1. Introduction

In epidemiological studies of infectious diseases, disease symptoms rarely occur in isolation but instead exhibit certain co-occurrence patterns. These patterns not only provide crucial clues for disease diagnosis but also serve as a basis for understanding underlying pathological mechanisms and developing effective public health strategies. For example, in COVID-19, loss of smell and taste often occur simultaneously, and this symptom combination may reflect the virus's direct impact on the olfactory system or immune-mediated neural dysfunction (Menni et al., 2020). Therefore, revealing the co-occurrence relationships between symptoms not only aids in the early identification and classification of diseases but also provides scientific support for personalized treatment and precise prevention and control (Sudre et al., 2021).

However, the occurrence and development of symptoms often exhibit high individual variability. Even when infected with the same pathogen, different individuals may exhibit significant differences in symptom type, number, duration, and severity. This heterogeneity may stem from various factors, including the host's genetic background, immune response capacity, prior health status, comorbidities, and the pathogen's variant characteristics. Additionally, symptoms may change over time, making symptom analysis based on a single time point insufficient to fully reflect the disease's true progression.

To more systematically characterize the structure and patterns of symptom evolution, researchers have begun to explore high-dimensional data analysis methods, integrating multiple symptoms and their temporal evolution into a unified analytical framework. In recent years, with the rapid accumulation of real-world data, particularly large-scale longitudinal symptom surveys and the widespread adoption of mobile health applications, researchers have gained access to datasets containing multi-dimensional features and long-term temporal spans. For example, Fyles et al. (2023) utilized multiple UK community infection datasets (including NHS Test & Trace and the COVID Symptom Study APP) and applied unsupervised clustering and dimensionality reduction methods to reveal diverse symptom phenotypes in SARS-CoV-2 infections. Their findings indicate that the primary variation in symptoms is not only related to quantity but also exhibits distinct typological groupings, such as respiratory symptoms, gastrointestinal symptoms, and other nonspecific symptoms, with these patterns differing across age groups. This discovery underscores the multidimensional nature of symptom structure and the influence of demographic factors on symptom patterns.

Dimension reduction methods play a crucial role in analyzing such high-dimensional symptom data. Traditional linear dimension reduction methods, such as Principal Component Analysis (PCA) (Jolliffe, 2002), can reduce dimensions while preserving the main variance of the data, thereby simplifying subsequent analysis. However, when data exhibit complex nonlinear manifold structures, PCA often struggles to capture underlying patterns. To address this, researchers have introduced nonlinear dimensionality reduction methods, such as t-distributed stochastic neighborhood embedding (t-SNE, van der Maaten & Hinton, 2008) and unified manifold approximation and projection (UMAP, McInnes et al., 2018), which excel at preserving the local neighborhood structure of data and revealing latent clustering structures, particularly for handling high-dimensional sparse data such as symptom co-occurrence matrices.

Additionally, non-negative matrix factorization (NMF; Lee & Seung, 1999) is favored in clinical data analysis due to the interpretability of its results. NMF decomposes the original data into two non-negative matrices: one representing feature components (e.g., symptom combinations) and the other representing component weights (e.g., the intensity of a patient's performance across symptom combinations), facilitating clinical interpretation and pattern recognition. When comparing symptom structures across different groups or time periods, methods like Aligned UMAP can construct embeddings that are comparable across multiple groups, thereby revealing structural differences between groups (McInnes et al., 2020).

Virtual data generation technology can simulate the structural characteristics of real data while protecting privacy. Researchers can generate simulated datasets that meet research requirements by setting data distributions, variable relationships, and noise levels (Patki et al., 2016). This method not only allows for the evaluation of different algorithms under controlled conditions but also enables systematic testing of method robustness across various data characteristics. For this study, the use of virtual data facilitates comparative experiments across multiple methods in an environment free of ethical risks and provides technical foundations for future transitions to real disease data analysis.

This study aims to use virtual data to systematically compare the performance of various dimension reduction and clustering methods in identifying symptom co-occurrence patterns. We designed two complementary experiments: (1) continuous trajectory-type nonlinear embedding data to simulate the dynamic changes of symptoms over time and test the methods' ability to recover complex structures; (2) sparse binary-type symptom co-occurrence data to simulate the sparsity and heterogeneity of real clinical data. On these two types of data, we will evaluate the performance of PCA, t-SNE, UMAP, NMF dimensionality reduction methods, as well as clustering

algorithms such as KMeans and hierarchical clustering. Through experimental comparisons across different data structures, we aim to provide reliable methodological foundations and practical guidance for the analysis of real longitudinal symptom data.

2. Literature Review

2.1 Research Background and Symptom Co-occurrence Patterns

In infectious disease epidemiology research, symptom co-occurrence patterns are an important entry point for understanding disease pathophysiology and clinical manifestations (Menni et al., 2020). The associations between symptoms not only aid in identifying specific disease subtypes but also provide scientific basis for personalized treatment. Taking COVID-19 as an example, numerous studies have found that its symptom presentation exhibits high diversity, with common symptom combinations including loss of smell and taste, fever and cough, diarrhea and decreased appetite, etc. (Sudre et al., 2021). These patterns may reflect underlying biological processes such as viral entry pathways, receptor distribution, and immune responses.

Fyles et al. (2023) utilized unsupervised clustering methods based on four community datasets from the UK to reveal different types of symptom phenotypes in SARS-CoV-2 infections. They found that, in addition to the main axis of symptom quantity, there were three major patterns: respiratory symptoms, gastrointestinal symptoms, and other symptoms. The distribution of these patterns also showed significant differences across age groups. This finding highlights the multidimensional nature of symptom co-occurrence structures and their association with demographic factors.

2.2 Overview of Dimension Reduction Methods

(1) Principal Component Analysis (PCA)

PCA is the most classic linear dimension reduction method, aiming to map high-dimensional data to a low-dimensional space through linear transformation while retaining the direction with the largest variance in the data (Jolliffe, 2002). In symptom co-occurrence analysis, PCA can be used to extract the principal components that explain symptom variation. However, due to its assumption that the relationships between variables are linear, its effectiveness is limited when dealing with highly nonlinear or manifold-structured data.

(2) t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-SNE is a nonlinear dimension reduction method particularly suited for visualizing high-dimensional data (van der Maaten & Hinton, 2008). It achieves the preservation of local structure by minimizing the difference in the distribution of data point neighborhoods between high-dimensional and low-dimensional spaces. t-SNE can reveal local clustering structures in symptom data analysis, but it lacks the ability to preserve global distances and is sensitive to parameters such as perplexity.

(3) Unified Manifold Approximation and Projection (UMAP)

UMAP is based on manifold learning theory and aims to simultaneously preserve both the local and global structures of data (McInnes et al., 2018). Compared to t-SNE, UMAP is more efficient when handling large datasets and performs better in terms of global structure preservation. UMAP is particularly suitable for revealing complex multi-cluster structures in symptom co-occurrence analysis, and its embedding results exhibit good stability.

(4) Non-negative Matrix Factorization (NMF)

NMF is a matrix decomposition method that decomposes original non-negative data into two non-negative matrices to extract interpretable latent components (Lee & Seung, 1999). In symptom data, the components extracted by NMF often correspond to specific symptom combination patterns, facilitating clinical interpretation. Its advantages include strong interpretability, but its disadvantages include sensitivity to noise and the need for reasonable setting of the number of components.

2.3 Clustering Methods and Similarity Measures

(1) K-Means Clustering

K-Means is a centroid-based clustering method that groups data by minimizing the within-cluster sum of squares (WCSS). Its advantages include high computational efficiency and simplicity of implementation, but it requires the number of clusters to be specified in advance and is sensitive to noise and outliers (MacQueen, 1967).

(2) Hierarchical Clustering

Hierarchical clustering constructs a hierarchical clustering structure by recursively merging (agglomerative) or splitting (divisive) clusters. In symptom

co-occurrence analysis, the advantage of hierarchical clustering lies in its ability to generate a dendrogram, which visually illustrates the similarity relationships between different symptoms. Its drawback is high computational complexity, particularly in large-scale datasets.

(3) Similarity Measures

In binary symptom data, the Jaccard distance is a commonly used similarity measure (Jaccard, 1901). It calculates similarity by determining the ratio of the intersection to the union of two sets, making it suitable for handling sparse data. Other commonly used similarity metrics include cosine similarity and Hamming distance, each suited for different data characteristics and analysis objectives.

2.4 Application of Virtual Data in Method Validation

In medical data analysis, data privacy and ethical constraints often pose obstacles to research progress (ONS, 2021). Virtual data (synthetic data) offers a feasible solution to this problem. By simulating the distribution and structural characteristics of real data, virtual data can be used for algorithm development and method validation without involving real personal information (Patki et al., 2016).

The advantages of virtual data include: (1) avoiding ethical review delays; (2) controllable data feature settings, facilitating the evaluation of method performance across different scenarios; and (3) high reproducibility, promoting transparency and verifiability in scientific research. In symptom co-occurrence pattern analysis, virtual data can be used to construct simulated samples with known clustering structures, thereby validating the effectiveness and robustness of dimension reduction and clustering methods.

3. Methodology

3.1 Experimental Design

The experimental part of this study is based on two types of virtual datasets, namely the Trajectory dataset and the Binary dataset, aiming to simulate the performance characteristics of different types of feature data during dimension reduction and clustering processes, and to provide methodological references for disease symptom co-occurrence analysis through systematic experiments. The Trajectory dataset consists of two-dimensional trajectories, where different drift vectors (J vectors) are introduced during the generation

process to create inter-class differences. Gaussian noise (σ=0.4) is added to increase data complexity and diversity. Each trajectory has a fixed length of 40 time steps, with initial positions randomly generated from a uniform distribution. Coordinates are then iteratively updated according to the given drift and noise.

The Binary dataset consists of high-dimensional sparse binary features, where each feature dimension represents the presence or absence of a specific symptom within a specific time window (1 indicates presence, 0 indicates absence). This data structure is widely found in clinical symptom records and survey questionnaires, and due to its high dimensionality and strong sparsity, it imposes stringent adaptability requirements on dimension reduction algorithms. In the dimensionality reduction stage, the binary dataset uses Jaccard distance to measure the similarity between samples, thereby more accurately reflecting the relationship between binary features in the embedding space.

In terms of experimental design, both types of datasets undergo a standardised processing workflow: first, data generation and preprocessing are performed, followed by the application of three methods—PCA, t-SNE, and UMAP—in the dimensionality reduction stage. For the binary dataset, a further UMAP parameter sensitivity analysis was conducted, selecting n_neighbors and min_dist as tuning variables. The clustering stage uniformly employed unsupervised K-means and agglomerative clustering methods, with clustering performance quantitatively evaluated using metrics such as the confusion matrix.

The advantage of this experimental design lies in two aspects: on the one hand, it allows for direct comparison of the adaptability and performance differences of different dimensionality reduction methods across two feature types; on the other hand, through hyperparameter tuning analysis, it identifies the optimal configuration range for UMAP across different data types, providing a reference for future real-world data analysis.

The overall process of this study includes:

1. Data Generation: Generate trajectory-type continuous data and sparse binary data using Python under controlled conditions.

2. Data Preprocessing: Standardize different types of data and calculate appropriate similarity metrics.

3. Dimensionality Reduction: In this phase, we will test various classical and emerging dimensionality reduction algorithms and explore their performance

under different data structures, including PCA, t-SNE, UMAP, NMF, and Aligned UMAP (for trajectory data only).

4. Clustering and Result Visualization: This is the visualization and comparative analysis stage of the embedding results, where graphical methods are used to intuitively present the embedding effects of different methods, combined with quantitative metrics for comparison.

3.2. Data Generation

3.2.1 Trajectory-based Synthetic Data

Trajectory-based data aims to simulate the continuous change patterns of symptoms over time. The generation process references trajectory construction methods from nonlinear manifold learning research (McInnes et al., 2018), with the following specific implementation details:

• Number of trajectory categories (n_classes): 3, with each category representing a distinct symptom evolution pattern corresponding to different disease progression trends.

• Number of trajectories per class (ntraj_per_class): 200, with a total number of trajectories ntraj = n_classes × ntraj_per_class = 600, ensuring sufficient representation of each category in the sample.

• Trajectory length (trajlength): 40, representing the number of time steps, corresponding to observations of symptoms at 40 time points.

• Drift vectors (drift vectors, J_options):

Category 1: J = (0.2, 0.1), simulating positive growth of symptoms across two feature dimensions.

Category 2: J = (-0.1, 0.2), simulating a trend of decreasing symptoms in the first feature dimension and increasing symptoms in the second dimension.

Category 3: J = (0.15, -0.15), simulating a pattern where the two feature dimensions move in opposite directions.

• Random noise intensity (sigma): 0.4, used to introduce individual differences and measurement errors, making the trajectories more diverse.

• Initial position: Randomly and uniformly distributed in the two-dimensional space [-1, 1], ensuring diversity in the initial state.

• Trajectory generation formula: X[i, j] = X[i, j-1] + J + np.random.normal(0, sigma, size=2), where np.random.normal(0, sigma, size=2) represents Gaussian noise with a mean of zero and covariance of sigma.

• Three-dimensional mapping: Map the two-dimensional trajectory to a three-dimensional S-curve space using the following mapping rules:

Z[i, j, 0] = np.sin(x_val / 2.0)

Z[i, j, 1] = 2.0 * y_val

Z[i, j, 2] = np.sign(x_val) * (np.cos(x_val / 2)-1)

This nonlinear mapping preserves the temporal sequence information of the original trajectory while adding geometric complexity, making the dimensionality reduction task more challenging.

2.2 Sparse Binary Data (Binary Symptom Co-occurrence Data)

Sparse binary data simulates the sparsity and heterogeneity commonly found in real clinical symptom records (Fyles et al., 2023). The generation process is as follows:

• Number of samples (n_samples): 1200, corresponding to 1200 patient samples.

• Number of symptoms (n_symptoms): 11, corresponding to 50 different symptom variables.

• Number of categories (n_clusters): 3, with each category corresponding to a typical symptom combination pattern.

• Probability of symptom occurrence within each cluster: For example, the probability of symptoms within a cluster p_in = 0.7, and the probability of symptoms outside a cluster p_out = 0.1, ensuring that the correlation of symptoms within a cluster is significantly higher than that outside the cluster.

• Data generation method: First, randomly assign cluster labels to each sample, then generate a 0/1 binary matrix using a binomial distribution based on the probability distribution of the assigned cluster.

• Sparsity control: Adjust the difference between p_in and p_out to control overall sparsity, thereby simulating the symptom distribution characteristics of different diseases or populations.

3.3 Data Preprocessing

The preprocessing steps for trajectory data include: flattening each trajectory into a high-dimensional vector of length trajlength × 3, thereby converting the time-series data into static feature vectors suitable for standard dimension reduction methods; Using z-score standardisation (sklearn.preprocessing.StandardScaler) to eliminate differences in feature scales, ensuring that dimensionality reduction algorithms are not biased by differences in feature scales (Jolliffe, 2002).

Preprocessing of binary co-occurrence data includes: calculating the Jaccard similarity between pairs of samples (sklearn.metrics.jaccard_score, average="binary"), whose formula is $J(A,B) = |A \cap B| / |A \cup B|$ (Jaccard, 1901; Choi et al., 2010). After obtaining the similarity matrix, it is converted into a distance matrix (1 - similarity) as input for distance-based dimensionality reduction methods (e.g., UMAP) to better reflect the sparsity characteristics of binary data.

3.4 Dimensionality Reduction Methods

This study employed four dimension reduction methods, including Principal Component Analysis (PCA), t-Distributed Stochastic Neighbourhood Embedding (t-SNE), Uniform Manifold Approximation and Projection (UMAP), and Non-Negative Matrix Factorisation (NMF), to encompass a range of dimension reduction strategies from linear to non-linear and from geometric mapping to matrix factorisation, thereby enabling a comprehensive performance comparison across different data structures.

(1) Principal Component Analysis (PCA) was implemented using sklearn.decomposition.PCA, with the number of principal components set to n_components=2 to project high-dimensional data into a two-dimensional space for visualisation and subsequent clustering. PCA is based on Singular Value Decomposition (SVD) to extract orthogonal directions that maximise data variance (Jolliffe, 2002). This method assumes that the data structure is approximately linear, so it can effectively capture the global variance structure in the trajectory data of this study. However, it may lose neighbourhood information when dealing with highly non-linear structures.

(2) t-distributed stochastic neighbour embedding (t-SNE) is implemented using sklearn.manifold.TSNE, with n_components=2, perplexity=30, learning rate=200, and n_iter=1000. t-SNE preserves local structure by minimising the Kullback–Leibler (KL) divergence between high-dimensional and low-dimensional neighbourhood distributions (van der Maaten & Hinton, 2008). The perplexity controls the number of effective neighbours considered for each sample when constructing the high-dimensional neighbourhood; a smaller value emphasises local cluster structure, while a larger value retains more global patterns; The learning rate determines the step size of gradient

descent. A value that is too small may result in slow convergence or even getting stuck in a local optimum, while a value that is too large may cause instability in the embedding. The parameters selected in this study strike a balance between maintaining the clarity of local clustering and the readability of global structure.

(3) Unified Manifold Approximation and Projection (UMAP) is implemented using umap.UMAP, with n_neighbors=15, min_dist=0.1, and Euclidean distance as the metric in trajectory data; For binary data, sensitivity analysis was conducted on n_neighbors and min_dist, with the former taking values {3, 4, 5, 7, 10, 20} and the latter taking values {0.0005, 0.001, 0.01, 0.1, 0.2}, and the distance metric using Jaccard distance. UMAP is based on manifold learning and topological data analysis, estimating the manifold structure of data in high-dimensional space by constructing a weighted k-nearest neighbour graph, and optimising the fidelity of the topological structure in low-dimensional space (McInnes et al., 2018). The parameter n_neighbours controls the balance between local and global structures; smaller values emphasise the compactness of local clusters, while larger values enhance the retention of global relationships. min_dist controls the minimum distance between samples in the low-dimensional space; smaller values produce more compact cluster structures, while larger values increase the spacing within clusters, facilitating the presentation of fine-grained patterns. For binary data, the Jaccard distance more accurately captures the similarity of ‘simultaneous presence’ while avoiding false similarities caused by ‘simultaneous absence’ (Choi et al., 2010).

(4) Non-negative matrix factorisation (NMF) is implemented using sklearn.decomposition.NMF, with ncomponents=3 and the initialisation method set to an improved version of non-negative double singular value decomposition (nndsvda), with a maximum iteration count of 500. NMF decomposes the original non-negative matrix X into a non-negative basis matrix W and a coefficient matrix H, i.e., $X \approx WH$, and optimises by minimising the Frobenius norm or KL divergence (Lee & Seung, 1999). This decomposition method has good interpretability because the decomposed basis matrix can be regarded as the potential feature patterns of the data, and the coefficient matrix represents the weights of the samples on these patterns. The parameters selected in this study aim to ensure convergence stability on sparse data and interpretability of the decomposition results.

3.5 Clustering Methods

This study primarily employed two classic and widely used clustering algorithms—K-means clustering and agglomerative clustering—to evaluate the performance of different dimension reduction methods in revealing underlying cluster structures. These two types of algorithms represent

prototype-based and hierarchical-based clustering approaches, respectively, and can characterise the distribution features of samples in the low-dimensional space from different perspectives.

(1) K-means clustering (MacQueen, 1967) is an iterative optimisation algorithm aimed at partitioning samples into K predefined clusters, minimising the within-cluster sum of squares (WCSS). Its optimisation objective can be expressed as:

$$\min_{C_1,\ldots,C_K} \sum_{i=1}^{K} \sum_{x_j \in C_i} \|x_j - \mu_i\|^2$$

where Ci denotes the sample set of the i-th cluster, and μi is the centroid vector of that cluster. In this study, the sklearn.cluster.KMeans module was used for implementation, and k-means++ (Arthur & Vassilvitskii, 2007) was employed as the initialisation strategy to enhance convergence speed and mitigate the issue of local optima caused by initial centroid selection. Regarding specific parameters, the number of clusters K is chosen to match the actual number of categories in the dataset to enable subsequent comparison with actual labels using external metrics.

(2) Agglomerative clustering is a clustering method based on dendrogram decomposition, starting with each sample as a separate cluster and gradually merging the closest clusters until the predefined number of clusters is reached. This study uses sklearn.cluster.AgglomerativeClustering and selects the Ward linkage criterion (Ward, 1963) as the metric for inter-cluster distance to minimise the total squared error of the merged clusters. The merger cost can be expressed as:

$$\Delta(A, B) = \frac{|A| \cdot |B|}{|A| + |B|} \|\mu_A - \mu_B\|^2$$

where |A| and |B| are the number of samples in clusters A and B, respectively, and μA and μB are the corresponding cluster centres. In the reduced-dimensional space, this method effectively preserves the global structure of the data and provides hierarchical clustering results through a dendrogram for interpretation.

3.6 Model Evaluation

To quantitatively analyse the effectiveness of dimension reduction and clustering results, this study selected normalised mutual information (NMI)

and confusion matrix as the main performance evaluation metrics in the trajectory dataset experiment. These metrics can respectively measure the correspondence between the clustering results after dimensionality reduction and the true category labels from the perspectives of information theory and classification consistency, thereby evaluating the performance of different dimensionality reduction methods in revealing cluster structures.

Normalized Mutual Information (NMI) is an external clustering evaluation metric based on information theory, used to measure the mutual information between clustering labels UU and true labels VV. Through normalisation, its value range is limited to [0,1] (Vinh et al., 2010). Its formula is:

$$\mathrm{NMI}(U,V) = \frac{2 \cdot I(U;V)}{H(U) + H(V)}$$

where I(U;V)denotes the mutual information between the clustering labels and the true labels, and H(U) and H(V) represent the Shannon entropy of the two sets of labels, respectively. The closer the NMI value is to 1, the more consistent the clustering results are with the true labels; when NMI approaches 0, it indicates that the two are almost independent. This study implements the calculation of this metric using sklearn.metrics.normalised_mutual_info_score and conducts a horizontal comparison between the clustering results of different dimension reduction methods to assess their differences in preserving cluster structure.

A confusion matrix is a tool that directly reflects the correspondence between clustering results and true labels. Its matrix element Mij represents the number of samples with true category ii assigned to cluster label j. A confusion matrix not only displays clustering accuracy but also reveals specific misclassification patterns, which helps analyse the separability between clusters and potential sources of confusion. In this study, row normalisation was applied when constructing the confusion matrix, ensuring that the sum of each row equals 1, thereby enabling a more intuitive comparison of the distribution proportions of samples across different categories within the predicted clusters.

It should be noted that in the binary symptom dataset (Binary Dataset) experiments, due to data characteristics and experimental design constraints, external labels were not introduced, so NMI or confusion matrix calculations were not performed. Instead, the analysis primarily relied on dimensionality reduction and clustering visualisation results for qualitative analysis. This strategy is common in unlabelled data analysis, as it allows for an indirect assessment of the dimensionality reduction method's ability to capture

underlying patterns by observing the cluster distribution and sample aggregation in the low-dimensional space.

In summary, the model evaluation methods in this study employed quantitative analysis for labelled trajectory data and qualitative analysis for unlabelled binary data. This differentiated evaluation approach ensures the statistical interpretability of the results while also accommodating the applicability across different data types.

4. Results

4.1 Trajectory Data Experiment Results

The Trajectory dataset consists of three types of trajectories, with drift vectors set as J1 = [0.2, 0.1], J2 = [-0.1, 0.2], and J3 = [0.15, -0.15] for each type, and the same number of trajectories for each type. The evolutionary patterns of trajectories in two-dimensional space are jointly determined by cumulative drift and random noise, resulting in a cluster structure that exhibits both global separation trends and some local overlap. This data structure simulates the potential pattern changes in actual disease symptoms over time, retaining temporal dependency while introducing random fluctuations.

4.1.1 Comparison of basic dimensionality reduction results

In dimensionality reduction experiments, PCA, as a linear dimensionality reduction method, projects high-dimensional features by maximising the direction of data variance. On the Trajectory dataset, PCA can effectively retain global structural information, with different clusters showing a certain degree of separation in the two-dimensional plane. However, due to the strong nonlinear structure of the data, PCA slightly lacks clarity in cluster boundaries, with some samples located at cluster edges and exhibiting cross-cluster mixing.

t-SNE, as a nonlinear dimensionality reduction method, excels in preserving local structure by minimising the Kullback–Leibler divergence between high-dimensional and low-dimensional distributions. Experimental results show that t-SNE can tightly aggregate similar trajectories, with clear cluster shapes and large inter-cluster distances, which facilitates the separation of clustering algorithms. However, it should be noted that t-SNE has limited ability to present global structures, and the relative positions of different clusters lack stability, which may affect interpretability when compared with real-time sequence data in subsequent analyses.

UMAP balances local and global structures in this dataset. By constructing a graph structure based on neighbourhoods and optimising topological preservation in a low-dimensional space, the two-dimensional embeddings generated by UMAP retain the compactness of clusters while maintaining a correspondence with the high-dimensional structure in the global layout. Experimental results show that the cluster boundaries generated by UMAP are clear and stable, with reasonable distances between clusters, providing a solid embedding foundation for subsequent clustering.

Figure 1: Comparison of two-dimensional visualisation results of Trajectory data under PCA, t-SNE, and UMAP.

4.1.2 Clustering Performance Analysis

K-means and hierarchical clustering algorithms were applied in the embedding spaces generated by the three dimensionality reduction methods. The clustering results were evaluated using a confusion matrix. The results showed that UMAP had the highest clustering purity, with most clusters corresponding one-to-one with the true labels; t-SNE had a high degree of cluster separation visually, but some samples were misclassified in the clustering labels; PCA had relatively low clustering accuracy due to blurred cluster boundaries.

Further analysis revealed that UMAP's stability advantage lies in its geometric structure in the embedding space, which is more conducive to clustering under Euclidean distance metrics. For trajectory data with strong time dependency, maintaining a balance between the overall trajectory trend and local fluctuation patterns is critical for clustering performance. UMAP's local neighbourhood retention mechanism effectively compresses similar trajectories into neighbouring positions in the low-dimensional space, thereby improving the performance of the clustering algorithm. Additionally, the calculated NMI values show that UMAP embeddings have the highest consistency between clustering labels and true labels.

Figure 2 : Comparison of clustering confusion matrices for trajectory data under three dimensionality reduction methods.

Overall, the experimental results on the Trajectory dataset indicate that UMAP has a significant advantage in tasks combining dimensionality reduction and clustering when handling data with distinct temporal dynamic characteristics. This conclusion provides important reference for method selection in subsequent analyses of disease symptom time-series data, particularly in

tasks requiring the simultaneous preservation of local similarity and global patterns.

4.2 Binary Data Experiment Results

The binary dataset consists of high-dimensional sparse binary matrices, where each dimension corresponds to the presence (1) or absence (0) of a particular symptom during a specific time period. This data type is characterised by a large number of dimensions and a high proportion of zero elements, i.e., strong sparsity. In practical applications, this data structure is widely found in clinical symptom survey questionnaires, electronic health record (EHR) coding data, and public health monitoring data. The analysis results are of significant importance for identifying and classifying symptom co-occurrence patterns.

To better accommodate the characteristics of this high-dimensional sparse data, this study selected the Jaccard distance as the similarity metric between samples when using UMAP for dimensionality reduction. The Jaccard distance measures the similarity between two samples based on the ‘set of features with a value of 1,’ making it particularly suitable for similarity calculations involving binary data. Compared to the risk of distortion in sparse data when using Euclidean distance, Jaccard distance avoids incorrectly treating ‘commonly missing’ features as sources of similarity, thereby improving the structural accuracy of the dimensionality reduction embedding. This section does not include quantitative clustering evaluations; the analysis is based on visualisation results.

4.2.1 Comparison of Basic Dimensionality Reduction Results

In dimensionality reduction experiments on binary datasets, PCA, as a method based on linear projection, has certain limitations when handling high-dimensional sparse binary data. This is primarily because PCA assumes features are continuous variables, and its variance maximisation principle cannot fully capture the semantic similarity of binary features in embeddings, leading to scattered sample distributions across different categories and blurred cluster boundaries in the dimensionality reduction results.

t-SNE has advantages in preserving local structure, as it minimises the KL divergence between high-dimensional and low-dimensional distributions, enabling similar samples to be mapped to nearby positions. However, t-SNE has weak capabilities in preserving global structure. Results on binary datasets show that the relative positions between different clusters may be

distorted, which poses challenges in tasks requiring interpretation of macro-level relationships between clusters.

UMAP demonstrates excellent performance when combined with Jaccard distance. This method constructs a neighbourhood graph between samples and optimises its topological structure in the low-dimensional space, thereby maintaining local cluster compactness while effectively restoring the distance relationships between different clusters on a global scale. Experimental results indicate that the cluster boundaries are clear and the intra-cluster compactness is high under UMAP embedding, providing high-quality low-dimensional representations for subsequent clustering analysis.

Figure 3: Comparison of two-dimensional visualisation results of binary data under PCA, t-SNE, and UMAP.

4.2.2 Sensitivity analysis of the UMAP n_neighbors parameter

n_neighbors is one of the core hyperparameters of UMAP, used to control the number of nearest neighbours considered when constructing the neighbourhood graph for each sample. This parameter directly influences the trade-off between local and global structure: smaller n_neighbors values emphasise local neighbourhoods, while larger values prioritise the preservation of global distribution.

In this experiment, the n_neighbors parameter was set to values of 3, 4, 5, 7, 10, and 20. When n_neighbors = 3 or 4, the embedding results for binary data exhibit highly compact cluster structures, with extremely small distances between samples within clusters and very clear boundaries between clusters. This structure is advantageous for clustering algorithms (such as K-means or hierarchical clustering) because the tight cluster structure reduces the probability of misclassifying samples.

However, an overly small n_neighbors may sacrifice global structural information, leading to less accurate macro-level relationships between different clusters. When n_neighbors is increased to 10 or 20, the cluster structure in the embedding results begins to loosen, with some clusters overlapping. While this helps preserve more global topological relationships, it may impair clustering accuracy. Considering all factors, n_neighbors = 4 to 5 is the optimal range for this dataset.

Figure 4: Comparison of UMAP embedding results for binary data at different n_neighbors values.

4.2.3 UMAP min_dist Parameter Sensitivity Analysis

The min_dist parameter controls the minimum distance between points in the embedding space, directly affecting the compactness of class clusters. Smaller min_dist values result in more compact mapping of samples within the same cluster, while larger min_dist values allow for greater spacing within class clusters.

In experiments with binary data, the range of min_dist values is 0.0005, 0.001, 0.01, 0.1, 0.2. When min_dist is extremely small (e.g., 0.0005, 0.001), the class clusters in the embedding results are very tightly clustered, which helps improve clustering accuracy. However, in this case, the fine-grained structure within the clusters is difficult to visualise, potentially obscuring potential sub-class patterns in the data.

When min_dist is large (e.g., 0.1, 0.2), the clusters become more loosely distributed, with increased distances between samples within clusters, better highlighting intra-class differences. However, the boundaries between clusters are less clear compared to when min_dist is lower. Comprehensive analysis indicates that min_dist in the range of 0.01 to 0.1 achieves a balance between maintaining cluster separation and revealing detailed structures.

Figure 5: Comparison of UMAP embedding results for binary data at different min_dist values.

4.3 Conclusion

This chapter systematically analysed the performance of binary and trajectory datasets under various dimension reduction and clustering methods. Experimental results show that UMAP outperforms PCA and t-SNE on both types of data, especially when combined with Jaccard distance on binary data, where cluster separation and clustering accuracy are significantly improved. The two core parameters of UMAP, n_neighbors and min_dist, have a significant impact on the embedding results. By adjusting these parameters appropriately, a balance can be achieved between cluster compactness, preservation of global structure, and detail representation.

5. Discussion

6. Conclusion and Future Work