–	Final cluster memberships found by k-means-FWPD
Z^*	–	Final cluster centroids found by k-means-FWPD
T	–	The convergent iteration of k-means-FWPD
–	τ	Any iteration preceding the current iteration t
$\mathscr{F}(Z)$	–	Set of feasible membership matrices for Z
$\mathscr{F}(U)$	–	Set of feasible centroid sets for U
$\mathscr{S}(U)$	–	Set of super-feasible centroids sets for U
$(\tilde{U}, \tilde{Z})$	–	A partial optimal solution of the k-means-FWPD problem
D	–	A feasible direction of movement for U^*
$\mathscr{O}$	–	Big O notation

algorithm functions by iteratively recalculating the k cluster centroids and reassigning the data instances to the nearest cluster (the cluster corresponding to the nearest cluster centroid), in an alternating manner. Selim and Ismail (1984) showed that the k-means algorithm converges to a local optimum of the non-convex optimization problem posed by the k-means problem, when the dissimilarity used is the Euclidean distance between data points.

The proposed formulation of the k-means problem for datasets with missing features using the proposed FWPD measure, referred to as the *k-means-FWPD problem* hereafter, differs from the standard k-means problem not only in that the underlying dissimilarity measure used is FWPD (instead of Euclidean distance), but also in the addition of a new constraint which ensures that a cluster centroid has observable values for exactly those features which are observed for at least one of the points in its corresponding cluster. Therefore, the k-means-FWPD problem to partition the dataset X into k clusters ($2 \le k < n$), can be formulated in the following way:

$$\text{P: minimize } f(U, Z) = \sum_{i=1}^{n} \sum_{j=1}^{k} u_{i,j} ((1 - \alpha) \times \frac{d(\mathbf{x}_i, \mathbf{z}_j)}{d_{max}} + \alpha \times p(\mathbf{x}_i, \mathbf{z}_j)), \quad (10a)$$

$$\text{subject to } \sum_{j=1}^{k} u_{i,j} = 1 \ \forall \ i \in \{1, 2, \cdots, n\}, \quad (10b)$$

$$u_{i,j} \in \{0, 1\} \ \forall \ i \in \{1, 2, \cdots, n\}, j \in \{1, 2, \cdots, k\}, \quad (10c)$$

$$\text{and } \gamma_{\mathbf{z}_j} = \bigcup_{\mathbf{x}_i \in C_j} \gamma_{\mathbf{x}_i} \ \forall \ j \in \{1, 2, \cdots, k\}, \quad (10d)$$

where $U = [u_{i,j}]$ is the $n \times k$ real matrix of memberships, d_{max} denotes the maximum observed distance between any two data points $\mathbf{x}_i, \mathbf{x}_i \in X$, $\gamma_{\mathbf{z}_j}$ denotes the set of observed features for $\mathbf{z}_j$ ($j \in \{1, 2, \cdots, k\}$), C_j denotes the j-th cluster (corresponding to the centroid $\mathbf{z}_j$), $Z = \{\mathbf{z}_1, \cdots, \mathbf{z}_k\}$, and it is said that $\mathbf{x}_i \in C_j$ when $u_{i,j} = 1$.

3.1 The k-means-FWPD algorithm

To find a solution to the problem P, which is a non-convex program, we propose a Lloyd's heuristic-like algorithm based on the FWPD (referred to as *k-means-FWPD algorithm*), as follows:

1. Start with a random initial set of cluster assignments U such that $\sum_{j=1}^{k} u_{i,j} = 1$. Set $t = 1$ and specify the maximum number of iterations $MaxIter$.
2. For each cluster C_j^t ($j = 1, 2, \cdots, k$), calculate the observed features of the cluster centroid $\mathbf{z}_j^t$. The value for the l-th feature of a centroid $\mathbf{z}_j^t$ should be the average of the corresponding feature values for all the data instances in the cluster C_j^t having observed values for the l-th feature. If none of the data instances in C_j^t have observed values for the feature in question, then the value $z_{j,l}^{t-1}$ of the feature from the previous iteration should be retained. Therefore, the feature values are calculated as follows:

$$z_{j,l}^t = \begin{cases} \left(\sum_{\mathbf{x}_i \in X_l} u_{i,j}^t \times x_{i,l} \right) \Big/ \left(\sum_{\mathbf{x}_i \in X_l} u_{i,j}^t \right), & \forall \ l \in \bigcup_{\mathbf{x}_i \in C_j^t} \gamma_{\mathbf{x}_i}, \\ z_{j,l}^{t-1}, & \forall \ l \in \gamma_{\mathbf{z}_j^{t-1}} \setminus \bigcup_{\mathbf{x}_i \in C_j^t} \gamma_{\mathbf{x}_i}, \end{cases} \quad (11)$$

where X_l denotes the set of all $\mathbf{x}_i \in X$ having observed values for the feature l.
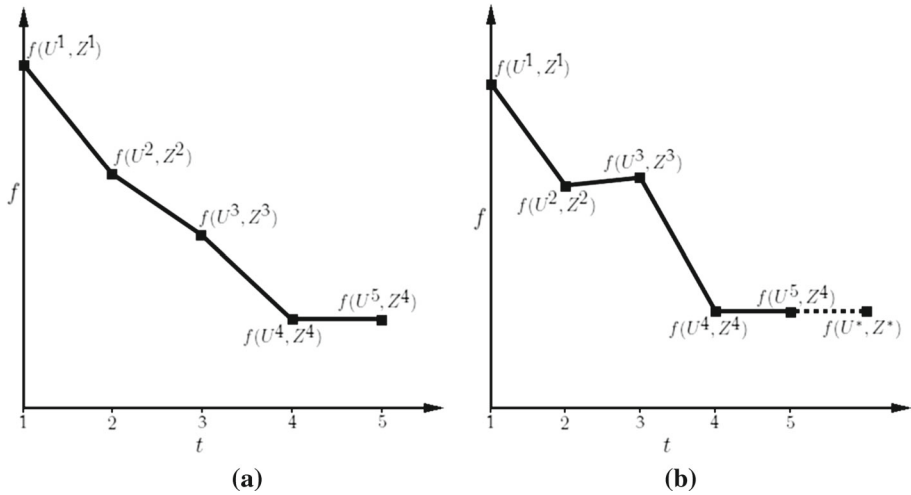
Fig. 2 Comparison of convergences of traditional k-means and k-means-FWPD algorithms. **a** Traditional k-means. **b** k-means-FWPD

3. Assign each data point $\mathbf{x}_i$ ($i = 1, 2, \cdots, n$) to the cluster corresponding to its nearest (in terms of FWPD) centroid, i.e.

$$u_{i,j}^{t+1} = \begin{cases} 1, & \text{if } \mathbf{z}_j^t = \arg\min_{\mathbf{z} \in Z^t} \delta(\mathbf{x}_i, \mathbf{z}), \\ 0, & \text{otherwise.} \end{cases}$$

Set $t = t + 1$. If $U^t = U^{t-1}$ or $t = MaxIter$, then go to Step 4; otherwise go to Step 2.

4. Calculate the final cluster centroid set Z^* as:

$$z_{j,l}^* = \frac{\sum_{\mathbf{x}_i \in X_l} u_{i,j}^{t+1} \times x_{i,l}}{\sum_{\mathbf{x}_i \in X_l} u_{i,j}^{t+1}} \ \forall l \in \bigcup_{\mathbf{x}_i \in C_j^{t+1}} \gamma_{\mathbf{x}_i}. \tag{12}$$

Set $U^* = U^{t+1}$.

Remark 1 The iterations of the traditional k-means algorithm are known to each result in a decrease in the value of the objective function f (Selim and Ismail 1984) (Fig. 2a). However, for the k-means-FWPD algorithm, the Z^t calculations for some of the iterations may result in a finite increase in f, as shown in Fig. 2b. We show in Theorem 3 that only a finite number of such increments may occur during a given run of the algorithm, thus ushering in ultimate convergence. Moreover, the final feasible, locally-optimal solution is obtained using Step 4 (denoted by dotted line) which does not result in any further change to the objective function value.

3.2 Notions of feasibility in problem P

Let $\mathscr{U}$ and $\mathscr{Z}$ respectively denote the sets of all possible U and Z. Unlike the traditional k-means problem, the entire $\mathscr{U} \times \mathscr{Z}$ space is not feasible for the Problem P. There exists a set of feasible U for a given Z. Similarly, there exist sets of feasible and super-feasible Z (a super-set of the set of feasible Z) for a given U. In this subsection, we formally define these notions.

Definition 6 Given a cluster centroid set Z, the set $\mathscr{F}(Z)$ of feasible membership matrices is given by

$$\mathscr{F}(\hat{Z}) = \{U : u_{i,j} = 0 \ \forall j \in \{1, 2, \cdots, k\} \text{ such that } \gamma_{\mathbf{z}_j} \subset \gamma_{\mathbf{x}_i}\},$$

i.e. $\mathscr{F}(Z)$ is the set of all such membership matrices which do not assign any $\mathbf{x}_i \in X$ to a centroid in Z missing some feature $l \in \gamma_{\mathbf{x}_i}$.

Definition 7 Given a membership matrix U, the set $\mathscr{F}(U)$ of feasible cluster centroid sets can be defined as

$$\mathscr{F}(U) = \{Z : Z \text{ satisfies constraint (10d)}\}.$$

Definition 8 Given a membership matrix U, the set $\mathscr{S}(U)$ of super-feasible cluster centroid sets is

$$\mathscr{S}(U) = \{Z : \gamma_{\mathbf{z}_j} \supseteq \bigcup_{\mathbf{x}_i \in C_j} \gamma_{\mathbf{x}_i} \ \forall j \in \{1, 2, \cdots, k\}\}, \tag{13}$$

i.e. $\mathscr{S}(U)$ is the set of all such centroid sets which ensure that any centroid has observed values at least for those features which are observed for any of the points assigned to its corresponding cluster in U.

Remark 2 The k-means-FWPD problem differs from traditional k-means in that not all $U \in \mathscr{U}$ are feasible for a given Z. Additionally, for a given U, there exists a set $\mathscr{S}(U)$ of super-feasible Z; $\mathscr{F}(U)$ a subset of $\mathscr{S}(U)$ being the set of feasible Z. The traversal of the k-means-FWPD algorithm is illustrated in Fig. 3 where the grey solid straight lines denote the
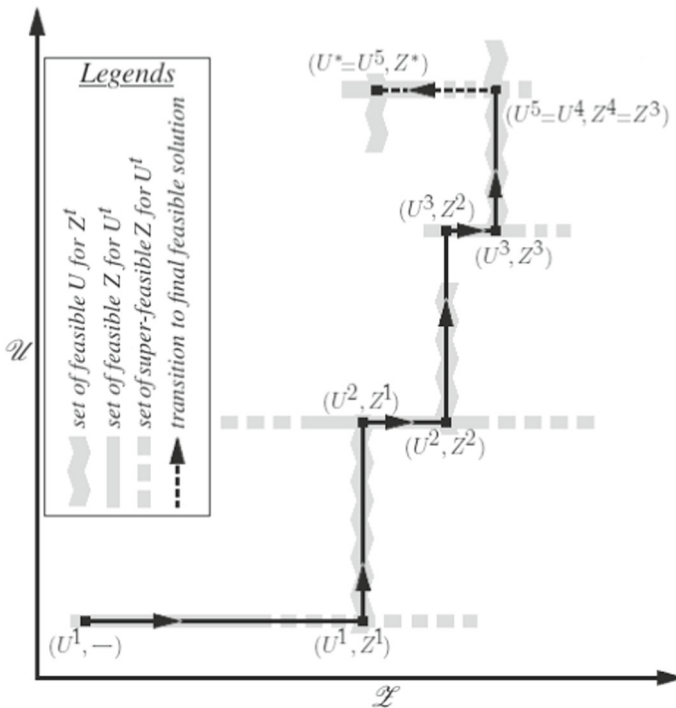

Fig. 3 Simplified representation of how the k-means-FWPD algorithm traverses the $\mathscr{U} \times \mathscr{Z}$ space ($\mathscr{U}$ and $\mathscr{Z}$ are shown to be unidimensional for the sake of visualizability)

set of feasible Z for the current U^t while the rest of the super-feasible region is marked by the corresponding grey dotted straight line. Furthermore, the grey jagged lines denote the feasible set of U for the current Z^t. Starting with a random $U^1 \in \mathscr{U}$ (Step 1), the algorithm finds $Z^1 \in \mathscr{S}(U^1)$ (Step 2), $U^2 \in \mathscr{F}(Z^1)$ (Step 3), and $Z^2 \in \mathscr{F}(U^2)$ (Step 2). However, it subsequently finds $U^3 \notin \mathscr{F}(Z^2)$ (Step 3), necessitating a feasibility adjustment (see Sect. 3.4) while calculating Z^3 (Step 2). Subsequently, the algorithm converges to (U^5, Z^4). For the convergent (U^{T+1}, Z^T), $U^{T+1} \in \mathscr{F}(Z^T)$ but it is possible that $Z^T \in \mathscr{S}(U^{T+1}) \backslash \mathscr{F}(U^T)$ (as seen in the case of Fig. 3). However, the final (U^*, Z^*) (obtained by the dotted black line transition denoting Step 4) is seen to be feasible in both respects and is shown (in Theorem 5) to be locally-optimal in the corresponding feasible region.

3.3 Partial optimal solutions

This subsection deals with the concept of *partial optimal solutions* of the problem P, to one of which the k-means-FWPD algorithm is shown to converge (prior to Step 4). The following definition formally presents the concept of a partial optimal solution.

Definition 9 A partial optimal solution $(\tilde{U}, \tilde{Z})$ of problem P, satisfies the following conditions (Wendel and Hurter Jr 1976):

$$f(\tilde{U}, \tilde{Z}) \leq f(U, \tilde{Z}) \ \forall \ U \in \mathscr{F}(\tilde{Z}) \text{ where } \tilde{U} \in \mathscr{F}(\tilde{Z}),$$
$$\text{and } f(\tilde{U}, \tilde{Z}) \leq f(\tilde{U}, Z) \ \forall \ Z \in \mathscr{S}(\tilde{U}) \text{ where } \tilde{Z} \in \mathscr{S}(\tilde{U}).$$

To obtain a partial optimal solution of P, the two following subproblems are defined:

> P1: Given $U \in \mathscr{U}$, minimize $f(U, Z)$ over $Z \in \mathscr{S}(U)$.

> P2: Given Z satisfying (10d), minimize $f(U, Z)$ over $U \in \mathscr{U}$.

The following lemmas establish that Steps 2 and 3 of the k-means-FWPD algorithm respectively solve the problems P1 and P2 for a given iterate. The subsequent theorem shows that the k-means-FWPD algorithm converges to a partial optimal solution of P.

Lemma 5 *Given a U^t, the centroid matrix Z^t calculated using Eq. (11) is an optimal solution of the Problem P1.*

Proof For a fixed $U^t \in \mathscr{U}$, the objective function is minimized when $\frac{\partial f}{\partial z_{j,l}^t} = 0 \ \forall j \in \{1, \cdots, k\}, l \in \gamma_{\mathbf{z}_j^t}$. For a particular $\mathbf{z}_j^t$, it follows from Definition 3 that $\{p(\mathbf{x}_i, \mathbf{z}_j^t) : \mathbf{x}_i \in C_j^t\}$ is independent of the values of the features of $\mathbf{z}_j^t$, as $\gamma_{\mathbf{x}_i} \bigcap \gamma_{\mathbf{z}_j^t} = \gamma_{\mathbf{x}_i} \ \forall \mathbf{x}_i \in C_j^t$. Since an observed feature $l \in \gamma_{\mathbf{z}_j^t} \backslash (\bigcup_{\mathbf{x}_i \in C_j^t} \gamma_{\mathbf{x}_i})$ of $\mathbf{z}_j^t$ has no contribution to the observed distances, $\frac{\partial f}{\partial z_{j,l}^t} = 0 \ \forall l \in \gamma_{\mathbf{z}_j^t} \backslash (\bigcup_{\mathbf{x}_i \in C_j^t} \gamma_{\mathbf{x}_i})$. For an observed feature $l \in \bigcup_{\mathbf{x}_i \in C_j^t} \gamma_{\mathbf{x}_i}$ of $\mathbf{z}_j^t$, differentiating $f(U^t, Z^t)$ w.r.t. $z_{j,l}^t$ we get

$$\frac{\partial f}{\partial z_{j,l}^t} = \frac{(1-\alpha)}{d_{max}} \times \sum_{\mathbf{x}_i \in X_l} u_{i,j}^t \left(\frac{x_{i,l} - z_{j,l}^t}{d(\mathbf{x}_i, \mathbf{z}_j^t)} \right).$$

Setting $\frac{\partial f}{\partial z_{j,l}^t} = 0$ and solving for $z_{j,l}^t$, we obtain

$$z_{j,l}^t = \frac{\sum_{\mathbf{x}_i \in X_l} u_{i,j}^t \times x_{i,l}}{\sum_{\mathbf{x}_i \in X_l} u_{i,j}^t}.$$

Since Eq. (11) is in keeping with this criterion and ensures that constraint (13) is satisfied, the centroid matrix Z^t calculated using Eq. (11) is an optimal solution of P1. $\qquad\square$

Lemma 6 *For a given Z^t, problem P2 is solved if $u_{i,j}^{t+1} = 1$ and $u_{i,j'}^{t+1} = 0 \ \forall \ i \in \{1, \cdots, n\}$ when $\delta(\mathbf{x}_i, \mathbf{z}_j^t) \leq \delta(\mathbf{x}_i, \mathbf{z}_{j'}^t)$, for all $j' \neq j$.*

Proof It is clear that the contribution of $\mathbf{x}_i$ to the total objective function is $\delta(\mathbf{x}_i, \mathbf{z}_j^t)$ when $u_{i,j}^{t+1} = 1$ and $u_{i,j'}^{t+1} = 0 \ \forall \ j' \neq j$. Since any alternative solution is an extreme point of $\mathscr{U}$ (Selim and Ismail 1984), it must satisfy (10c). Therefore, the contribution of $\mathbf{x}_i$ to the objective function for an alternative solution will be some $\delta(\mathbf{x}_i, \mathbf{z}_{j'}^t) \geq \delta(\mathbf{x}_i, \mathbf{z}_j^t)$. Hence, the contribution of $\mathbf{x}_i$ is minimized by assigning $u_{i,j}^{t+1} = 1$ and $u_{i,j'}^{t+1} = 0 \ \forall \ j' \neq j$. This argument holds true for all $\mathbf{x}_i \in X$, i.e. $\forall \ i \in \{1, \cdots, n\}$. This completes the proof. $\qquad\square$

Theorem 2 *The k-means-FWPD algorithm finds a partial optimal solution of P.*

Proof Let T denote the terminal iteration. Since Step 2 and Step 3 of the k-means-FWPD algorithm respectively solve P1 and P2, the algorithm terminates only when the obtained iterate (U^{T+1}, Z^T) solves both P1 and P2. Therefore, $f(U^{T+1}, Z^T) \leq f(U^{T+1}, Z) \ \forall Z \in \mathscr{S}(U^{T+1})$. Since Step 2 ensures that $Z^T \in \mathscr{S}(U^T)$ and $U^{T+1} = U^T$, we must have $Z^T \in \mathscr{S}(U^{T+1})$. Moreover, $f(U^{T+1}, Z^T) \leq f(U, Z^T) \ \forall U \in \mathscr{U}$ which implies $f(U^{T+1}, Z^T) \leq f(U, Z^T) \ \forall U \in \mathscr{F}(Z^T)$. Now, Step 2 ensures that $\gamma_{\mathbf{z}_j^T} \supseteq \bigcup_{\mathbf{x}_i \in C_j^T} \gamma_{\mathbf{x}_i} \ \forall \ j \in \{1, 2, \cdots, k\}$. Since we must have $U^{T+1} = U^T$ for convergence to occur, it follows that $\gamma_{\mathbf{z}_j^T} \supseteq \bigcup_{\mathbf{x}_i \in C_j^{T+1}} \gamma_{\mathbf{x}_i} \ \forall \ j \in \{1, 2, \cdots, k\}$, hence $u_{i,j}^{T+1} = 1$ implies $\gamma_{\mathbf{z}_j^T} \supseteq \gamma_{\mathbf{x}_i}$. Therefore, $U^{T+1} \in \mathscr{F}(Z^T)$. Consequently, the terminal iterate of Step 3 of the k-means-FWPD algorithm must be a partial optimal solution of P. $\qquad\square$

3.4 Feasibility adjustments

Since it is possible for the number of observed features of the cluster centroids to increase over the iterations to maintain feasibility w.r.t. constraint (10d), we now introduce the concept of *feasibility adjustment*, the consequences of which are discussed in this subsection.

Definition 10 A feasibility adjustment for cluster j ($j \in \{1, 2, \cdots, k\}$) is said to occur in iteration t if $\gamma_{\mathbf{z}_j^t} \supset \gamma_{\mathbf{z}_j^{t-1}}$ or $\gamma_{\mathbf{z}_j^t} \backslash \gamma_{\mathbf{z}_j^{t-1}} \neq \phi$, i.e. if the centroid $\mathbf{z}_j^t$ acquires an observed value for at least one feature which was unobserved for its counter-part $\mathbf{z}_j^{t-1}$ in the previous iteration.

The following lemma shows that feasibility adjustment can only occur for a cluster as a result of the addition of a new data point previously unassigned to it.

Lemma 7 *Feasibility adjustment occurs for a cluster C_j in iteration t iff at least one data point $\mathbf{x}_i$, such that $\gamma_{\mathbf{x}_i} \backslash \gamma_{\mathbf{z}_j^\tau} \neq \phi \ \forall \tau < t$, which was previously unassigned to C_j (i.e. $u_{i,j}^\tau = 0$ $\forall \tau < t$) is assigned to it in iteration t.*

Proof Due to Eq. (11), all features defined for $\mathbf{z}_j^{t-1}$ are also retained for $\mathbf{z}_j^t$. Therefore, for $\gamma_{\mathbf{z}_j^t} \backslash \gamma_{\mathbf{z}_j^{t-1}} \neq \phi$ there must exist some $\mathbf{x}_i$ such that $u_{i,j}^t = 1$, $u_{i,j}^{t-1} = 0$, and $\gamma_{\mathbf{x}_i} \backslash \gamma_{\mathbf{z}_j^{t-1}} \neq \phi$. Since the set of defined features for any cluster centroid is a monotonically growing set, we have $\gamma_{\mathbf{x}_i} \backslash \gamma_{\mathbf{z}_j^\tau} \neq \phi \ \forall \tau < t$. It then follows from constraint (10d) that $u_{i,j}^\tau = 0 \ \forall \tau < t$. Now, to

prove the converse, let us assume the existence of some $\mathbf{x}_i$ such that $\gamma_{\mathbf{x}_i} \backslash \gamma_{\mathbf{z}_j^\tau} \neq \phi \ \forall \tau < t$ and $u_{i,j}^\tau = 0 \ \forall \tau < t$. Since $\gamma_{\mathbf{x}_i} \backslash \gamma_{\mathbf{z}_j^{t-1}} \neq \phi$ and $\gamma_{\mathbf{z}_j^t} \supseteq \gamma_{\mathbf{x}_i} \bigcup \gamma_{\mathbf{z}_j^{t-1}}$, it follows that $\gamma_{\mathbf{z}_j^t} \backslash \gamma_{\mathbf{z}_j^{t-1}} \neq \phi$. □

The following theorem deals with the consequences of the feasibility adjustment phenomenon.

Theorem 3 *For a finite number of iterations during a single run of the k-means-FWPD algorithm, there may be a finite increment in the objective function f, due to the occurrence of feasibility adjustments.*

Proof It follows from Lemma 5 that $f(U^t, Z^t) \leq f(U^t, Z) \ \forall Z \in \mathscr{S}(U^t)$. If there is no feasibility adjustment in iteration t, $\mathscr{S}(U^{t-1}) = \mathscr{S}(U^t)$. Hence, $f(U^t, Z^t) \leq f(U^t, Z^{t-1})$. However, if a feasibility adjustment occurs in iteration t, then $\gamma_{\mathbf{z}_j^t} \subset \gamma_{\mathbf{z}_j^{t-1}}$ for at least one $j \in \{1, 2, \cdots, k\}$. Hence, $Z^{t-1} \in \mathscr{Z} \backslash \mathscr{S}(U^t)$ and we may have $f(U^t, Z^t) > f(U^t, Z^{t-1})$. Since both $f(U^t, Z^t)$ and $f(U^t, Z^{t-1})$ are finite, $(f(U^t, Z^t) - f(U^t, Z^{t-1}))$ must also be finite. Now, the maximum number of feasibility adjustments occur in the worst case scenario where each data point, having an unique set of observed features (which are unobserved for all other data points), traverses all the clusters before convergence. Therefore, the maximum number of possible feasibility adjustments during a single run of the k-means-FWPD algorithm is $n(k-1)$, which is finite. □

3.5 Convergence of the k-means-FWPD algorithm

We now show that the k-means-FWPD algorithm converges to the partial optimal solution, within a finite number of iterations. The following lemma and the subsequent theorem are concerned with this.

Lemma 8 *Starting with a given iterate (U^t, Z^t), the k-means-FWPD algorithm either reaches convergence or encounters a feasibility adjustment, within a finite number of iterations.*

Proof Let us first note that there are a finite number of extreme points of $\mathscr{U}$. Then, we observe that an extreme point of $\mathscr{U}$ is visited at most once by the algorithm before either convergence or the next feasibility adjustment. Suppose, this is not true, and let $U^{t_1} = U^{t_2}$ for distinct iterations t_1 and t_2 ($t_1 \geq t, t_1 < t_2$) of the algorithm. Applying Step 2 of the algorithm, we get Z^{t_1} and Z^{t_2} as optimal centroid sets for U^{t_1} and U^{t_2}, respectively. Then, $f(U^{t_1}, Z^{t_1}) = f(U^{t_2}, Z^{t_2})$ since $U^{t_1} = U^{t_2}$. However, it is clear from Lemmas 5, 6 and Theorem 3 that f strictly decreases subsequent to the iterate (U^t, Z^t) and prior to either the next feasibility adjustment (in which case the value of f may increase) or convergence (in which case f remains unchanged). Hence, $U^{t_1} \neq U^{t_2}$. Therefore, it is clear from the above argument that the k-means-FWPD algorithm either converges or encounters a feasibility adjustment within a finite number of iterations. □

Theorem 4 *The k-means-FWPD algorithm converges to a partial optimal solution within a finite number of iterations.*

Proof It follows from Lemma 8 that the first feasibility adjustment is encountered within a finite number of iterations since initialization and that each subsequent feasibility adjustment occurs within a finite number of iterations of the previous. Moreover, we know from Theorem 3 that there can only be a finite number of feasibility adjustments during a single run of

the algorithm. Therefore, the final feasibility adjustment must occur within a finite number of iterations. Moreover, it follows from Lemma 8 that the algorithm converges within a finite number of subsequent iterations. Hence, the k-means-FWPD algorithm must converge within a finite number of iterations. □

3.6 Local optimality of the final solution

In this subsection, we establish the local optimality of the final solution obtained in Step 4 of the k-means-FWPD algorithm, subsequent to convergence in Step 3.

Lemma 9 Z^* *is the unique optimal feasible cluster centroid set for* U^*, *i.e.* $Z^* \in \mathscr{F}(U^*)$ *and* $f(U^*, Z^*) \leq f(U^*, Z) \, \forall Z \in \mathscr{F}(U^*)$.

Proof Since Z^* satisfies constraint (10d) for U^*, $Z^* \in \mathscr{F}(U^*)$. We know from Lemma 5 that for $f(U^*, Z^*) \leq f(U^*, Z) \, \forall Z \in \mathscr{F}(U^*)$, we must have

$$z_{j,l}^* = \frac{\sum_{\mathbf{x}_i \in X_l} u_{i,j}^* \times x_{i,l}}{\sum_{\mathbf{x}_i \in X_l} u_{i,j}^*}.$$

As this is ensured by Step 4, Z^* must be the unique optimal feasible cluster centroid set for U^*. □

Lemma 10 *If* Z^* *is the unique optimal feasible cluster centroid set for* U^*, *then* $f(U^*, Z^*) \leq f(U, Z^*) \, \forall U \in \mathscr{F}(Z^*)$.

Proof We know from Theorem 2 that $f(U^*, Z^T) \leq f(U, Z^T) \, \forall U \in \mathscr{F}(Z^T)$. Now, $\gamma_{\mathbf{z}_j^*} \subseteq \gamma_{\mathbf{z}_j^T} \, \forall j \in \{1, 2, \cdots, k\}$. Therefore, $\mathscr{F}(Z^*) \subseteq \mathscr{F}(Z^T)$ must hold. It therefore follows that $f(U^*, Z^*) \leq f(U, Z^*) \, \forall U \in \mathscr{F}(Z^*)$. □

Now, the following theorem shows that the final solution obtained by Step 4 of the k-means-FWPD algorithm is locally optimal.

Theorem 5 *The final solution* (U^*, Z^*) *obtained by Step 4 of the k-means-FWPD algorithm is a local optimal solution of P.*

Proof We have from Lemma 10 that $f(U^*, Z^*) \leq f(U, Z^*) \, \forall U \in \mathscr{F}(Z^*)$. Therefore, $f(U^*, Z^*) \leq \min_U \{f(U, Z^*) : U \in \mathscr{F}(Z^*)\}$ which implies that for all feasible directions D at U^*, the one-sided directional derivative (Lasdon 2013),

$$trace(\nabla_U f(U^*, Z)^\mathrm{T} D) \geq 0. \tag{14}$$

Now, since Z^* is the unique optimal feasible cluster centroid set for U^* (Lemma 9), (U^*, Z^*) is a local optimum of problem P. □

3.7 Time complexity of the k-means-FWPD algorithm

In this subsection, we present a brief discussion on the time complexity of the k-means-FWPD algorithm. The k-means-FWPD algorithm consists of four basic steps, which are repeated iteratively. These steps are

1. *Centroid Calculation*: As a maximum of m features of each centroid must be calculated, the complexity of centroid calculation is at most $\mathscr{O}(kmn)$.

Table 3 Some important notations used in Sect. 4 and beyond

Notation	Meaning
B^t	Set of hierarchical clusters obtained in iteration t of HAC-FWPD
β_i^t	i-th hierarchical cluster in B^t
Q^t	Matrix of dissimilarities between the hierarchical clusters in B^t
$q^{(i,j)}$	(i,j)-th element of Q^t
q_{min}^t	Smallest non-zero value in Q^t
M	List of location in Q^t having value q_{min}^t
G	New hierarchical cluster formed by merging two of the closest hierarchical clusters in B^t
i_G	Location of G in the set B^{t+1}
$L(G, \beta)$	Linkage between two hierarchical clusters G and β

2. *Distance Calculation*: As each distance calculation involves at most m features, the observed distance calculation between n data instances and k cluster centroids is at most $\mathcal{O}(kmn)$.
3. *Penalty Calculation*: The penalty calculation between a data point and a cluster centroid involves at most m summations. Hence, penalty calculation over all possible pairings is at most $\mathcal{O}(kmn)$.
4. *Cluster Assignment*: The assignment of n data points to k clusters consists of the comparisons of the dissimilarities of each point with k clusters, which is $\mathcal{O}(nk)$.

Therefore, if the algorithm runs for T iterations, the total computational complexity is $\mathcal{O}(kmnT)$ which is the same as that of the standard k-means algorithm.

4 Hierarchical agglomerative clustering for datasets with missing features using the proposed FWPD

In this section we present HAC clustering methods using the proposed FWPD measure that can be directly applied to data with missing features. The important notations used in this section (and beyond) are summarized in Table 3.

Hierarchical agglomerative schemes for data clustering seek to build a multi-level hierarchy of clusters, starting with each data point as a single cluster, by combining two (or more) of the most proximal clusters at one level to obtain a lower number of clusters at the next (higher) level. A survey of HAC methods can be found in Murtagh and Contreras (2012). However, these methods cannot be directly applied to datasets with missing features. Therefore, in this section, we develop variants of HAC methods, based on the proposed FWPD measure. Various proximity measures may be used to merge the clusters in an agglomerative clustering method. Modifications of the three most popular of such proximity measures [Single Linkage (SL), Complete Linkage (CL) and Average Linkage (AL)] so as to have FWPD as the underlying dissimilarity measure, are as follows:

1. *Single Linkage with FWPD (SL-FWPD)*: The SL between two clusters C_i and C_j is $\min\{\delta(\mathbf{x}_i, \mathbf{x}_j) : \mathbf{x}_i \in C_i, \mathbf{x}_j \in C_j\}$.
2. *Complete Linkage with FWPD (CL-FWPD)*: The CL between two clusters C_i and C_j is $\max\{\delta(\mathbf{x}_i, \mathbf{x}_j) : \mathbf{x}_i \in C_i, \mathbf{x}_j \in C_j\}$.

3. *Average Linkage with FWPD (AL-FWPD)*: $\frac{1}{|C_i|\times|C_j|}\sum_{\mathbf{x}_i\in C_i}\sum_{\mathbf{x}_j\in C_j}\delta(\mathbf{x}_i,\mathbf{x}_j)$ is the AL between two clusters C_i and C_j, where $|C_i|$ and $|C_j|$ are respectively the number of instances in the clusters C_i and C_j.

4.1 The HAC-FWPD algorithm

To achieve hierarchical clusterings in the presence of unstructured missingness, the HAC method based on SL-FWPD, CL-FWPD, or AL-FWPD (referred to as the *HAC-FWPD algorithm* hereafter) is as follows:

1. Set $B^0 = X$. Compute pairwise dissimilarities $\delta(\mathbf{x}_i,\mathbf{x}_j)$, $\forall\,\mathbf{x}_i,\mathbf{x}_j \in X$ and construct the dissimilarity matrix Q^0 so that $q^0(i,j) = \delta(\mathbf{x}_i,\mathbf{x}_j)$. Set $t = 0$.
2. Search Q^t to identify the set $M = \{(i_1,j_1),(i_2,j_2),\cdots,(i_k,j_k)\}$ containing all the pairs of indexes such that $q^t(i_r,j_r) = q^t_{min}$ $\forall\,r \in \{1,2,\cdots,k\}$, q^t_{min} being the smallest non-zero element in Q^t.
3. Merge the elements corresponding to any one pair in M, say $\beta^t_{i_r}$ and $\beta^t_{j_r}$ corresponding to the pair (i_r,j_r), into a single group $G = \{\beta^t_{i_r},\beta^t_{j_r}\}$. Construct B^{t+1} by removing $\beta^t_{i_r}$ and $\beta^t_{j_r}$ from B^t and inserting G.
4. Define Q^{t+1} on $B^{t+1} \times B^{t+1}$ as $q^{t+1}(i,j) = q^t(i,j)$ $\forall\,i,j$ such that $\beta^t_i,\beta^t_j \neq G$ and $q^{t+1}(i,i_G) = q^{t+1}(i_G,i) = L(G,\beta^t_i)$, where i_G denotes the location of G in B^{t+1} and

$$L(G,\beta) = \begin{cases} \min\limits_{\mathbf{x}_i\in G,\mathbf{x}_j\in\beta}\delta(\mathbf{x}_i,\mathbf{x}_j) & \text{for SL-FWPD,} \\ \max\limits_{\mathbf{x}_i\in G,\mathbf{x}_j\in\beta}\delta(\mathbf{x}_i,\mathbf{x}_j) & \text{for CL-FWPD,} \\ \frac{1}{|G|\times|\beta|}\sum\limits_{\mathbf{x}_i\in G}\sum\limits_{\mathbf{x}_j\in\beta}\delta(\mathbf{x}_i,\mathbf{x}_j) & \text{for AL-FWPD.} \end{cases}$$

Set $t = t + 1$.
5. Repeat Steps 2-4 until B^t contains a single element.

FWPD being the underlying dissimilarity measure (instead of other metrics such as the Euclidean distance), the HAC-FWPD algorithm can be directly applied to obtain SL, CL, or AL based hierarchical clustering of datasets with missing feature values.

4.2 Time complexity of the HAC-FWPD algorithm

Irrespective of whether SL-FWPD, AL-SWPD or CL-FWPD is used as the proximity measure, the HAC-FWPD algorithm consists of the following three basic steps:

1. *Distance Calculation*: As each distance calculation involves at most m features, the calculation of all pairwise observed distances among n data instances is at most $\mathcal{O}(n^2 m)$.
2. *Penalty Calculation*: The penalty calculation between a data point and a cluster centroid involves at most m summations. Hence, penalty calculation over all possible pairings is at most $\mathcal{O}(n^2 m)$.
3. *Cluster Merging*: The merging of two clusters takes place in each of the $n-1$ steps of the algorithm, and each merge at most has a time complexity of $\mathcal{O}(n^2)$.

Therefore, the total computational complexity of the HAC-FWPD algorithm is $\mathcal{O}(n^2 m)$ which is the same as that of the standard HAC algorithm based on SL, CL or AL.

5 Experimental results

In this section, we report the results of several experiments carried out to validate the merit of the proposed k-means-FWPD and HAC-FWPD clustering algorithms.[2] In the following subsections, we describe the experimental setup used to validate the proposed techniques. The results of the experiments for the k-means-FWPD algorithm and the HAC-FWPD algorithm, are respectively presented thereafter.

5.1 Experiment setup

Adjusted Rand Index (ARI) (Hubert and Arabie 1985) is a popular validity index used to judge the merit of the clustering algorithms. When the true class labels are known, ARI provides a measure of the similarity between the cluster partition obtained by a clustering technique and the true class labels. Therefore, a high value of ARI is thought to indicate better clusterings. But, the class labels may not always be in keeping with the natural cluster structure of the dataset. In such cases, good clusterings are likely to achieve lower values of these indexes compared to possibly erroneous partitions (which are more akin to the class labels). However, the purpose of our experiments is to find out how close the clusterings obtained by the proposed methods (and the contending techniques) are to the clusterings obtained by the standard algorithms (k-means algorithm and HAC algorithm); the proposed methods (and its contenders) being run on the datasets with missingness, while the standard methods are run on corresponding fully observed datasets. Hence, the clusterings obtained by the standard algorithms are used as the ground-truths using which the ARI values are calculated for the proposed methods (and their contenders). The performances of ZI, MI, kNNI (with $k \in \{3, 5, 10, 20\}$) and SVDI (using the most significant 10% of the eigenvectors) are used for comparison with the proposed methods. The variant of MI that we impute with for these experiments differs from the traditional technique in that we use the average of the averages for individual classes, instead of the overall average. This is done to minimize the effects of severe class imbalances that may exist in the datasets. We also conduct the Wilcoxon's signed rank test (Wilcoxon 1945) to evaluate the statistical significance of the observed results.

The performance of k-means depends on the initial cluster assignment. Therefore, to ensure fairness, we use the same set of random initial cluster assignments for both the standard k-means algorithm on the fully observed dataset as well as the proposed k-means-FWPD method (and its contenders). The maximum number of iterations of the k-means variants is set as $MaxIter = 500$. Results are recorded in terms of average ARI values over 50 different runs on each dataset. The number of clusters is assumed to be same as the number of classes.

For HAC experiments, Results are reported as average ARI values obtained over 20 independent runs. AL is chosen over SL and CL as it is observed to generally achieve higher ARI values. The number of clusters is assumed to be same as the number of classes.

5.1.1 Datasets

We take 20 real-world datasets from the University of California at Irvine (UCI) repository (Dheeru and Karra Taniskidou 2017) and the Jin Genomics Dataset (JGD) repository (Jin 2017). Each feature of each dataset is normalized so as to have zero mean and unit standard deviation. The details of these 20 datasets are listed in Table 4.

---

[2] Source codes are available at https://github.com/Shounak-D/Clustering-Missing-Features.

Table 4 Details of the 20 real-world datasets

Dataset	#Instances	#Features	#Classes	Repository
Chronic kidney	800	24	2	UCI
Colon	62	2000	2	JGD
GSAD* 1[†]	445	128	6	UCI
Glass	214	9	6	UCI
Iris	150	4	3	UCI
Isolet 5[†]	1559	617	26	UCI
Landsat	6435	36	6	UCI
Leaf	340	15	36	UCI
Libras	360	90	15	UCI
Lung	181	12, 533	2	JGD
Lung Cancer	27	56	3	UCI
Lymphoma	62	4026	3	JGD
Pendigits	10, 992	16	10	UCI
Prostate	102	6033	2	JGD
Seeds	210	7	3	UCI
Sensorless[†]	6000	48	11	UCI
Sonar	208	60	2	UCI
Theorem proving[†]	3059	51	6	UCI
Vehicle	94	18	4	UCI
Vowel context	990	14	11	UCI

*Gas sensor array drift

[†]Only a meaningful subset of the dataset is used

5.1.2 Simulating missingness mechanisms

Experiments are conducted by removing features from each of the datasets according to the four missingness mechanisms, namely MCAR, MAR, MNAR-I and MNAR-II (Datta et al. 2016b). The detailed algorithm for simulating the four missingness mechanisms is as follows:

1. Specify the number of entries $MissCount$ to be removed from the dataset. Select the missingness mechanism as one out of MCAR, MAR, MNAR-I or MNAR-II.
2. If the mechanism is MAR or MNAR-II, select a random subset $\gamma_{miss} \subset S$ containing half of the features in S (i.e. $|\gamma_{miss}| = \frac{m}{2}$ if $|S|$ is even or $\frac{m+1}{2}$ if $|S|$ is odd). If the mechanism is MNAR-I, set $\gamma_{miss} = S$. Identify $\gamma_{obs} = S \backslash \gamma_{miss}$. Otherwise, go to Step 5.
3. If the mechanism is MAR or MNAR-II, for each feature $l \in \gamma_{miss}$, randomly select a feature $l_c \in \gamma_{obs}$ on which the missingness of feature l may depend.
4. For each feature $l \in \gamma_{miss}$ randomly choose a type of missingness $MissType_l$ as one out of CENTRAL, INTERMEDIATE or EXTREMAL.
5. Randomly select a non-missing entry $x_{i,l}$ from the data matrix. If the mechanism is MCAR, mark the entry as missing and decrement $MissCount = MissCount - 1$ and go to Step 11.
6. If the mechanism is MAR, set $\lambda = x_{i,l_c}$, $\mu = \mu_{l_c}$ and $\sigma = \sigma_{l_c}$, where μ_{l_c} and σ_{l_c} are the mean and standard deviation of the l_c-th feature over the dataset. If the mechanism

is MNAR-I, set $\lambda = x_{i,l}$, $\mu = \mu_l$ and $\sigma = \sigma_l$. If the mechanism is MNAR-II, randomly set either $\lambda = x_{i,l}$, $\mu = \mu_l$ and $\sigma = \sigma_l$ or $\lambda = x_{i,l_c}$, $\mu = \mu_{l_c}$ and $\sigma = \sigma_{l_c}$.

7. Calculate $z = \frac{1}{\sigma}\sqrt{(\lambda - \mu)^2}$.

8. If $MissType_l = $ CENTRAL, set $\mu_z = 0$. If $MissType_l = $ INTERMEDIATE, set $\mu_z = 1$. If $MissType_l = $ EXTREMAL, set $\mu_z = 2$. Set $\sigma_z = 0.35$.

9. Calculate $pval = \frac{1}{\sqrt{2\pi}\sigma_z}exp(-\frac{(z-\mu_z)^2}{2\sigma_z^2})$.

10. Randomly generate a value $qval$ in the interval $[0, 1]$. If $pval \geq qval$, then mark the entry $x_{i,l}$ as missing and decrement $MissCount = MissCount - 1$.

11. If $MissCount > 0$, then go to Step 5.

In the above algorithm, the dependence of the missingness on feature values for MAR, MNAR-I and MNAR-II is achieved by removing entries based on the values of control features for their corresponding data points. The control feature may be the feature itself (for MNAR-I and MNAR-II) or may be another feature for the same data point (as in the case of MAR and MNAR-II). The dependence can be of three types, namely CENTRAL, INTERMEDIATE or EXTREMAL. CENTRAL dependence removes a feature when its corresponding control feature has a value close to the mean of the control feature over the dataset. EXTREMAL dependence removes a feature when the value of its control feature lies near the extremes. INTERMEDIATE dependence removes a feature when the value of the control lies between the mean and the extremes.

For our experiments, we set $MissCount = \frac{nm}{4}$ to remove 25% of the features from each dataset. Thus, an average of $\frac{m}{4}$ of the features are missing from each data instance.

5.1.3 Selecting the parameter α

In order to conduct experiments using the FWPD measure, we need to select a value of the parameter α. Proper selection of α may help to boost the performance of the proposed k-means-FWPD and HAC-FWPD measures. Therefore, in this section, we undertake a study on the effects of α on the performance of FWPD. Experiments are conducted using $\alpha \in \{0.1, 0.25, 0.5, 0.75, 0.9\}$ on the datasets listed in Table 4 using the experimental setup detailed above. The summary of the results of this study is shown in Table 5 in terms of average ARI values.

A choice of $\alpha = 0.25$ performs best overall as well as individually except for MAR missingness (where $\alpha = 0.1$ proves to be a better choice). This seems to indicate some correlation between the extent of missingness and the optimal value of α (25% of the features are missing in our experiments as mentioned in Sect. 5.1.2). However, the correlation is rather weak for k-means-FWPD where all values of alphas seem to have competitive performance. On the other hand, the correlation is seen to be much stronger for HAC-FWPD. This indicates that the optimal α varies considerably with the pattern of missingness for k-means-FWPD but not as much for HAC-FWPD. Another interesting observation is that performance of HAC-FWPD deteriorates considerably for high values of α implying that the distance term is FWPD must be given greater importance for HAC methods. As $\alpha = 0.25$ has the best performance overall, we report the detailed experimental results in the subsequent sections for this choice of α.

5.2 Experiments with the k-means-FWPD algorithm

We compare the proposed k-means-FWPD algorithm to the standard k-means algorithm run on the datasets obtained after performing ZI, MI, SVDI and kNNI. All runs of k-means-

Table 5 Summary of results for different choices of α in terms of average ARI values

Clustering Algorithm	Type of Missingness	α				
		0.1	0.25	0.5	0.75	0.9
k-means-FWPD	MCAR	0.682	0.712	0.664	0.691	0.683
	MAR	0.738	0.730	0.723	0.729	0.711
	MNAR-I	0.649	0.676	0.675	0.613	0.666
	MNAR-II	0.711	0.718	0.689	0.665	0.678
	Overall	0.695	**0.709**	0.688	0.675	0.685
HAC-FWPD	MCAR	0.665	0.709	0.389	0.073	0.017
	MAR	0.740	0.724	0.441	0.210	0.094
	MNAR-I	0.720	0.721	0.458	0.158	0.036
	MNAR-II	0.708	0.716	0.443	0.140	0.025
	Overall	0.709	**0.718**	0.433	0.145	0.043

Best values shown in boldface

FWPD were found to converge within the stipulated budget of $MaxIter = 500$. The results of the experiments are listed in terms of the means and standard deviations of the obtained ARI values, in Tables 6, 7, 8 and 9. Only the best results for kNNI are reported, along with the best k values. The statistically significance of the listed results are summarized at the bottom of the table in terms of average ranks as well as signed rank test hypotheses and p values (H_0 signifying that the ARI values achieved by the proposed method and the contending method originate from identical distributions having the same medians; H_1 implies that the ARI values achieved by the proposed method and the contender originate from different distributions).

We know from Theorem 3 that the maximum number of feasibility adjustments that can occur during a single run of k-means-FWPD is $n(k-1)$. This begs the question of whether one should choose $MaxIter \geq n(k-1)$. However, k-means-FWPD was observed to converge within the stipulated $MaxIter = 500$ iterations even for datasets like Isolet 5, Pendigits, Sensorless, etc. which have relatively large values of $n(k-1)$. This indicates that the number of feasibility adjustments that occur during a run is much lower in practice. Therefore, we conclude that it is not required to set $MaxIter \geq n(k-1)$ for practical problems.

It is seen from Tables 6, 7, 8 and 9 that the k-means-FWPD algorithm performs best, indicated by the consistently minimum average rankings on all types of missingness. The proposed method performs best on the majority of datasets for all kinds of missingness. kNNI is overall seen to be the second best performer (being statistically comparable to k-means-FWPD in case of MAR). It is also interesting to observe that the performance of MI is improved in case of MAR and MNAR-II, indicating that MI tends to be useful for partitional clustering when the missingness depends on the observed features. Moreover, SVDI is generally observed to perform poorly irrespective of the type of missingness, implying that the linear model assumed by SVDI is unable to conserve the convexity of the clusters (which is essential for good performance in case of partitional clustering).

5.3 Experiments with the HAC-FWPD algorithm

The experimental setup described in Sect. 5.1 is also used to compare the HAC-FWPD algorithm (with AL-FWPD as the proximity measure) to the standard HAC algorithm (with

Table 6 Means and standard deviations of ARI values for the k-means-FWPD algorithm against MCAR

Dataset	k-means-FWPD	ZI	MI	SVDI	kNNI	Best k (for kNNI)
Chronic Kidney	0.807 ± 0.002	0.813 ± 0.005	0.229 ± 0.013	0.763 ± 0.005	**0.815** ± 0.003	5
Colon	**0.681** ± 0.341	0.656 ± 0.323	0.659 ± 0.322	0.662 ± 0.314	0.656 ± 0.323	3
GSAD 1	**0.798** ± 0.236	0.625 ± 0.199	0.722 ± 0.172	0.552 ± 0.203	0.711 ± 0.195	3
Glass	0.488 ± 0.097	0.466 ± 0.119	0.131 ± 0.066	0.417 ± 0.114	**0.505** ± 0.134	5
Iris	**0.799** ± 0.119	0.672 ± 0.113	0.116 ± 0.083	0.732 ± 0.159	0.758 ± 0.157	3
Isolet 5	**0.679** ± 0.117	0.623 ± 0.093	0.625 ± 0.097	0.626 ± 0.105	0.614 ± 0.072	3
Landsat	**0.937** ± 0.001	0.807 ± 0.001	0.798 ± 0.001	0.838 ± 0.104	0.937 ± 0.010	5
Leaf	0.455 ± 0.014	0.328 ± 0.010	0.339 ± 0.019	0.354 ± 0.037	**0.465** ± 0.029	5
Libras	**0.656** ± 0.070	0.642 ± 0.069	0.103 ± 0.019	0.619 ± 0.067	0.625 ± 0.077	20
Lung	**0.731** ± 0.341	0.718 ± 0.261	0.659 ± 0.341	0.694 ± 0.318	0.718 ± 0.261	3
Lung Cancer	**0.542** ± 0.249	0.541 ± 0.202	0.537 ± 0.214	0.529 ± 0.192	0.525 ± 0.189	5
Lymphoma	**0.755** ± 0.167	0.743 ± 0.175	0.700 ± 0.165	0.733 ± 0.175	0.743 ± 0.175	3
Pendigits	0.729 ± 0.089	0.659 ± 0.082	0.083 ± 0.013	0.604 ± 0.063	**0.832** ± 0.105	3
Prostate	**0.961** ± 0.025	0.944 ± 0.043	0.944 ± 0.043	0.946 ± 0.041	0.944 ± 0.043	3
Seeds	**0.866** ± 0.030	0.735 ± 0.021	0.242 ± 0.039	0.745 ± 0.041	0.865 ± 0.025	5
Sensorless	**0.765** ± 0.031	0.684 ± 0.028	0.687 ± 0.021	0.719 ± 0.060	0.726 ± 0.051	20
Sonar	**0.697** ± 0.195	0.681 ± 0.187	0.672 ± 0.188	0.434 ± 0.234	0.656 ± 0.162	5
Theorem proving	**0.714** ± 0.197	0.671 ± 0.229	0.661 ± 0.188	0.565 ± 0.139	0.672 ± 0.197	20
Vehicle	0.715 ± 0.143	0.674 ± 0.139	0.114 ± 0.060	0.646 ± 0.105	**0.723** ± 0.134	10
Vowel context	0.458 ± 0.031	0.366 ± 0.028	0.360 ± 0.029	0.352 ± 0.022	**0.461** ± 0.060	3
Average ranks	**1.38**	3.33	4.20	3.65	2.45	
Signed rank hypotheses (p values)		H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	H_1 (0.03)	

H_1 significantly different from k-means-FWPD
H_0 statistically similar to k-means-FWPD
Best values shown in boldface

Table 7 Means and standard deviations of ARI values for the k-means-FWPD algorithm against MAR

Dataset	k-means-FWPD	ZI	MI	SVDI	kNNI	Best k (for kNNI)
Chronic kidney	0.793 ± 0.006	0.792 ± 0.004	0.803 ± 0.003	0.787 ± 0.008	**0.838** ± 0.011	20
Colon	0.812 ± 0.177	0.826 ± 0.149	**0.827** ± 0.157	0.796 ± 0.202	0.826 ± 0.149	3
GSAD 1	**0.801** ± 0.163	0.737 ± 0.187	0.673 ± 0.179	0.719 ± 0.165	0.760 ± 0.144	3
Glass	**0.617** ± 0.125	0.455 ± 0.168	0.565 ± 0.128	0.411 ± 0.171	0.479 ± 0.152	5
Iris	0.776 ± 0.185	0.776 ± 0.185	**0.851** ± 0.144	0.776 ± 0.163	0.764 ± 0.176	5
Isolet 5	**0.729** ± 0.072	0.713 ± 0.051	0.691 ± 0.046	0.704 ± 0.027	0.713 ± 0.051	3
Landsat	**0.940** ± 0.002	0.828 ± 0.127	0.850 ± 0.123	0.773 ± 0.150	0.899 ± 0.058	3
Leaf	0.510 ± 0.022	0.392 ± 0.042	0.440 ± 0.046	0.501 ± 0.021	**0.532** ± 0.046	3
Libras	0.731 ± 0.076	0.700 ± 0.077	**0.778** ± 0.065	0.697 ± 0.082	0.675 ± 0.071	3
Lung	**0.754** ± 0.131	0.711 ± 0.230	0.717 ± 0.232	0.625 ± 0.312	0.711 ± 0.230	3
Lung cancer	**0.606** ± 0.223	0.526 ± 0.224	0.476 ± 0.219	0.509 ± 0.202	0.493 ± 0.231	3
Lymphoma	0.790 ± 0.164	0.883 ± 0.109	**0.885** ± 0.102	0.875 ± 0.120	0.883 ± 0.109	3
Pendigits	0.717 ± 0.068	0.494 ± 0.072	0.852 ± 0.062	0.705 ± 0.067	**0.903** ± 0.065	5
Prostate	**0.990** ± 0.021	0.984 ± 0.036	0.986 ± 0.036	0.918 ± 0.017	0.984 ± 0.036	3
Seeds	0.785 ± 0.026	0.755 ± 0.025	0.774 ± 0.027	0.752 ± 0.025	**0.834** ± 0.033	3
Sensorless	**0.759** ± 0.038	0.663 ± 0.089	0.629 ± 0.145	0.662 ± 0.132	0.685 ± 0.087	3
Sonar	**0.620** ± 0.289	0.599 ± 0.326	0.598 ± 0.325	0.524 ± 0.315	0.574 ± 0.359	10
Theorem proving	**0.672** ± 0.179	0.636 ± 0.165	0.617 ± 0.182	0.618 ± 0.189	0.649 ± 0.145	10
Vehicle	**0.699** ± 0.142	0.545 ± 0.155	0.644 ± 0.147	0.566 ± 0.154	0.537 ± 0.149	3
Vowel context	0.497 ± 0.044	0.461 ± 0.072	0.436 ± 0.064	0.408 ± 0.054	**0.577** ± 0.043	3
Average ranks	**1.85**	3.33	2.90	4.25	2.67	
Signed rank hypotheses (p values)		H_1(0.00)	H_0(0.13)	H_1(0.00)	H_0(0.37)	

H_1 significantly different from k-means-FWPD
H_0 statistically similar to k-means-FWPD
Best values shown in boldface

Table 8 Means and standard deviations of ARI values for the k-means-FWPD algorithm against MNAR-I

Dataset	k-means-FWPD	ZI	MI	SVDI	kNNI	Best k (for kNNI)
Chronic Kidney	**0.729** ± 0.011	0.714 ± 0.010	0.399 ± 0.051	0.599 ± 0.005	0.616 ± 0.022	3
Colon	0.789 ± 0.214	0.781 ± 0.202	0.770 ± 0.205	**0.801** ± 0.147	0.781 ± 0.202	3
GSAD 1	0.791 ± 0.112	**0.799** ± 0.097	0.790 ± 0.110	0.689 ± 0.175	**0.799** ± 0.097	3
Glass	**0.439** ± 0.101	0.391 ± 0.117	0.152 ± 0.048	0.388 ± 0.097	0.438 ± 0.105	10
Iris	0.662 ± 0.073	0.709 ± 0.144	0.137 ± 0.077	**0.739** ± 0.132	0.658 ± 0.168	5
Isolet 5	**0.708** ± 0.098	0.680 ± 0.103	0.680 ± 0.082	0.663 ± 0.067	0.680 ± 0.103	3
Landsat	**0.869** ± 0.048	0.701 ± 0.149	0.712 ± 0.159	0.858 ± 0.058	0.813 ± 0.001	10
Leaf	0.493 ± 0.052	0.416 ± 0.029	0.403 ± 0.020	0.439 ± 0.023	**0.522** ± 0.040	3
Libras	**0.717** ± 0.083	0.667 ± 0.076	0.378 ± 0.058	0.638 ± 0.070	0.656 ± 0.067	3
Lung	**0.636** ± 0.201	0.592 ± 0.199	0.606 ± 0.210	0.578 ± 0.192	0.592 ± 0.199	3
Lung cancer	**0.529** ± 0.235	0.497 ± 0.184	0.459 ± 0.129	0.457 ± 0.217	0.497 ± 0.184	3
Lymphoma	0.796 ± 0.118	**0.798** ± 0.133	0.764 ± 0.130	0.764 ± 0.129	**0.798** ± 0.133	3
Pendigits	0.666 ± 0.079	0.635 ± 0.067	0.135 ± 0.025	0.619 ± 0.054	**0.756** ± 0.093	5
Prostate	**0.975** ± 0.032	0.958 ± 0.075	0.962 ± 0.075	0.910 ± 0.061	0.958 ± 0.075	3
Seeds	0.776 ± 0.019	0.705 ± 0.044	0.298 ± 0.065	0.725 ± 0.042	**0.819** ± 0.052	20
Sensorless	**0.693** ± 0.041	0.638 ± 0.036	0.636 ± 0.039	0.610 ± 0.069	0.593 ± 0.051	10
Sonar	**0.600** ± 0.297	0.537 ± 0.287	0.546 ± 0.292	0.326 ± 0.282	0.537 ± 0.287	3
Theorem proving	**0.540** ± 0.211	0.399 ± 0.196	0.388 ± 0.178	0.513 ± 0.224	0.465 ± 0.195	3
Vehicle	**0.639** ± 0.141	0.551 ± 0.123	0.298 ± 0.078	0.613 ± 0.103	0.536 ± 0.107	3
Vowel context	0.473 ± 0.044	0.412 ± 0.049	0.435 ± 0.056	0.383 ± 0.043	**0.512** ± 0.036	3
Average ranks	**1.55**	3.02	4.08	3.67	2.67	
Signed Rank Hypotheses (p values)		H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	H_1 (0.05)	

H_1 significantly different from k-means-FWPD

H_0 statistically similar to k-means-FWPD

Best values shown in boldface

Table 9 Means and standard deviations of ARI values for the k-means-FWPD algorithm against MNAR-II

Dataset	k-means-FWPD	ZI	MI	SVDI	kNNI	Best k (for kNNI)
Chronic kidney	0.751 ± 0.014	0.659 ± 0.032	0.744 ± 0.015	0.636 ± 0.037	**0.770** ± 0.017	3
Colon	**0.804** ± 0.210	0.797 ± 0.226	0.798 ± 0.220	0.781 ± 0.208	0.797 ± 0.226	3
GSAD 1	**0.731** ± 0.198	0.665 ± 0.242	0.712 ± 0.212	0.689 ± 0.215	0.665 ± 0.242	3
Glass	**0.530** ± 0.089	0.423 ± 0.101	0.413 ± 0.099	0.395 ± 0.109	0.451 ± 0.101	5
Iris	**0.773** ± 0.165	0.718 ± 0.172	0.635 ± 0.189	0.702 ± 0.174	0.756 ± 0.175	10
Isolet 5	**0.789** ± 0.061	0.765 ± 0.076	0.747 ± 0.056	0.728 ± 0.063	0.765 ± 0.076	3
Landsat	**0.892** ± 0.083	0.871 ± 0.084	0.868 ± 0.083	0.794 ± 0.130	0.836 ± 0.083	3
Leaf	**0.476** ± 0.021	0.385 ± 0.036	0.381 ± 0.031	0.389 ± 0.033	0.454 ± 0.028	3
Libras	**0.698** ± 0.079	0.675 ± 0.078	0.681 ± 0.077	0.648 ± 0.080	0.669 ± 0.081	3
Lung	0.686 ± 0.220	0.649 ± 0.226	**0.707** ± 0.089	0.674 ± 0.216	0.649 ± 0.226	3
Lung cancer	**0.641** ± 0.200	0.640 ± 0.283	0.567 ± 0.243	0.568 ± 0.190	0.640 ± 0.283	3
Lymphoma	**0.856** ± 0.115	0.824 ± 0.136	0.842 ± 0.123	0.818 ± 0.153	0.824 ± 0.136	3
Pendigits	0.608 ± 0.082	0.589 ± 0.095	0.561 ± 0.097	0.557 ± 0.096	**0.825** ± 0.083	5
Prostate	**0.984** ± 0.032	**0.984** ± 0.032	**0.984** ± 0.032	0.969 ± 0.053	**0.984** ± 0.032	3
Seeds	**0.884** ± 0.028	0.738 ± 0.039	0.772 ± 0.038	0.771 ± 0.038	0.831 ± 0.029	10
Sensorless	**0.747** ± 0.041	0.667 ± 0.130	0.727 ± 0.030	0.704 ± 0.056	0.698 ± 0.068	3
Sonar	**0.704** ± 0.227	0.662 ± 0.235	0.658 ± 0.221	0.314 ± 0.152	0.662 ± 0.235	3
Theorem proving	**0.640** ± 0.141	0.600 ± 0.105	0.605 ± 0.077	0.610 ± 0.175	0.593 ± 0.106	3
Vehicle	0.677 ± 0.145	**0.734** ± 0.132	0.571 ± 0.167	0.635 ± 0.153	0.712 ± 0.141	3
Vowel context	0.478 ± 0.048	0.404 ± 0.041	0.396 ± 0.064	0.345 ± 0.032	**0.511** ± 0.056	3
Average ranks	**1.38**	3.30	3.27	4.25	2.80	
Signed rank hypotheses (p values)		H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	H_1 (0.03)	

H_1 significantly different from k-means-FWPD

H_0 statistically similar to k-means-FWPD

Best values shown in boldface

AL as the proximity measure) in conjunction with ZI, MI, SVDI and kNNI. Results are reported as means and standard deviations of obtained ARI values over the 20 independent runs. AL is preferred here over SL and CL as it is observed to generally achieve higher ARI values. The results of the experiments are listed in Tables 10, 11, 12 and 13. The statistically significance of the listed results are also summarized at the bottom of the respective tables in terms of average ranks as well as signed rank test hypotheses and p values (H_0 signifying that the ARI values achieved by the proposed method and the contending method originate from identical distributions having the same medians; H_1 implies that the ARI values achieved by the proposed method and the contender originate from different distributions).

It is seen from Tables 10, 11, 12 and 13 that the HAC-FWPD algorithm is able to perform best on all types of missingness, as evident from the consistently minimum average ranking. The proposed method performs best on the majority of datasets for all types of missingness. Moreover, the performance of HAC-FWPD is observed to be significantly better than kNNI which performs poorly overall, indicating that kNNI may not be useful for hierarchical clustering applications with missingness. Interestingly, in case of MAR and MNAR-II, both of which are characterized by the dependence of the missingness on the observed features, ZI, MI as well as SVDI show improved performance. This indicates that the dependence of the missingness on the observed features aids these imputation methods in case of hierarchical clustering. Another intriguing observation is that all the contending HAC methods consistently achieved the best possible performance on the high-dimensional datasets Lung and Prostate. This may indicate that while convexity of the cluster structures may be harmed due to missingness, the local proximity among the points is preserved owing to the sparse nature of such high-dimensional datasets.

6 Conclusions

In this paper, we propose to use the FWPD measure as a viable alternative to imputation and marginalization approaches to handle the problem of missing features in data clustering. The proposed measure attempts to estimate the original distances between the data points by adding a penalty term to those pair-wise distances which cannot be calculated on the entire feature space due to missing features. Therefore, unlike existing methods for handling missing features, FWPD is also able to distinguish between distinct data points which look identical due to missing features. Yet, FWPD also ensures that the dissimilarity for any data instance from itself is never greater than its dissimilarity from any other point in the dataset. Intuitively, these advantages of FWPD should help us better model the original data space which may help in achieving better clustering performance on the incomplete data.

Therefore, we use the proposed FWPD measure to put forth the k-means-FWPD and the HAC-FWPD clustering algorithms, which are directly applicable to datasets with missing features. We conduct extensive experimentation on the new techniques using various benchmark datasets and find the new approach to produce generally better results (for both partitional as well as hierarchical clustering) compared to some of the popular imputation methods which are generally used to handle the missing feature problem. In fact, it is observed from the experiments that the performance of the imputation schemes varies with the type of missingness and/or the clustering algorithm being used (for example, kNNI is useful for k-means clustering but not for HAC clustering; SVDI is useful for HAC clustering but not for k-means clustering; MI is effective when the missingness depends on the observed features). The proposed approach, on the other hand, exhibits good performance across all types of

Table 10 Means and standard deviations of ARI values for the HAC-FWPD algorithm against MCAR

Dataset	HAC-FWPD	ZI	MI	SVDI	kNNI	Best k (for kNNI)
Chronic kidney	**1.000** ± 0.000	0.967 ± 0.031	0.933 ± 0.033	0.933 ± 0.033	0.000 ± 0.000	3
Colon	**0.690** ± 0.240	0.469 ± 0.145	0.286 ± 0.174	0.380 ± 0.304	0.000 ± 0.000	3
GSAD 1	**0.454** ± 0.309	0.367 ± 0.088	0.271 ± 0.112	0.311 ± 0.087	0.022 ± 0.018	3
Glass	**0.737** ± 0.081	0.671 ± 0.089	0.680 ± 0.089	0.638 ± 0.090	0.033 ± 0.032	3
Iris	0.885 ± 0.072	**0.922** ± 0.047	0.831 ± 0.053	0.917 ± 0.049	0.559 ± 0.129	20
Isolet 5	**0.855** ± 0.111	0.044 ± 0.003	0.046 ± 0.003	0.081 ± 0.037	0.064 ± 0.003	3
Landsat	**0.712** ± 0.098	0.228 ± 0.034	0.254 ± 0.012	0.217 ± 0.033	0.300 ± 0.018	10
Leaf	**0.497** ± 0.046	0.200 ± 0.016	0.221 ± 0.017	0.290 ± 0.077	0.140 ± 0.011	3
Libras	**0.845** ± 0.054	0.276 ± 0.031	0.298 ± 0.033	0.381 ± 0.030	0.156 ± 0.050	10
Lung	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.001 ± 0.002	3
Lung cancer	**0.458** ± 0.193	0.408 ± 0.223	0.356 ± 0.229	0.436 ± 0.335	0.034 ± 0.035	3
Lymphoma	**0.885** ± 0.058	0.718 ± 0.373	0.335 ± 0.498	0.713 ± 0.372	0.547 ± 0.297	3
Pendigits	**0.712** ± 0.082	0.242 ± 0.194	0.228 ± 0.224	0.252 ± 0.260	0.365 ± 0.147	3
Prostate	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.001 ± 0.001	3
Seeds	0.534 ± 0.173	0.332 ± 0.046	0.317 ± 0.055	0.436 ± 0.127	**0.563** ± 0.110	10
Sensorless	**0.416** ± 0.303	0.196 ± 0.024	0.203 ± 0.017	0.249 ± 0.102	0.005 ± 0.008	3
Sonar	**0.440** ± 0.419	0.329 ± 0.473	0.128 ± 0.298	0.261 ± 0.365	0.001 ± 0.000	3
Theorem proving	**0.802** ± 0.085	0.691 ± 0.088	0.691 ± 0.088	0.654 ± 0.082	0.002 ± 0.001	5
Vehicle	**0.807** ± 0.108	0.315 ± 0.295	0.315 ± 0.295	0.645 ± 0.232	0.084 ± 0.008	5
Vowel context	**0.453** ± 0.081	0.248 ± 0.042	0.211 ± 0.066	0.194 ± 0.029	0.101 ± 0.019	3
Average ranks	**1.30**	2.95	3.52	2.88	4.35	
Signed rank hypotheses (p values)		H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	

H_1 significantly different from HAC-FWPD
H_0 statistically similar to HAC-FWPD
Best values shown in boldface

Table 11 Means and standard deviations of ARI values for the HAC-FWPD algorithm against MAR

Dataset	HAC-FWPD	ZI	MI	SVDI	kNNI	Best k (for kNNI)
Chronic kidney	**0.799** ± 0.394	0.398 ± 0.494	0.398 ± 0.494	0.398 ± 0.494	0.003 ± 0.001	3
Colon	**1.000** ± 0.000	0.463 ± 0.516	0.463 ± 0.516	0.601 ± 0.423	0.016 ± 0.002	3
GSAD 1	**0.619** ± 0.230	0.359 ± 0.115	0.419 ± 0.227	0.346 ± 0.150	0.007 ± 0.005	3
Glass	**0.650** ± 0.188	0.617 ± 0.124	0.603 ± 0.129	0.590 ± 0.184	0.057 ± 0.082	20
Iris	**0.949** ± 0.051	0.893 ± 0.083	0.893 ± 0.083	0.854 ± 0.146	0.587 ± 0.068	10
Isolet 5	**0.725** ± 0.186	0.491 ± 0.011	0.491 ± 0.011	0.464 ± 0.076	0.076 ± 0.005	3
Landsat	**0.734** ± 0.044	0.638 ± 0.323	0.601 ± 0.301	0.721 ± 0.142	0.162 ± 0.113	3
Leaf	0.463 ± 0.107	0.420 ± 0.099	0.415 ± 0.058	**0.471** ± 0.053	0.154 ± 0.013	3
Libras	**0.864** ± 0.098	0.855 ± 0.048	0.815 ± 0.081	0.813 ± 0.064	0.469 ± 0.012	3
Lung	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.008 ± 0.024	3
Lung cancer	**0.709** ± 0.270	0.522 ± 0.438	0.538 ± 0.422	0.443 ± 0.358	0.036 ± 0.019	3
Lymphoma	**1.000** ± 0.000	0.903 ± 0.089	0.772 ± 0.132	0.890 ± 0.104	0.788 ± 0.000	3
Pendigits	0.493 ± 0.138	0.351 ± 0.124	0.292 ± 0.076	0.483 ± 0.103	0.407 ± 0.035	3
Prostate	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.001 ± 0.000	3
Seeds	**0.564** ± 0.131	0.487 ± 0.016	0.444 ± 0.103	0.535 ± 0.224	0.556 ± 0.135	3
Sensorless	**0.439** ± 0.288	0.276 ± 0.163	0.174 ± 0.053	0.256 ± 0.136	0.000 ± 0.000	10
Sonar	**0.396** ± 0.551	0.005 ± 0.000	0.005 ± 0.000	0.094 ± 0.222	0.001 ± 0.000	3
Theorem proving	**0.725** ± 0.102	0.677 ± 0.031	0.685 ± 0.107	0.641 ± 0.121	0.001 ± 0.006	3
Vehicle	**0.827** ± 0.123	0.431 ± 0.279	0.431 ± 0.278	0.825 ± 0.105	0.075 ± 0.044	3
Vowel context	**0.517** ± 0.127	0.451 ± 0.095	0.445 ± 0.126	0.401 ± 0.207	0.104 ± 0.024	3
Average ranks	**1.20**	2.83	3.27	3.00	4.70	
Signed rank hypotheses (p values)		H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	

H_1 significantly different from HAC-FWPD
H_0 statistically similar to HAC-FWPD
Best values shown in boldface

Table 12 Means and standard deviations of ARI values for the HAC-FWPD algorithm against MNAR-I

Dataset	HAC-FWPD	ZI	MI	SVDI	kNNI	Best k (for kNNI)
Chronic kidney	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.002 ± 0.000	3
Colon	**0.926** ± 0.166	0.025 ± 0.000	0.025 ± 0.000	0.025 ± 0.000	0.014 ± 0.000	3
GSAD 1	**0.473** ± 0.237	0.326 ± 0.085	0.343 ± 0.098	0.271 ± 0.112	0.005 ± 0.003	3
Glass	0.736 ± 0.135	0.697 ± 0.119	**0.738** ± 0.133	0.614 ± 0.145	0.018 ± 0.006	10
Iris	0.852 ± 0.142	0.527 ± 0.437	0.543 ± 0.456	**0.881** ± 0.180	0.540 ± 0.026	20
Isolet 5	**0.586** ± 0.085	0.326 ± 0.207	0.401 ± 0.169	0.223 ± 0.153	0.048 ± 0.033	3
Landsat	**0.786** ± 0.085	0.420 ± 0.298	0.443 ± 0.375	0.765 ± 0.086	0.072 ± 0.004	5
Leaf	**0.514** ± 0.043	0.345 ± 0.065	0.267 ± 0.080	0.437 ± 0.054	0.110 ± 0.035	5
Libras	**0.843** ± 0.109	0.750 ± 0.101	0.750 ± 0.101	0.782 ± 0.087	0.419 ± 0.042	3
Lung	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.008 ± 0.024	3
Lung cancer	**0.651** ± 0.316	0.516 ± 0.269	0.516 ± 0.269	0.561 ± 0.336	0.020 ± 0.010	3
Lymphoma	**0.942** ± 0.130	0.861 ± 0.000	0.861 ± 0.000	0.861 ± 0.000	0.651 ± 0.000	3
Pendigits	**0.641** ± 0.172	0.405 ± 0.255	0.341 ± 0.238	0.399 ± 0.139	0.416 ± 0.043	5
Prostate	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.001 ± 0.000	3
Seeds	**0.584** ± 0.154	0.429 ± 0.141	0.479 ± 0.047	0.421 ± 0.084	0.555 ± 0.082	3
Sensorless	**0.300** ± 0.298	0.225 ± 0.025	0.217 ± 0.027	0.216 ± 0.009	0.000 ± 0.000	3
Sonar	**0.598** ± 0.550	0.196 ± 0.449	0.196 ± 0.449	0.329 ± 0.473	0.001 ± 0.000	3
Theorem proving	**0.775** ± 0.121	0.762 ± 0.068	0.731 ± 0.025	0.742 ± 0.031	0.002 ± 0.000	3
Vehicle	**0.797** ± 0.121	0.321 ± 0.194	0.522 ± 0.264	0.699 ± 0.290	0.068 ± 0.040	3
Vowel context	**0.447** ± 0.168	0.300 ± 0.073	0.282 ± 0.068	0.306 ± 0.114	0.095 ± 0.029	3
Average ranks	**1.33**	3.15	3.05	2.83	4.65	
Signed rank hypotheses (p values)		H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	

H_1 significantly different from HAC-FWPD
H_0 statistically similar to HAC-FWPD
Best values shown in boldface

Table 13 Means and standard deviations of ARI values for the HAC-FWPD algorithm against MNAR-II

Dataset	HAC-FWPD	ZI	MI	SVDI	kNNI	Best k (for kNNI)
Chronic kidney	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.002 ± 0.000	3
Colon	**1.000** ± 0.000	0.147 ± 0.384	0.147 ± 0.384	0.147 ± 0.384	0.013 ± 0.004	3
GSAD 1	**0.429** ± 0.190	0.353 ± 0.011	0.356 ± 0.021	0.353 ± 0.017	0.009 ± 0.001	3
Glass	0.717 ± 0.125	**0.733** ± 0.103	0.696 ± 0.113	0.690 ± 0.172	0.020 ± 0.008	20
Iris	0.885 ± 0.079	0.718 ± 0.394	0.718 ± 0.394	1 ± 0.024	0.552 ± 0.012	10
Isolet 5	**0.711** ± 0.047	0.296 ± 0.214	0.479 ± 0.007	0.184 ± 0.174	0.043 ± 0.034	3
Landsat	**0.763** ± 0.048	0.229 ± 0.004	0.229 ± 0.004	0.651 ± 0.278	0.082 ± 0.003	3
Leaf	**0.477** ± 0.036	0.255 ± 0.109	0.277 ± 0.081	0.342 ± 0.124	0.110 ± 0.044	3
Libras	**0.817** ± 0.053	0.742 ± 0.057	0.758 ± 0.065	0.780 ± 0.097	0.392 ± 0.040	3
Lung	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.009 ± 0.019	3
Lung cancer	**0.527** ± 0.158	0.515 ± 0.194	0.515 ± 0.194	0.503 ± 0.216	0.035 ± 0.021	3
Lymphoma	**0.925** ± 0.122	0.916 ± 0.076	0.876 ± 0.034	0.876 ± 0.034	0.706 ± 0.075	3
Pendigits	**0.485** ± 0.132	0.219 ± 0.053	0.300 ± 0.117	0.427 ± 0.108	0.387 ± 0.030	20
Prostate	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	**1.000** ± 0.000	0.001 ± 0.000	3
Seeds	**0.581** ± 0.154	0.398 ± 0.079	0.319 ± 0.149	0.491 ± 0.149	0.580 ± 0.125	3
Sensorless	**0.475** ± 0.378	0.204 ± 0.061	0.210 ± 0.044	0.214 ± 0.045	0.000 ± 0.000	3
Sonar	**0.295** ± 0.449	0.196 ± 0.450	0.196 ± 0.450	0.261 ± 0.436	0.001 ± 0.000	3
Theorem proving	**0.885** ± 0.078	0.681 ± 0.057	0.681 ± 0.057	0.711 ± 0.059	0.001 ± 0.002	3
Vehicle	**0.821** ± 0.072	0.518 ± 0.289	0.664 ± 0.251	0.700 ± 0.278	0.041 ± 0.037	3
Vowel context	**0.533** ± 0.154	0.377 ± 0.232	0.430 ± 0.109	0.425 ± 0.109	0.103 ± 0.024	3
Average ranks	**1.33**	3.33	3.00	2.60	4.75	
Signed rank hypotheses (p values)		H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	H_1 (0.00)	

H_1 significantly different from HAC-FWPD
H_0 statistically similar to HAC-FWPD
Best values shown in boldface

missingness as well as both partitional and hierarchical clustering paradigms. The experimental results attest to the ability of FWPD to better model the original data space, compared to existing methods.

However, it must be stressed, that the performance of all these methods, including the FWPD based ones, can vary depending on the structure of the dataset concerned, the choice of the proximity measure used (for HAC), and the pattern and extent of missingness plaguing the data. Fortunately, the α parameter embedded in FWPD can be varied in accordance with the extent of missingness to achieve desired results. The results in Sect. 5.1.3 indicate that it may be useful to choose a high value of α when a large fraction of the features are unobserved, and to choose a smaller value when only a few of the features are missing. However, in the presence of a sizable amount of missingness and the absence of ground-truths to validate the merit of the achieved clusterings, it is safest to choose a value of α proportional to the percentage of missing features restricted within the range [0.1, 0.25]. We also present an appendix dealing with an extension of the FWPD measure to problems with absent features and show that this modified form of FWPD is a semi-metric.

An obvious follow-up to this work is the application of the proposed PDM variant to practical clustering problems which are characterized by large fractions of unobserved data that arise in various fields such as economics, psychiatry, web-mining, etc. Studies can be undertaken to better understand the effects that the choice of α has on the clustering results. Another rewarding topic of research is the investigation of the abilities of the FWPD variant for absent features (see "Appendix A") by conducting proper experiments using benchmark applications characterized by this rare form of missingness (structural missingness).

Appendix A: Extending the FWPD to problems with absent features

This appendix proposes an extension of the FWPD measure to the case of absent features or structural missingness. The principal difference between missing and absent features lies in the fact that the unobserved features are known to be undefined in the latter case, unlike the former. Therefore, while it makes sense to add penalties for features which are observed for only one of the data instances (as the very existence of such a feature sets the points apart), it makes little sense to add penalties for features which are undefined for both the data points. This is in contrast to problems with unstructured missingness where a feature missing from both the data instances is known to be defined for both points (which potentially have distinct values of this feature). Thus, the fundamental difference between the problems of missing and absent features is that two points observed in the same subspace and having identical observed features should (unlike the missing data problem) essentially be considered identical instances in the case of absent features, as the unobserved features are known to be non-existent. But, in case of the unobserved features being merely unknown (rather than being non-existent), such data points should be considered distinct because the unobserved features are likely to have distinct values (making the points distinct when completely observed). Hence, it is essential to add penalties for features missing from both points in the case of missing features, but not in the case of absent features. Keeping this in mind, we can modify the proposed FWPD (essentially modifying the proposed FWP) as defined in the following text to serve as a dissimilarity measure for structural missingness.

Let the dataset X_{abs} consist of n data instances $\mathbf{x}_i$ ($i \in \{1, 2, \cdots, n\}$). Let $\zeta_{\mathbf{x}_i}$ denote the set of features on which the data point $\mathbf{x}_i \in X_{abs}$ is defined.

Definition 11 The FWP between the instances $\mathbf{x}_i$ and $\mathbf{x}_j$ in X_{abs} is defined as

$$p_{abs}(\mathbf{x}_i, \mathbf{x}_j) = \frac{\sum_{l \in (\zeta_{\mathbf{x}_i} \cup \zeta_{\mathbf{x}_j}) \setminus (\zeta_{\mathbf{x}_i} \cap \zeta_{\mathbf{x}_j})} v_l}{\sum_{l' \in \zeta_{\mathbf{x}_i} \cup \zeta_{\mathbf{x}_j}} v_{l'}} \tag{15}$$

where $v_s \in (0, n]$ is the number of instances in X_{abs} that are characterized by the feature s. Like in the case of unstructured missingness, this FWP also exacts greater penalty for the non-existence of commonly features.

Then, the definition of the FWPD modified for structural missingness is as follows.

Definition 12 The FWPD between $\mathbf{x}_i$ and $\mathbf{x}_j$ in X_{abs} is

$$\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) = (1 - \alpha) \times \frac{d(\mathbf{x}_i, \mathbf{x}_j)}{d_{max}} + \alpha \times p_{abs}(\mathbf{x}_i, \mathbf{x}_j), \tag{16}$$

where $\alpha \in (0, 1)$ is a parameter which determines the relative importance between the two terms and $d(\mathbf{x}_i, \mathbf{x}_j)$ and d_{max} retain their former definitions (but, in the context of structural missingness).

Now, having modified the FWPD to handle structural missingness, we show in the following theorem that the modified FWPD is a semi-metric.

Theorem 6 *The FWPD for absent features is a semi-metric, i.e. it satisfies the following important properties:*

1. *$\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) \geq 0 \ \forall \ \mathbf{x}_i, \mathbf{x}_j \in X_{abs}$,*
2. *$\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) = 0$ iff $\mathbf{x}_i = \mathbf{x}_j$, i.e. $\zeta_{\mathbf{x}_i} = \zeta_{\mathbf{x}_j}$ and $x_{i,l} = x_{j,l} \ \forall \ l \in \zeta_{\mathbf{x}_i}$, and*
3. *$\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) = \delta_{abs}(\mathbf{x}_j, \mathbf{x}_i) \ \forall \ \mathbf{x}_i, \mathbf{x}_j \in X_{abs}$.*

Proof 1. From Eq. (15) we can see that $p_{abs}(\mathbf{x}_i, \mathbf{x}_j) \geq 0 \ \forall \ \mathbf{x}_i, \mathbf{x}_j \in X_{abs}$ and Eq. (1) implies that $d(\mathbf{x}_i, \mathbf{x}_j) \geq 0 \ \forall \ \mathbf{x}_i, \mathbf{x}_j \in X_{abs}$. Hence, it follows that $\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) \geq 0 \ \forall \ \mathbf{x}_i, \mathbf{x}_j \in X_{abs}$.

2. It is easy to see from Eq. (15) that $p_{abs}(\mathbf{x}_i, \mathbf{x}_i) = 0$ iff $\zeta_{\mathbf{x}_i} = \zeta_{\mathbf{x}_j}$. Now, if $x_{i,l} = x_{j,l} \ \forall \ l \in \zeta_{\mathbf{x}_i}$, then $d(\mathbf{x}_i, \mathbf{x}_j) = 0$. Hence, $\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) = 0$ when $\zeta_{\mathbf{x}_i} = \zeta_{\mathbf{x}_j}$ and $x_{i,l} = x_{j,l} \ \forall \ l \in \zeta_{\mathbf{x}_i}$. The converse is also true as $\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) = 0$ implies $\zeta_{\mathbf{x}_i} = \zeta_{\mathbf{x}_j}$ and $d(\mathbf{x}_i, \mathbf{x}_j) = 0$; the latter in turn implying that $x_{i,l} = x_{j,l} \ \forall \ l \in \zeta_{\mathbf{x}_i}$.

3. From Eq. (16) we have

$$\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) = (1 - \alpha) \times \frac{d(\mathbf{x}_i, \mathbf{x}_j)}{d_{max}} + \alpha \times p_{abs}(\mathbf{x}_i, \mathbf{x}_j),$$

$$\text{and } \delta_{abs}(\mathbf{x}_j, \mathbf{x}_i) = (1 - \alpha) \times \frac{d(\mathbf{x}_j, \mathbf{x}_i)}{d_{max}} + \alpha \times p_{abs}(\mathbf{x}_j, \mathbf{x}_i).$$

But, $d(\mathbf{x}_i, \mathbf{x}_j) = d(\mathbf{x}_j, \mathbf{x}_i)$ and $p_{abs}(\mathbf{x}_i, \mathbf{x}_j) = p(\mathbf{x}_j, \mathbf{x}_i) \ \forall \ \mathbf{x}_i, \mathbf{x}_j \in X_{abs}$ (by definition). Therefore, it can be easily seen that $\delta_{abs}(\mathbf{x}_i, \mathbf{x}_j) = \delta_{abs}(\mathbf{x}_j, \mathbf{x}_i) \ \forall \ \mathbf{x}_i, \mathbf{x}_j \in X_{abs}$.   □

References

Acuña, E., & Rodriguez, C. (2004). The treatment of missing values and its effect on classifier accuracy. In D. Banks, F. R. McMorris, P. Arabie, & W. Gaul (Eds.), *Classification, clustering, and data mining applications, studies in classification, data analysis, and knowledge organisation* (pp. 639–647). Berlin, Heidelberg: Springer.

Ahmad, S., & Tresp, V. (1993). Some solutions to the missing feature problem in vision. In S. Hanson, J. Cowan, & C. Giles (Eds.), *Advances in neural information processing systems 5* (pp. 393–400). Los Altos, CA: Morgan-Kaufmann.

Barceló, C. (2008). The impact of alternative imputation methods on the measurement of income and wealth: Evidence from the spanish survey of household finances. In *Working paper series*. Banco de España.

Bo, T. H., Dysvik, B., & Jonassen, I. (2004). Lsimpute: Accurate estimation of missing values in microarray data with least squares methods. *Nucleic Acid Research*, *32*(3).

Broder, A. Z., Glassman, S. C., Manasse, M. S., & Zweig, G. (1997). Syntactic clustering of the web. *Computer Networks and ISDN Systems*, *29*(8–13), 1157–1166.

Chan, L. S., & Dunn, O. J. (1972). The treatment of missing values in discriminant analysis-1. The sampling experiment. *Journal of the American Statistical Association*, *67*(338), 473–477.

Chaturvedi, A., Carroll, J. D., Green, P. E., & Rotondo, J. A. (1997). A feature-based approach to market segmentation via overlapping k-centroids clustering. *Journal of Marketing Research*, pp. 370–377.

Chechik, G., Heitz, G., Elidan, G., Abbeel, P., & Koller, D. (2008). Max-margin classification of data with absent features. *Journal of Machine Learning Research*, *9*, 1–21.

Chen, F. (2013). Missing no more: Using the mcmc procedure to model missing data. In *Proceedings of the SAS global forum 2013 conference*, pp. 1–23. SAS Institute Inc.

Datta, S., Bhattacharjee, S., & Das, S. (2016a). Clustering with missing features: A penalized dissimilarity measure based approach. *CoRR*, arXiv:1604.06602.

Datta, S., Misra, D., & Das, S. (2016b). A feature weighted penalty based dissimilarity measure for k-nearest neighbor classification with missing features. *Pattern Recognition Letters*, *80*, 231–237.

Dempster, A. P., & Rubin, D. B. (1983). *Incomplete data in sample surveys, vol. 2*, chap. Part I: Introduction, pp. 3–10. New York: Academic Press.

Dheeru, D., & Taniskidou, E. K. (2017). UCI machine learning repository. Online repository at http://archive.ics.uci.edu/ml.

Dixon, J. K. (1979). Pattern recognition with partly missing data. *IEEE Transactions on Systems, Man and Cybernetics*, *9*(10), 617–621.

Donders, A. R. T., van der Heijden, G. J. M. G., Stijnen, T., & Moons, K. G. M. (2006). Review: A gentle introduction to imputation of missing values. *Journal of Clinical Epidemiology*, *59*(10), 1087–1091.

Forgy, E. W. (1965). Cluster analysis of multivariate data: Efficiency versus interpretability of classifications. *Biometrics*, *21*, 768–769.

Grzymala-Busse, J. W., & Hu, M. (2001). A comparison of several approaches to missing attribute values in data mining. In *Rough sets and current trends in computing*, pp. 378–385. Berlin: Springer.

Hathaway, R. J., & Bezdek, J. C. (2001). Fuzzy c-means clustering of incomplete data. *IEEE Transactions on Systems, Man, and Cybernetics: Part B: Cybernetics*, *31*(5), 735–744.

Haveliwala, T., Gionis, A., & Indyk, P. (2000). Scalable techniques for clustering the web. Tech. rep.: Stanford University.

Heitjan, D. F., & Basu, S. (1996). Distinguishing "missing at random" and "missing completely at random". *The American Statistician*, *50*(3), 207–213.

Himmelspach, L., & Conrad, S. (2010). Clustering approaches for data with missing values: Comparison and evaluation. In *Digital Information Management (ICDIM), 2010 fifth international conference on*, pp. 19–28.

Horton, N. J., & Lipsitz, S. R. (2001). Multiple imputation in practice: Comparison of software packages for regression models with missing variables. *The American Statistician*, *55*(3), 244–254.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, *2*(1), 193–218.

Jin, J. (2017). Genomics dataset repository. Online Repository at http://www.stat.cmu.edu/~jiashun/Research/software/GenomicsData/.

Juszczak, P., & Duin, R. P. W. (2004). Combining one-class classifiers to classify missing data. In *Multiple classifier systems*, pp. 92–101. Berlin: Springer.

Krause, S., & Polikar, R. (2003). An ensemble of classifiers approach for the missing feature problem. In *Proceedings of the international joint conference on neural networks, vol. 1*, pp. 553–558. IEEE.

Lasdon, L. S. (2013). *Optimization theory for large systems*. Courier Corporation.

Lei, L. (2010). Identify earthquake hot spots with 3-dimensional density-based clustering analysis. In *Geoscience and remote sensing symposium (IGARSS), 2010 IEEE international*, pp. 530–533. IEEE.

Little, R. J. A., & Rubin, D. B. (1987). *Statistical analysis with missing data*. New York: Wiley.

Lloyd, S. P. (1982). Least squares quantization in pcm. *IEEE Transactions on Information Theory*, *28*(2), 129–137.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, pp. 281–297. University of California Press.

Marlin, B. M. (2008). *Missing data problems in machine learning*. Ph.D. thesis, University of Toronto.

Millán-Giraldo, M., Duin, R. P., & Sánchez, J. S. (2010). Dissimilarity-based classification of data with missing attributes. In *Cognitive information processing (CIP), 2010 2nd international workshop on*, pp. 293–298. IEEE.

Murtagh, F., & Contreras, P. (2012). Algorithms for hierarchical clustering: An overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *2*(1), 86–97.

Myrtveit, I., Stensrud, E., & Olsson, U. H. (2001). Analyzing data sets with missing data: An empirical evaluation of imputation methods and likelihood-based methods. *IEEE Transactions on Software Engineering*, *27*(11), 999–1013.

Nanni, L., Lumini, A., & Brahnam, S. (2012). A classifier ensemble approach for the missing feature problem. *Artificial Intelligence in Medicine*, *55*(1), 37–50.

Porro-Muñoz, D., Duin, R. P., & Talavera, I. (2013). Missing values in dissimilarity-based classification of multi-way data. In *Iberoamerican congress on pattern recognition*, pp. 214–221. Berlin: Springer.

Rubin, D. B. (1976). Inference and missing data. *Biometrika*, *63*(3), 581–592.

Rubin, D. B. (1987). *Multiple imputation for nonresponse in surveys*. London: Wiley.

Sabau, A. S. (2012). Survey of clustering based financial fraud detection research. *Informatica Economica*, *16*(1), 110.

Schafer, J. L. (1997). *Analysis of incomplete multivariate data*. Boca Raton, FL: CRC Press.

Schafer, J. L., & Graham, J. W. (2002). Missing data: Our view of the state of the art. *Psychological Methods*, *7*(2), 147–177.

Sehgal, M. S. B., Gondal, I., & Dooley, L. S. (2005). Collateral missing value imputation: a new robust missing value estimation algorithm fpr microarray data. *Bioinformatics*, *21*(10), 2417–2423.

Selim, S. Z., & Ismail, M. A. (1984). K-means-type algorithms: A generalized convergence theorem and characterization of local optimality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *6*(1), 81–87.

Shelly, D. R., Ellsworth, W. L., Ryberg, T., Haberland, C., Fuis, G. S., Murphy, J., et al. (2009). Precise location of san andreas fault tremors near cholame, california using seismometer clusters: Slip on the deep extension of the fault? *Geophysical Research Letters*, *36*(1).

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., et al. (2001). Missing value estimation methods for dna microarrays. *Bioinformatics*, *17*(6), 520–525.

Wagstaff, K. L. (2004). Clustering with missing values: No imputation required. In *Proceedings of the meeting of the international Federation of classification societies*, pp. 649–658.

Wagstaff, K. L., & Laidler, V. G. (2005). Making the most of missing values: Object clustering with partial data in astronomy. In *Astronomical data analysis software and systems XIV*, ASP Conference Series, pp. 172–176. Astronomical Society of the Pacific.

Wang, Q., & Rao, J. N. K. (2002a). Empirical likelihood-based inference in linear models with missing data. *Scandinavian Journal of Statistics*, *29*(3), 563–576.

Wang, Q., & Rao, J. N. K. (2002b). Empirical likelihood-based inference under imputation for missing response data. *The Annals of Statistics*, *30*(3), 896–924.

Weatherill, G., & Burton, P. W. (2009). Delineation of shallow seismic source zones using k-means cluster analysis, with application to the aegean region. *Geophysical Journal International*, *176*(2), 565–588.

Wendel, R. E., & Hurter, A. P., Jr. (1976). Minimization of a non-separable objective function subject to disjoint constraints. *Operations Research*, *24*, 643–657.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, *1*(6), 80–83.

Zhang, W., Yang, Y., & Wang, Q. (2012). A comparative study of absent features and unobserved values in software effort data. *International Journal of Software Engineering and Knowledge Engineering*, *22*(02), 185–202.