Hierarchical Bayesian Site Clustering for a Geotechnical Database

Jianye Ching
Professor, Dept. of Civil Engineering, National Taiwan University, Taipei, Taiwan

Atma Sharma
Research Assistant, Dept. of Civil Engineering, National Taiwan University, Taipei, Taiwan

ABSTRACT: This paper presents a new clustering method for sites in a geotechnical database. This clustering method is based on the hierarchical Bayesian site similarity measure (HBSSM) recently proposed by the authors. Utilizing the HBSSM, this study presents a spectral algorithm for clustering geotechnical sites with similar multivariate probability distributions. Using numerical examples, it is shown that the proposed algorithm can produce reasonable clustering results.

1. INTRODUCTION

Data-driven methods are growing in popularity in geotechnical engineering (Phoon et al. 2019; Yuen et al. 2021; Phoon and Ching, 2021; Phoon and Zhang, 2022; Phoon et al. 2022a, b, c; Wang et al. 2022). While physics-based methods are desirable, it can be difficult to gain insight in situations where the physical mechanisms are unclear. In these cases, data-driven methods prove useful, such as in the characterization of a geotechnical site's subsurface using limited data (Wang et al. 2022). The importance of data in the advancement and application of data-driven methods is clear, though the role of physics in these approaches also requires investigation (Tao et al. 2023).

Geotechnical databases play a crucial role in data-driven methodologies. In situations where site-specific data is limited, which is often the case in geotechnical engineering, databases provide valuable preliminary/prior information. To foster the creation and accessibility of these databases, the Technical Committee on Engineering Practices for Risk Assessment and Management (TC304) under the International Society for Soil Mechanics and Geotechnical Engineering (ISSMGE) recommended the launch of the 304dB soil database project in 2017. Several geotechnical databases are now accessible through the 304dB website (TC304 2018). Along with databases of soil/rock properties, performance databases for data-driven research are also available (e.g. Hancock 2018; Phoon and Tang 2019; Tang and Phoon 2021).

While most databases, such as TC304 (2018), contain data from various sites across different countries, geotechnical planning and decision-making require site-specific data. Ideally, only data specific to the site should be used for characterization. If ample site-specific data is available, site-specific data-driven models can be created. However, in geotechnical projects, data is often sparse, incomplete, or corrupted, leading to significant statistical uncertainty. Phoon et al. (2019) described this type of data as MUSIC-X (Multivariate, Uncertain, Unique, Sparse, Incomplete, and Corrupt, with X denoting the spatial/temporal dimension). One solution to this issue is to supplement site-specific data with data from common databases.

Phoon et al. (2022c) defined "data-driven site characterization" (DDSC) as a site characterization method relying solely on measured data, including site-specific data and pre-existing data from the other projects. Three challenges were identified in DDSC: (1) handling of poor quality data, (2) site recognition, and (3) stratification. Site recognition involves quantification of "site-uniqueness" in a global database. Classical probabilistic multiple regression methods (PMR) combine general databases with site-specific data without taking into account the "unique" attribute of site (Phoon
and Ching 2022). PMR does not consider site labels in the database. Towards the site recognition challenge, recently the authors of this study proposed a Hierarchical Bayesian site similarity measure (HBSSM) in Sharma et al. (2022). The HBSSM quantifies site uniqueness, allowing technicians to tailor regression results to specific site conditions based on local experience.

Although it is recognized that every geotechnical site is unique, in geotechnical practice, grouping sites into “regions” is quite common. Data-driven clustering/grouping can result in solutions beyond traditional geographical boundaries. The clustering approach identifies a subset of a general purpose database for decision-making. Engineers often review site survey reports from neighboring site groups, typically identified through local knowledge and judgment. Thus, site clustering is relevant with current practice.

Clustering is a class of unsupervised learning problems whose goal is to explore groups in a dataset. The application of clustering in geotechnical engineering has a long history dating back to the 1970s (e.g., Cubitt et al. 1978; Judd et al. 1980; Imamura 1994; Hegazy 1998; Hegazy and Mayne 2002; Liao and Mayne 2007; Han et al. 2022; Collico et al. 2022; Wu et al. 2022). Based on the type of data to be clustered, most of the above work in geotechnical engineering falls into two categories: (1) clustering using single-site data and, (2) clustering using multi-site data. Type 1 is sometimes called intra-site clustering because the clustering is based on the similarity between records within a single site. For example, the aim of Hegazy and Mayne (2002) is to delineate different subsurface zones of a site based on Cone Penetration Test (CPT) data. In type 2 clustering, data from multiple sites are combined together in the form of a database (e.g., Collico et al. 2022; Han et al. 2022) and the purpose is to cluster similar records. Type 1 clustering is common in geotechnical engineering, but type 2 clustering has attracted attention only in recent years. Collico et al. (2022) improved prediction of soil unit weight using clustering analysis on the database. Han et al. (2022) performed clustering on the NUS/SpreadFound/919 lift load test database and adjusted model factors. It is to be noted that the works of Collico et al. (2022) and Han et al. (2022) do not involve the notion of site: records from different sites are aggregated and similar records are grouped together. Moreover, most previous studies related to type 2 clustering have been limited to univariate full records and may not be readily generalizable to MUSIC data. Geotechnical site clustering in this study is closely related to type 2. That is, data from multiple sites are considered. However, instead of grouping similar records (“record-level clustering”), we group similar sites according to HBSSM (“site-level clustering”).

Site clustering requires three components; (1) a site-labelled geotechnical database, (2) a similarity measure between sites, and (3) a clustering algorithm. The main focus of this study is related to the component 3. Component 2 was the main focus in Sharma et al. (2022). In the current work, a spectral clustering algorithm based on HBSSM is proposed. Using numerical examples, it is demonstrated that the proposed HBSSM clustering algorithm can provide reasonable results. This algorithm can be applied to a site-labelled database with sparse and incomplete (MUSIC) data.

2. CLUSTERING ALGORITHM

The proposed site clustering algorithm, is based on spectral clustering. It is chosen over other popular clustering algorithms such as k-means (Lyod 1982) because it enables clustering using a customized similarity measure like HBSSM, and there is no strict requirement for the similarity measure to be based on a distance metric. Spectral clustering is a simple and widely used clustering algorithm in machine learning (Ng et al. 2001). In this study, HBSSM is used to develop a site clustering algorithm. Let us assume that $n$ soil properties are of concern and there are $m_i$ observed records at the $i$th site. Let $\mathbf{y}_{ik} \in \mathbb{R}^{nx1}$ denote the $k$-th record ($k = 1, \ldots, m_i$) at the $i$th
As a result, \( Y_i = [y_{i1}, y_{i2}, \ldots, y_{im}] \in \mathbb{R}^{n \times m_i} \) denote the collection of all data from the \( i \)th (\( i = 1:n_s \)) site. Let \( X_i \) denotes the \( Y_i \) data transformed into normal space using Johnson transformation (Ching et al. 2014). Let us further denote the collection of all database data by \( X_g = \{X_1, X_2, \ldots, X_{n_s}\} \). The clustering algorithm is outlined as follows.

**Input:** Database \( X_g = \{X_1, X_2, \ldots, X_{n_s}\} \) and number of clusters \( k \).

**Output:** Clusters \( C_1, C_2, \ldots, C_k \).

**Step 1:** Construction of pairwise similarity matrix \( S^{sym} \) using HBSSM

- For \( j = 1:n_s \) and \( i = 1:n_s \), calculate \( S^r \in \mathbb{R}^{n_s \times n_s} \), where each matrix entry \( S^r_{ij} = S_{ij}/S_{ji} \), where \( S_{ij} \) is the similarity between the \( i \)th and \( j \)th site obtained based on HBSSM.
- Construct the symmetric matrix \( S^{sym} \in \mathbb{R}^{n_s \times n_s} \), where \( S^{sym}_{ij} = \sqrt{S^r_{ij}S^r_{ji}} \).

**Step 2:** Transformation to obtain normalized Laplacian matrix \( L_{norm} \in \mathbb{R}^{n_s \times n_s} \)

- Construct the affinity matrix \( A = S^{sym} \)
- Construct the unnormalized Laplacian matrix \( L = D - A \), \( D_{ii} = \sum_{j(i \neq j)}A_{ij} \) and \( D_{ij} = 0 \), if \( i \neq j \).
- Compute the \( L_{norm} = D^{-\frac{1}{2}}LD^{-\frac{1}{2}} \).

**Step 3:** Spectral decomposition of \( L_{norm} \)

- Compute \( U \in \mathbb{R}^{n_s \times k} = [u_1, u_2, \ldots, u_k] \), the first \( k \) eigen vectors of \( L_{norm} \).
- Construct the matrix \( T \in \mathbb{R}^{n_s \times k} \), where \( T_{ij} = U_{ij}/(\sum_j U_{ij}^2)^{1/2} \).

**Step 4:** Clustering in eigen vector space

- For \( i = 1:n_s \), let \( t_{i} \in \mathbb{R}^{k \times 1} \) be the \( i \)th column of the transpose of \( T \).
- Perform k-means clustering on \( \{t_i; i = 1:n_s\} \) to obtain the clusters \( C_1, C_2, \ldots, C_k \).

The clustering algorithm is based on \( S^{sym} \) constructed using HBSSM. It is expected that the sites assigned to the same cluster exhibit similar site-specific probability densities, and sites assigned to different clusters exhibit dissimilar site-specific probability densities.

3. **NUMERICAL EXAMPLES**

In this section, the performance of the proposed clustering algorithm is evaluated through numerical examples in 2-dimensional space. The examples are designed to demonstrate the ability of the algorithm to detect clusters with differences in mean, variance, and correlation. The first set of examples is based on complete and abundant data, representing an ideal scenario, while the second set of examples deals with sparse and incomplete data, which is a more realistic scenario. The aim is to demonstrate the effectiveness of the algorithm in both scenarios.

3.1. **Clusters with different means**

For the demonstration example, three synthetic clusters were created using three HBM models with varying hyperparameters \( \Theta \). \( \Theta = \{\mu_0, C_0, \nu_0, \Sigma_0\} \), where \( \{\mu_0, C_0\} \) govern the probability distribution of \( \{\mu_0, \mu_1, \ldots, \mu_{n_s}\} \), and \( \{\nu_0, \Sigma_0\} \) govern the probability distribution of \( \{C_1, C_2, \ldots, C_{n_s}\} \). \( \mu_i \) (site-specific mean vector) and \( C_i \) (site-specific covariance matrix) model the intra-site variability, and the hyper-parameters \( \Theta \) model the inter-site variability. Please refer to Ching et al. (2021) for more details about the HBM structure. The reason for using an HBM structure to generate \( X_g \) was for convenience, but the clusters could have also been generated in any other way without relying on an HBM structure. \( \mu_0 \) for the three clusters are taken as \( \mu_0 = [-4, -4] \), \( \mu_{1} = [0, 0] \) and \( \mu_{3} = [4, 4] \). All other hyperparameters except \( \mu_0 \) are fixed: \( C_0 = 1 \), and \( \{C_i = 1: i = 1, \ldots, n_s\} \) (1 is the identity matrix). 20 sites are independently generated for each cluster, and for each site 50 (bivariate) records are independently generated. The collection of the generated data is treated as the database \( (X_g) \).

Fig. 1(a) shows the generated \( X_g \), where different colors indicate different clusters. Now using step 1 of the clustering algorithm, similarity
matrix \( S_{sym} \) is calculated for \( X_g \). Fig. 1(b) presents the site-specific joint densities \( f_{i|X_g}(x) \) for the 60 generated sites i.e., \( i = 1:60 \) and \( \max(S_{sym}) \) and \( \min(S_{sym}) \) to the interval [0, 1]. Note that the HBSSM does not need the cluster-label information. The cluster-label information (colors in Figs. 1[a-b]) is regarded as unknown during the computation of \( S_{sym} \). Fig. 1(c) presents the obtained \( S_{sym} \). Bright diagonal spots denote the self-similarity values \( S_{ii}^{sym} = 1 \). The color intensity is scaled between \( \max(S_{sym}) \) and \( \min(S_{sym}) \) to the interval [0, 1].

Figure 1: Demonstration of the presented clustering algorithm to detect clusters with different means: (a) generated clusters; (b) estimated densities \( f_{i|X_g}(x) \) for \( i = 1:60 \); (c) similarity matrix \( S_{sym} \); and (d) identified clusters.

Three clusters can be identified visually in Fig. 1(c). The clusters obtained by the proposed clustering algorithm with \( k = 3 \) are shown in Fig. 1(d). It should be noted that the results in Fig. 1(d) are obtained using site clustering (60 sites), not record clustering. It should be noted that the obtained clusters in Fig. 1(d) resemble the actual clusters in Fig. 1(a). Overall, the results in Fig. 1 indicate that the proposed clustering algorithm is sensitive to differences in cluster mean position.

3.2. Clusters with different variances
It is also important to assess the proposed method's performance in differentiating between clusters that differ not in their mean positions but in their variance or correlation (shape). To investigate the capability to detect clusters with different variances, \( \{\mu_i = [0,0]; i = 1, ..., n_3\} \) and \( \nu_0 = 8 \) are fixed. \( \Sigma_0 \) for the three clusters was taken as \( \Sigma_{01} = 5I, \Sigma_{02} = 2I \) and \( \Sigma_{03} = 0.2I \). Fig. 2 presents the clustering analysis and results. In comparison to Fig. 1(c), there are more off-diagonal bright spots in Fig. 2(c). In Fig. 2(d), it can be observed that the clustering algorithm can roughly reproduce the actual clusters in Fig. 2(a).

Figure 2: Demonstration of the presented clustering algorithm to detect clusters with different variances: (a) generated clusters; (b) estimated densities \( f_{i|X_g}(x) \) for \( i = 1:60 \); (c) similarity matrix \( S_{sym} \); and (d) identified clusters.

3.3. Clusters with different correlations
To investigate the capability to detect clusters with different correlations, \( \{\mu_i = [0,0]; i = 1, ..., n_3\} \) and \( \nu_0 = 8 \) are fixed. \( \Sigma_0 \) for the three clusters was taken as \( \Sigma_{01} = 5[1 - 0.95; -0.95 1], \Sigma_{02} = 5[1 0; 0 1] \) and \( \Sigma_{03} = 5[1 0.95; 0.95 1] \). Fig. 3(a) shows the generated clusters. In Fig. 3(d), it is shown that the proposed clustering algorithm can reproduce the actual clusters in Fig. 3(d). Based on the overall results
in Figs. 1-3, it can be concluded that the presented algorithm is effective in clustering the sites according to their actual underlying clusters for the numerical examples.

![Figure 3](image.png)

**Figure 3:** Demonstration of the presented clustering algorithm to detect clusters with different correlations: (a) generated clusters; (b) estimated densities $f_i|X_0(x)$ for $i = 1:60$; (c) similarity matrix $S^{sym}$; and (d) identified clusters.

### 3.4. Clusters with sparse (limited) data

The results shown in Figs. 1-3 are based on a high amount of data, with 50 records per site. To examine the impact of limited data, the same experiment in Figs. 1-3 was conducted, but this time with varying numbers of records per site. Fig. 4 displays the clustering results, with the three rows representing the impact on detecting differences in mean, variance, and correlation for different amounts of records per site ($m_s$). As seen in Fig. 4, as data becomes sparser, the number of incorrectly clustered sites increases due to increased statistical uncertainty.

#### 3.5. Clusters with incomplete (missing) data

The proposed clustering method can also handle missing data. To investigate the impact of missing data, a similar experiment as in Fig. 4 was conducted. Starting from a reference number of records ($m_s = 50$), incompleteness was introduced by randomly deleting some of the original records. Fig. 5 displays the clustering results, where $\%NaN$ represents the percentage of missing records. For example, $\%NaN=10$ means that 10 entries (out of 100) are randomly missing. As noted with the limited data case in Fig. 4, the number of incorrectly clustered sites increases with an increase in $\%NaN$ due to increased statistical uncertainty, as seen in Fig. 5.

![Figure 4](image.png)

**Figure 4:** Impact of sparse data on detecting cluster differences in (row a) mean, (row b) variance, and (row c) correlation.
4. COMPARISON TO CONVENTIONAL RECORD CLUSTERING

This section compares the “site clustering” approach presented in the current study with the traditional “record clustering” method that has been documented in previous literature (e.g. Collico et al. 2022; Han et al. 2022). If the similarity matrix in the first step is calculated based on the similarity between records while ignoring the site labels, the algorithm is referred to as “spectral record clustering”. The comparison between the two methods was made using similar numerical examples as shown in Figs. 1-3. The results of the comparison are presented in Fig. 6. The figure has three rows that represent clusters with varying mean, variance, and correlation. As seen in Fig. 6, it can be concluded that the spectral record clustering method fails to recreate the original site clusters. It is important to note that “site clustering” and “record clustering” are different concepts and if the goal is to cluster sites, a site clustering method (such as the one presented in this study) should be adopted. Using a record clustering method like spectral record clustering or k-means record clustering may result in irrelevant outcomes. This preliminary clustering study highlights the importance of retaining the site label in a geotechnical database as it is crucial for obtaining accurate results.

Figure 5: Impact of incomplete data on detecting cluster differences in (row a) mean, (row b) variance, and (row c) correlation.

Figure 6: Comparison of the proposed method with the conventional record clustering method to detect clusters that differ in mean (1st row), variance (2nd row), and correlation (3rd row): (a) generated site clusters; (b) clusters obtained by spectral record clustering and; (c) clusters obtained by the proposed spectral site clustering method.
5. CONCLUSIONS
Sharma et al. (2022) presented a hierarchical Bayesian site similarity measure (HBSSM) for retrieving geotechnical sites. The HBSSM was used to determine the similarity between a target site and a database site. However, the current study has a different objective, which is to group sites within a geotechnical database. The spectral clustering method based on the HBSSM was proposed for site clustering in a geotechnical database. The results of numerical examples indicate that the proposed site clustering algorithm can produce acceptable clustering results. Note that the focus of this study is “site clustering” as opposed to the traditional approach of “record clustering” that ignores site labels.

6. REFERENCES
Assessment and Management of Risk for Engineered Systems and Geohazards, 1-16.


