Engineering Applications of Artificial Intelligence 128 (2024) 107378

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Network-based exploratory data analysis and explainable three-stage deep clustering for financial customer profiling

Insu Choi [a,1], Woosung Koh [b,1], Bonwoo Koo [a,1], Woo Chang Kim [a,*]

[a] Department of Industrial and Systems Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea
[b] Department of Economics/Department of Computer Science, Yonsei University, Seoul, Republic of Korea

ARTICLE INFO

Keywords:
Network science
Autoencoders
Deep clustering
Dimension reduction
Explainable AI
Financial expert system

ABSTRACT

Effective customer segmentation and communication of these findings to non-experts is a pressing task in the financial services sector, with the potential for widespread applications. This study employs a three-stage dimension reduction and clustering technique to segment a large, high-dimensional dataset, emphasizing explainability and intuitive visualization. We present the high-dimensional data and feature set using novel network-based visualization methods and identify the multi-stage process's optimal configuration. The approach segments 14,837 potential customers, each with 163 categorical and 143 numerical features. The first stage of the dimension reduction process employs deep neural network-based autoencoders. The second and third stage uses a non-neural network-based dimension reduction algorithm and clustering algorithm contingent on clustering performance. Subsequently, game theory-inspired Shapley values are computed for each feature to enhance explainability. The optimal approach involves an autoencoder, isometric mapping to three dimensions, and K-means clustering. Lastly, we derive investment portfolios for each segment to demonstrate an expert system application in financial investment advisory to underscore the importance of explainable segmentations.

1. Introduction

Effective customer segmentation is integral to product offering customizations and marketing strategies for firms with a massive customer body (Nguyen, 2021). This is especially the case for Business-to-Customer (B2C) firms, as their customer base is typically very large (Liao et al., 2011). For example, financial firms targeting retail banking customers may have millions of customers (Son et al., 2019). Naturally, the large customer base is highly heterogeneous across many dimensions, such as preference for products and services.

Accordingly, in the past, firms would discretionarily segment their customers based on domain knowledge from day-to-day observations (Seret et al., 2015). However, this practice has faded away with data-driven customer segmentation approaches (Yanik and Elmorsy, 2019). A common approach in modern customer segmentation is clustering, an unsupervised machine-learning technique (Li et al., 2021a). A wide range of clustering algorithms is commonly used in industry and academia, such as K-means, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and the Gaussian Mixture Model (GMM) (Ezugwu et al., 2021). However, conventional clustering approaches work poorly in high-dimensional data due to the curse of dimensionality and the algorithms not being designed to operate in high-dimensional spaces (Kriegel et al., 2021). Furthermore, conventional clustering approaches have been shown to work poorly for data sets containing both numerical and categorical data (Kovács et al., 2021).

This burgeoning issue resonates profoundly within expansive databanks (Eslami et al., 2020), particularly in the era of digital finance, epitomizing a critical facet of the digital economy (DE). Firms, especially large financial entities, are transitioning from physical branches to digital platforms, exemplifying a paradigmatic shift toward an electronically dominated, data-driven financial landscape (Aitken et al., 2021b). Notably, this is evident in South Korea, the origin country of our research data. The Korean 'MyData' initiative, supporting 'Open Finance' and 'Open Banking', enables a list of major financial institutions and digital platforms to amalgamate consumer data within a singular hub, with individual consent (Park et al., 2021). This government-led program, aiming to foster FinTech innovation and competition, mirrors analogous initiatives globally, thereby spotlighting the synergistic relationship between DE and artificial intelligence (AI) in elevating and transforming economic activities and structures. The

* Corresponding author.
E-mail addresses: jl.cheivly@kaist.ac.kr (I. Choi), reiss.koh@yonsei.ac.kr (W. Koh), kbw9896@kaist.ac.kr (B. Koo), wkim@kaist.ac.kr (W. Chang Kim).
[1] First Authors.

https://doi.org/10.1016/j.engappai.2023.107378
Received 25 May 2023; Received in revised form 25 October 2023; Accepted 25 October 2023
Available online 3 December 2023

escalation in dimensionality transcends geographical and industrial boundaries, permeating various sectors such as Manufacturing, Materials, Healthcare, and Communication (Kim et al., 2015; Wu et al., 2018; Brown et al., 2021; Kai and Jin, 2022). A ubiquitous strategy adopted by researchers, which embeds a dimensionality reduction component within the clustering procedure (Wang and Sun, 2015), not only addresses the challenges presented by high-dimensional datasets but also epitomizes the seamless integration of AI in harnessing and optimizing data – a core element of DE. Therefore, this strategy both tackles the complexities associated with vast and multi-faceted data, but also illuminates the pivotal role of AI in driving the DE, especially in an era where data is deemed a fundamental economic catalyst.

Various state-of-the-art dimensionality reduction and clustering techniques work well on some datasets over others, with no clear theoretical or empirical support for using one over the other. This is why many dimensionality reduction and clustering approaches are still used in academia and industry practice. Despite the recently proposed highly competitive three-stage clustering framework, there needs empirical or theoretical evidence for using specific algorithms that make up the framework. With this backdrop, we suggest taking an approach that systematically identifies the optimal components given a data set.

This research emphasizes the explainability and visualization factor, as it is a commonly highlighted obstacle in the financial industry due to the nature of the business (Caffo et al., 2022). The financial industry is one of the most regulated industries, resulting in stringent risk management (Prasch and Warin, 2016). This environment is a backdrop to a risk-averse and conservative culture (Kihm and Kamal, 2021), resulting in obstacles for data-driven machine learning methodologies to fully integrate into all processes within the industry (Bussmann et al., 2020). Data-driven approaches, especially those with neural-network architectures, are viewed as black boxes (Guidotti et al., 2019). Furthermore, even if other more explainable models are used, effectively communicating, and persuading non-technical managers remains a pervasive obstacle (Nambisan et al., 2019). The problem is especially potent for financial firms with a long history, but much less for emerging digital-native financial firms (Temelkov, 2018). Nevertheless, data-driven processes and decision-making, commonly called digital transformation, have been an industry-wide trend (Vial, 2019) that this paper aims to aid.

We incorporate interpretability and visualization in every part of the proposed end-to-end customer data mining system:

In the exploratory data analysis (EDA) stage, we propose two novel network-based approaches that complement existing EDA methods. We incorporate Shapley value-based explanations and visualize the findings across two dimensions: (i) segment dimension and (ii) feature dimension. To demonstrate the significance of interpretable segmentation, investment portfolios are derived for each segment as an expert system application demonstration to the area of financial investment advisory.

Investment advisory is a large part of the financial services industry, from tailored wealth management and private banking for high-net-worth individuals to machine-assisted asset allocation in the form of robo-advisors and related digital banking services (Bhatia et al., 2022). Nevertheless, the customer segmentation and profiling methodology presented here can be applied to other industries and use cases, including data-driven marketing, recommendation systems, expert systems, and decision support systems.

To summarize, our contributions to the literature are:

- Two novel network-based high-dimensional dataset visualization methods
- Systematic search for optimal configuration of the three-stage clustering framework
- Including interpretability and visualization factors in the three-stage clustering process
- Demonstrate a post-clustering application that can benefit from the proposed explainable system

This paper is organized as follows: Section 2 introduces the data. Section 3 presents the data mining system and the methodological components. Section 4 presents the novel network-based EDA methodologies and applies them to the data set. Moreover, we share the results of the end-to-end system with commentary in Section 4. Section 5 discusses the results presented in Section 4. Finally, Section 6 concludes the paper and provides suggestions for future research.

2. Data description and prior research

High-dimensional cross-sectional data is sourced from the Korea Institute of Public Finance (KIPF), an official governmental organization. The KIPF conducts an annual survey named the National Survey of Tax and Benefit (NaSTaB), in which the 14th annual survey, corresponding to the year 2021, was taken as the raw data for this research. A total of 8798 individual households and 14,837 unique individuals are sampled. A total of 238 features were selected from the household data set, and a total of 286 features were chosen from the individual data set. In short, the household data set, $H \in R^{8,798\times238}$, and the individuals' data set, $H \in R^{14,837\times286}$.

2.1. Data preprocessing and feature engineering

This research integrates two related data sets from an identical source. All individuals in the first data set are part of households available in the second data set. To minimize the loss of unique data points, household data is directly mapped or feature-engineered onto individuals' observations.

Before data set integration, all features in each data set are cleaned, prepared, and, when necessary, feature engineered to best proxy individuals' underlying characteristics in a machine-interpretable and data-minable format. Examples include feature merging and one hot encoding categorical features. Vis-à-vis missing features, significant features, especially those with low levels of missing data points, are imputed with various approaches such as mean substitution, linear regression, and k-nearest neighbors, depending on the percentage of missing data and relationship with other features.

Post integration, the total data set, $T \in R^{14,837\times306}$. Of the 306 features, 162 are categorical, and 144 are numerical. A summary of the data set is provided in Table 1, while detailed documentation is provided in Table A1. Finally, the data is scaled with the minimum and maximum (MinMax) normalization method, where the transformed values are bound by [0, 1].

Table 1
Data category.

No	Category	Factor Affecting Investment	Categorical	Numerical	Total Count
1	Financial	Asset	31	39	70
2		Liability	1	1	2
3		Income	14	18	32
4		Insurance Income	15	2	17
5		Expenditure	0	16	16
6		Insurance Expenditure	1	12	13
7		Subsidy	4	5	9
8	Demographic	Residence	19	0	19
9		Age	1	3	4
10		Occupation	59	18	77
11		Education	3	9	12
12	Decision-making Traits	Risk Tolerance	0	1	1
13		Others	5	0	5
14	Lifestyle	Lifestyle	9	16	25
15		Health	1	3	4
	Total		**163**	**143**	**306**

2.2. Prior works

Extensive work has been done on (i) dimensionality reduction, (ii) clustering, and (iii) model interpretability and explainability. Vis-à-vis dimensionality reduction and clustering, most works focus on improving a quantitative performance metric. These literatures typically work on the algorithmic level and develop novel methods in isolation of upstream or downstream algorithms. This approach is reasonable as the modern machine-learning pipeline is highly modular. Notable examples include LI et al. (2021b) and Li et al. (2022), where their contrastive neural clustering approach comprehensively improves on other recent baselines. Many works also focus on domain-dependent methods. Dong et al. (2023) presents a dimensionality reduction approach specific to the vision and robotics domain. Works where dimensionality reduction and clustering are combined do exist. The approach shared by (McConville et al., 2021) tackles both dimensionality reduction and clustering simultaneously and achieves competitive results. Accordingly, we use their approach as a component in our explainable modeling and application pipeline.

Model interpretability and explainability works innately do not work in isolation, as it naturally accompanies an underlying model that requires post-inference explainability or directly improves interpretability on the algorithmic level. A significant recent work in improving the interpretability of a clustering model on the algorithmic level is (Peng et al., 2022). Peng et al.'s work allows for model decomposition and algorithmic transparency by allowing the modeler to interpret the cluster layers' input, neural network parameters, and activation function. While interpretability approaches are helpful for academics and technical practitioners, they remain unintuitive for non-experts.

Intuitive explainability is studied in the research field of post-inference explainability. Explainability in clustering remains of open interest, as demonstrated by Hwang et al. (2023)'s explainability-first clustering. Within this field, a widely used approach is Shapley-values for feature explainability. The intuitive nature of Shapley-values has extended to the field of computer vision. In Kuroki and Yamasaki (2023), their Shapley-based approach allows for intuitive feature explainability on vision tasks. With the effectiveness of Shapley-based approaches in mind, we incorporate it into our explainability pipeline.

The expansive domain of data analytics has highlighted customer profiling as an indispensable tool. It aids businesses in unraveling patterns inherent in customer behavior, preferences, and prospective buying routes. Drawing from both quantitative and qualitative data sources, customer profiling meticulously categorizes distinct customer cohorts, informed by variables like demographics, buying trends, and prior brand engagements. These insights empower organizations to design offerings and strategies that resonate profoundly with specific customer demographics, promoting a nuanced approach (Hague et al., 2013; Tabianan et al., 2022; Nagaraj et al., 2022; Narayana et al., 2022).

In this quest for customer-centric understanding, data-driven personas have carved out a significant niche. Distinct from their traditional counterparts, which may lean on overgeneralizations or conjectures, these personas draw life from real-world user data. This empirical foundation guarantees that the personas resonate with authentic user experiences, granting actionable insights to businesses. Harnessing advanced clustering and segmentation tools, sprawling and seemingly disjointed datasets are transformed into lucid, representative user outlines. Each crafted persona reflects the nuanced patterns and tendencies observed in raw data, becoming an indispensable guide for a gamut of professionals, from product developers to marketers and UX designers (Alkhayarat et al., 2020; Brahmana et al., 2020; Zhang et al., 2022; Abbasimehr and Bahrini, 2022).

The convergence of customer profiling offers a panoramic view of the customer landscape. While the former sketches an overarching view of customer clusters, the latter zooms into the unique stories each representative user encapsulates. Together, they furnish a thorough grasp of the customer spectrum. Businesses, armed with this knowledge, can evolve from broad-brush strategies to tailor-made campaigns and solutions. This strategic pivot enhances customer rapport, fortifying loyalty and driving growth. With the data deluge characterizing today's digital milieu, the amalgamation of customer profiling with data-driven persona crafting stands poised to redefine customer-oriented paradigms (Micheaux and Bosio, 2019; Camilleri, 2020; Jansen et al., 2020; Saura et al., 2021).

3. Experiment design and methodology

3.1. Experiment design for Three Stage Clustering

The three-stage clustering approach begins with an Autoencoder-driven deep dimension reduction, taking high-dimensional data to a lower, middle dimension. Our experiment sets the middle dimension to 16. Then, from the middle dimension of 16, a non-neural network dimension reduction approach is applied to end up in the second or third dimension. The final dimension is set to the second and third dimension for two reasons. First, conventional clustering algorithms work well under low dimensions. Second, the clusters become intuitively visualizable to non-experts—a key characteristic needed in the financial services industry, as mentioned in Section 1. Fig. 1 summarizes the three-stage clustering approach visually and Algorithm 1 and 2 presents the high-level pseudocode for our systematic search.

Algorithm 1. Optimal Configuration Search

Algorithm 1: Optimal Configuration Search
Input: $data_{raw}$, $config\_search\_space_{AutoEncoder}$, $config\_search\_space_{Stage_2,Stage_3}$
Output: $config$
Function `ThreeStageClusterSearch`($data_{raw}$):
  $Initialize\ config \longleftarrow Dict()$
  $data_{processed} \longleftarrow DataPreprocessor(data_{raw})$
  $Optimizer \longleftarrow RandomGridSearch(config\_search\_space_{AutoEncoder})$
  $config_{AutoEncoder} \longleftarrow Optimizer.tune(AutoEncoder(data_{processed}))$
  $config \longleftarrow config.append(config_{AutoEncoder})$
  $AutoEncoder(data_{processed}).train(config_{AutoEncoder})$
  $data_{dimension\_reduced\_0} \longleftarrow AutoEncoder.get\_latent\_vector()$
  $Optimizer \longleftarrow GlobalSearch(config\_search\_space_{Stage_2,Stage_3})$
  $config_{Stage_2,Stage_3} \longleftarrow Optimizer.tune(DimensionReduction(\cdot), Cluster(\cdot))$
  $config \longleftarrow config.append(config_{Stage_2,Stage_3})$
  return $config$
End Function

Algorithm 2. Three Stage Cluster

Algorithm 2: Three Stage Cluster
Input: $data_{raw}$, $config$
Output: $data_{clustered}$
Function `ThreeStageCluster`($data_{raw}$, $config$):
  $config_{AutoEncoder} \longleftarrow config.get\_config_{AutoEncoder}()$
  $DimensionReduction() \longleftarrow config.get\_DimensionReduction()$
  $Cluster() \longleftarrow config.get\_Cluster()$
  $data_{processed} \longleftarrow DataPreprocessor(data_{raw})$
  $AutoEncoder(config_{AutoEncoder}).train(data_{processed})$
  $data_{dimension\_reduced\_0} \longleftarrow AutoEncoder.get\_latent\_vector()$
  $data_{dimension\_reduced\_1} \longleftarrow DimensionReduction(dimension\_reduced\_0)$
  $data_{clustered} \longleftarrow Cluster(data_{dimension\_reduced\_1})$
  return $data_{clustered}$
End Function

The optimal configuration of the first stage is searched via random grid search, detailed in Section 3.2. After the configuration of the autoencoder is optimized, stages two and three are optimized together. Optimal stage two and three configurations are searched via global argmax of the aggregate scaled score of each configuration combination. Three cluster scoring metrics, Silhouette, Calinski-Harabasz score, and Davies-Bouldin score, are scaled, then equal weight aggregated. The Davies-Bouldin score is transformed to be in the ordinally same direction as the Silhouette and Calinski-Harabasz score. The scores are MinMax scaled to give relative and fair aggregation weights. The second and third-stage configuration optimization approach is summarized by

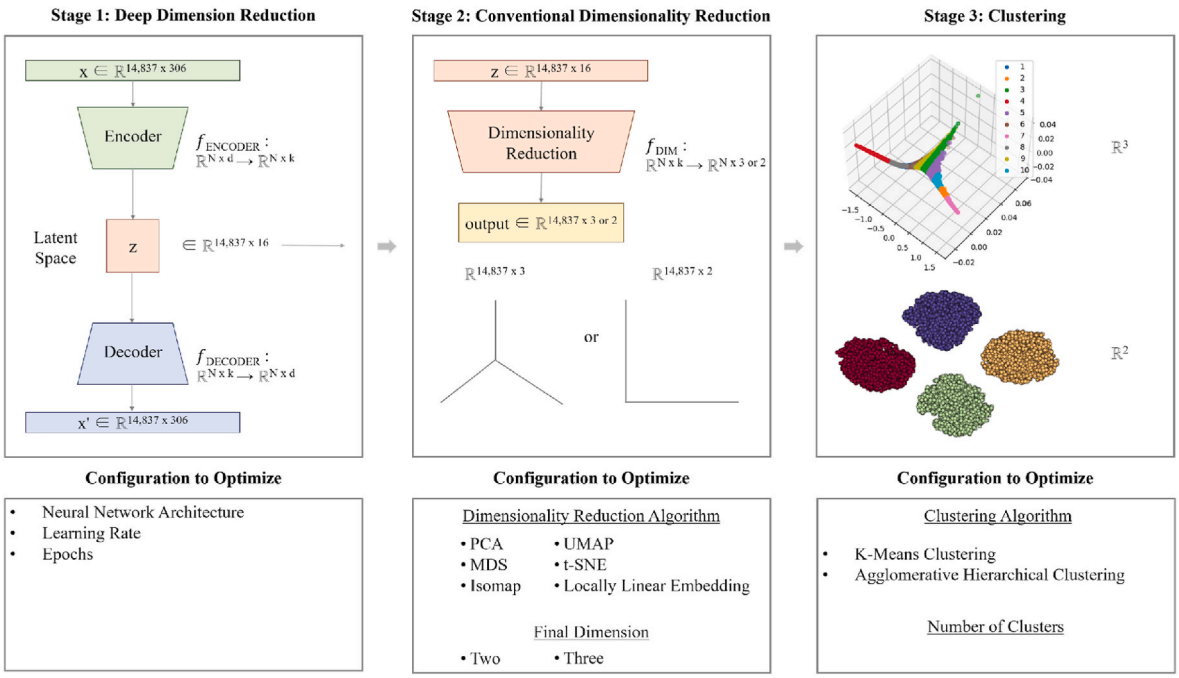

Fig. 1. Proposed three stage clustering method for NaSTaB data.

Equation (1).

$$\sum_{s \in S} MinMax(s|\ C_2, C_3) \tag{1}$$

$\boldsymbol{C}_2$ and $\boldsymbol{C}_3$ refer to the sets of configurations to optimize in Stage 2 and 3, respectively. Score set, $\boldsymbol{S}$ = {Silhouette, Calinski-Harabasz, 1 - Davies-Bouldin}

After identifying the optimal configuration, the clusters are interpreted via SHapley Additive exPlanation (SHAP) values. These values systematically identify the most influential features within the high-dimensional feature set in the clustering process. Next, based on the significant features, customer segmentations are profiled. Lastly, we demonstrate a potential financial expert system application—portfolio curation and recommendation.

3.2. Stage 1: deep dimensionality reduction

3.2.1. Autoencoders

Based on the Encoder-Decoder architecture, the Autoencoder is an artificial neural network architecture applied to machine learning problems (Ballard, 1987). First, the encoder compresses the data to a lower dimension, and afterward, the decoder reconstructs the data to the original dimension. The latent space between the encoder and decoder is taken as the dimension-reduced representation of the data (Li et al., 2020). The autoencoder trained by minimizing the reconstruction error captures complex non-linear patterns in high-dimensional data and represents them in a lower dimension. The mathematical approach is as follows.

$$\begin{aligned} f_{ENC} &: R^{N\times d} \rightarrow R^{N\times k} \\ f_{DEC} &: R^{N\times k} \rightarrow R^{N\times d} \\ l_{AE} &= \| X - f_{DEC}(f_{ENC}(X)) \|_F^2 \end{aligned} \tag{2}$$

In the given notation, N represents the number of samples we are working with. d refers to the original dimension of the data, while k is the lower dimension to which we want to reduce the data. Lastly, $\| \cdot \|_F^2$ is the Frobeniums norm. The autoencoder's hyperparameters are automatically optimized via random grid search (Liaw et al., 2018) with a total trial size of 450, where 15 parallel trials are run simultaneously for computational efficiency. Random grid search is an optimization approach where trial samples are initialized via grid search, then randomly sampled afterward. The search space and optimal configuration are provided in Table 2. The total number of combinations equals $2 \times 7 \times 10 = 280$, making the total trial size more than adequate. A five cross-validation approach is used to identify the optimal hyperparameters to prevent overfitting (Allen, 1974). The neural network architectures are provided in Table A3 and Table 3.

3.3. Stage 2: conventional dimensionality reduction

3.3.1. Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a classical dimensionality reduction technique that linearly transforms the high-dimensional data into a new coordinate system where the variation in the data can be described with fewer dimensions than the initial data. Specifically, it identifies orthogonal vectors of data points, named principal components, and aims to retain components with large variance and remove components with small variance (Ghodsi, 2006). It has the advantages of removing correlated features in a large data set and preserving valuable information through a sequence of best linear approximations to the data (Hasan and Abdulazeez, 2021).

3.3.2. Manifold learning

Although PCA is a popular dimensionality reduction technique, its limitation lies in its assumption that the data is linearly separable, whereas many datasets also exhibit non-linear structures. Manifold learning is a type of non-linear dimensionality reduction technique that seeks to describe the data as low-dimensional manifolds embedded in

Table 2
Autoencoder search space and optimal Configuration.

Linear Autoencoder		
Hyperparameter	Search Set	Optimal
Neural Network Architecture	{A, B}	B
Learning Rate	{0.0000001, 0.000001, …, 0.001, 0.01}	0.01
Epochs	{1, 2, …, 19, 20}	18

Table 3
Neural network B architecture.

Neural Network B				
Layer	Input Shape	Output Shape	# of Parameters	# of Operations
Encoder				
Linear	(14837, 306)	(14837, 256)	78,592	78,336
ReLU	(14837, 256)	(14837, 256)		
Linear	(14837, 256)	(14837, 128)	32,896	32,768
ReLU	(14837, 128)	(14837, 128)		
Linear	(14837, 128)	(14837, 64)	8256	8192
ReLU	(14837, 64)	(14837, 64)		
Linear	(14837, 64)	(14837, 32)	2080	2048
ReLU	(14837, 32)	(14837, 32)		
Linear	(14837, 32)	(14837, 16)	528	512
ReLU	(14837, 16)	(14837, 16)		
Decoder				
Linear	(14837, 16)	(14837, 32)	544	512
ReLU	(14837, 32)	(14837, 32)		
Linear	(14837, 32)	(14837, 64)	2112	2048
ReLU	(14837, 64)	(14837, 64)		
Linear	(14837, 64)	(14837, 128)	8320	8192
ReLU	(14837, 128)	(14837, 128)		
Linear	(14837, 128)	(14837, 256)	33,024	32,768
ReLU	(14837, 256)	(14837, 256)		
Linear	(14837, 256)	(14837, 306)	78,642	78,336
Sigmoid	(14837, 306)	(14837, 306)		

high-dimensional spaces. Since manifold is a topological concept in which every point is locally connected, manifold learning can capture local features effectively. Along with PCA, we used 5 other manifold learning techniques in this paper: Multidimensional Scaling, Isometric Mapping, Locally Linear Embedding, t-distributed Stochastic Neighbor Embedding, and Uniform Manifold Approximation and Projection, to determine the best dimensionality reduction technique for the three-stage clustering.

3.3.2.1. Multidimensional Scaling (MDS). Multidimensional Scaling (MDS) is a type of manifold learning that visualizes the relationships between data points based on their similarities. It consists of two main types: Metric MDS and Non-Metric MDS (Torgerson, 1952; Kruskal, 1964).

Metric MDS assumes that the input data consists of pairwise distances between data points, and seeks to find a low-dimensional representation that preserves these distances. The distance metric used in the low-dimensional space is the same as the distance metric used in the high-dimensional space. The error function (or stress) for metric MDS is computed as (Williams, 2000):

$$S = \frac{\sum_{i,j} w_{ij}\left(d_{ij} - f\left(\delta_{ij}\right)\right)^2}{\sum_{ij} {d_{ij}}^2} \tag{3}$$

where d_{ij} is the interpoint distance, $f(\delta_{ij})$ is the analytic function of the dissimilarities, and w_{ij} is the appropriately chosen weight.

However, non-metric MDS does not assume that the pairwise distances between data points are metric. Non-Metric MDS uses a rank-based measure of similarity to preserve the ordinal relationships between data points. It uses a monotonic function to transform the dissimilarity values into a new distance metric in the low-dimensional space. Unlike PCA which focuses on identifying the direction in the data sets that account for the most variation, MDS aims to preserve the similarities between data points and to capture non-linear relationships, which results in better data visualization in the low-dimensional space.

3.3.2.2. Isometric mapping (Isomap). Isometric Mapping (Tenenbaum et al., 2000), henceforth referred to as Isomap, is a non-linear dimensionality reduction algorithm that combines PCA and MDS. If the high-dimensional data lies on or near a curved manifold, MDS might consider two data points as near points, whereas their distance over the manifold is much larger than the typical interpoint distance. Isomap resolves this problem by attempting to preserve pairwise geodesic distances between data points (Van der Maaten et al., 2009). The Isomap criterion for shortest geodesic distance is as follows, where $\tau(D_Y)$ denotes the inner product matrix of Euclidean distances and $\tau(D_G)$ is the shortest path inner product matrix, $||$ denotes $L2$-norm, and $\|_F$ is the Frobenius matrix norm (Zhang et al., 2012).

$$J(Y) = \min_{Y} \| \tau(D_G) - \tau(D_Y) \|_F \tag{4}$$

Then, the low-dimensional representations of the datapoints are computed by applying MDS on the resulting pairwise geodesic distance matrix.

3.3.2.3. Locally linear embedding with Hessian Eigenmapping (HLLE). Locally Linear Embedding (LLE), proposed by Roweis and Saul (2000), is a different manifold learning approach for dimensionality reduction that seeks a lower dimensional projection of the data which preserves distances within local neighborhoods. Unlike MDS and Isomap, LLE eliminates the need to estimate pairwise distances between widely separated data points. LLE involves three steps (Roweis and Saul, 2000): first, selecting neighbors which we expect to lie on or close to a locally linear patch of the manifold; second, characterizing the local geometry of these patches by linear coefficients that reconstruct each data point from neighbors; third, mapping into embedded coordinates by minimizing the cost function which is computed as:

$$\varepsilon(W) = \sum_i \left| \vec{X}_i - \Sigma_j W_{ij} \vec{X}_j \right|^2 \tag{5}$$

where X_i is the matrix of original data points, X_j is the matrix of low-dimensional embedded coordinates, and W_{ij} is the weights that summarize the contribution of the j th data point to the i th reconstruction. Specifically, our experiments used Hessian Eigenmapping, which solves regularization problems in locally linear embedding techniques (Donoho and Grimes, 2003). Hessian Eigenmapping utilizes the following Hessian matrix and Hessian Eigenmap.

$$(Hy)(0): = \left[\frac{\partial^2 y(x)}{\partial x(i)\partial x(j)}\right]_{x=0} = 0 \min_{y \perp 1} \int \| Hy\|^2, \| y \| = 1 \tag{6}$$

where $y(x)$ is a linear transformation of the coordinates $x \in R^p$ in the tangent space at x_i. LLE has demonstrated a significant benefit in capturing the structure of manifolds generated by images of faces or documents of text by utilizing the local symmetries of linear reconstructions and learning the global structure of nonlinear manifolds (Roweis and Saul, 2000).

3.3.2.4. t-distributed Stochastic Neighbor Embedding (t-SNE). t-distributed Stochastic Neighbor Embedding, often referred to as t-SNE, is another manifold learning approach that converts affinities of data points to probabilities. This allows t-SNE to be particularly sensitive to reveal local structure. While Isomap and LLE are best suited to unfold a single continuous low-dimensional manifold, t-SNE focuses on the local structure of the data and tends to extract clustered local groups of samples. This ability makes it a powerful tool for data visualization, clustering, and anomaly detection.

t-SNE follows two steps (Vn der Maaten and Hinton, 2008). First, t-SNE constructs a probability distribution on pairs in higher dimensions such that similar objects are assigned a higher probability, and dissimilar objects are assigned a lower probability. The conditional probability $p_{j|i}$ can be interpreted as the probability that a point x_j is a neighbor of point x_i in the high-dimensional space, and is computed as:

$$p_{j|i} = \frac{exp\left(-\| x_i - x_j \|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} exp\left(-\| x_i - x_k \|^2 / 2\sigma_i^2\right)} \tag{7}$$

where σ_i is the variance of the Gaussian distribution that is centered on the data point x_i. To overcome the crowding problem in the Gaussian distribution, it uses a Student-t distribution to compute the pairwise similarities of two data points y_i, y_j in the low dimension,

$$q_{ij} = \frac{\left(1 + \| y_i - y_j \|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \| y_k - y_l \|^2\right)^{-1}} \tag{8}$$

Then, t-SNE aims to find a low-dimensional data representation that minimizes a single Kullback-Leibler divergence between a joint probability distribution, P, in the high-dimensional space and a joint probability distribution, Q, in the low-dimensional space:

$$C = KL(P \parallel Q) = \sum_i \sum_j p_{ij}\, log \frac{p_{ij}}{q_{ij}} \tag{9}$$

$$\frac{\partial C}{\partial y_i} = 4 \sum_j \left(p_{ij} - q_{ij}\right) q_{ij} \left(y_i - y_j\right)$$

computing p_{ij} as the symmetrized joint probabilities of $p_{j|i}$ and $p_{i|j}$ such that $p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N}$ and setting p_{ii} and q_{ii} to zero.

3.3.2.5. Uniform Manifold Approximation and Projection (UMAP). Uniform Manifold Approximation and Projection (UMAP) was proposed by McInnes et al. (2018), a newer alternative to t-SNE. While t-SNE has shown promising performance for preserving the underlying structure of complex datasets, it has been often criticized for being too locally focused and lacking scalability (McConville et al., 2021). UMAP addresses the limitations of t-SNE by using a graph-based approach that balances the preservation of both local and global structure in the data. Furthermore, UMAP can scale to significantly larger data set sizes and has no computational restrictions on the embedding dimension, making it more flexible for general-purpose dimensionality reduction techniques than t-SNE.

UMAP follows three steps (Hwang and Whang, 2023). First, a graph representation of $Z \in R^{N \times k}$ is presented with the probability of two data points $Z_i, Z_j \in R^k$ as

$$p_{i|j} = exp\left(-\frac{d\left(z_i, z_j\right) - \rho_i}{\sigma_i}\right) \tag{10-1}$$

and global probability p_{ij} as

$$p_{ij} = \left(p_{i|j} + p_{j|i}\right) - p_{i|j} p_{j|i} \tag{10-2}$$

where d is a distance measure, ρ_i is a local connectivity parameter, and σ_i is a normalization factor. Second, the graph is embedded by the pairwise probability q_{ij} with $Z_i^{'}, Z_j^{'} \in R^2$:

$$q_{ij} = \frac{1}{1 + a \| Z_i^{'} - Z_j^{'} \|^{2b}} \tag{11}$$

where a and b are hyperparameters and || a norm function. Finally, the optimal weights of edges in the low-dimensional representation are found by the cross-entropy cost function, computed as:

$$l_{Umap} = \sum_{i \neq j} p_{ij}\, log\left(\frac{p_{ij}}{q_{ij}}\right) + \left(1 - p_{ij}\right) log\left(\frac{\left(1 - p_{ij}\right)}{\left(1 - q_{ij}\right)}\right) \tag{12}$$

3.4. Stage 3: clustering

3.4.1. K-means clustering

K-means clustering is a popular unsupervised clustering algorithm that groups the unlabeled dataset into clusters. It is an iterative process of assigning each data point to K different groups—gradually, data points get clustered based on similar features. The objective is to minimize the sum of distances between the data points and the cluster centroid, the mean of the data points in each cluster (Hartigan and Wong, 1979). The objective function can be computed as:

$$arg\ min_s \sum_{i=1}^{k} |S_i|^* Var(S_i) \tag{13}$$

when S_i is the cluster set, $\{S_1, S_2, ..., S_k\}$, given a set of observations $\{x_1, x_2, ..., x_n\}$ and the mean μ_i of points in S_i and $Var(S_i)$ is the squared Euclidean distances between the mean and the data point x.

$$\mu_i = \frac{1}{|S_i|} \sum_{x \in S_i} x \tag{14}$$

The K-means clustering algorithm starts by randomly initializing K number of centroids, assigning each data point to its closest centroid, and then iteratively updating them until they convergence, where the centroids no longer change their position. Since the algorithm works with predefined value K, determining the optimal number K is another major factor in the performance of clustering algorithms. Various performance metrics for determining the optimal number include the Silhouette, Calinski-Harabasz, and Davies-Bouldin scores. These performance metrics will be discussed at the end of the Section 6.

K-means clustering has been a popular method in data mining and machine learning due to its computational efficiency and scalability to large datasets. It has successful applications in customer segmentation in marketing, social network analysis, and more (Kuo et al., 2002; Alsayat and El-Sayed, 2016).

3.4.2. Agglomerative clustering

Hierarchical clustering is a clustering algorithm that aims to group similar data points into clusters in a hierarchical or tree-like structure. There are two main types of hierarchical clustering algorithms: agglomerative and divisive. Agglomerative clustering is a bottom-up clustering method that starts with each point and iteratively merges the most similar clusters until a stopping criterion is met, such as when the predefined number K of clusters is reached. In this paper, we used agglomerative clustering to test hierarchical clustering against K-means. Another type of hierarchical clustering is divisive clustering, also known as top-down clustering, which starts with all data points in a single cluster and recursively splits the clusters into smaller sub-clusters until a stopping criterion is met (Murtagh et al., 2012).

There are two hyperparameters in agglomerative clustering: metric, which is the distance function to calculate the distance between individual data points, and linkage, which is the distance function to calculate the distance between two clusters of data points. In our experiment, we used two metrics: Euclidean and Manhattan distance, which are computed as $\sqrt{\sum_i (a_i - b_i)^2}$ and $\sum_i |a_i - b_i|$, respectively. We used four linkage methods: Single, Complete, Average, and Ward. 'Single' uses the minimum of the distances between all observations of the two sets. 'Complete' uses the maximum distances between all observations of the two sets. 'Average' uses the average of the distances of each observation of the two sets, and 'Ward' minimizes the variance of the merged clusters.

Hierarchical clustering has been shown to outperform K-means clustering when dealing with non-linearly separable data, such as in categorical data or data with clusters of varying sizes. Additionally, hierarchical clustering provides a simple and understandable dendrogram of clusters, where theoretical properties for dividing the clusters are easily identified. Relative to K-means clustering, hierarchical clustering may provide different perspectives of clustering algorithms in certain applications, such as customer segmentation in marketing or social network analysis (Hung et al., 2019).

3.4.3. Performance metrics

There are three metrics for measuring the performance of unsupervised clustering: Silhouette score, Calinski-Harabasz score, and Davies-Bouldin score (Baarsch et al., 2012). Silhouette score measures the similarity of data points within a cluster compared to the similarity of data points in different clusters. The silhouette score ranges from −1 to 1, where 1 indicates a perfect clustering result, 0 indicates overlapping clusters, and negative scores indicate that data points are assigned to the wrong clusters. The silhouette width $s(x_i)$ for the point x_i is defined as (Shutaywi and Kachouie, 2021):

$$s(x_i) = \frac{b(x_i) - a(x_i)}{max\{b(x_i), a(x_i)\}} \tag{15-1}$$

where x_i is an element in cluster π_k, $a(x_i)$ is the average distance of x_i to all other elements in the cluster π_k, and $b(x_i)$ defined as:

$$b(x_i) = min\{d_l(x_i)\}, \text{among all clusters} l \neq k \tag{15-2}$$

where $d_l(x_i)$ is the average distance from x_i to all points in cluster π_k for $l \neq k$.

Calinski-Harabasz score measures the inter-cluster to intra-cluster variance ratio. A higher Calinski-Harabasz score indicates that the clusters are well separated and compact. Calinski-Harabasz score is computed as:

$$\frac{trace(BCSM)}{trace(WCSM)} * \frac{N-k}{k-1} \tag{16}$$

where BCSM is a between-cluster scatter matrix, WCSM is a within-cluster scatter matrix, N is the number of points in the data set, and k is the number of clusters. The trace BCSM is the sum of the squares of the distances between the center of each cluster and the centroid of the data set, weighted by the cluster size. The trace WCSM is the sum of the squares of the distances between the center of each cluster and every point in the cluster.

Davies-Bouldin measures the similarity between clusters based on the distance between the cluster centers and the distance between the points in each cluster. A lower Davies-Bouldin score indicates that the clusters are well separated and that the data points within a cluster are similar. The maximum values of the sum of the average distances of each point in the two clusters to its respective center for each cluster are averaged, resulting in a score:

$$\frac{1}{k}\sum_{i=1}^{k} R_i \tag{17}$$

where $R_i = \max R_{ij}$, $i \neq j$, $R_{ij} = (S_i + S_j)/M_j$. S_i is the sum of the average distances from each point in cluster i to the centroid of its cluster, and M_{ij} is the distance between the two cluster centers.

In our experiment, we compare the clustering performance via the aggregated metric score as in Equation (1). We consider six dimensionality reduction technique types (PCA, MDS, Isomap, UMAP, t-SNE, and LLE), two clustering algorithm types (K-means and Agglomerative), and seven combinations of metric and linkage type in Agglomerative clustering. We range our number of clusters K from 2 to 10 clusters for interpretability and downstream tasks. In aggregate, 1944 unique experiments were conducted to identify the best three-stage deep clustering algorithm for our high-dimensional data. We select the best three-stage deep clustering algorithm based on the results of these experiments.

3.5. SHapley Additive exPlanations (SHAP)

Machine learning offers tremendous potential for classification and regression problems; however, the lack of interpretability in predictions has hindered its widespread adoption. Lundberg and Lee (2017a, 2017b) introduced the SHAP approach to address this issue, which provides interpretable clustering resuls for various machine learning algorithms. SHAP enables users to understand complex model predictions better. The foundation of this approach, the Shapley value, was first proposed by American economist Lloyd Shapley and is rooted in game theory (Shapley, 1953). It has gained prominence as an explainable artificial intelligence (XAI) technique capable of elucidating the influence of each feature on a specific prediction. By employing game theory, we can obtain the mean of the estimated Shapley values by averaging the conditional expected values for each data column.

$$\varphi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(M - |S| - 1)!}{M!} [f_x(S \cup \{i\}) - f_x(S)] \tag{18}$$

φ_i is the Shapley value for the data and N is the set of total input variable. S is the set except i-th variable in the total input variable and $v(S)$ is the contribution that the remaining subset except the i-th data contributed to the result, and $f_x(S \cup \{i\})$ is the total contribution including the i-th data. A high-level pseudocode is available in Algorithm 3 (Lundberg et al., 2020; Laberge and Pequignot, 2022).

Algorithm 3. Feature Importance Derivation using TreeSHAP

Algorithm 3: Feature Importance Derivation using TreeSHAP
Input: $data_{raw}, config$
Output: $tree\_mean\_absolute\_values$
Function `FeatureImportance` $(data_{raw}, config)$:
  $m \longleftarrow data_{raw}.num\_rows()$
  $n \longleftarrow size(features)$
  $data_{clustered} \longleftarrow ThreeStageCluster(data_{raw}, config)$
  $target \longleftarrow data_{clustered}$
  $model \longleftarrow XGBoostClassifier().train(data_{raw}, target)$
  $features \longleftarrow data_{raw}.get\_features()$
  $importance\_matrix \longleftarrow m \times n\ matrix, 0 \longleftarrow element_{i,j},\ \forall i, j \in matrix$
  for $\forall row\ r \in 1$ **to** m **do**
    for $\forall feature\ f \in 1$ **to** n **do**
      $importance\_matrix[r][f] \longleftarrow$ `TreeSHAP` $(model, r, f)$
    end
  end
  $sum \longleftarrow n\ vector, 0 \longleftarrow element_i,\ \forall i \in vector$
  for $\forall feature\ f \in 1$ **to** n **do**
    for $\forall row\ r \in 1$ **to** m **do**
      $sum[f] \longleftarrow sum[f] + |importance\_matrix[r][f]|$
    end
    $tree\_mean\_absolute\_values[f] \longleftarrow \frac{sum[f]}{m}$
  end
  return $tree\_mean\_absolute\_values$
End Function

3.6. Cluster-based portfolio curation

We use the quadratic utility model as the cluster-based portfolio curation method. Maximizing quadratic utility model is a portfolio optimization technique used in portfolio management to find the optimal asset allocation that maximizes an investor's utility. Utility, in this context, is a measure of satisfaction or happiness derived from holding a particular portfolio. The model considers both the expected returns of the assets and the associated risks, as represented by the covariance matrix.

The quadratic utility-based portfolio allocation is defined as:

$$w^T \mu - \frac{\delta}{2} w^T \Sigma w \tag{19}$$

w is the portfolio weight vector, μ is the expected return vector, and Σ is the covariance matrix of asset returns. δ is the risk aversion coefficient (must be greater than 0). The model aims to maximize this utility function, subject to certain constraints. The risk aversion coefficient (δ) is crucial in balancing the trade-off between expected returns and risk, which has a positive value. A higher value of δ indicates a higher level of risk aversion, causing the investor to prioritize minimizing risk over maximizing returns.

The max quadratic utility model also allows for an optional market-neutral constraint, ensuring that the sum of the portfolio weights equals zero. This requires a negative lower weight bound and is helpful for

investors who want to hedge against overall market movements. The quadratic utility model has its roots in the mean-variance portfolio optimization theory introduced by Markowitz (1952). The theory has since been extended and refined, but maximizing an investor's utility by balancing risk and return remains central to modern portfolio management. Tobin (1958) extended Harry Markowitz's portfolio selection theory (1952) by introducing the concept of the "Separation Theorem." The Separation Theorem states that an investor's optimal portfolio can be determined in two steps:

- Identifying the optimal risky portfolio that maximizes the risk-return trade-off can be found using the quadratic utility function.
- Deciding on the proportion of their investment to allocate to a risky portfolio and a risk-free asset based on their risk preferences.

The work of Tobin (1958) expanded upon the quadratic utility model by integrating the risk-free asset and outlining a two-step process for investors to determine their optimal portfolio. This approach is significant because it emphasizes the importance of the investor's risk preferences when making portfolio allocation decisions.

3.7. Time complexity of methodologies

Understanding the computational resources of our methods is crucial, particularly when working with large data sets. This section provides a comprehensive examination of the time complexities associated with the various algorithms employed in our research.

3.7.1. Auto-encoder (AE)

The time complexity of neural networks, particularly autoencoders, is a topic of considerable interest in deep learning literature. Predominantly, this complexity is dictated by the operations involved in forward propagation. One of the most computationally intensive operations during this propagation is matrix multiplication, especially pronounced in fully connected or linear layers. Specifically, in such layers, the computational cost is directly related to the dimensions of the matrices involved.

Let n represent the number of data points or samples. In the table provided earlier, this was quantified as 14,837. Let m represent the number of input features and k represent the number of hidden layers.

With the notation above, a typical fully connected layer, entails matrix multiplication between an input data matrix X of dimensions (n, m) and a weight matrix dimensions $(m, output\_features)$. This results in an output matrix of dimensions $(n, output\_features)$.

Given the architecture we use, it is evident that the operational complexity for each linear layer can be approximated as $(n \cdot m \cdot k \cdot output\_features)$. In big O notation, which aims to describe the upper bounds of an algorithm in terms of its most significant elements, this complexity is simplified to $O(n \cdot m)$, as the multiplication with the constant output features is overshadowed by the more dominant terms. In the provided details for Neural Network B, the dimensionality of input features, denoted by m, peaks at 306. Thus, the forward pass of our proposed model has a complexity of $O(n \cdot 306)$. More generally, when the size of the feature set varies, this complexity can be abstracted to $O(n \cdot m \cdot k)$.

3.7.2. Principal Component Analysis (PCA)

PCA operates with a time complexity of $O(min(m^3, n^3))$. The complexity stems from:

- Covariance matrix computation: $O(mn^2)$.
- Eigenvalue decomposition of the covariance matrix: $O(m^3)$.

If $m(number\ of\ features)$ exceeds $n(number\ of\ data\ points)$, eigen analysis runs at $O(n^3)$ leading to an overall complexity of $O(min(m^3, n^3))$.

3.7.3. Multidimensional Scaling (MDS)

The primary steps in MDS are distance matrix computation, $O(n^2)$, and matrix factorization, which can reach $O(n^3)$. Thus, MDS's cumulative time complexity is $O(n^3)$.

3.7.4. Isomap

Isomap's time complexity can be broken down into:
Parameters

- D: This represents the dimensionality of the original data space, which translates to the number of features or variables in the dataset.
- d: This stands for the reduced dimensionality in the embedded space. Ideally, for effective dimensionality reduction, d should be much less than D, implying that the embedded space has substantially fewer dimensions than the original.

Complexity Breakdown:

- Nearest Neighbor Search: When dealing with high-dimensional data, Isomap determines the nearest neighbors by evaluating all features or dimensions of each data point.
  - This step has a time complexity of $O(Dlog(k)nlog(n))$, particularly due to the challenges posed by high dimensionality.
- Shortest-Path Graph Search: The algorithm computes geodesic distances, which are the shortest paths, between all pairs of points in the dataset.
  - Its complexity is $O(n^2(k + log\ \ log(n)))$, stemming from the pairwise distance computations that become computationally intensive as the dataset grows.
- Lower-Dimensional Embedding: This phase embeds the data into a reduced-dimensional space, and it's commonly achieved using Multidimensional Scaling (MDS).
  - Its associated complexity is $O(dn^2)$.

When we aggregate the complexities of all these steps, the overall time complexity for Isomap becomes $O(Dlog(k)nlog(n) + n^2(k + log\ \ log(n)) + dn^2)$. This aggregated complexity gives an upper bound on the computational demands of the Isomap algorithm. It's evident that both the original dimensionality D and the number of data points n play significant roles in determining the efficiency and feasibility of applying Isomap to a given dataset.

3.7.5. Locally linear embedding with Hessian Eigenmapping (HLLE)

The standard LLE complexity is $O(Dlog(k)nlog(n)) + O(Dnk^3) + O(dn^2)$. Given HLLE's additional computations due to the Hessian-based approach, its time complexity is presumed to be in a similar or slightly higher range (Roweis and Saul, 2000; Donoho and Grimes, 2003).

3.7.6. t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-SNE's complexity is influenced heavily by the data points, n. Specifically:

- t-SNE demands $3n^2$ in memory allocation due to the creation of two $n \times n$ matrices for pairwise distances. Additionally, another $n \times n$ matrix, termed PQ, is derived during gradient computation.
- The convolution step in t-SNE, which can be expedited using the fast Fourier transform through grid interpolation, is its most time-consuming component.
- The runtime for t-SNE can fluctuate based on the perplexity parameter.

However, the actual complexity might differ based on the specific implementation and applied optimization techniques (Van Der Maaten

and Hinton, 2008).

3.7.7. Uniform Manifold Approximation and Projection (UMAP)

The UMAP (Uniform Manifold Approximation and Projection) algorithm is characterized by two primary computational phases that determine its time complexity: graph construction and optimization.

Graph Construction:

- For larger datasets, UMAP employs approximate nearest neighbors to speed up the graph construction process. This results in a time complexity of $O(nlogn)$.
- For smaller datasets, where the benefits of approximate nearest neighbors diminish or if they are not employed, the complexity tends to be quadratic, $O(n^2)$, due to exhaustive pairwise distance computations.

Optimization Phase:

- The optimization step, responsible for embedding the data into a lower-dimensional space, has a complexity similar to t-SNE per iteration. Given that t-SNE has a typical complexity of $O(n^2)$, the UMAP optimization phase follows suit. The exact time it takes can vary based on the number of iterations and specific parameters used.

Considering both phases, the overall time complexity of UMAP can be described as:

- $O(nlogn)$ for larger datasets.
- $O(n^2)$ for smaller datasets, with an added complexity per optimization iteration akin to that of t-SNE.

In summary, while the theoretical complexities offer an upper limit on computational demands, actual runtimes may deviate based on optimizations, implementations, and specific dataset attributes (McInnes et al., 2018).

4. Results

4.1. Network-based exploratory data analysis

Two unique networks are created to represent the data and feature set intuitively. The network in Fig. 2 captures all 306 features in a three-dimensional space via feature categorization. The figure captures this three-dimensional network by visually flattening the space to two dimensions. Each feature corresponds to one of the four categories in set C = {Financial, Demographic, Lifestyle, Decision-making Traits}. Large nodes represent the corresponding category, while the smaller nodes represent each feature in each category.

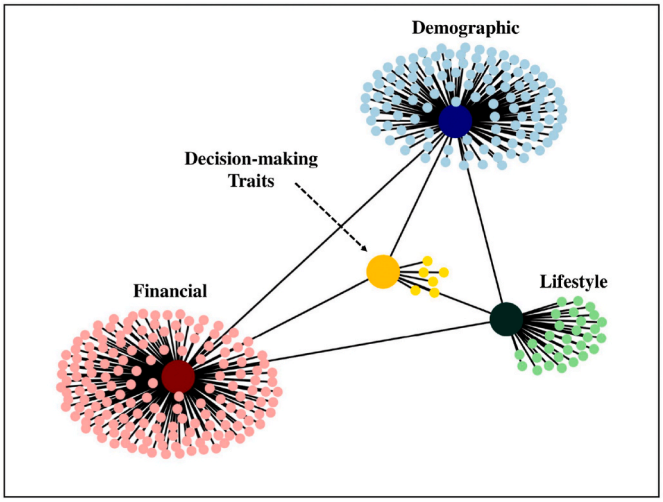

Fig. 2. All features' category network.

The edges connecting network nodes are Euclidean distances computed given a value in Pearson's correlation units α as shown in Equation 20-1. The Euclidean distance follows a similar logic for both intra-category and inter-category edges. Equation 20-2 mathematically describes the computation for each intra-category edge length. The summation function aggregates the Pearson correlation ρ across all other intra-category features, $i^{'}$, given feature i. The mean of the aggregation is computed by dividing by the size of set c less one. Equation 20-3 mathematically describes the computation for each inter-category edge length. The summation function aggregates the Pearson correlation ρ across all other categories $c^{'}$, and its elements, $j^{'}$, given a category c and all its elements, j,. The mean of the aggregation is computed by dividing by the size of set c multiplied by the summation of the size of all other sets, $c^{'}$.

$$euclidean\ distance = \sqrt{2}\sqrt{1-\alpha} \tag{20-1}$$

$$\alpha_{i\in c} = \frac{1}{|c|-1} \sum_{i\in c\setminus\{i\} \epsilon C, j\neq i} |\rho(i,i^{'})| \tag{20-2}$$

$$\alpha_{c\in C, c^{'}\in C\setminus\{c\}} = \frac{1}{|c|\cdot\sum_{c^{'}\in C}|c^{'}|} \sum_{j\in c}\sum_{j\in c^{'}} |\rho(j,j^{'})| \tag{20-3}$$

Based on the intra-category and inter-category Euclidean distances, no statistically meaningful linear relationships can be identified. Nevertheless, the most closely related categories are 'Decision-making Traits' and 'Lifestyle.' The least related pair is 'Financial' and 'Demographic.' For visualization, the inter-category Euclidean distance is scaled by the power of three. Therefore, the Euclidean distance visualized in Fig. 2 should only be compared within the intra-category and inter-category. We made this tradeoff to ensure the intuitive visualization of all four categories and 306 features. A high-level pseudocode is available in Algorithm 4.

Algorithm 4. Category Network

Algorithm 4: Category Network
Input: $data_{raw}$, $category\_feature\_map$, $scale$
Output: G
Function `CategoryNetwork` ($data_{raw}$, $category\_feature\_map$):
  $G \longleftarrow Graph()$

  for $category$, $feature_{category} \in category\_feature\_map$ **do**
    for $feature_i \in feature_{category}$ **do**
      $\rho_{sum} \longleftarrow 0$
      $counter \longleftarrow 0$
      for $feature_j \in feature_{category}$ **do**
        if $feature_i \neq feature_j$ **then**
          ρ_{sum} += $|\rho(feature_i, feature_j; data_{raw})|$
          $counter$ ++
      end
      $\rho_{mean} \longleftarrow \rho_{sum}/counter$
      $G.add\_edge(category, feature_i, length \leftarrow EuclideanDistance(\rho_{mean}))$
    end
  end

  $nodes_{central} = category\_feature\_map.get\_categories()$
  $combinations_{categories} = hashset(combinations(nodes_{central}, groups\_of \leftarrow 2))$

  for $category_I$, $features_I \in category\_feature\_map$ **do**
    for $category_J$, $features_J \in category\_feature\_map$ **do**
      if $(category_I, category_J) \in combinations_{categories}$ **then**
        $\rho_{sum} \longleftarrow 0$
        $counter \longleftarrow 0$
        for $feature_i \in features_I$ **do**
          for $feature_j \in features_J$ **do**
            ρ_{sum} += $|\rho(feature_i, feature_j; data_{raw})|$
            $counter$ ++
          end
        end
        $\rho_{mean} \longleftarrow \rho_{sum}/counter$
        $G.add\_edge(category_I, category_J, length \leftarrow$
        $EuclideanDistance(\rho_{mean})^{scale})$
    end
  end

  return G
End Function

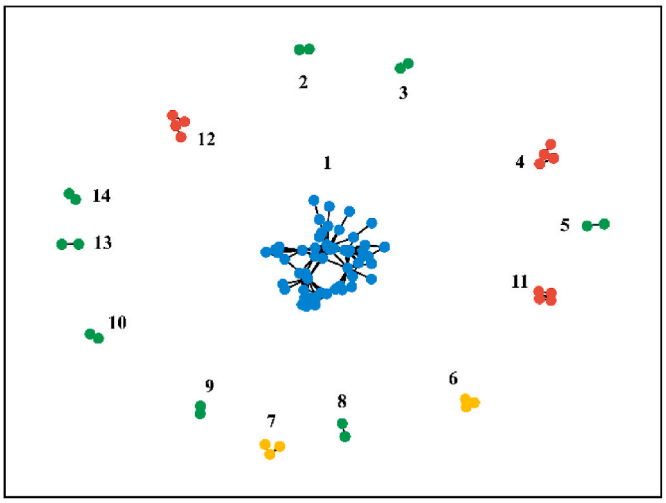

Fig. 3. Numerical features' relationship cluster network ($\beta = 0.5$).

Figs. 3 and 4 are illustrations of our relationship cluster network and the commonly used relationship heatmap, for features of numerical data type. The proposed network in Fig. 3 is a supplementary visualization method to the commonly used relationship heatmap and matrix. Given a high-dimensional data set, like the 306 featured data set used in this experiment, it is challenging to fit a relationship heatmap into a paper or screen. Even if it was legible, it would be unintuitive for the viewer.

In response, we developed a novel visualization approach to supplement the relationship heatmap. Given a threshold, β, for a relationship metric, γ, the relationships are transformed into a distance measure, then mapped onto a network in three-dimensional space. Finally, it is flattened to two dimensions for visualization. Naturally, $\beta \in dom(\gamma)$. We use Pearson's correlation coefficient $\gamma := \rho \in [-1, 1]$ as the relationship metric. The relationship to distance measure mapping is discretionary depending on the chosen relationship metric and distance measure. We use the approach displayed in Equation 20. The absolute value of the relationship is used to capture the magnitude of the relationship rather than both direction and magnitude. The computational approach for our demonstration is summarized as follows.

A non-directed graph is initialized, as $G(E,N)$, where E is the edge set, and N is the node set comprised of all features. **E** is formed based on Equation A conditional on $|\rho(i, \acute{i})| > |\beta|$, for representing strong relationships, and $|\rho(i,\acute{i})| < |\beta|$, for representing weak relationships. Here, $i \neq \acute{i} \in N$. The network visualization in Fig. 3 represents strong relationships, equivalently smaller distances. A high-level pseudocode is available in Algorithm 5.

Algorithm 5. Relationship Network

Algorithm 5: Relationship Network
Input: $data_{raw}$, $relationship\_matrix$
Output: G
Function `RelationshipNetwork` ($data_{raw}$, $relationship\_matrix$, $threshold$) **:**
  $G \longleftarrow Graph()$

  for $feature_i, feature_j \in relationship\_matrix$ **do**
    if $|\rho(feature_i, feature_j; data_{raw})| > threshold$ **then**
      $G.add\_edge(feature_i, feature_j, length \leftarrow EuclideanDistance(|\rho(feature_i, feature_j; data_{raw})|))$
  end

  return G
End Function

Each cluster in the network is color-coded based on cluster size—that is, the number of nodes. Table A2 details the relationship clusters, with a sample available in Table 4. The table's background color of the cluster column corresponds to the node colors, equivalently, cluster size, in Fig. 3. Also, shades of gray are used in the table's background to intuitively distinguish between features of different categories within a cluster. A total of 14 clusters are identified given $\beta = 0.5$. Within these 14 clusters, four different cluster sizes are observed, 2, 3, 4, and 135, with corresponding colors green, yellow, red, and blue, respectively. In line with expectations, all clusters of sizes 2, 3, and 4 are homogeneous vis-à-vis associated categories. On the other hand, the large cluster of size 135 is heterogeneous, with features from 'Financial,' 'Demographic,' and 'Lifestyle.'

Insights can be mined based on medium-sized clusters. Medium-

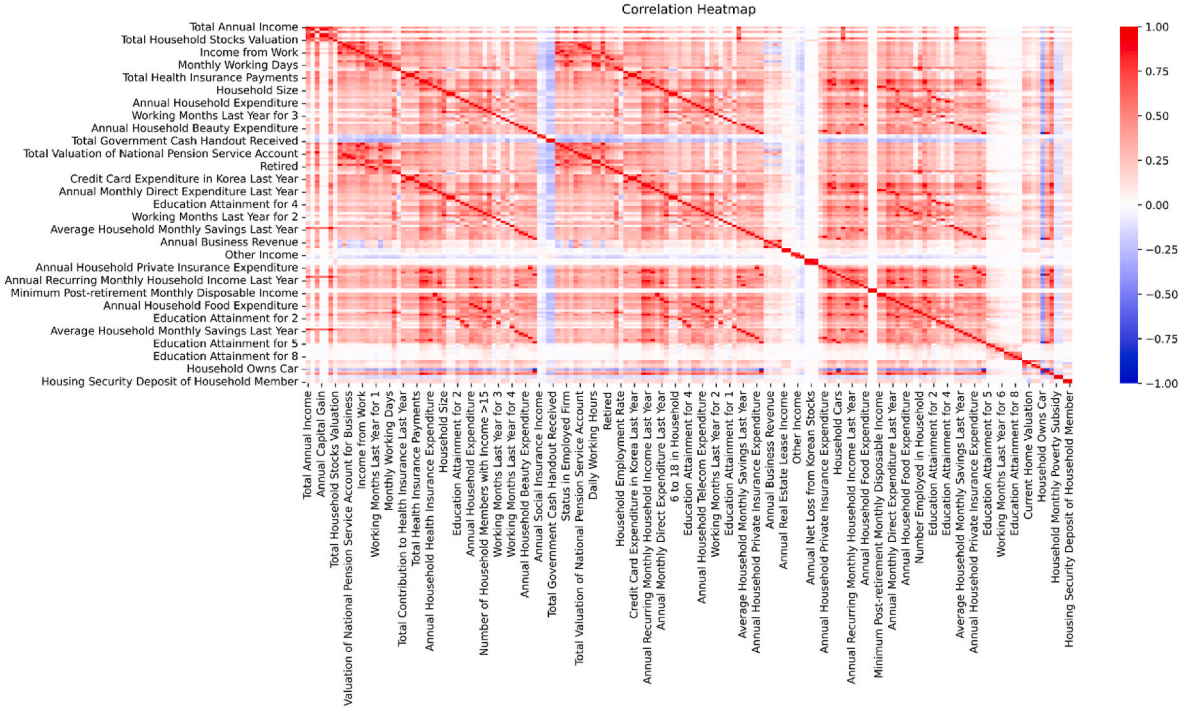

Fig. 4. Numerical features' relationship heatmap.

Table 4
Sample relationship cluster Table.

Feature	Category	Cluster	Cluster Size
Employed Firm's Headcount	Demographic	Cluster 1	135
Income from Work	Financial	Cluster 1	135
Status in Employed Firm	Demographic	Cluster 1	135
Retired	Lifestyle	Cluster 1	135
Working Months Last Year for 1	Demographic	Cluster 1	135
Regularity of Working Hours	Demographic	Cluster 1	135
...			
Annual Social Insurance Income	Financial	Cluster 2	2
Total Social Insurance Subsidy Received	Financial	Cluster 2	2
Annual Government Cash Subsidy Received	Financial	Cluster 3	2
Total Government Cash Handout Received	Financial	Cluster 3	2
Annual Net Income from Business	Financial	Cluster 4	4
Annual Business Revenue	Financial	Cluster 4	4
Valuation of Local District National Pension Service Account	Financial	Cluster 4	4
Annual Business Net Income	Financial	Cluster 4	4
Annual Real Estate Lease Income	Financial	Cluster 5	2
Total Real Estate Lease Income	Financial	Cluster 5	2
Annual Other Income	Financial	Cluster 6	3
Other Income	Financial	Cluster 6	3
Annual Other, Non-household Living, Schooling Aid Received	Financial	Cluster 6	3

sized clusters refer to clusters greater than two, as these are pairs, but not the outlier-like large clusters. In our data set, the largest cluster engulfs a majority of features simply due to its size effect. Clusters of size greater than two capture higher-dimensional relationship structures improving upon simple relationship pairs available in two-dimensional relationship matrices. The relationship cluster network, relationship cluster table, and commonly implemented relationship heatmap or matrix can be used hand-in-hand to intuitively examine high-dimensional data sets, prior to the modeling process. Examples of unique insights are as follows.

Cluster 12 of size four, available in Table A2, sheds insight into the positive relationship structure between real estate wealth and household debt. This strong positive relationship suggests that individuals with outsized real estate asset valuations are typically funded by correspondingly large amounts of debt. Another cluster, Cluster 4 of size four sheds insights on the positive relationship between business earnings and local district pension service account valuation. One hypothesis could be regulatory-induced behavior, perhaps a tax-related benefit to business owners promoting contributions to their local district pension service accounts. Future data analysts can visualize and examine higher-dimensional relationship structures through this approach in the EDA stage.

We also examined the scalability of the two proposed network-based EDA approaches. The computational complexity input to the Category Network presented in Algorithm 4 and Fig. 2 is Categories, C, Features, F, and Samples N. The worst-case time complexity of Algorithm 4 is $O(n(C^2 + F^2)$.

Here, since *features* $\in$ *categories*, typically, C$\ll$F. Strictly, C $\leq$ F, thus the complexity can be represented as $O(nF^2)$. As we are using a machine learning approach, which naturally entails a data-driven approach, typically, $n \gg F$, therefore, the scalability of this algorithm is reasonable observing $\Delta n > \Delta F$ as we scale the dataset. On top of this, the $O(n)$ component that represents computing the relationship coefficient can be distributed over computing threads. In our dataset and experiment, we do not use parallel computing, yet still achieve 51.52 s in computing the relationship metrics for each combination and 0.14 s in creating the Category Network. The relationship metric was computed via the Python package Pandas, which is known for notoriously slow compute. Despite this, with a single household-grade CPU, it can compute the edges and create the graph within 52 s.

The scalability of the Relationship Network presented in Fig. 3 and Algorithm 5 is similar to the Category Network. Here, since C is irrelevant the worst-case time complexity is $O(nF^2)$.For the same reason mentioned above, with very large datasets, parallel computing can help, but in most cases can be completed in a reasonable amount of time. Also, assuming the same relationship metric is used, the previously computed relationship matrix can be applied. Meaning, the compute time for the Relationship Matrix, which takes up most of the computational time, may be skipped. In our case, the additional compute time for creating the Relationship Network, when the relationship matrix is given is 0.21 s. Again, we use a single household-grade CPU. When applying our method to larger datasets, we expect significant improvement in computational efficiency when a more efficient backend data processing infrastructure is used, i.e., one that is not the Pandas package. Additionally, with the possibility of computing relationship matrices via multiple CPUs, the scalability of this method is highly reasonable.

Modified versions of the presented relationship network can be applied to specific EDA tasks. For instance, instead of constructing the network only based on relationship magnitude, it can also be constructed with both direction and magnitude, or direction alone. Additional high-dimensional relationship networks can be part of the EDA: (i) strong negative relationship, (ii) strong positive relationship, (iii) negative relationship, and (iv) positive relationship. This should be discretionarily implemented given the objective of the EDA and the underlying data set.

4.2. Optimal Three Stage Clustering

In total, 1944 clustering experiments have been conducted for Stage 1 (Autoencoder), 2 (Dimensionality Reduction), and 3 (Clustering) configuration combinations. The top 5 configurations of the aggregated scaled score are presented in Table 5 and top 100 are presented in Table A5 in Appendix. These experiments consisted of 13 different Stage 1 and 2 configurations; 2 different final dimensions before Stage 3; 9 different final clusters ranging from 2 to 10; and 7 different distance metrics in Agglomerative clustering. Additionally, we conducted 9 baseline experiments using K-means clustering and 63 different baseline experiments using Agglomerative clustering with various distance metrics, totaling 243 K-means experiments and 1701 Agglomerative experiments.

$$9 + (9 \cdot 13 \cdot 2) + (9 \cdot 7) + (9 \cdot 13 \cdot 2 \cdot 7) = 1944$$

The optimal three-stage clustering configuration involves first using a Linear Autoencoder to extract the data's latent vector into a 16-dimensional space. This is followed by an intermediate stage of dimensionality reduction through Isometric Mapping compressing the data further into a 3-dimensional space. Lastly, K-means clustering is used to define the final ten clusters. The final ten clusters, in 3 dimensions, are visualized in Fig. 5.

To compare the relative performance of three-stage clustering to that of one-stage clustering, which serves as the baseline algorithm utilizing only K-means or Agglomerative clustering, and two-stage clustering, which includes either Autoencoder or Dimensionality reduction technique before Stage 3, we computed the mean of the aggregated score of all experiments under the same stage configuration. As depicted in Table 6, three-stage configurations (Autoencoder + PCA into 3rd dimension + K-means or Autoencoder + Isomap into 3rd dimension + K-means) achieved the highest mean aggregated score of 0.9342. Furthermore, when comparing the two Stage 3 configurations, K-means

Table 5
Top 5 scaled scores.

Methodology						Scaled Score			
Stage 1	Middle Dimension	Stage 2	Final Dimension	Stage 3	Clusters	Silhouette	Calinski-Harabasz	Davies-Bouldin	Aggregated Score
Autoencoder	16	**Isomap**	3	**K-means**	10	0.3971	1.0000	0.4191	1.0000
Autoencoder	16	Isomap	2	K-means	10	0.3983	0.9974	0.4188	0.9994
Autoencoder	16	PCA	2	K-means	10	0.3982	0.9963	0.4188	0.9992
Autoencoder	16	PCA	3	K-means	10	0.3950	0.9970	0.4173	0.9992
Autoencoder	16	MDS	3	K-means	10	0.3781	0.9765	0.3950	0.9929

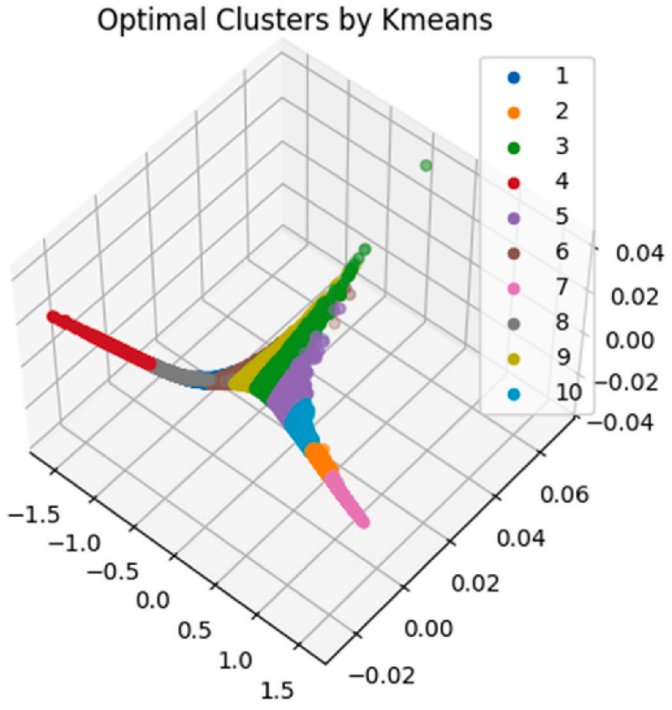

Fig. 5. Final clusters.

clustering demonstrated superior clustering performance compared to Agglomerative Clustering. The mean of K-means clustering in both 2nd and 3rd final dimension experiments exceed 0.8 in Table 6, while the mean of Agglomerative clustering was under 0.8.

Across 13 different Stage 1 and 2 configurations and the baseline of clustering the original data with 305th dimension in the absence of Stage 1 and 2, all types of Stage 1 and 2 configurations yielded higher scores than 0.6048 which is the mean score of the baseline. However, not all three-stage clustering combinations resulted in superior performance than two-stage clustering. Some configurations such as Isomap only and Locally Linear Embedding only in the absence of Autoencoder, overperformed compared to some three-stage clustering configurations.

Nevertheless, the three-stage clustering algorithms were able to discern differences between clusters more effectively by maintaining the latent features and gradually reducing the data dimensionality through autoencoder and dimensionality reduction technique.

The next step is identifying the most impactful features of each cluster in this multi-stage clustering process, aiming to uncover the heterogeneity across the clusters. The average values of absolute SHAP values of features are aggregated across each cluster and presented in Fig. 6. The feature's mean impact on each cluster is color-coded and labeled on the figure. The details of the features in Fig. 6 are presented in Table 7. Both numerical and categorical features are shown to impact the clustering process. While 53% of the features are categorical and 47% are numerical, a disproportionately larger number of numerical features are in the top ten. This is reasonable as more complex

Table 6
Mean aggregated scores.

Stage 1 and 2 configurations	K-means Clustering		Agglomerative Clustering		Mean (± 1STD)
Baseline: No Stage 1 and 2	0.5782		0.6314		0.6048 ± 0.0376
	Final Input Dimension for Clustering				
	2	3	2	3	Mean (± 1STD)
Autoencoder only	0.8313	0.7933	0.7609	0.7282	0.7784 ± 0.0441
PCA only	0.7773	0.7569	0.7796	0.7781	0.7730 ± 0.0108
MDS only	0.7323	0.6959	0.8024	0.7837	0.7536 ± 0.0485
Isomap only	0.8672	0.7929	0.8269	0.7760	0.8158 ± 0.0403
HLLE only	0.8710	0.8539	0.8810	0.8656	0.8679 ± 0.0113
t-SNE only	0.7500	0.7066	0.7178	0.6171	0.6979 ± 0.0569
UMAP only	0.7711	0.7339	0.7636	0.7182	0.7467 ± 0.0249
Autoencoder + PCA	0.9333	**0.9342**	0.8424	0.8436	0.8884 ± 0.0524
Autoencoder + MDS	0.9306	0.9312	0.8352	0.8197	0.8792 ± 0.0601
Autoencoder + Isomap	0.9334	**0.9342**	0.7907	0.7824	0.8602 ± 0.0851
Autoencoder + HLLE	0.8389	0.8386	0.8145	0.7897	0.8204 ± 0.0235
Autoencoder + t-SNE	0.7548	0.7087	0.7281	0.6689	0.7151 ± 0.0362
Autoencoder + UMAP	0.7481	0.7528	0.7168	0.7178	0.7338 ± 0.0192
Mean (± 1STD)	0.8261 ± 0.0758	0.8025 ± 0.0885	0.7892 ± 0.0510	0.7607 ± 0.0692	

dependency structures are present in numerical features, which the clustering process recognizes. All feature categories are in the top ten, excluding 'Lifestyle.' While 37% of the features are 'Demographic' and 2% are 'Decision-making Traits', a disproportionately larger amount of 'Demographic' and 'Decision-making Traits' features are present in the top ten. In contrast, other categories show relatively lower relevance in the clustering process. Lastly, various factors affecting investment decisions are seen in the top ten, including 'Occupation,' 'Income,' 'Asset,' 'Expenditure,' and 'Others (Decision-making Traits).'

The top ten SHAP values for each cluster are available in Figure A1.

4.3. Experiment results for analysis of customer profiling and portfolio curation

4.3.1. Customer profiling

Fig. 7 illustrates the heterogeneity of clusters across features by keeping the feature constant on each bar graph. The aggregated top ten significant features, presented in Fig. 6, are selected for depiction here. The top ten most influential features of each cluster, that did not make it in the aggregated list, but are included in the top ten for each cluster are presented in Figure A2.

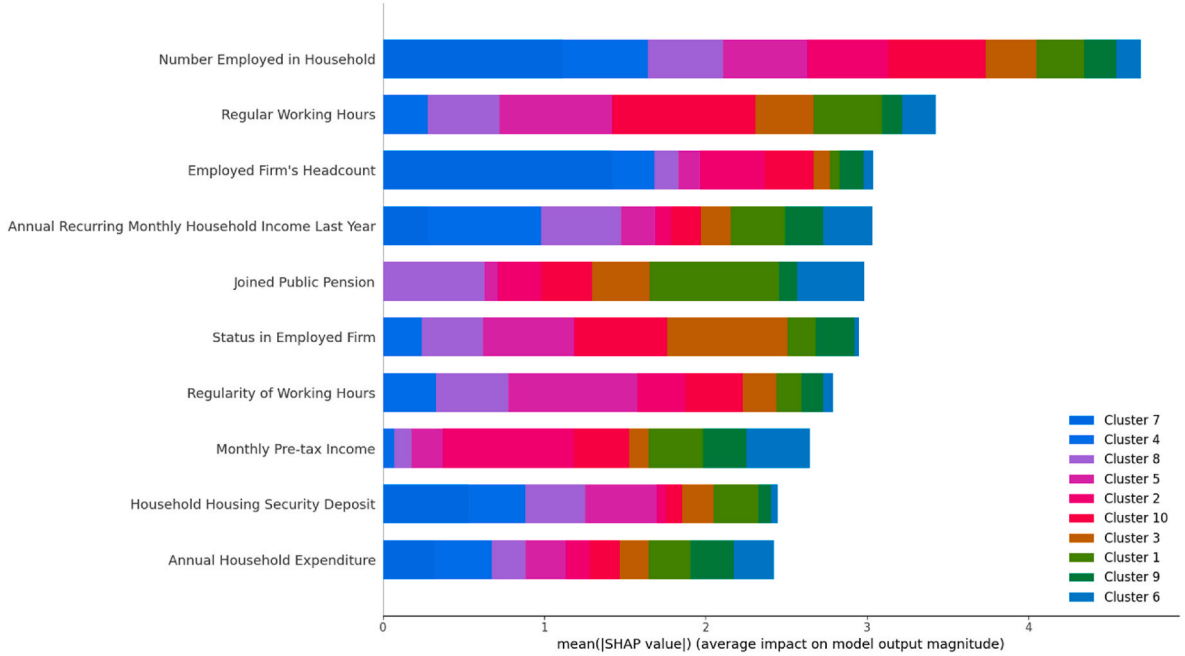

Fig. 6. Significant features ranking by SHAP values.

Table 7
Significant features and their descriptions.

Feature	Data Type	Category	Factor Affecting Investment
Number Employed in Household	Numerical	Demographic	Occupation
Regular Working Hours	Categorical	Demographic	Occupation
Employed Firm's Headcount	Numerical	Demographic	Occupation
Annual Recurring Monthly Household Income Last Year	Numerical	Financial	Income
Joined Public Pension	Categorical	Decision-making Traits	Others
Status in Employed Firm	Numerical	Demographic	Occupation
Regularity of Working Hours	Numerical	Demographic	Occupation
Monthly Pre-tax Income	Numerical	Financial	Income
Household Housing Security Deposit	Numerical	Financial	Asset
Annual Household Expenditure	Numerical	Financial	Expenditure

The units of each feature are heterogeneous and challenging to include in the axis label. Therefore, the text will mention the units of each feature. Descriptions of all other features' units are available online in the public governmental data set mentioned in Section 2. The first feature, 'Number Employed in Household,' in units of headcount, varies significantly, with the lowest cluster nearing zero and the greatest cluster around 2.5. The second feature, 'Regular Working Hours,' is a dummy variable, where zero represents irregular working hours, and one represents regular working hours. Similarly, the results are highly heterogeneous, where some clusters exhibit near zero regularity while others exhibit near-perfect regularity.

The unit of the third feature, 'Employed Firm's Headcount,' is in bucket fashion, consistent with the data source. Numerical values correspond to the following headcount buckets: {1: [1, 4], 2: [5, 9], 3: [10, 29], 4: [30, 99], 5: [100, 299], 6: [300, 499], 7: ⩾500}. The average employed firm's headcount for Cluster 4 is greatest at 4.277, translating to firms with 30–299 employees. The unit of the fourth feature, 'Annual Recurring Monthly Household Income Last Year,' is 10,000 won. On average, the lowest earning cluster, Cluster 7, earns approximately 750 monthly recurring U.S. Dollars (USD). On average, the highest earning cluster, Cluster 4, earns approximately 8405 monthly recurring USD.

The fifth feature, 'Joined Public Pension,' is a dummy variable. The unit of the sixth feature, 'Status in Employed Firm,' is categorical but ordinal. The numerical values correspond to {0: None, 1: Unpaid Household Worker, 2: Daily Worker, 3: Temporary Worker, 4: Self-employed with No Employees, 5: Full-time Worker, 6: Self-employed with Employees}. The seventh feature, 'Regularity of Working Hours,' is in numerical values $\in$ [0, 2], where the larger the value, the greater the regularity of the individual's working hours. The remaining three features are in units of 10,000 won and are self-explanatory.

Next, potential customer segment groups, as represented by clusters, are profiled via the top ten features in aggregate, and corresponding to the cluster. The profiles are visually summarized via radial charts in Fig. 8. Radial charts on the left with the aggregate top ten feature set are easily comparable across clusters. On the contrary, radial charts on the right have a heterogeneous feature set as the top ten features corresponding to its own cluster vary across clusters. The units of the radial charts are ranks, where the mean value of the corresponding feature is ranked from 1st to 10th in descending order, against other clusters. Additionally, the features are presented in clockwise descending order of SHAP values, with the most impactful feature presented right below the title of the diagram.

Based on the radial charts and supplementary bar graphs, clusters can be discretionarily labeled and profiled via intuitive, natural language. We present sample profile names in Table 8.

Through the presented approach, firms can effectively target the automatically segmented customer population. A wide array of potential applications, including machine-learning-assisted expert systems and decision support systems, can benefit from the presented methodology. This research presents one potential widespread application of the segmented customer base—an investment advisory expert system.

4.3.2. *Portfolio curation*

Based on the methodology used in prior literature (Mankiw and Zeldes, 1991; Grable and Lytton, 1999; Barber and Odean, 2001; Guiso et al., 2008; Gennaioli et al., 2015), the risk aversion coefficient corresponding to each cluster is used to derive sample portfolio

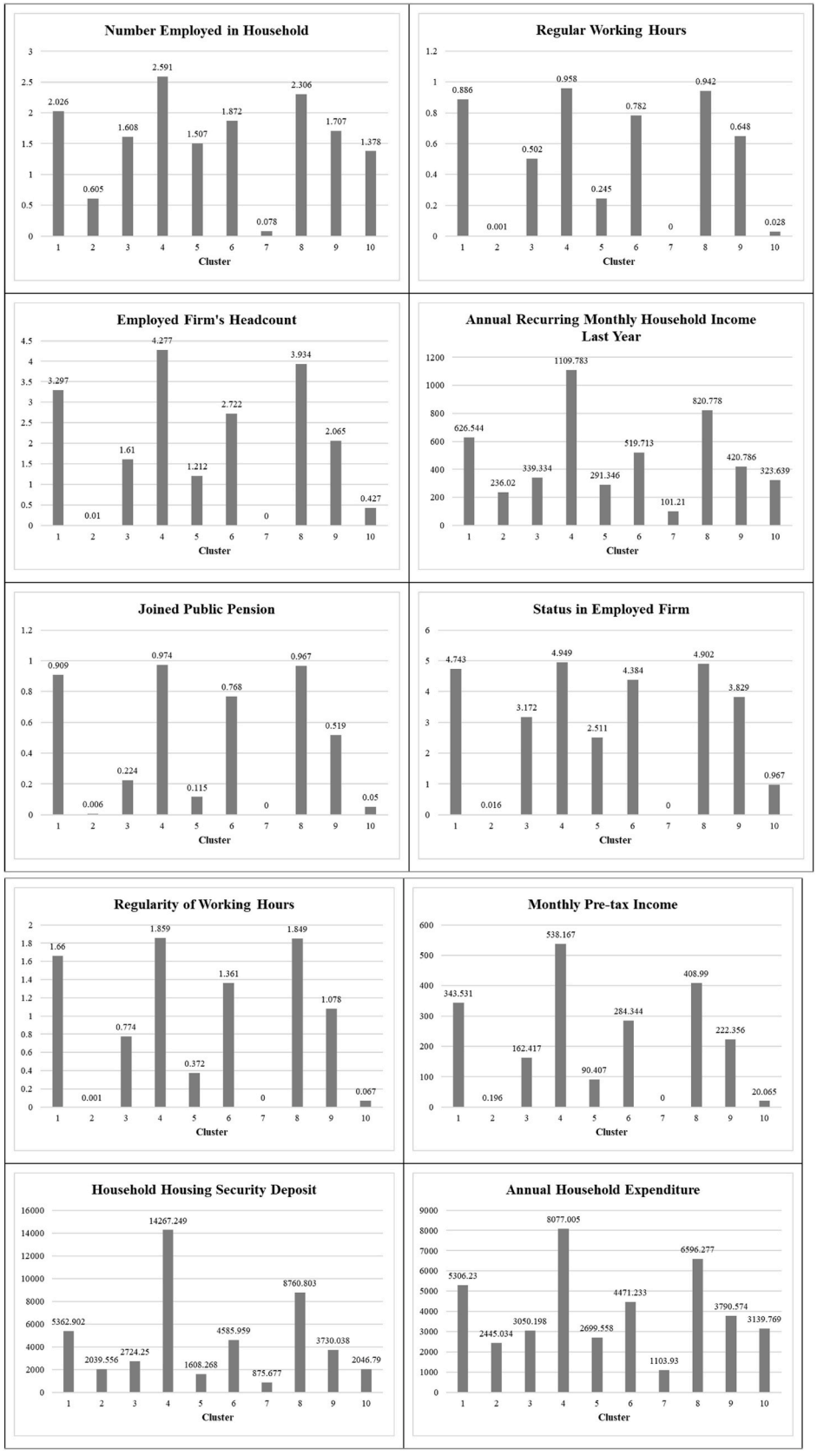

Fig. 7. Heterogeneity of clusters across top 10 important features based on SHAP values.

recommendations. The risk tolerance of customer segments can be divided into a spectrum from most risk-averse to most risk-tolerant. Based on these relative values, sample portfolio recommendations are automatically generated. The longitudinal portfolio recommendation, spanning t to t+11 years, is presented in Fig. 9. The figures illustrate the output of portfolio optimization with annual rebalancing. The investment universe is a collection of the 50 largest exchange-traded funds (ETFs), including equities, bonds, commodities, and currencies. The details of the investment universe are presented in Table A4.

4.3.2.1. ***Most risk-averse group.*** This group, composed of the most risk-averse individuals, is recommended to allocate their investments to a limited number of low-risk ETFs focused on stability and capital preservation. Despite the number of ETFs with non-zero weights varying each year, these investors should diversify across a smaller set of ETFs, ensuring minimal risk exposure. The three clusters associated with this risk profile are Cluster 2, 7, and 10, labeled as 'Non-affiliated Lower Class,' 'Highly Risk-averse Poorest,' and 'High-expenditure Lower Class.' Based on the profiling in the previous section, Section 4.2, it can be noticed that all lower-class clusters are associated with being most risk-averse, regardless of Cluster 2 exhibiting high levels of non-affiliation and Cluster 10 exhibiting relatively high levels of expenditure. The sample portfolio recommendations appropriately reflect this characteristic, supporting their socioeconomic lifestyle.

4.3.2.2. ***Moderately risk-averse group.*** Individuals in this group can withstand modestly higher risk than the previous group. While still prioritizing low-risk ETFs, a broader range of ETFs can be selected as higher volatility ETFs can be chosen. While maintaining a relatively low-risk profile, investors in this group will be able to achieve improved returns. A balanced approach to risk management and return potential is recommended for this group. The two clusters associated with this risk profile are Clusters 3 and 5, labeled 'Middle Class with Government Aid,' and 'Frugal Middle Class' respectively. Based on the customer profiling section, the clusters in this group exhibit higher levels of socioeconomic distress relative to other middle-class clusters, displaying higher levels of government aid and frugality. Accordingly, sample portfolio recommendations appropriately reflect this characteristic, supporting their socioeconomic lifestyle.

4.3.2.3. ***Minimally risk-averse group.*** Individuals in this group display a low level of risk aversion. They are more willing to diversify their investments across a more extensive range of ETFs, including those with higher risk-return potential. The cluster associated with this risk profile is Cluster 9, labeled 'Typical Middle Class.' Unlike Cluster 3 and 5, which represent relatively distressed middle-class individuals, the typical middle-class individual can tolerate higher levels of risk. Their stable socioeconomic status allows for a more diverse range of financial products to be recommended.

4.3.2.4. ***Moderately risk-tolerant group.*** Individuals in this group will be interested in high-risk, high-return ETFs. They can allocate most of their capital into ETFs with higher risk-return potential. In turn, the number of ETFs with non-zero weights is larger than in previous groups. The two clusters associated with this risk profile are Clusters 1 and 6, labeled 'Typical Upper-middle Class' and 'Self-employed Hard-Working Middle Class.' Consistent with greater socioeconomic stability, upper-middle-class individuals tolerate investing in more aggressive financial products. Interestingly, middle-class self-employed individuals with high working days exhibit higher risk tolerance, comparable to upper-middle-class individuals. Most South Korean self-employed people with high working hours are small business owners (Kim et al., 2015). Their experience with starting and running a business may be associated with higher risk tolerance levels. The sample recommended portfolios reflect each cluster's socioeconomic characteristics.

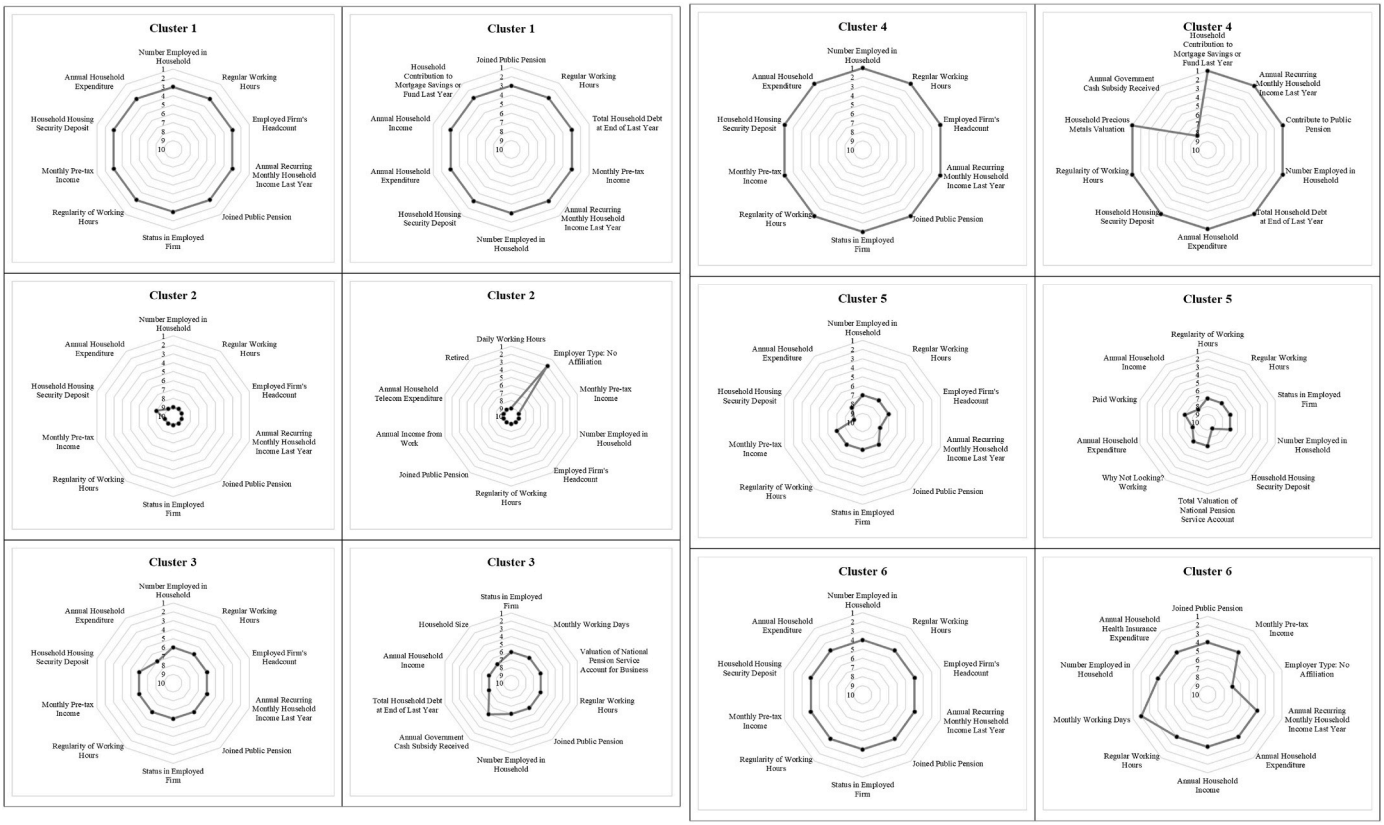

Fig. 8. Aggregate top ten features and corresponding top ten features by clusters.

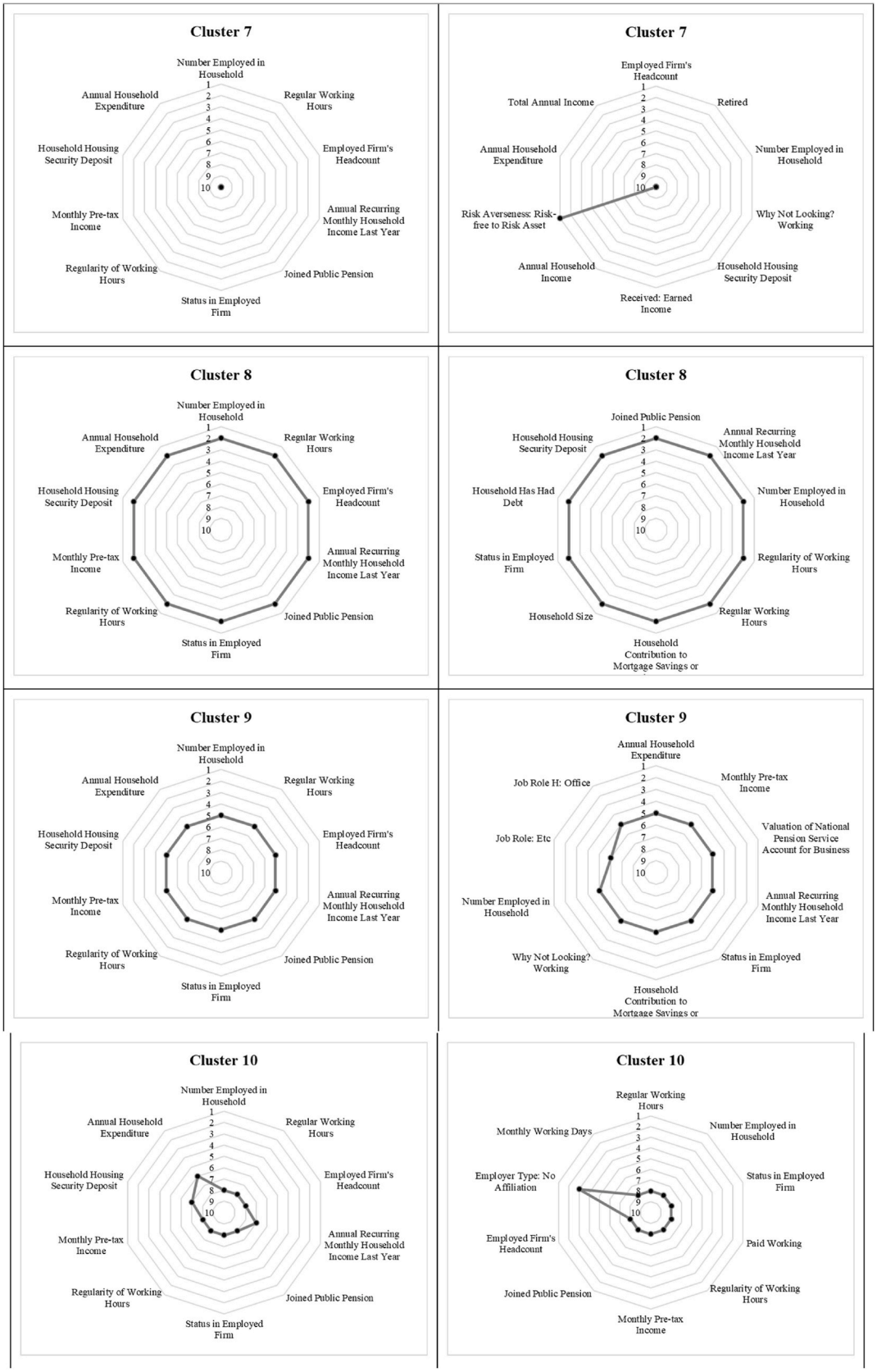

Fig. 8. (*continued*).

Table 8
Cluster-based customer segment labeling.

Cluster	Discretionary Labeling: Profile
1	Typical Upper-middle Class
2	Non-affiliated Lower Class
3	Middle Class with Government Aid
4	Wealthiest
5	Frugal Middle Class
6	Self-employed Hard-Working Middle Class
7	Highly Risk-averse Poorest
8	Typical Upper Class
9	Typical Middle Class
10	High-expenditure Lower Class

4.3.2.5. ***Most risk-tolerant group.*** This group represents the most risk-tolerant individuals. They can invest in numerous high-risk ETFs, actively pursuing higher returns and growth opportunities. The number of ETFs with non-zero weights in this group is the largest, demonstrating their capacity to tolerate aggressive investment strategies. Portfolios that maximize returns should be recommended to this group. The two clusters associated with this risk profile are Clusters 4 and 8, labeled 'Wealthiest, and 'Typical Upper Class.' Consistent with their high net assets and income levels, they can pursue the most aggressive financial product offerings. The sample recommended portfolios' expected volatilities and returns across all clusters are depicted in Fig. 10 in descending order.

The diversity in allocation strategies and risk preferences among different cluster groups exemplifies the wide range of approaches investors can adopt when investing in ETFs across various asset classes and sectors. This flexibility enables investors to construct portfolios that suit their risk tolerance, investment objectives, and market outlook. Asset classes such as equities, fixed income, commodities, currencies, and precious metals offer unique risk-reward profiles to help investors manage risk and optimize returns (Elton, Gruber, Brown and Goetzmann, 2014). The more detailed description of Fig. 9 are in

A study by Barber and Odean (2000) supports that investors' risk aversion levels significantly impact their investment choices. Furthermore, Statman (2004) argued that investors exhibit behavioral biases that affect their investment choices, leading to varying preferences and risk-taking behavior. These findings highlight the importance of understanding one's risk tolerance and considering behavioral factors when constructing a well-diversified portfolio across multiple asset classes and sectors.

Therefore, our analysis of investment groups is crucial in understanding investors' various financial-related features. By examining the allocation patterns within various cluster groups, we can gain valuable insights into how different investors approach the investment process, depending on their risk tolerance and objectives. This information is essential for portfolio managers and financial advisors, enabling them to tailor investment recommendations and strategies to suit individual clients' unique needs and preferences. Furthermore, cluster analysis can help identify emerging trends and shifts in the market, allowing investors to adjust their portfolios accordingly and seize potential opportunities. Employing customer-profiling clusters in the context of investment groups contributes to a more informed and strategic decision-making process, enhancing the likelihood of achieving desired investment outcomes.

5. Discussion

This research presents a practical and visually explainable clustering approach for high-dimensional socioeconomic data, serving as a vital tool in the digital economy. In this context, data assumes a pivotal role, and AI emerges as a strategic instrument for its management. While the proposed method systematically addresses the issues of high-dimensionality and ensures practical usability in various socio-economic contexts, it also carves a pathway in the DE, enhancing the management and utility of vast, multifaceted data. This research not only underpins the necessity of AI in optimizing and managing data but also elucidates how a structured, modular approach to data mining can be integral in driving digital economic activities and transformations. In the light of DE's characteristic of treating data resources as key elements, our methodology ensures that voluminous and high-dimensional data is utilized in a manner that drives economic decision-making and strategy formulation in the digital realm. In the pre-modeling stage, we introduce two data visualization methods for conducting EDA. First, as illustrated in Fig. 2, we represent high-dimensional data through category networks using correlation-based Euclidean distance. This approach necessitates minimal manual engineering if the high-dimensional dataset includes categories or other labels. Despite the potential need for manual labeling, which constitutes a supervised computing problem, the intuitive visualization of high-dimensional data segregated by labels can serve as a valuable tool for examining the underlying features that compose the dataset.

Secondly, as depicted in Fig. 3 and Table 4, we offer a network-based visualization tool that complements the commonly used relationship heatmap or matrix, typically applied with a correlation coefficient. The relationship cluster network provides network clusters given a relationship threshold. Again, the length of the edges corresponds to correlation-based Euclidean distances. Modelers can quickly grasp the multi-dimensional dependency structure of high-dimensional datasets by using the correlation heatmap, relationship cluster network, and corresponding table in the EDA process. This result significantly improves relationship matrices, which only display pairwise relationships and become increasingly illegible as the feature count increases.

We then search for an optimal configuration in the three-stage clustering approach, which is superior to two-stage dimensionality reduction and clustering approaches in previous literature. Compared to the existing deep clustering of large socioeconomic datasets in the remarkable work of Hwang et al. (2023), our study offers a granular approach, particularly in dimensionality reduction and clustering, and proves the effectiveness of the three-stage deep clustering with another large socioeconomic data set.

To ensure the scalability of our proposed system, we first analyze the computational complexity of the two novel network-based EDA algorithms in-depth. Here, both algorithms can scale reasonably under mild assumptions. Due to the graphical nature of the algorithms, with no assumptions, the computational complexity upper-bound is not exceptionally fast. Any introduction of new algorithms will require an additional examination of the computational complexity. All in all, none of the algorithms presented in the paper here are exponential. Therefore, we claim that it can scale reasonably.

Via SHAP values, our research delves deeper with a three-stage clustering, culminating in more refined customer segments with greater explainability. Based on our scaled aggregate scoring method in Equation (1), the potential customer population was effectively segregated into segments. To profile these customer segments and identify the influential features that form them, the top ten SHAP values are identified and visualized in aggregate and across clusters. The SHAP values help ascertain the data type, category, and factor affecting investment most relevant in customer profiling. The heterogeneity of the features in clusters is visualized through bar graphs. Subsequently, profiles are displayed using relative ranking-based radial charts corresponding to aggregate and cluster-specific top-ten features. Employing radial charts and bar graphs enables data scientists to describe each customer segment with intuitive, natural language.

While these methods present strengths through intuitive visualization and effective segmentation, potential limitations arise from the need for manual labeling in datasets lacking predefined categories and the need for comparing performance with other existing methods. The necessity for manual labeling in certain datasets diverges from the anticipation of a wholly automated approach. This deviation invites

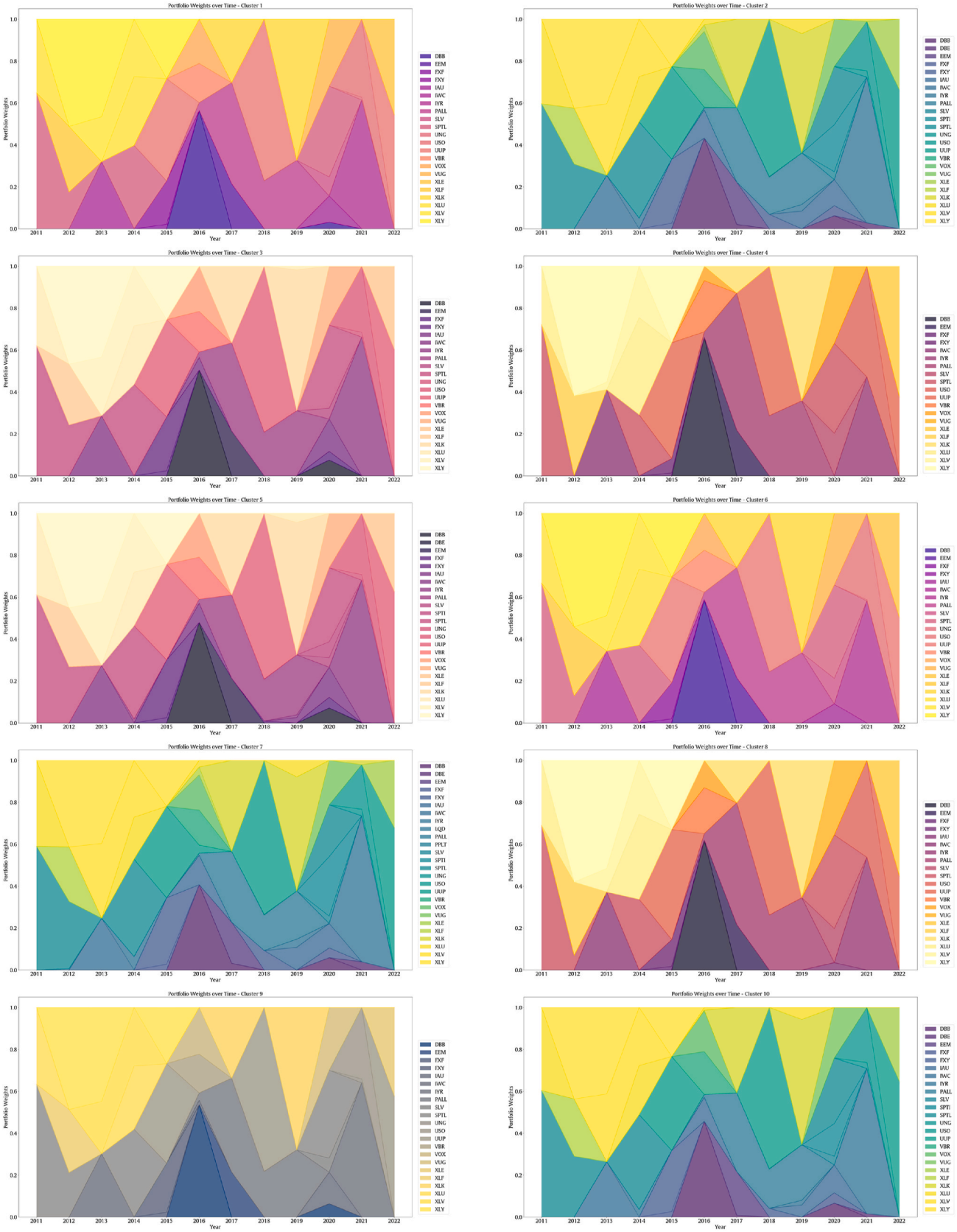

Fig. 9. Constructed portfolios by clusters.

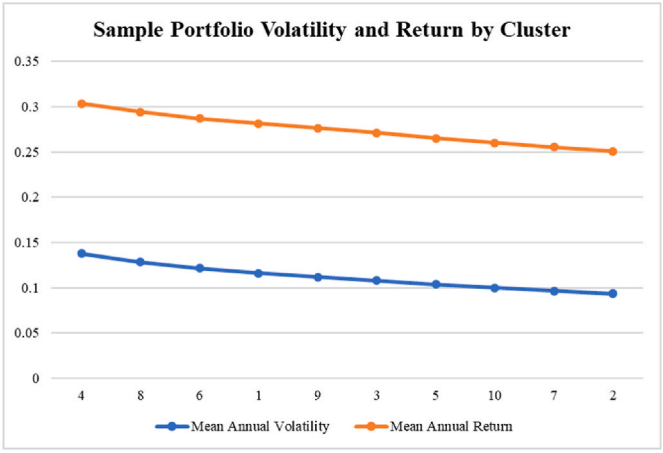

Fig. 10. Sample portfolio volatility and return by cluster.

further exploration into mechanisms that can autonomously categorize unlabeled high-dimensional data sets, minimizing human intervention. Rather than comparing different data mining algorithms, our experiments focused on an extensive evaluation of various autoencoder configurations and dimensionality reduction techniques within the three-stage algorithm in order to highlight the system's explainability and flexibility for users. By assessing the clustering performance of the original high-dimensional data without reduction, we demonstrated that both the autoencoder and dimensionality reduction techniques contribute to concrete segmentation with distinct heterogeneities.

Our research propounds an innovative approach, pivoted around a three-stage clustering methodology tailored for high-dimensional socioeconomic data. While the robustness and efficacy of this methodology are evident in its application to our specific data set, it is unlikely that the resulting specification is optimal for all data distributions. For instance, a data set with a much larger data set, perhaps due to the inclusion of a temporal dimension, may realistically benefit from a smaller subset of more computationally efficient algorithms. Therefore, we highlight that users apply the systematic approach we present for the three-stage clustering algorithm with a set of component algorithms reasonable for their specific data set.

Our research findings have practical implications for analyzing high-dimensional data in various fields. While innovative algorithmic-level deep clustering methods, such as in Contrastive Clustering (Li et al., 2021) and Interpretable Neural Clustering (Peng et al., 2022), have been already introduced, data mining algorithms for socio-economic and demographic datasets have lagged behind those for text, images, and multi-modal data sets. Our proposed data mining system, which includes network-based EDA and explainable three-stage clustering, demonstrates promising potential for an end-to-end expert system pipeline. We envisage our approach as being inherently modular, allowing each stage to be adaptable based on the specific requirements of the data set and the end goal of the analysis. This modularity grants the flexibility to either adopt the entire three-stage process or selectively apply individual stages based on the inherent structure and nuances of the data. For example, In our pursuit of elucidating high-dimensional data structures, we can apply the work of Demšar (2006) in advancing statistical analyses across multiple classifiers and datasets, and He et al. (2022) in their intricate development of a dependency-oriented method for classifying mixed-attribute data. Spanning from socioeconomics to business intelligence, our proposed method can be integrated depending on the specific domain and problem set. This includes data-driven marketing strategies and recommendations that emphasize the need for in-depth interpretability of AI strategies. To demonstrate this, our method-based curated portfolio recommendations promise more intuitive customer segment engagement. The intuitive analysis derived from our proposed data mining system enriches examining and discussing recommendations for customer segments, making them more accessible to non-experts.

6. Conclusion

In a world progressively steering towards a digital economy, the data mining of large customer bases is not merely a technological endeavour but a vital economic activity. Our research underscores the utility and necessity of a system in navigating through the voluminous and high-dimensional data spaces encountered by firms, especially those with expansive customer populations, in the digital age. The system, while being firmly rooted in the principles and functionalities of AI, goes beyond mere data management. It facilitates a structured and economically insightful navigation through data, thereby acting as a catalyst in driving data-based decision-making and strategies in the DE. The proposed system in this research ensures that AI is not just a computational tool but a strategic ally in managing, optimizing, and deriving actionable insights from data, which is fundamental in a DE. By maintaining a delicate balance between computational efficiency, data management, and economic insight, the research both propounds a viable pathway for managing high-dimensional data and ensures that the pathway is economically insightful and aligned with the principles and strategies pivotal in a DE.

A primary challenge in data mining and data-driven decision-making is the curse of dimensionality. This issue not only hampers the effectiveness and computational efficiency of traditional data-mining algorithms but also presents an obstacle in visualizing and interpreting high-dimensional datasets. This problem is especially pertinent in finance, where interpretable models and intuitive communication with non-expert stakeholders are crucial.

We propose an end-to-end system with network-based EDA and explainable three-stage clustering approach to provide a more detailed understanding of high-dimensional data using numerous visualization approaches. Our incorporation of SHAP values in customer profiling reflects an ongoing effort to provide businesses with tools that aid in-firm and out-firm stakeholders like non-technical managers and regulators in understanding black-box-like computational systems. We also demonstrate a potential downstream application in the form of portfolio recommendations for profiled potential customers.

However, there are limitations to this study. Firstly, only some configuration combinations in the three-stage clustering framework are tested globally. The research does not examine all dimensionality reduction and clustering methods. Due to limited computational resources, a trade-off between global optimization and computational efficiency was necessary. Future research could test more combinations with different types of autoencoder, other dimensionality reduction techniques, and other clustering methods. However, the impact on final clustering performance and downstream applications is likely minimal. Another limitation is the transferability of the optimal configuration to different datasets. Instead of replicating this study's optimal configuration, future research and applications should emulate the optimal configuration search process, ensuring the three-stage process is optimally configured for the corresponding dataset. Moreover, challenges like manual labeling in specific datasets and balancing optimization with computational efficiency are ever-present reminders of the intricacies inherent in our domain. Such challenges not only ground our work but also inspire future explorations and refinements.

Further research can build upon our contributions and limitations by exploring alternative dimensionality reduction and clustering methods and evaluating their effects on clustering performance and downstream applications. Additionally, future studies could investigate the generalizability of our approach to other datasets such as text, images, and multi-modal datasets, and develop innovative methods to optimize the configuration search process. Such research would strengthen the robustness and applicability of our proposed end-to-end framework and

broaden its potential impact across multiple sectors.

In the broader context of data science, our study represents one of many steps in its continuous evolution. We hope our approach adds a complementary perspective to existing methods, offering a way to interpret and categorize data. We also hope the emphasis on interpretability and incorporation of visualization in every aspect of the end-to-end data mining pipeline sets the stage for more literature development on methods that enhance the communication of data analytics to non-experts. This will help accelerate the pace of digitization in the field of finance.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Appendix

Table A1
All Features of NaSTaB Data

No	Feature	Data Type	Category	Factor Affecting Investment
1	Has Income	Categorical	Financial	Income
2	Usually Has Income	Categorical	Financial	Income
3	Applicable for Social Insurance and Support	Categorical	Financial	Subsidy
4	Receives Payment: Personal Pension Fund	Categorical	Financial	Insurance Income
5	Receives Payment: Personal Pension Insurance	Categorical	Financial	Insurance Income
6	Receives Payment: Personal Injury Insurance	Categorical	Financial	Insurance Income
7	Receives Payment: Personal Deposit Insurance	Categorical	Financial	Insurance Income
8	Receives Payment: Personal Pension or Insurance	Categorical	Financial	Insurance Income
9	Receives Payment: Personal Farmland Pension	Categorical	Financial	Insurance Income
10	Received: Government Handout	Categorical	Financial	Subsidy
11	Received: Retirement Pension	Categorical	Financial	Insurance Income
12	Received: Civil Servant Retirement Pension	Categorical	Financial	Insurance Income
13	Received: Private School Retirement Pension	Categorical	Financial	Insurance Income
14	Received: Military Retirement Pension	Categorical	Financial	Insurance Income
15	Received: Police Officer Retirement Pension	Categorical	Financial	Insurance Income
16	Received: Military Retirement Pension Lumpsum	Categorical	Financial	Insurance Income
17	Received: Police Officer Retirement Pension Lumpsum	Categorical	Financial	Insurance Income
18	Received: Personal Retirement Pension	Categorical	Financial	Insurance Income
19	Received: Personal Retirement Pension Lumpsum	Categorical	Financial	Insurance Income
20	Received: Earned Income	Categorical	Financial	Income
21	Received: Business Revenue	Categorical	Financial	Income
22	Received: Business Net Income	Categorical	Financial	Income
23	Received: Business Net Loss	Categorical	Financial	Asset
24	Received: Any Rental Earning	Categorical	Financial	Income
25	Received: Land Rental Earning	Categorical	Financial	Income
26	Received: Housing Rental Earning	Categorical	Financial	Income
27	Received: Office Rental Earning	Categorical	Financial	Income
28	Received: Other Rental Earning	Categorical	Financial	Income
29	Received: Interest or Dividend Earning	Categorical	Financial	Income
30	Received: Any Other Earning	Categorical	Financial	Income
31	Received: Other, Nonhousehold Earning	Categorical	Financial	Income
32	Received: Other, Elder Care Earning	Categorical	Financial	Subsidy
33	Received: Other, Other Earning	Categorical	Financial	Income
34	Received: Capital Gain	Categorical	Financial	Asset
35	Received: Net Capital Gain, Korean Stocks	Categorical	Financial	Asset
36	Received: Net Capital Loss, Korean Stocks	Categorical	Financial	Asset
37	Received: Net Capital Gain, Overseas Stocks	Categorical	Financial	Asset
38	Received: Net Capital Loss, Overseas Stocks	Categorical	Financial	Asset
39	Received: Net Capital Gain, Other Securities	Categorical	Financial	Asset
40	Received: Net Capital Loss, Other Securities	Categorical	Financial	Asset
41	Received: Net Capital Gain, Crypto	Categorical	Financial	Asset
42	Received: Net Capital Loss, Crypto	Categorical	Financial	Asset
43	Registered Rental Operator	Categorical	Financial	Asset
44	Contribute to Public Pension	Categorical	Financial	Insurance Expenditure
45	Household Poverty Last Year	Categorical	Financial	Subsidy
46	Household Has Savings or Deposit Account	Categorical	Financial	Asset
47	Household Invested in Financial Fund	Categorical	Financial	Asset
48	Household Invested in Fixed Income	Categorical	Financial	Asset
49	Household Invested in Stocks	Categorical	Financial	Asset
50	Household Has Deposit Insurance	Categorical	Financial	Asset
51	Household Has Retirement Savings Plan	Categorical	Financial	Asset
52	Household Invested in Crypto	Categorical	Financial	Asset
53	Household Loaned Out Money	Categorical	Financial	Asset
54	Household Other Financial Assets	Categorical	Financial	Asset
55	Household Contributed to Self-housing Savings or Fund Last Year	Categorical	Financial	Asset
56	Household Home Owner	Categorical	Financial	Asset
57	Household Multiple Home Owner	Categorical	Financial	Asset
58	Household Land or Building Owner	Categorical	Financial	Asset

(*continued on next page*)

Table A1 (*continued*)

No	Feature	Data Type	Category	Factor Affecting Investment
59	Household Precious Metals Owner	Categorical	Financial	Asset
60	Household Automobile Owner	Categorical	Financial	Asset
61	Household Other Asset Owner	Categorical	Financial	Asset
62	Household Residential Rental Deposit Owner	Categorical	Financial	Asset
63	Household Member is Residential Rental Deposit Owner	Categorical	Financial	Asset
64	Household Member is Other Residential Rental Deposit Owner	Categorical	Financial	Asset
65	Household Non-residential Rental Deposit Owner	Categorical	Financial	Asset
66	Household Has Had Debt	Categorical	Financial	Liability
67	In City	Categorical	Demographic	Residence
68	>15 Years Old	Categorical	Demographic	Age
69	Paid Working	Categorical	Demographic	Occupation
70	Regular Working Hours	Categorical	Demographic	Occupation
71	Works Overtime	Categorical	Demographic	Occupation
72	Paid for Overtime	Categorical	Demographic	Occupation
73	Employer Type: Etc	Categorical	Demographic	Occupation
74	Employer Type: Foreign	Categorical	Demographic	Occupation
75	Employer Type: Governmental	Categorical	Demographic	Occupation
76	Employer Type: Incorporation	Categorical	Demographic	Occupation
77	Employer Type: No Affiliation	Categorical	Demographic	Occupation
78	Employer Type: Nongovernmental Public	Categorical	Demographic	Occupation
79	Employer Type: Nonprofit	Categorical	Demographic	Occupation
80	Employer Type: Private	Categorical	Demographic	Occupation
81	Employer Industry Type: Accomodation and Dining	Categorical	Demographic	Occupation
82	Employer Industry Type: Agriculture and Fish	Categorical	Demographic	Occupation
83	Employer Industry Type: Arts, Sports, Leisure	Categorical	Demographic	Occupation
84	Employer Industry Type: Business Support	Categorical	Demographic	Occupation
85	Employer Industry Type: Construction	Categorical	Demographic	Occupation
86	Employer Industry Type: Education	Categorical	Demographic	Occupation
87	Employer Industry Type: Electricity, Gas, Waterworks	Categorical	Demographic	Occupation
88	Employer Industry Type: Environmental	Categorical	Demographic	Occupation
89	Employer Industry Type: Etc	Categorical	Demographic	Occupation
90	Employer Industry Type: Finance	Categorical	Demographic	Occupation
91	Employer Industry Type: Household and Others	Categorical	Demographic	Occupation
92	Employer Industry Type: International	Categorical	Demographic	Occupation
93	Employer Industry Type: Manufacturing	Categorical	Demographic	Occupation
94	Employer Industry Type: Media	Categorical	Demographic	Occupation
95	Employer Industry Type: Mining	Categorical	Demographic	Occupation
96	Employer Industry Type: Public	Categorical	Demographic	Occupation
97	Employer Industry Type: Real Estate	Categorical	Demographic	Occupation
98	Employer Industry Type: Repair	Categorical	Demographic	Occupation
99	Employer Industry Type: Science and Technology	Categorical	Demographic	Occupation
100	Employer Industry Type: Social	Categorical	Demographic	Occupation
101	Employer Industry Type: Transportation	Categorical	Demographic	Occupation
102	Employer Industry Type: Wholesale and Retail	Categorical	Demographic	Occupation
103	Job Role: Agriculture and Fishing	Categorical	Demographic	Occupation
104	Job Role: Etc	Categorical	Demographic	Occupation
105	Job Role: Expert	Categorical	Demographic	Occupation
106	Job Role: Machine Equipment	Categorical	Demographic	Occupation
107	Job Role: Management	Categorical	Demographic	Occupation
108	Job Role: Military	Categorical	Demographic	Occupation
109	Job Role: Office	Categorical	Demographic	Occupation
110	Job Role: Sales	Categorical	Demographic	Occupation
111	Job Role: Service	Categorical	Demographic	Occupation
112	Job Role: Simple Labor	Categorical	Demographic	Occupation
113	Job Role: Technician	Categorical	Demographic	Occupation
114	Job Role H: Manager	Categorical	Demographic	Occupation
115	Job Role H: Expert	Categorical	Demographic	Occupation
116	Job Role H: Office	Categorical	Demographic	Occupation
117	Job Role H: Service	Categorical	Demographic	Occupation
118	Job Role H: Sales	Categorical	Demographic	Occupation
119	Job Role H: Agriculture	Categorical	Demographic	Occupation
120	Job Role H: Skill	Categorical	Demographic	Occupation
121	Job Role H: Machine	Categorical	Demographic	Occupation
122	Job Role H: Labor	Categorical	Demographic	Occupation
123	Job Role H: Soldier	Categorical	Demographic	Occupation
124	Job Role H: Etc	Categorical	Demographic	Occupation
125	Job Role H: Housewife	Categorical	Demographic	Occupation
126	Job Role H: Jobless	Categorical	Demographic	Occupation
127	Job Role H: Student	Categorical	Demographic	Education
128	Job Role H: Preschool	Categorical	Demographic	Education
129	Residence City: Busan	Categorical	Demographic	Residence
130	Residence City: Chungbuk	Categorical	Demographic	Residence
131	Residence City: Chungnam	Categorical	Demographic	Residence
132	Residence City: Daegu	Categorical	Demographic	Residence
133	Residence City: Daejeon	Categorical	Demographic	Residence
134	Residence City: Gangwon	Categorical	Demographic	Residence

(*continued on next page*)

Table A1 (*continued*)

No	Feature	Data Type	Category	Factor Affecting Investment
135	Residence City: Gwangju	Categorical	Demographic	Residence
136	Residence City: Gyeongbuk	Categorical	Demographic	Residence
137	Residence City: Gyeongi	Categorical	Demographic	Residence
138	Residence City: Gyeongnam	Categorical	Demographic	Residence
139	Residence City: Incheon	Categorical	Demographic	Residence
140	Residence City: Jeju	Categorical	Demographic	Residence
141	Residence City: Jeonbuk	Categorical	Demographic	Residence
142	Residence City: Jeonnam	Categorical	Demographic	Residence
143	Residence City: Sejong	Categorical	Demographic	Residence
144	Residence City: Seoul	Categorical	Demographic	Residence
145	Residence City: Ulsan	Categorical	Demographic	Residence
146	Urban Residence	Categorical	Demographic	Residence
147	Education Attainment for 10	Categorical	Demographic	Education
148	Working Months Last Year for 10	Categorical	Demographic	Occupation
149	Unemployed but Looking	Categorical	Decision-making Traits	Others
150	Joined Public Pension	Categorical	Decision-making Traits	Others
151	Not Looking for Work	Categorical	Decision-making Traits	Others
152	Looking for Part Time Work	Categorical	Decision-making Traits	Others
153	Looking for Full Time Work	Categorical	Decision-making Traits	Others
154	Why Not Looking? Education	Categorical	Lifestyle	Lifestyle
155	Why Not Looking? Etc	Categorical	Lifestyle	Lifestyle
156	Why Not Looking? Working	Categorical	Lifestyle	Lifestyle
157	Why Not Looking? Housework	Categorical	Lifestyle	Lifestyle
158	Why Not Looking? No Ideal Work	Categorical	Lifestyle	Lifestyle
159	Why Not Looking? Poor Health	Categorical	Lifestyle	Health
160	Why Not Looking? Preschool	Categorical	Lifestyle	Lifestyle
161	Why Not Looking? Retirement	Categorical	Lifestyle	Lifestyle
162	Why Not Looking? Take Time Off	Categorical	Lifestyle	Lifestyle
163	Why Not Looking? Wealthy	Categorical	Lifestyle	Lifestyle
164	Total Annual Income	Numerical	Financial	Income
165	Annual Social Insurance Income	Numerical	Financial	Insurance Income
166	Annual Private Insurance Income	Numerical	Financial	Insurance Income
167	Annual Government Cash Subsidy Received	Numerical	Financial	Subsidy
168	Annual Income from Work	Numerical	Financial	Income
169	Annual Net Income from Business	Numerical	Financial	Income
170	Annual Real Estate Lease Income	Numerical	Financial	Income
171	Annual Interest, Dividend, Capital Gains Income	Numerical	Financial	Income
172	Annual Other Income	Numerical	Financial	Income
173	Monthly Pre-tax Income	Numerical	Financial	Income
174	Income from Work	Numerical	Financial	Income
175	Annual Business Revenue	Numerical	Financial	Income
176	Annual Business Net Income	Numerical	Financial	Income
177	Total Real Estate Lease Income	Numerical	Financial	Income
178	Annual Interest and Dividend Income	Numerical	Financial	Income
179	Other Income	Numerical	Financial	Income
180	Annual Other, Non-household Living, Schooling Aid Received	Numerical	Financial	Income
181	Annual Capital Gain	Numerical	Financial	Asset
182	Annual Capital Loss	Numerical	Financial	Asset
183	Annual Net Income from Korean Stocks	Numerical	Financial	Asset
184	Annual Net Loss from Korean Stocks	Numerical	Financial	Asset
185	Annual Net Income from Overseas Stocks	Numerical	Financial	Asset
186	Annual Net Loss from Overseas Stocks	Numerical	Financial	Asset
187	Annual Net Income from Fixed Income, Funds, and Traded Securities	Numerical	Financial	Asset
188	Annual Net Loss from Fixed Income, Funds, and Traded Securities	Numerical	Financial	Asset
189	Annual Net Income Crypto	Numerical	Financial	Asset
190	Annual Net Loss Crypto	Numerical	Financial	Asset
191	Annual Real Estate Lease Business Earning Last Year	Numerical	Financial	Asset
192	Total Valuation of National Pension Service Account	Numerical	Financial	Insurance Expenditure
193	Valuation of National Pension Service Account for Business	Numerical	Financial	Insurance Expenditure
194	Valuation of Local District National Pension Service Account	Numerical	Financial	Insurance Expenditure
195	Total Contribution to Health Insurance Last Year	Numerical	Financial	Insurance Expenditure
196	Annual Contribution to Unemployment Insurance	Numerical	Financial	Insurance Expenditure
197	Valuation of Private Pension Plans	Numerical	Financial	Insurance Expenditure
198	Valuation of Private Pension Insurance	Numerical	Financial	Insurance Expenditure
199	Valuation of Private Injury and Life Insurance	Numerical	Financial	Insurance Expenditure
200	Valuation of Automobile Insurance	Numerical	Financial	Insurance Expenditure
201	Total Credit and Debit Card Expenditure in Korea Last Year	Numerical	Financial	Expenditure
202	Credit Card Expenditure in Korea Last Year	Numerical	Financial	Expenditure
203	Debit Card Expenditure in Korea Last Year	Numerical	Financial	Expenditure
204	Total Credit and Debit Card Overseas Expenditure Last Year	Numerical	Financial	Expenditure
205	Credit Card Overseas Expenditure Last Year	Numerical	Financial	Expenditure
206	Debit Card Overseas Expenditure Last Year	Numerical	Financial	Expenditure
207	Credit or Debit Card Expenditure in Traditional Market Last Year	Numerical	Financial	Expenditure
208	Total Social Insurance Subsidy Received	Numerical	Financial	Subsidy
209	Total Government Cash Handout Received	Numerical	Financial	Subsidy
210	Total Health Insurance Payments	Numerical	Financial	Insurance Expenditure

(*continued on next page*)

Table A1 (*continued*)

No	Feature	Data Type	Category	Factor Affecting Investment
211	Luxury Level of Housing	Numerical	Financial	Asset
212	Total Household Real Estate Wealth	Numerical	Financial	Asset
213	Household Owns Car	Numerical	Financial	Asset
214	Monthly Household Income Last Year	Numerical	Financial	Income
215	Annual Household Income	Numerical	Financial	Income
216	Annual Household Expenditure	Numerical	Financial	Expenditure
217	Annual Recurring Monthly Household Income Last Year	Numerical	Financial	Income
218	Annual Monthly Direct Expenditure Last Year	Numerical	Financial	Expenditure
219	Average Household Monthly Savings Last Year	Numerical	Financial	Asset
220	Annual Household Housing Expense	Numerical	Financial	Expenditure
221	Annual Household Food Expenditure	Numerical	Financial	Expenditure
222	Annual Household Alcoholic Beverage Expenditure	Numerical	Financial	Expenditure
223	Annual Household Transportation Expenditure	Numerical	Financial	Expenditure
224	Annual Household Telecom Expenditure	Numerical	Financial	Expenditure
225	Annual Household Electronics Expenditure	Numerical	Financial	Expenditure
226	Annual Household Communication Equipment Expenditure	Numerical	Financial	Expenditure
227	Annual Household Private Insurance Expenditure	Numerical	Financial	Insurance Expenditure
228	Annual Household Health Insurance Expenditure	Numerical	Financial	Insurance Expenditure
229	Household Inheritance Received	Numerical	Financial	Asset
230	Household Inheritance Given	Numerical	Financial	Asset
231	Household Monthly Poverty Subsidy	Numerical	Financial	Subsidy
232	Household Annual Poverty Subsidy	Numerical	Financial	Subsidy
233	Total Household Savings and Deposits	Numerical	Financial	Asset
234	Total Household Funds Valuation	Numerical	Financial	Asset
235	Total Household Fixed Income Valuation	Numerical	Financial	Asset
236	Total Household Stocks Valuation	Numerical	Financial	Asset
237	Total Household Savings and Pension Insurance Valuation	Numerical	Financial	Asset
238	Total Household Pension Valuation	Numerical	Financial	Asset
239	Total Household Crypto Valuation	Numerical	Financial	Asset
240	Total Household Lendings	Numerical	Financial	Asset
241	Total Household Other Financial Assets	Numerical	Financial	Asset
242	Household Contribution to Mortgage Savings or Fund Last Year	Numerical	Financial	Asset
243	Current Home Valuation	Numerical	Financial	Asset
244	Household Non-living Housing Valuation	Numerical	Financial	Asset
245	Household Land and Building Valuation	Numerical	Financial	Asset
246	Household Precious Metals Valuation	Numerical	Financial	Asset
247	Household Car Valuation	Numerical	Financial	Asset
248	Household Other Assets Valuation	Numerical	Financial	Asset
249	Household Housing Security Deposit	Numerical	Financial	Asset
250	Housing Security Deposit of Household Member	Numerical	Financial	Asset
251	Non-living Housing Security Deposit of Household Member	Numerical	Financial	Asset
252	Non-housing Real Estate Security Deposit	Numerical	Financial	Asset
253	Total Household Debt at End of Last Year	Numerical	Financial	Liability
254	Number of Household Members with Income >15	Numerical	Financial	Income
255	Household Cars	Numerical	Financial	Asset
256	Household Motorcycles	Numerical	Financial	Asset
257	Employed Firm's Headcount	Numerical	Demographic	Occupation
258	Status in Employed Firm	Numerical	Demographic	Occupation
259	Regularity of Working Hours	Numerical	Demographic	Occupation
260	Monthly Overtime Hours	Numerical	Demographic	Occupation
261	Daily Working Hours	Numerical	Demographic	Occupation
262	Monthly Working Days	Numerical	Demographic	Occupation
263	Change in Working Hours, against Last Year	Numerical	Demographic	Occupation
264	Number Employed in Household	Numerical	Demographic	Occupation
265	Household Employment Rate	Numerical	Demographic	Occupation
266	<18 in Household	Numerical	Demographic	Age
267	0 to 6 in Household	Numerical	Demographic	Age
268	6 to 18 in Household	Numerical	Demographic	Age
269	Education Attainment for 1	Numerical	Demographic	Education
270	Working Months Last Year for 1	Numerical	Demographic	Occupation
271	Education Attainment for 2	Numerical	Demographic	Education
272	Working Months Last Year for 2	Numerical	Demographic	Occupation
273	Education Attainment for 3	Numerical	Demographic	Education
274	Working Months Last Year for 3	Numerical	Demographic	Occupation
275	Education Attainment for 4	Numerical	Demographic	Education
276	Working Months Last Year for 4	Numerical	Demographic	Occupation
277	Education Attainment for 5	Numerical	Demographic	Education
278	Working Months Last Year for 5	Numerical	Demographic	Occupation
279	Education Attainment for 6	Numerical	Demographic	Education
280	Working Months Last Year for 6	Numerical	Demographic	Occupation
281	Education Attainment for 7	Numerical	Demographic	Education
282	Working Months Last Year for 7	Numerical	Demographic	Occupation
283	Education Attainment for 8	Numerical	Demographic	Education
284	Working Months Last Year for 8	Numerical	Demographic	Occupation
285	Education Attainment for 9	Numerical	Demographic	Education
286	Working Months Last Year for 9	Numerical	Demographic	Occupation

(*continued on next page*)

Table A1 (*continued*)

No	Feature	Data Type	Category	Factor Affecting Investment
287	Risk Averseness: Risk-free to Risk Asset	Numerical	Decision-making Traits	Risk Tolerance
288	Why Looking for Work?	Numerical	Lifestyle	Lifestyle
289	Credit or Debit Card Expenditure on Public Transportation Last Year	Numerical	Lifestyle	Lifestyle
290	Total Cash Receipt Last Year	Numerical	Lifestyle	Lifestyle
291	Current Health	Numerical	Lifestyle	Health
292	Self Life Expectancy	Numerical	Lifestyle	Health
293	Retired	Numerical	Lifestyle	Lifestyle
294	Expected Retirement Age	Numerical	Lifestyle	Lifestyle
295	Minimum Post-retirement Monthly Disposable Income	Numerical	Lifestyle	Lifestyle
296	Ideal Post-retirement Monthly Disposable Income	Numerical	Lifestyle	Lifestyle
297	Household Size	Numerical	Lifestyle	Lifestyle
298	Total Household Size	Numerical	Lifestyle	Lifestyle
299	Annual Household Tobacco Expenditure	Numerical	Lifestyle	Health
300	Annual Household Leisure Expenditure	Numerical	Lifestyle	Lifestyle
301	Annual Household Equipment for Leisure Expenditure	Numerical	Lifestyle	Lifestyle
302	Annual Household Clothing Expenditure	Numerical	Lifestyle	Lifestyle
303	Annual Household Beauty Expenditure	Numerical	Lifestyle	Lifestyle
304	Annual Household Travel Expenditure	Numerical	Lifestyle	Lifestyle
305	Annual Household Education Expenditure	Numerical	Lifestyle	Lifestyle
306	Annual Household Health Expenditure	Numerical	Lifestyle	Lifestyle

Table A2
Supplement to Fig. 2

Feature	Category	Cluster	Cluster Size
Total Annual Income	Financial	Cluster 1	135
Annual Interest, Dividend, Capital Gains Income	Financial	Cluster 1	135
Monthly Pre-tax Income	Financial	Cluster 1	135
Annual Capital Gain	Financial	Cluster 1	135
Annual Net Income from Korean Stocks	Financial	Cluster 1	135
Annual Household Income	Financial	Cluster 1	135
Total Household Stocks Valuation	Financial	Cluster 1	135
Annual Income from Work	Financial	Cluster 1	135
Total Valuation of National Pension Service Account	Financial	Cluster 1	135
Valuation of National Pension Service Account for Business	Financial	Cluster 1	135
Annual Contribution to Unemployment Insurance	Financial	Cluster 1	135
Employed Firm's Headcount	Demographic	Cluster 1	135
Income from Work	Financial	Cluster 1	135
Status in Employed Firm	Demographic	Cluster 1	135
Retired	Lifestyle	Cluster 1	135
Working Months Last Year for 1	Demographic	Cluster 1	135
Regularity of Working Hours	Demographic	Cluster 1	135
Daily Working Hours	Demographic	Cluster 1	135
Monthly Working Days	Demographic	Cluster 1	135
Number Employed in Household	Demographic	Cluster 1	135
Household Employment Rate	Demographic	Cluster 1	135
Total Contribution to Health Insurance Last Year	Financial	Cluster 1	135
Total Credit and Debit Card Expenditure in Korea Last Year	Financial	Cluster 1	135
Credit Card Expenditure in Korea Last Year	Financial	Cluster 1	135
Total Health Insurance Payments	Financial	Cluster 1	135
Monthly Household Income Last Year	Financial	Cluster 1	135
Annual Recurring Monthly Household Income Last Year	Financial	Cluster 1	135
Annual Household Health Insurance Expenditure	Financial	Cluster 1	135
Minimum Post-retirement Monthly Disposable Income	Lifestyle	Cluster 1	135
Annual Monthly Direct Expenditure Last Year	Financial	Cluster 1	135
Household Size	Lifestyle	Cluster 1	135
<18 in Household	Demographic	Cluster 1	135
6 to 18 in Household	Demographic	Cluster 1	135
Education Attainment for 2	Demographic	Cluster 1	135
Education Attainment for 3	Demographic	Cluster 1	135
Education Attainment for 4	Demographic	Cluster 1	135
Annual Household Expenditure	Financial	Cluster 1	135
Annual Household Food Expenditure	Financial	Cluster 1	135
Annual Household Telecom Expenditure	Financial	Cluster 1	135
Number of Household Members with Income >15	Financial	Cluster 1	135
Total Household Size	Lifestyle	Cluster 1	135
Working Months Last Year for 2	Demographic	Cluster 1	135
Working Months Last Year for 3	Demographic	Cluster 1	135
Annual Household Education Expenditure	Lifestyle	Cluster 1	135
Education Attainment for 1	Demographic	Cluster 1	135
Working Months Last Year for 4	Demographic	Cluster 1	135
Average Household Monthly Savings Last Year	Financial	Cluster 1	135

(*continued on next page*)

Table A2 (*continued*)

Feature	Category	Cluster	Cluster Size
Annual Household Clothing Expenditure	Lifestyle	Cluster 1	135
Annual Household Beauty Expenditure	Lifestyle	Cluster 1	135
Annual Household Private Insurance Expenditure	Financial	Cluster 1	135
Household Cars	Financial	Cluster 1	135
Annual Income from Work	Financial	Cluster 1	135
Employed Firm's Headcount	Demographic	Cluster 1	135
Status in Employed Firm	Demographic	Cluster 1	135
Regularity of Working Hours	Demographic	Cluster 1	135
Monthly Pre-tax Income	Financial	Cluster 1	135
Total Valuation of National Pension Service Account	Financial	Cluster 1	135
Valuation of National Pension Service Account for Business	Financial	Cluster 1	135
Annual Contribution to Unemployment Insurance	Financial	Cluster 1	135
Daily Working Hours	Demographic	Cluster 1	135
Monthly Working Days	Demographic	Cluster 1	135
Income from Work	Financial	Cluster 1	135
Retired	Lifestyle	Cluster 1	135
Working Months Last Year for 1	Demographic	Cluster 1	135
Number Employed in Household	Demographic	Cluster 1	135
Household Employment Rate	Demographic	Cluster 1	135
Total Contribution to Health Insurance Last Year	Financial	Cluster 1	135
Total Credit and Debit Card Expenditure in Korea Last Year	Financial	Cluster 1	135
Credit Card Expenditure in Korea Last Year	Financial	Cluster 1	135
Total Health Insurance Payments	Financial	Cluster 1	135
Monthly Household Income Last Year	Financial	Cluster 1	135
Annual Recurring Monthly Household Income Last Year	Financial	Cluster 1	135
Annual Household Health Insurance Expenditure	Financial	Cluster 1	135
Minimum Post-retirement Monthly Disposable Income	Lifestyle	Cluster 1	135
Annual Monthly Direct Expenditure Last Year	Financial	Cluster 1	135
Household Size	Lifestyle	Cluster 1	135
<18 in Household	Demographic	Cluster 1	135
6 to 18 in Household	Demographic	Cluster 1	135
Education Attainment for 2	Demographic	Cluster 1	135
Education Attainment for 3	Demographic	Cluster 1	135
Education Attainment for 4	Demographic	Cluster 1	135
Annual Household Expenditure	Financial	Cluster 1	135
Annual Household Food Expenditure	Financial	Cluster 1	135
Annual Household Telecom Expenditure	Financial	Cluster 1	135
Number of Household Members with Income >15	Financial	Cluster 1	135
Total Household Size	Lifestyle	Cluster 1	135
Working Months Last Year for 2	Demographic	Cluster 1	135
Working Months Last Year for 3	Demographic	Cluster 1	135
Annual Household Education Expenditure	Lifestyle	Cluster 1	135
Education Attainment for 1	Demographic	Cluster 1	135
Working Months Last Year for 4	Demographic	Cluster 1	135
Annual Household Income	Financial	Cluster 1	135
Average Household Monthly Savings Last Year	Financial	Cluster 1	135
Annual Household Clothing Expenditure	Lifestyle	Cluster 1	135
Annual Household Beauty Expenditure	Lifestyle	Cluster 1	135
Annual Household Private Insurance Expenditure	Financial	Cluster 1	135
Household Cars	Financial	Cluster 1	135
Valuation of Private Injury and Life Insurance	Financial	Cluster 1	135
Annual Household Private Insurance Expenditure	Financial	Cluster 1	135
Monthly Household Income Last Year	Financial	Cluster 1	135
Annual Household Health Insurance Expenditure	Financial	Cluster 1	135
Household Cars	Financial	Cluster 1	135
Annual Household Income	Financial	Cluster 1	135
Annual Household Expenditure	Financial	Cluster 1	135
Annual Recurring Monthly Household Income Last Year	Financial	Cluster 1	135
Annual Monthly Direct Expenditure Last Year	Financial	Cluster 1	135
Average Household Monthly Savings Last Year	Financial	Cluster 1	135
Annual Household Food Expenditure	Financial	Cluster 1	135
Minimum Post-retirement Monthly Disposable Income	Lifestyle	Cluster 1	135
Ideal Post-retirement Monthly Disposable Income	Lifestyle	Cluster 1	135
Monthly Household Income Last Year	Financial	Cluster 1	135
Annual Monthly Direct Expenditure Last Year	Financial	Cluster 1	135
Household Size	Lifestyle	Cluster 1	135
Annual Household Expenditure	Financial	Cluster 1	135
Annual Household Food Expenditure	Financial	Cluster 1	135
Annual Household Telecom Expenditure	Financial	Cluster 1	135
Number of Household Members with Income >15	Financial	Cluster 1	135
Number Employed in Household	Demographic	Cluster 1	135
Annual Recurring Monthly Household Income Last Year	Financial	Cluster 1	135
Total Household Size	Lifestyle	Cluster 1	135
Education Attainment for 2	Demographic	Cluster 1	135
Education Attainment for 3	Demographic	Cluster 1	135
Working Months Last Year for 3	Demographic	Cluster 1	135

(*continued on next page*)

Table A2 (*continued*)

Feature	Category	Cluster	Cluster Size
Education Attainment for 4	Demographic	Cluster 1	135
Working Months Last Year for 4	Demographic	Cluster 1	135
Annual Household Income	Financial	Cluster 1	135
Average Household Monthly Savings Last Year	Financial	Cluster 1	135
Annual Household Clothing Expenditure	Lifestyle	Cluster 1	135
Annual Household Beauty Expenditure	Lifestyle	Cluster 1	135
Annual Household Private Insurance Expenditure	Financial	Cluster 1	135
Annual Household Health Insurance Expenditure	Financial	Cluster 1	135
Household Cars	Financial	Cluster 1	135
Household Owns Car	Financial	Cluster 1	135
Household Cars	Financial	Cluster 1	135
Monthly Household Income Last Year	Financial	Cluster 1	135
Annual Social Insurance Income	Financial	Cluster 2	2
Total Social Insurance Subsidy Received	Financial	Cluster 2	2
Annual Government Cash Subsidy Received	Financial	Cluster 3	2
Total Government Cash Handout Received	Financial	Cluster 3	2
Annual Net Income from Business	Financial	Cluster 4	4
Annual Business Revenue	Financial	Cluster 4	4
Valuation of Local District National Pension Service Account	Financial	Cluster 4	4
Annual Business Net Income	Financial	Cluster 4	4
Annual Real Estate Lease Income	Financial	Cluster 5	2
Total Real Estate Lease Income	Financial	Cluster 5	2
Annual Other Income	Financial	Cluster 6	3
Other Income	Financial	Cluster 6	3
Annual Other, Non-household Living, Schooling Aid Received	Financial	Cluster 6	3
Annual Capital Loss	Financial	Cluster 7	3
Annual Net Loss from Korean Stocks	Financial	Cluster 7	3
Annual Net Loss from Fixed Income, Funds, and Traded Securities	Financial	Cluster 7	3
Total Credit and Debit Card Overseas Expenditure Last Year	Financial	Cluster 8	2
Credit Card Overseas Expenditure Last Year	Financial	Cluster 8	2
Education Attainment for 5	Demographic	Cluster 9	2
Working Months Last Year for 5	Demographic	Cluster 9	2
Education Attainment for 6	Demographic	Cluster 10	2
Working Months Last Year for 6	Demographic	Cluster 10	2
Education Attainment for 7	Demographic	Cluster 11	4
Working Months Last Year for 7	Demographic	Cluster 11	4
Education Attainment for 8	Demographic	Cluster 11	4
Working Months Last Year for 8	Demographic	Cluster 11	4
Total Household Real Estate Wealth	Financial	Cluster 12	4
Current Home Valuation	Financial	Cluster 12	4
Household Non-living Housing Valuation	Financial	Cluster 12	4
Total Household Debt at End of Last Year	Financial	Cluster 12	4
Household Monthly Poverty Subsidy	Financial	Cluster 13	2
Household Annual Poverty Subsidy	Financial	Cluster 13	2
Household Housing Security Deposit	Financial	Cluster 14	2
Housing Security Deposit of Household Member	Financial	Cluster 14	2

Table A3
Neural Network A Architecture

Neural Network A				
Layer	Input Shape	Output Shape	# of Parameters	# of Operations
Encoder				
Linear	(14837, 306)	(14837, 128)	39,296	39,168
ReLU	(14837, 128)	(14837, 128)		
Linear	(14837, 128)	(14837, 64)	8256	8192
ReLU	(14837, 64)	(14837, 64)		
Linear	(14837, 64)	(14837, 32)	2080	2048
ReLU	(14837, 32)	(14837, 32)		
Linear	(14837, 32)	(14837, 16)	528	512
ReLU	(14837, 16)	(14837, 16)		
Decoder				
Linear	(14837, 16)	(14837, 32)	544	512
ReLU	(14837, 32)	(14837, 32)		
Linear	(14837, 32)	(14837, 64)	2112	2048
ReLU	(14837, 64)	(14837, 64)		
Linear	(14837, 64)	(14837, 128)	8320	8192
ReLU	(14837, 128)	(14837, 128)		
Linear	(14837, 128)	(14837, 306)	39,474	39,168
Sigmoid	(14837, 306)	(14837, 306)		

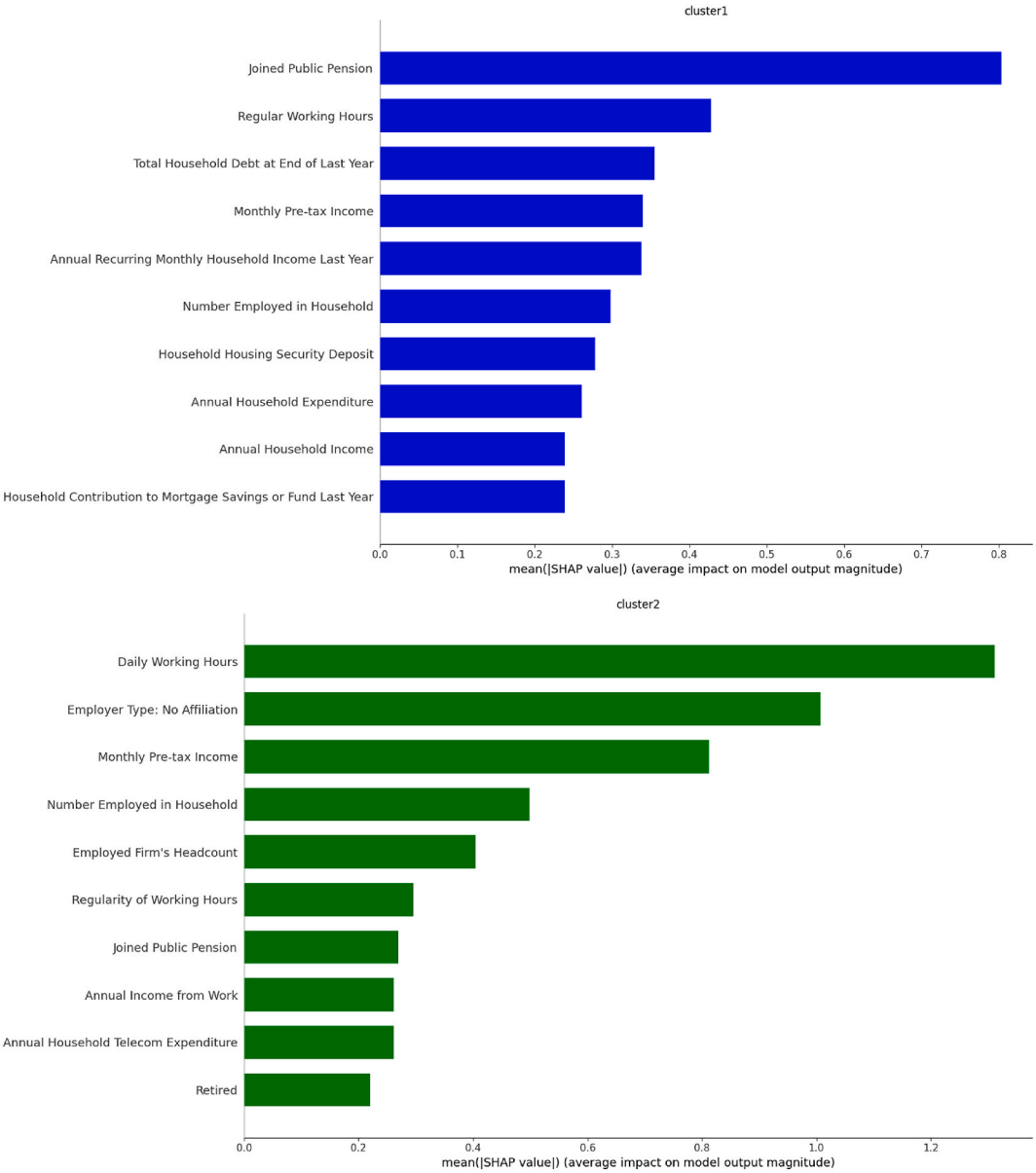

Fig. A1. Core Determinant Features of Clusters (From Top to Bottom: 1–10).

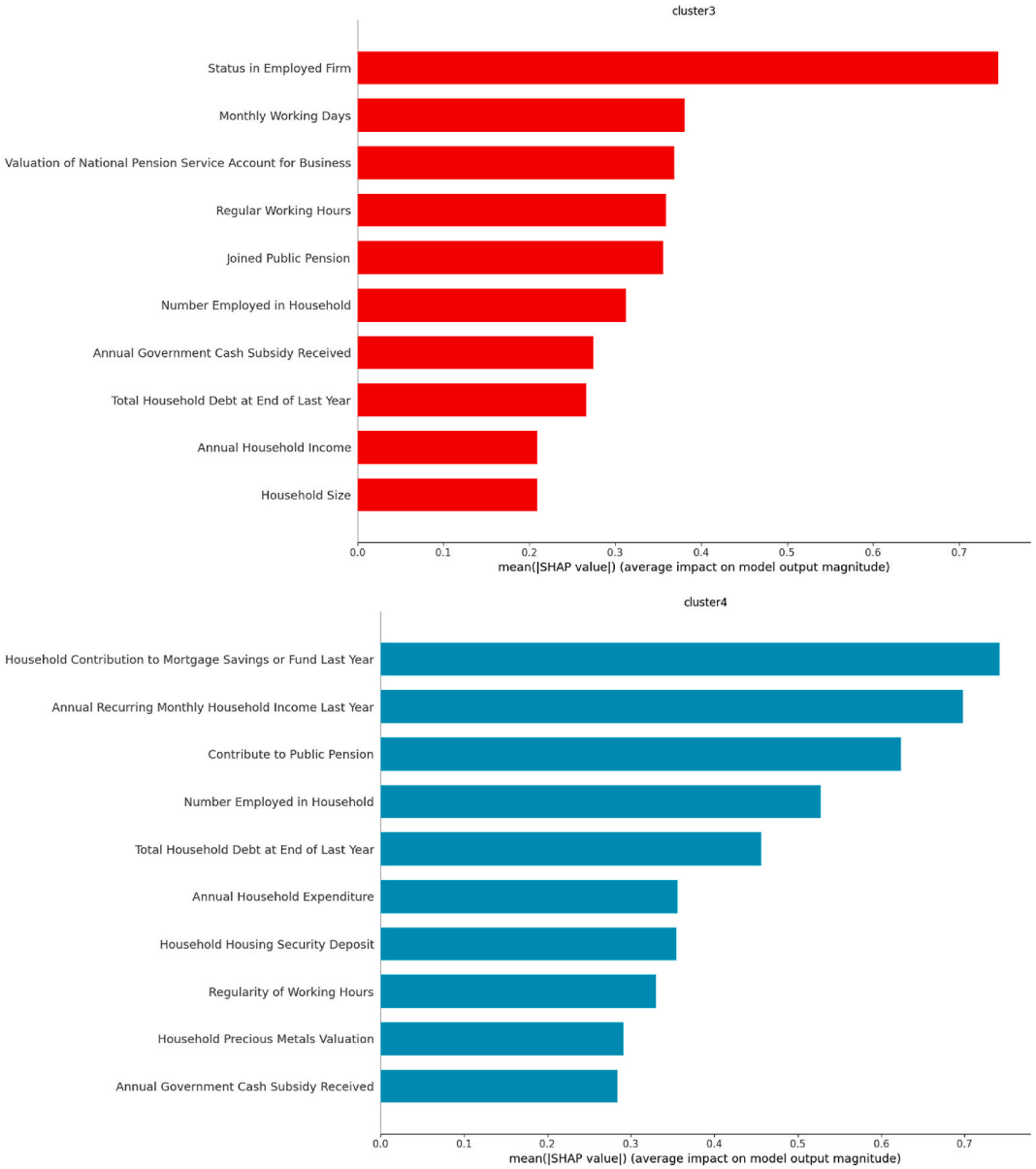

Fig. A1. (*continued*).

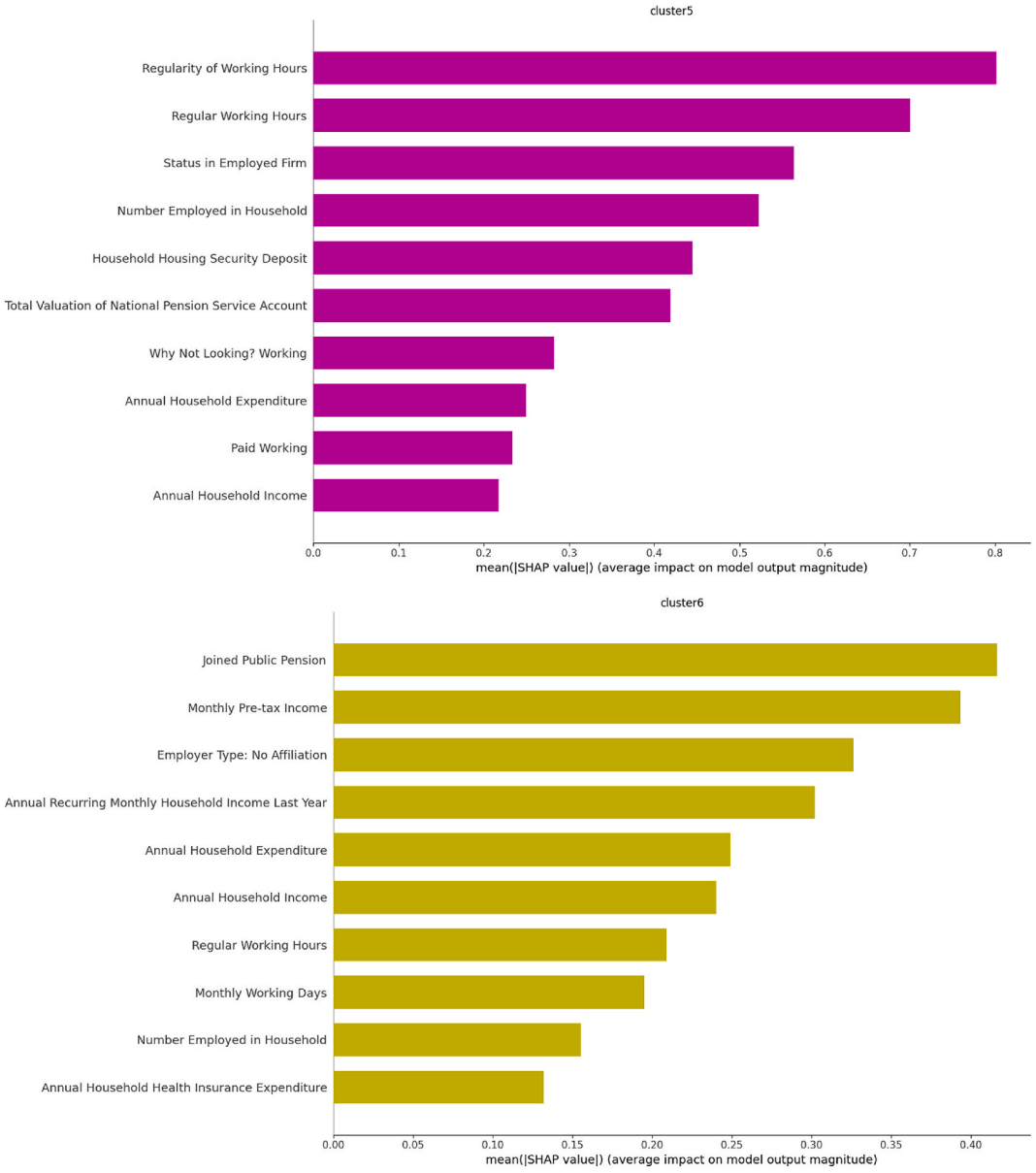

Fig. A1. (*continued*).

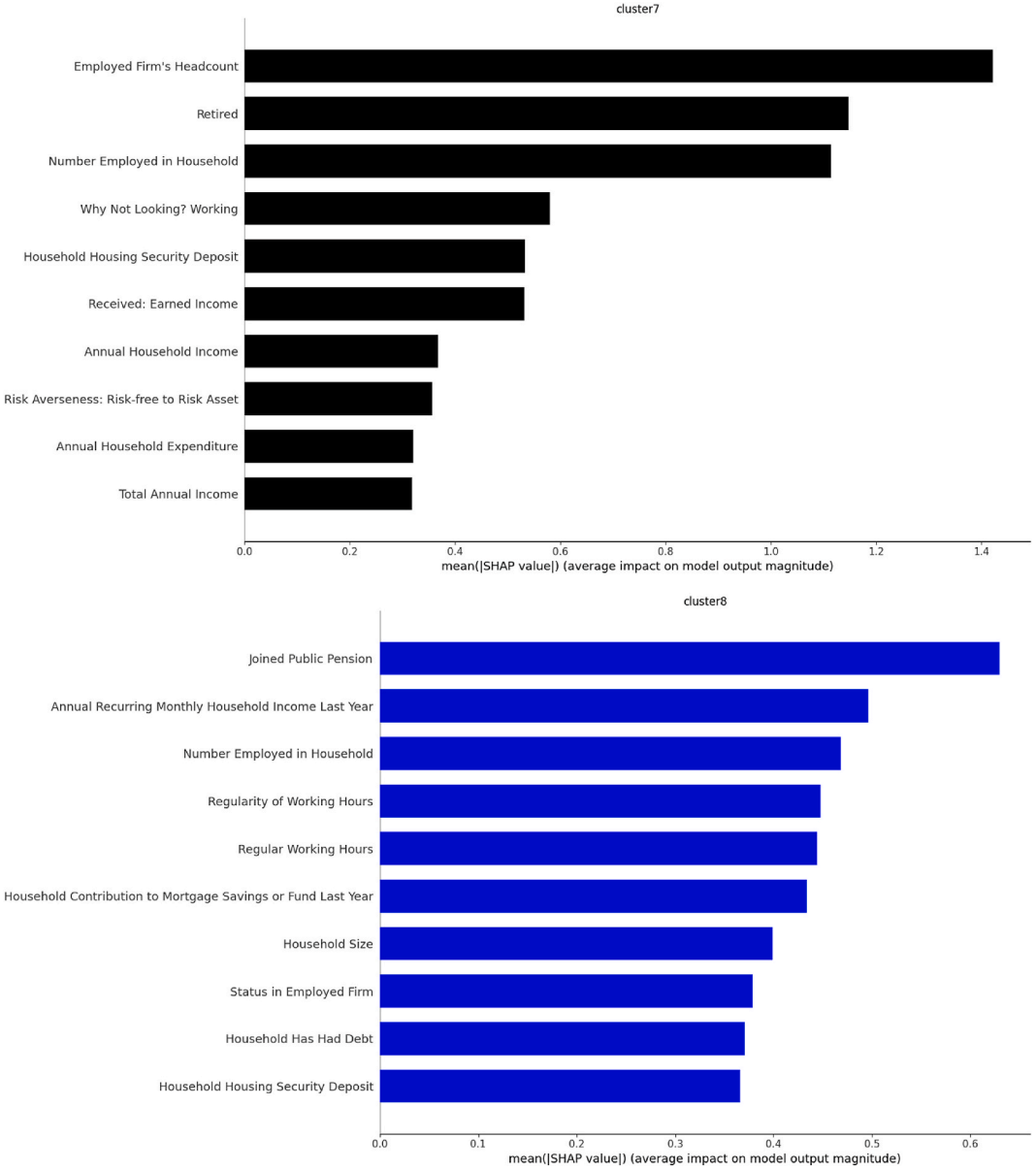

Fig. A1. (*continued*).

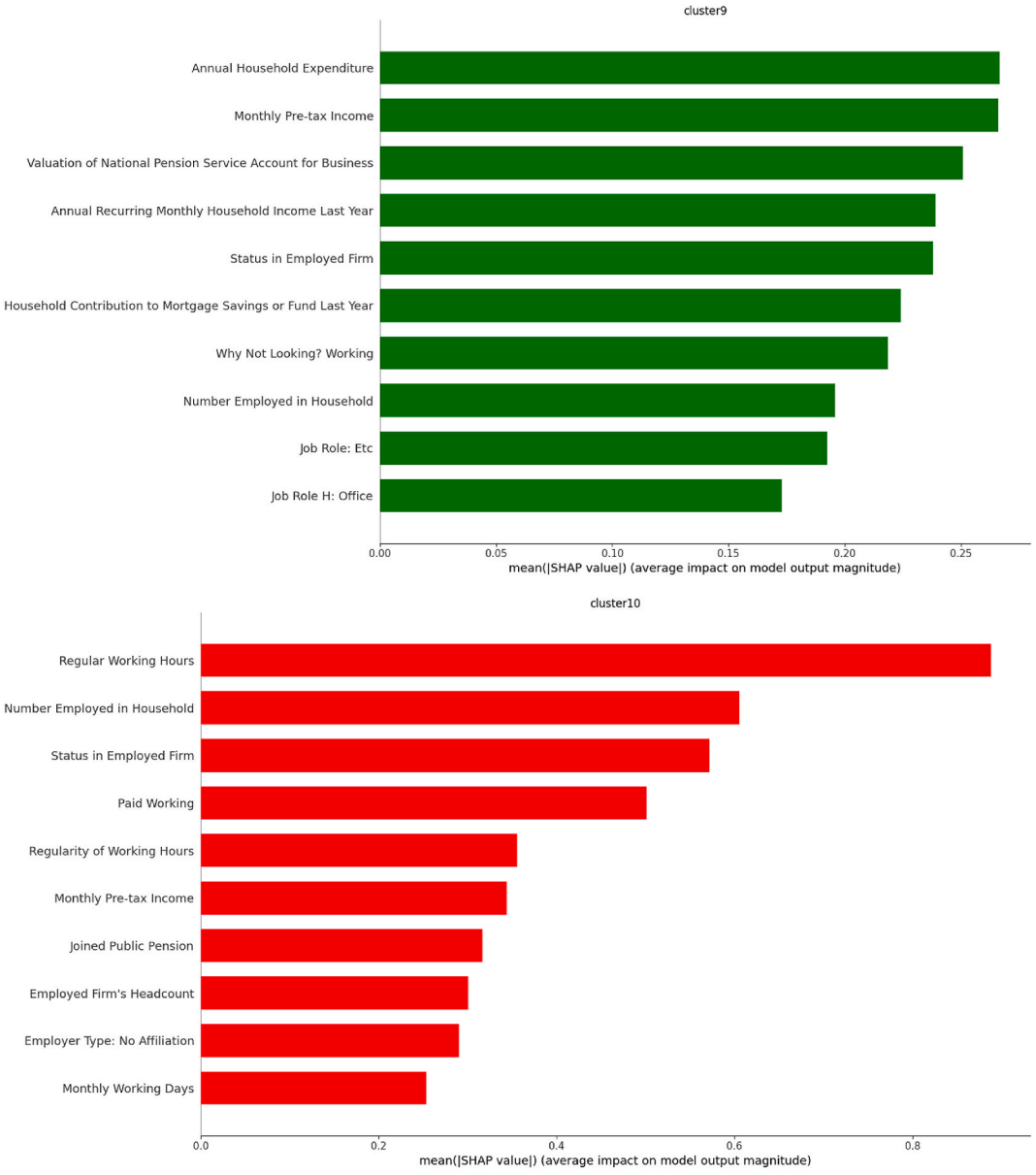

Fig. A1. (*continued*).

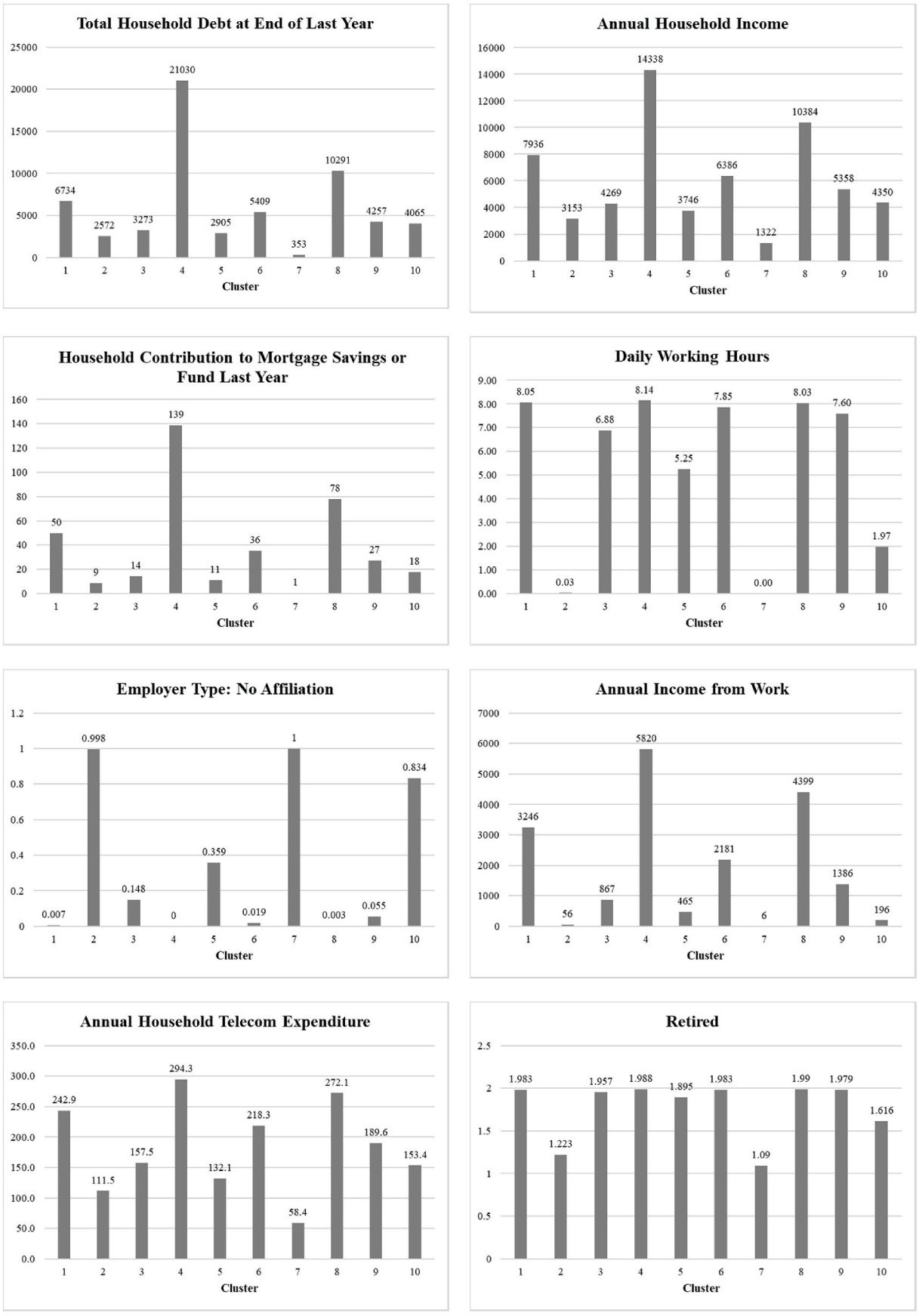

Fig. A2. The top ten most influential features of each cluster that did not make it in the aggregated list, but are included in the top ten for each cluster.

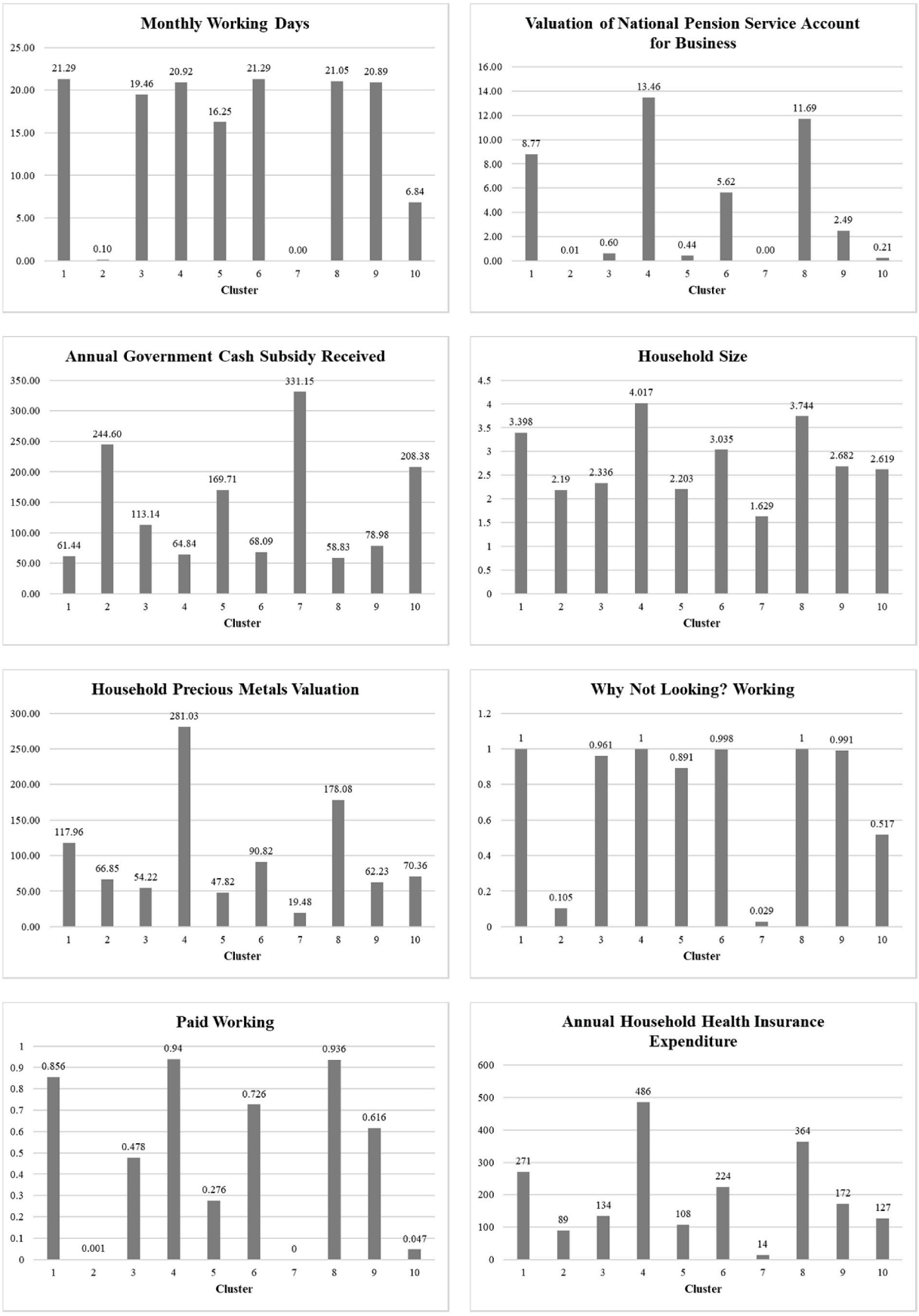

Fig. A2. (*continued*).

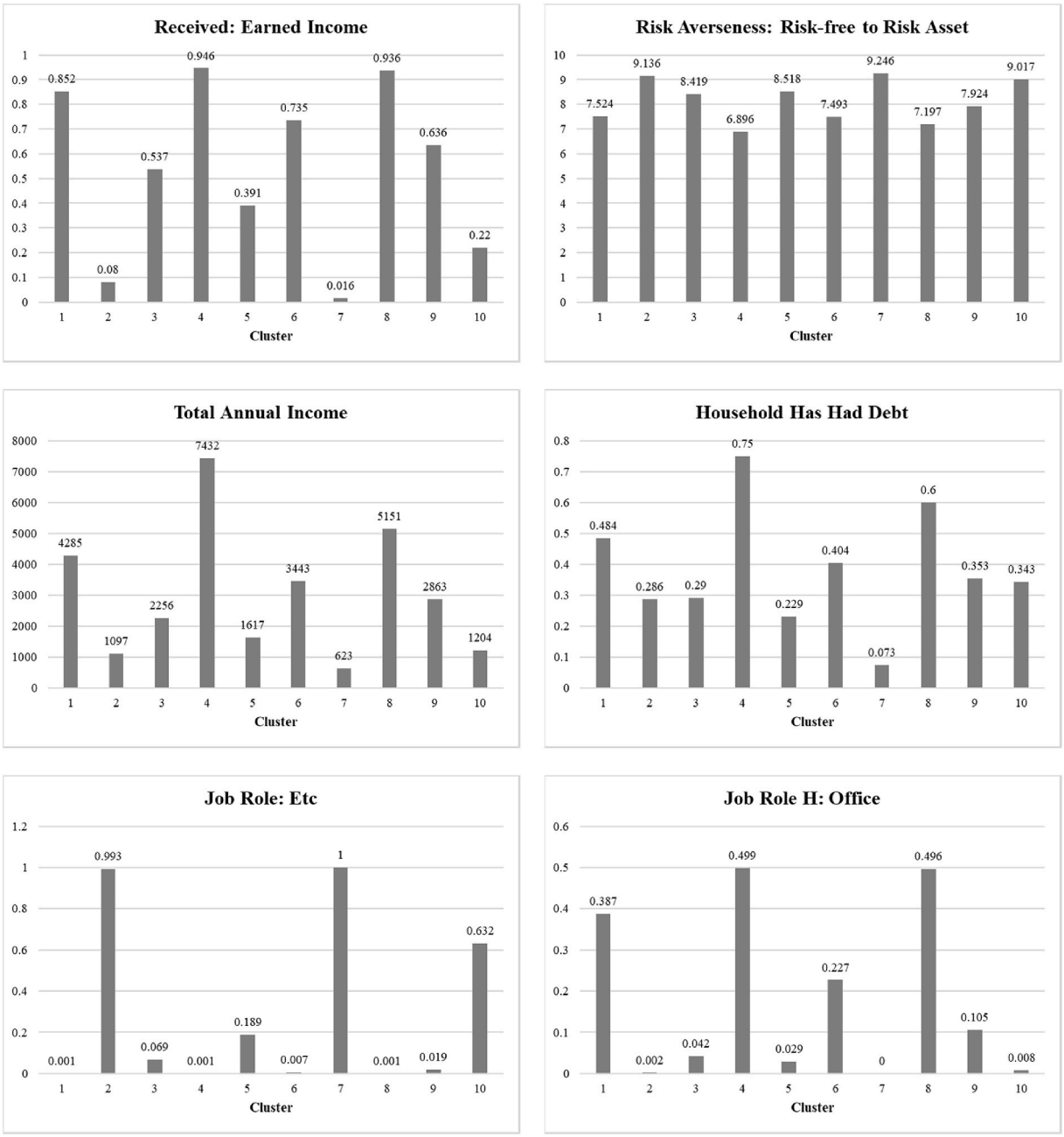

Fig. A2. (*continued*).

Table A4
Investment Universe

Full Name	Abbreviation	Issuer	Tracking Index	Index Maker (Parent Company)	Asset Class
iShares iBoxx USD High Yield Corporate Bond ETF	HYG	BlackRock	iBoxx USD Liquid High Yield Index	Standard & Poor's (S&P)	Bond
iShares iBoxx USD Investment Grade Corporate Bond ETF	LQD	BlackRock	iBoxx USD Liquid Investment Grade Index	Standard & Poor's (S&P)	Bond
SPDR Bloomberg 1-3 Month T-Bill ETF	BIL	State Street Global Advisors	Bloomberg 1-3 Month U.S. Treasury Bill Index	Bloomberg	Bond
Schwab Short-Term U.S. Treasury ETF	SCHO	Charles Schwab	Bloomberg U.S. Treasury 1-3 Year Index	Bloomberg	Bond
SPDR Portfolio Intermediate Term Treasury ETF	SPTI	State Street Global Advisors	Bloomberg U.S. 3-10 Year Treasury Bond Index	Bloomberg	Bond
SPDR Portfolio Long Term Treasury ETF	SPTL	State Street Global Advisors	Bloomberg U.S. Long Treasury Index	Bloomberg	Bond

(*continued on next page*)

Table A4 (*continued*)

Full Name	Abbreviation	Issuer	Tracking Index	Index Maker (Parent Company)	Asset Class
abrdn Physical Palladium Shares ETF	PALL	Abrdn Plc	LBMA Palladium PM Price (USD/Troy ounce)	London Bullion Market Association (LBMA)	Commodity
abrdn Physical Platinum Shares ETF	PPLT	Abrdn Plc	LBMA Platinum PM Price (USD/Troy ounce)	London Bullion Market Association (LBMA)	Commodity
iShares S&P GSCI Commodity-Indexed Trust	GSG	BlackRock	S&P GSCI Index	Standard & Poor's (S&P)	Commodity
iShares Gold Trust	IAU	BlackRock	LBMA Gold PM Price (USD/Troy ounce)	London Bullion Market Association (LBMA)	Commodity
iShares Silver Trust	SLV	BlackRock	LBMA Silver Price (USD/Troy ounce)	London Bullion Market Association (LBMA)	Commodity
Invesco DB Agriculture Fund	DBA	Invesco	DBIQ Diversified Agriculture Index	Deutsche Bank (DB)	Commodity
Invesco DB Base Metals Fund	DBB	Invesco	DBIQ Optimum Yield Industrial Metals Index	Deutsche Bank (DB)	Commodity
Invesco DB Energy Fund	DBE	Invesco	DBIQ Optimum Yield Energy Index	Deutsche Bank (DB)	Commodity
United States Natural Gas Fund LP	UNG	Marygold	NYMEX Natural Gas Front Month Price (USD/Million British thermal unit)	New York Mercantile Exchange (NYMEX)	Commodity
United States Oil Fund LP	USO	Marygold	NYMEX Crude Oil Front Month Price (USD/Barrel)	New York Mercantile Exchange (NYMEX)	Commodity
Invesco CurrencyShares Australian Dollar Trust	FXA	Invesco	USD/AUD Exchange Rate	Federal Reserve System (Fed)	Currency
Invesco CurrencyShares British Pound Sterling Trust	FXB	Invesco	USD/GBP Exchange Rate	Federal Reserve System (Fed)	Currency
Invesco CurrencyShares Canadian Dollar Trust	FXC	Invesco	USD/CAD Exchange Rate	Federal Reserve System (Fed)	Currency
Invesco CurrencyShares Euro Trust	FXE	Invesco	USD/EUR Exchange Rate	Federal Reserve System (Fed)	Currency
Invesco CurrencyShares Swiss Franc Trust	FXF	Invesco	USD/CHF Exchange Rate	Federal Reserve System (Fed)	Currency
Invesco Currencyshares Japanese Yen Trust	FXY	Invesco	USD/JPY Exchange Rate	Federal Reserve System (Fed)	Currency
Invesco DB U.S. Dollar Index Bullish Fund	UUP	Invesco	Deutsche Bank Long USD Currency Portfolio Index	Deutsche Bank (DB)	Currency
iShares MSCI Emerging Markets ETF	EEM	BlackRock	MSCI Emerging Markets Index	Morgan Stanley Capital International (MSCI)	Equity
iShares MSCI EAFE ETF	EFA	BlackRock	MSCI EAFE Index	Morgan Stanley Capital International (MSCI)	Equity
iShares Micro-Cap ETF	IWC	BlackRock	Russell Microcap Index	London Stock Exchange Group (LSEG)	Equity
iShares U.S. Real Estate ETF	IYR	BlackRock	Dow Jones U.S. Real Estate Capped Index	Standard & Poor's (S&P)	Equity
SPDR MSCI ACWI ex-U.S. ETF	CWI	State Street Global Advisors	MSCI ACWI ex U.S.A Index	Morgan Stanley Capital International (MSCI)	Equity
SPDR Dow Jones Industrial Average ETF Trust	DIA	State Street Global Advisors	Dow Jones Industrial Average	Standard & Poor's (S&P)	Equity
SPDR S&P 500 ETF Trust	SPY	State Street Global Advisors	S&P 500 Index	Standard & Poor's (S&P)	Equity
Materials Select Sector SPDR Fund	XLB	State Street Global Advisors	S&P Materials Select Sector Index	Standard & Poor's (S&P)	Equity
Energy Select Sector SPDR Fund	XLE	State Street Global Advisors	S&P Energy Select Sector Index	Standard & Poor's (S&P)	Equity
Financial Select Sector SPDR Fund	XLF	State Street Global Advisors	S&P Financial Select Sector Index	Standard & Poor's (S&P)	Equity
Industrial Select Sector SPDR Fund	XLI	State Street Global Advisors	S&P Industrial Select Sector Index	Standard & Poor's (S&P)	Equity
Technology Select Sector SPDR Fund	XLK	State Street Global Advisors	S&P Technology Select Sector Index	Standard & Poor's (S&P)	Equity
Consumer Staples Select Sector SPDR Fund	XLP	State Street Global Advisors	S&P Consumer Staples Select Sector Index	Standard & Poor's (S&P)	Equity
Utilities Select Sector SPDR Fund	XLU	State Street Global Advisors	S&P Utilities Select Sector Index	Standard & Poor's (S&P)	Equity
Health Care Select Sector SPDR Fund	XLV	State Street Global Advisors	S&P Health Care Select Sector Index	Standard & Poor's (S&P)	Equity
Consumer Discretionary Select Sector SPDR Fund	XLY	State Street Global Advisors	S&P Consumer Discretionary Select Sector Index	Standard & Poor's (S&P)	Equity
Vanguard FTSE All-World ex-US ETF	VEU	Vanguard	FTSE All-World ex US Index	London Stock Exchange Group (LSEG)	Equity
Vanguard Communication Services ETF	VOX	Vanguard	MSCI U.S. Investable Market Communication Services 25/50 Index	Morgan Stanley Capital International (MSCI)	Equity
Vanguard Small-Cap ETF	VB	Vanguard	CRSP U.S. Small Cap Index	Center for Research in Security Prices (CRSP)	Equity
Vanguard Small-Cap Growth ETF	VBK	Vanguard	CRSP U.S. Small Cap Growth Index	Center for Research in Security Prices (CRSP)	Equity
Vanguard Small-Cap Value ETF	VBR	Vanguard	CRSP U.S. Small Cap Value Index	Center for Research in Security Prices (CRSP)	Equity
Vanguard Mid-Cap ETF	VO	Vanguard	CRSP U.S. Mid Cap Index	Center for Research in Security Prices (CRSP)	Equity
Vanguard Mid-Cap Growth ETF	VOT	Vanguard	CRSP U.S. Mid Cap Growth Index	Center for Research in Security Prices (CRSP)	Equity

(*continued on next page*)

Table A4 (*continued*)

Full Name	Abbreviation	Issuer	Tracking Index	Index Maker (Parent Company)	Asset Class
Vanguard Mid-Cap Value ETF	VOE	Vanguard	CRSP U.S. Mid Cap Value Index	Center for Research in Security Prices (CRSP)	Equity
Vanguard Large-Cap ETF	VV	Vanguard	CRSP U.S. Large Cap Index	Center for Research in Security Prices (CRSP)	Equity
Vanguard Growth ETF	VUG	Vanguard	CRSP U.S. Large Cap Growth Index	Center for Research in Security Prices (CRSP)	Equity
Vanguard Value ETF	VTV	Vanguard	CRSP U.S. Large Cap Value Index	Center for Research in Security Prices (CRSP)	Equity

Table A5
Top 100 Models from an Evaluation of 1944 Candidate Performances

Rank	Stage	Dimension Reduction Method	Embedding Dimension	Number of Clusters	Clustering Methods	Distance Measure	Silhouette Coefficient	Calinski-Harabasz Score (Min-max Scaled, then Standardized)	Davies-Bouldin Index (1 - Original Value Transformed, then Standardized)	Final Score (Min-Max Scaled)
1	Autoencoder + Dimension Reduction	IsoMap	3	10	K-Means Algorithm	-	0.5243	4.4756	0.4191	1.0000
2	Autoencoder + Dimension Reduction	IsoMap	2	10	K-Means Algorithm	-	0.5246	4.4619	0.4188	0.9994
3	Autoencoder + Dimension Reduction	Principal Component Analysis	2	10	K-Means Algorithm	-	0.5246	4.4564	0.4188	0.9992
4	Autoencoder + Dimension Reduction	Principal Component Analysis	3	10	K-Means Algorithm	-	0.5236	4.4599	0.4173	0.9992
5	Autoencoder + Dimension Reduction	MDS	3	10	K-Means Algorithm	-	0.5184	4.3533	0.3950	0.9929
6	Autoencoder + Dimension Reduction	MDS	2	10	K-Means Algorithm	-	0.5205	4.3235	0.3958	0.9920
7	Autoencoder + Dimension Reduction	IsoMap	3	9	K-Means Algorithm	-	0.5238	4.1027	0.4068	0.9835
8	Autoencoder + Dimension Reduction	Principal Component Analysis	3	9	K-Means Algorithm	-	0.5229	4.0983	0.4053	0.9831
9	Autoencoder + Dimension Reduction	MDS	3	9	K-Means Algorithm	-	0.5184	3.9856	0.3879	0.9770
10	Autoencoder + Dimension Reduction	IsoMap	2	10	Agglomerative Clustering	Ward - Euclidean Distance	0.4984	4.0103	0.4099	0.9762
11	Autoencoder + Dimension Reduction	IsoMap	2	9	K-Means Algorithm	-	0.5261	3.8992	0.4201	0.9757
12	Autoencoder + Dimension Reduction	Principal Component Analysis	2	9	K-Means Algorithm	-	0.5259	3.8988	0.4191	0.9756
13	Autoencoder + Dimension Reduction	MDS	2	9	K-Means Algorithm	-	0.5222	3.8066	0.3979	0.9703
14	Autoencoder + Dimension Reduction	IsoMap	3	10	Agglomerative Clustering	Ward - Euclidean Distance	0.4750	3.8748	0.4075	0.9671
15	Autoencoder + Dimension Reduction	MDS	2	10	Agglomerative Clustering	Ward - Euclidean Distance	0.5028	3.8006	0.3800	0.9666
16	Autoencoder + Dimension Reduction	IsoMap	2	8	K-Means Algorithm	-	0.5304	3.6400	0.4135	0.9650
17	Autoencoder + Dimension Reduction	Principal Component Analysis	3	8	K-Means Algorithm	-	0.5288	3.6493	0.4091	0.9650

(*continued on next page*)

Table A5 (*continued*)

Rank	Stage	Dimension Reduction Method	Embedding Dimension	Number of Clusters	Clustering Methods	Distance Measure	Silhouette Coefficient	Calinski-Harabasz Score (Min-max Scaled, then Standardized)	Davies-Bouldin Index (1 - Original Value Transformed, then Standardized)	Final Score (Min-Max Scaled)
18	Autoencoder + Dimension Reduction	Principal Component Analysis	2	8	K-Means Algorithm	-	0.5309	3.6343	0.4138	0.9648
19	Autoencoder + Dimension Reduction	IsoMap	3	8	K-Means Algorithm	-	0.5313	3.6311	0.4147	0.9648
20	Autoencoder + Dimension Reduction	Principal Component Analysis	2	10	Agglomerative Clustering	Ward - Euclidean Distance	0.4835	3.7981	0.4024	0.9648
21	Autoencoder + Dimension Reduction	IsoMap	2	9	Agglomerative Clustering	Ward - Euclidean Distance	0.5031	3.6819	0.4076	0.9627
22	Autoencoder + Dimension Reduction	MDS	2	8	K-Means Algorithm	-	0.5271	3.5731	0.3986	0.9610
23	Autoencoder + Dimension Reduction	MDS	3	8	K-Means Algorithm	-	0.5256	3.5581	0.3954	0.9601
24	Autoencoder + Dimension Reduction	MDS	2	9	Agglomerative Clustering	Ward - Euclidean Distance	0.5092	3.6141	0.3893	0.9599
25	Autoencoder + Dimension Reduction	Principal Component Analysis	2	9	Agglomerative Clustering	Ward - Euclidean Distance	0.5013	3.5573	0.3958	0.9567
26	Autoencoder + Dimension Reduction	Principal Component Analysis	3	10	Agglomerative Clustering	Ward - Euclidean Distance	0.4717	3.5683	0.4175	0.9540
27	Autoencoder + Dimension Reduction	Principal Component Analysis	2	8	Agglomerative Clustering	Ward - Euclidean Distance	0.5137	3.3892	0.4136	0.9520
28	Autoencoder + Dimension Reduction	MDS	3	10	Agglomerative Clustering	Ward - Euclidean Distance	0.4679	3.4873	0.4070	0.9495
29	Autoencoder + Dimension Reduction	Principal Component Analysis	3	7	K-Means Algorithm	-	0.5341	3.2405	0.4180	0.9487
30	Autoencoder + Dimension Reduction	IsoMap	2	7	K-Means Algorithm	-	0.5354	3.2323	0.4210	0.9486
31	Autoencoder + Dimension Reduction	Principal Component Analysis	2	7	K-Means Algorithm	-	0.5352	3.2334	0.4202	0.9486
32	Autoencoder + Dimension Reduction	IsoMap	3	7	K-Means Algorithm	-	0.5313	3.2406	0.4045	0.9477
33	Autoencoder + Dimension Reduction	IsoMap	2	8	Agglomerative Clustering	Ward - Euclidean Distance	0.5121	3.2376	0.4273	0.9459
34	Autoencoder + Dimension Reduction	MDS	2	7	K-Means Algorithm	-	0.5298	3.1887	0.3987	0.9450
35	Autoencoder + Dimension Reduction	MDS	3	7	K-Means Algorithm	-	0.5278	3.1863	0.3953	0.9445
36	Autoencoder + Dimension Reduction	MDS	3	9	Agglomerative Clustering	Ward - Euclidean Distance	0.4804	3.2836	0.3960	0.9421
37	Dimension Reduction Only	IsoMap	2	3	K-Means Algorithm	-	0.6557	2.6175	0.4830	0.9417
38	Autoencoder + Dimension Reduction	IsoMap	3	9	Agglomerative Clustering	Ward - Euclidean Distance	0.4841	3.1893	0.4138	0.9394
39	Autoencoder + Dimension Reduction	Principal Component Analysis	3	9	Agglomerative Clustering	Ward - Euclidean Distance	0.4890	3.1491	0.4314	0.9391
40	Dimension Reduction Only	IsoMap	2	3	Agglomerative Clustering	Complete - Manhattan Distance	0.6541	2.5034	0.4968	0.9372

(*continued on next page*)

Table A5 (*continued*)

Rank	Stage	Dimension Reduction Method	Embedding Dimension	Number of Clusters	Clustering Methods	Distance Measure	Silhouette Coefficient	Calinski-Harabasz Score (Min-max Scaled, then Standardized)	Davies-Bouldin Index (1 - Original Value Transformed, then Standardized)	Final Score (Min-Max Scaled)
41	Dimension Reduction Only	IsoMap	2	3	Agglomerative Clustering	Ward - Euclidean Distance	0.6535	2.4858	0.4972	0.9364
42	Autoencoder + Dimension Reduction	IsoMap	2	10	Agglomerative Clustering	Average - Euclidean Distance	0.4807	3.1130	0.4272	0.9362
43	Autoencoder + Dimension Reduction	MDS	2	8	Agglomerative Clustering	Ward - Euclidean Distance	0.4913	3.0940	0.3935	0.9354
44	Autoencoder + Dimension Reduction	IsoMap	3	6	K-Means Algorithm	-	0.5404	2.8680	0.4265	0.9340
45	Dimension Reduction Only	IsoMap	2	3	Agglomerative Clustering	Average - Manhattan Distance	0.6512	2.4257	0.4984	0.9336
46	Autoencoder + Dimension Reduction	Principal Component Analysis	3	6	K-Means Algorithm	-	0.5392	2.8630	0.4240	0.9335
47	Autoencoder + Dimension Reduction	IsoMap	2	6	K-Means Algorithm	-	0.5392	2.8619	0.4242	0.9335
48	Autoencoder + Dimension Reduction	Principal Component Analysis	2	6	K-Means Algorithm	-	0.5391	2.8617	0.4239	0.9335
49	Dimension Reduction Only	IsoMap	2	3	Agglomerative Clustering	Average - Euclidean Distance	0.6507	2.4197	0.4984	0.9332
50	Autoencoder + Dimension Reduction	MDS	2	6	K-Means Algorithm	-	0.5373	2.8290	0.4193	0.9316
51	Autoencoder + Dimension Reduction	MDS	3	6	K-Means Algorithm	-	0.5369	2.8256	0.4184	0.9314
52	Autoencoder + Dimension Reduction	Principal Component Analysis	2	7	Agglomerative Clustering	Ward - Euclidean Distance	0.5160	2.8604	0.4313	0.9305
53	Autoencoder + Dimension Reduction	Principal Component Analysis	2	10	Agglomerative Clustering	Average - Euclidean Distance	0.4940	2.9008	0.4555	0.9302
54	Autoencoder + Dimension Reduction	MDS	3	8	Agglomerative Clustering	Ward - Euclidean Distance	0.4885	2.9829	0.3905	0.9302
55	Autoencoder + Dimension Reduction	Principal Component Analysis	3	8	Agglomerative Clustering	Ward - Euclidean Distance	0.5063	2.8393	0.4410	0.9287
56	Autoencoder + Dimension Reduction	IsoMap	3	9	Agglomerative Clustering	Average - Euclidean Distance	0.4963	2.8146	0.4592	0.9270
57	Autoencoder + Dimension Reduction	Principal Component Analysis	3	7	Agglomerative Clustering	Ward - Euclidean Distance	0.5165	2.7373	0.4419	0.9258
58	Autoencoder + Dimension Reduction	MDS	2	7	Agglomerative Clustering	Ward - Euclidean Distance	0.5110	2.7852	0.3905	0.9249
59	Autoencoder + Dimension Reduction	IsoMap	2	7	Agglomerative Clustering	Ward - Euclidean Distance	0.5132	2.7363	0.4257	0.9246
60	Autoencoder + Dimension Reduction	IsoMap	3	8	Agglomerative Clustering	Ward - Euclidean Distance	0.4876	2.8053	0.4189	0.9237
61	Autoencoder + Dimension Reduction	Principal Component Analysis	3	6	Agglomerative Clustering	Ward - Euclidean Distance	0.5253	2.6781	0.4160	0.9234
62	Autoencoder + Dimension Reduction	Principal Component Analysis	3	10	Agglomerative Clustering	Complete - Manhattan Distance	0.4803	2.7987	0.4309	0.9229
63	Autoencoder + Dimension Reduction	IsoMap	3	8	Agglomerative Clustering	Average - Euclidean Distance	0.4946	2.7112	0.4462	0.9218

(*continued on next page*)

Table A5 (*continued*)

Rank	Stage	Dimension Reduction Method	Embedding Dimension	Number of Clusters	Clustering Methods	Distance Measure	Silhouette Coefficient	Calinski-Harabasz Score (Min-max Scaled, then Standardized)	Davies-Bouldin Index (1 - Original Value Transformed, then Standardized)	Final Score (Min-Max Scaled)
64	Autoencoder + Dimension Reduction	Principal Component Analysis	3	10	Agglomerative Clustering	Complete - Euclidean Distance	0.4786	2.7234	0.4480	0.9202
65	Autoencoder + Dimension Reduction	MDS	3	7	Agglomerative Clustering	Ward - Euclidean Distance	0.5054	2.6720	0.3827	0.9189
66	Autoencoder + Dimension Reduction	IsoMap	2	9	Agglomerative Clustering	Complete - Manhattan Distance	0.4686	2.7482	0.4255	0.9189
67	Autoencoder + Dimension Reduction	IsoMap	3	5	K-Means Algorithm	-	0.5496	2.4817	0.4236	0.9187
68	Autoencoder + Dimension Reduction	IsoMap	2	5	K-Means Algorithm	-	0.5489	2.4823	0.4224	0.9186
69	Autoencoder + Dimension Reduction	Principal Component Analysis	2	5	K-Means Algorithm	-	0.5488	2.4818	0.4221	0.9185
70	Autoencoder + Dimension Reduction	Principal Component Analysis	3	5	K-Means Algorithm	-	0.5487	2.4820	0.4217	0.9185
71	Autoencoder + Dimension Reduction	MDS	3	6	Agglomerative Clustering	Ward - Euclidean Distance	0.5241	2.5754	0.4046	0.9184
72	Autoencoder + Dimension Reduction	Principal Component Analysis	3	9	Agglomerative Clustering	Average - Manhattan Distance	0.4998	2.6069	0.4342	0.9176
73	Autoencoder + Dimension Reduction	MDS	2	5	K-Means Algorithm	-	0.5476	2.4626	0.4192	0.9174
74	Autoencoder + Dimension Reduction	MDS	3	9	Agglomerative Clustering	Complete - Manhattan Distance	0.4786	2.7972	0.3036	0.9172
75	Autoencoder + Dimension Reduction	MDS	3	5	K-Means Algorithm	-	0.5471	2.4588	0.4179	0.9171
76	Autoencoder + Dimension Reduction	Principal Component Analysis	3	10	Agglomerative Clustering	Average - Manhattan Distance	0.4847	2.5992	0.4384	0.9154
77	Autoencoder + Dimension Reduction	IsoMap	2	10	Agglomerative Clustering	Complete - Euclidean Distance	0.4853	2.5915	0.4407	0.9152
78	Autoencoder + Dimension Reduction	IsoMap	3	9	Agglomerative Clustering	Complete - Manhattan Distance	0.4698	2.6456	0.4332	0.9150
79	Autoencoder + Dimension Reduction	IsoMap	2	6	Agglomerative Clustering	Ward - Euclidean Distance	0.5227	2.4634	0.4049	0.9134
80	Autoencoder + Dimension Reduction	MDS	2	6	Agglomerative Clustering	Ward - Euclidean Distance	0.5196	2.4382	0.4368	0.9133
81	Dimension Reduction Only	IsoMap	2	2	Agglomerative Clustering	Complete - Manhattan Distance	0.7364	1.4339	0.7290	0.9129
82	Dimension Reduction Only	IsoMap	2	2	Agglomerative Clustering	Ward - Euclidean Distance	0.7364	1.4339	0.7290	0.9129
83	Dimension Reduction Only	IsoMap	2	2	K-Means Algorithm	-	0.7361	1.4356	0.7259	0.9128
84	Dimension Reduction Only	IsoMap	2	2	Agglomerative Clustering	Average - Manhattan Distance	0.7363	1.4307	0.7295	0.9128
85	Dimension Reduction Only	IsoMap	2	2	Agglomerative Clustering	Average - Euclidean Distance	0.7362	1.4280	0.7297	0.9127
86	Dimension Reduction Only	IsoMap	2	2	Agglomerative Clustering	Complete - Euclidean Distance	0.7345	1.4203	0.7202	0.9117

(*continued on next page*)

Table A5 (*continued*)

Rank	Stage	Dimension Reduction Method	Embedding Dimension	Number of Clusters	Clustering Methods	Distance Measure	Silhouette Coefficient	Calinski-Harabasz Score (Min-max Scaled, then Standardized)	Davies-Bouldin Index (1 - Original Value Transformed, then Standardized)	Final Score (Min-Max Scaled)
87	Dimension Reduction Only	IsoMap	2	3	Agglomerative Clustering	Complete - Euclidean Distance	0.6059	2.1673	0.3895	0.9116
88	Autoencoder + Dimension Reduction	IsoMap	3	10	Agglomerative Clustering	Average - Euclidean Distance	0.4916	2.4413	0.4575	0.9104
89	Autoencoder + Dimension Reduction	MDS	3	8	Agglomerative Clustering	Complete - Manhattan Distance	0.4770	2.5167	0.3939	0.9089
90	Autoencoder + Dimension Reduction	Principal Component Analysis	2	9	Agglomerative Clustering	Average - Euclidean Distance	0.5074	2.3018	0.4836	0.9077
91	Autoencoder + Dimension Reduction	IsoMap	3	7	Agglomerative Clustering	Complete - Euclidean Distance	0.4908	2.4086	0.4073	0.9067
92	Autoencoder + Dimension Reduction	IsoMap	2	10	Agglomerative Clustering	Average - Manhattan Distance	0.4827	2.3883	0.4482	0.9065
93	Autoencoder + Dimension Reduction	IsoMap	3	7	Agglomerative Clustering	Complete - Manhattan Distance	0.4929	2.3582	0.4407	0.9063
94	Dimension Reduction Only	IsoMap	2	4	K-Means Algorithm	-	0.5491	2.3809	0.2271	0.9060
95	Autoencoder + Dimension Reduction	Principal Component Analysis	3	9	Agglomerative Clustering	Average - Euclidean Distance	0.4820	2.3468	0.4689	0.9055
96	Autoencoder + Dimension Reduction	IsoMap	2	10	Agglomerative Clustering	Complete - Manhattan Distance	0.4662	2.4144	0.4502	0.9054
97	Autoencoder + Dimension Reduction	IsoMap	2	4	K-Means Algorithm	-	0.5625	2.0906	0.4365	0.9044
98	Autoencoder + Dimension Reduction	Principal Component Analysis	2	4	K-Means Algorithm	-	0.5624	2.0903	0.4362	0.9043
99	Autoencoder + Dimension Reduction	Principal Component Analysis	3	4	K-Means Algorithm	-	0.5622	2.0901	0.4359	0.9043
100	Autoencoder + Dimension Reduction	IsoMap	3	4	K-Means Algorithm	-	0.5617	2.0871	0.4352	0.9040

References

Abbasimehr, H., Bahrini, A., 2022. An analytical framework based on the recency, frequency, and monetary model and time series clustering techniques for dynamic segmentation. Expert Syst. Appl. 192, 116373.

Aitken, M., Ng, M., Horsfall, D., Coopamootoo, K., Van Moorsel, A., Elliott, K., 2021b. Pursuit of socially-minded data-intensive innovation in banking: a focus group study of public expectations of digital innovation in banking. Technol. Soc. 66, 101666 https://doi.org/10.1016/j.techsoc.2021.101666.

Allen, D.M., 1974. The relationship between variable selection and data agumentation and a method for prediction. Technometrics 16 (1), 125–127.

Alsayat, A., El-Sayed, H., 2016. Social media analysis using optimized K-means clustering. In: 2016 Ieee 14th International Conference on Software Engineering Research, Management and Applications (Sera), pp. 61–66. Ieee.

Baarsch, J., Celebi, M.E., 2012. Investigation of internal validity measures for K-means clustering. In: Proceedings of the International Multiconference of Engineers and Computer Scientists, 1, pp. 14–16. SN.

Ballard, D.H., 1987. Modular learning in neural networks. Proceedings of the Sixth National Conference on Artificial Intelligence 1, 279–284.

Barber, B.M., Odean, T., 2000. Trading is hazardous to your wealth: the common stock investment performance of individual investors. J. Finance 55 (2), 773–806. https://doi.org/10.1111/0022-1082.00226.

Barber, B.M., Odean, T., 2001. Boys will Be boys: gender, overconfidence, and common stock investment. Q. J. Econ. 116 (1), 261–292.

Bhatia, A., Chandani, A., Divekar, R., Mehta, M., Vijay, N., 2022. Digital innovation in wealth management landscape: the moderating role of robo advisors in behavioural biases and investment decision-making. Int. J. Innovat. Sci. 14 (3/4), 693–712.

Brahmana, R.S., Mohammed, F.A., Chairuang, K., 2020. Customer segmentation based on rfm model using K-means, K-medoids, and dbscan methods. Lontar komput. J. Ilm. Teknol. Inf 11 (1), 32.

Brown, S., Ghosh, P., Gray, D., Pareek, B., Roberts, J., 2021. Saving behaviour and health: a high-dimensional bayesian analysis of British panel data. Eur. J. Finance 27 (16), 1581–1603.

Bussmann, N., Giudici, P., Marinelli, D., Papenbrock, J., 2020. Explainable AI in fintech risk management. Frontiers in Artificial Intelligence 3, 26.

Caffo, B.S., D'Asaro, F.A., Garcez, A., Raffinetti, E., 2022. Explainable artificial intelligence models and methods in finance and Healthcare. Frontiers in Artificial Intelligence 5, 970246.

Camilleri, M.A., 2020. The use of data-driven technologies for customer-centric marketing. Int. J. Biomed. Data Min 1 (1), 50–63.

Demšar, J., 2006. Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7, 1–30.

Dong, H., Chen, X., Dusmanu, M., Larsson, V., Pollefeys, M., Stachniss, C., 2023. Learning-based dimensionality reduction for computing compact and effective local feature descriptors. In: 2023 Ieee International Conference on Robotics and Automation (Icra), pp. 6189–6195. Ieee.

Donoho, D.L., Grimes, C., 2003. Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. Proc. Natl. Acad. Sci. USA 100 (10), 5591–5596.

Eslami, M., Mahmoodian, V., Dayarian, I., Charkhgard, H., Tu, Y., 2020. Query batching optimization in database systems. Comput. Oper. Res. 121, 104983 https://doi.org/10.1016/j.cor.2020.104983.

Gennaioli, N., Shleifer, A., Vishny, R., 2015. Money doctors. J. Finance 70 (1), 91–114.

Ghodsi, A., 2006. Dimensionality reduction a short tutorial. Department of Statistics and Actuarial Science, Univ. Of Waterloo 37 (38), 2006. Ontario, Canada.

Grable, J.E., Lytton, R.H., 1999. Financial risk tolerance revisited: the development of a risk assessment instrument. Financ. Serv. Rev. 8 (3), 163–181.
Guiso, L., Sapienza, P., Zingales, L., 2008. Trusting the stock market. J. Finance 63 (6), 2557–2600.
Hague, P.N., Hague, N., Morgan, C.A., 2013. Market Research in Practice: How to Get Greater Insight from Your Market. Kogan Page Publishers.
Hartigan, J.A., Wong, M.A., 1979. Algorithm as 136: a K-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics) 28 (1), 100–108.
Hasan, B.M.S., Abdulazeez, A.M., 2021. A review of principal component analysis algorithm for dimensionality reduction. Journal of Soft Computing and Data Mining 2 (1), 20–30.
He, Y.L., Ou, G.L., Fournier-Viger, P., Huang, J.Z., Suganthan, P.N., 2022. A novel dependency-oriented mixed-attribute data classification method. Expert Syst. Appl. 199, 116782.
Hung, P.D., Lien, N.T.T., Ngoc, N.D., 2019. Customer segmentation using hierarchical agglomerative clustering. In: Proceedings of the 2nd International Conference on Information Science and Systems, pp. 33–37.
Hwang, Y., Lee, Y., Fabozzi, F.J., 2023. Identifying household finance heterogeneity via deep clustering. Ann. Oper. Res. 325 (2), 1255–1289.
Hwang, H., Whang, S.E., 2023. Xclusters: explainability-first clustering. In: Proceedings of the Aaai Conference on Artificial Intelligence, 37, pp. 7962–7970, 7.
Jansen, B.J., Salminen, J.O., Jung, S.G., 2020. Data-driven personas for enhanced user understanding: combining empathy with rationality for better insights to analytics. Data and Information Management 4 (1), 1–17.
Kai, S., Jin, W., 2022. Semiconductor chip's quality analysis based on its high dimensional test data. Ann. Oper. Res. 1–12.
Kim, S., Joh, H., Choi, S., Ryoo, I., 2015. Energy Efficient Mac Scheme for Wireless Sensor Networks with High-Dimensional Data Aggregate. Mathematical Problems in Engineering, 2015.
Kovács, T., Kő, A., Asemi, A., 2021. Exploration of the investment patterns of potential retail banking customers using two-stage cluster analysis. Journal of Big Data 8 (1). https://doi.org/10.1186/S40537-021-00529-4.
Kruskal, J.B., 1964. Nonmetric multidimensional scaling: a numerical method. Psychometrika 29 (2), 115–129.
Kuo, R.J., Ho, L.M., Hu, C.M., 2002. Integration of self-organizing feature map and K-means algorithm for market segmentation. Comput. Oper. Res. 29 (11), 1475–1493.
Kuroki, M., Yamasaki, T., 2023. Bsed: Baseline Shapley-Based Explainable Detector. Arxiv Preprint arxiv:2308.07490.
Laberge, G., Pequignot, Y., 2022. Understanding Interventional Treeshap: How and Why it Works. Arxiv Preprint arxiv:2209.15123.
LI, X., Zhang, T., Zhao, X., Yi, Z., 2020. Guided autoencoder for dimensionality reduction of pedestrian features. Appl. Intell. 50, 4557–4567.
LI, Y., Chu, X., Tian, D., Feng, J., Mu, W., 2021a. Customer segmentation using K-means clustering and the adaptive particle swarm optimization algorithm. Appl. Soft Comput. 113, 107924 https://doi.org/10.1016/j.asoc.2021.107924.
LI, Y., Hu, P., Liu, Z., Peng, D., Zhou, J.T., Peng, X., 2021b. Contrastive clustering. In: Proceedings of the Aaai Conference on Artificial Intelligence, 35, pp. 8547–8555, 10.
Li, Y., Yang, M., Peng, D., Li, T., Huang, J., Peng, X., 2022. Twin contrastive learning for online clustering. Int. J. Comput. Vis. 130 (9), 2205–2221.
Liao, S., Chen, Y., Lin, Y., 2011. Mining customer knowledge to implement online shopping and home delivery for hypermarkets. Expert Syst. Appl. 38 (4), 3982–3991. https://doi.org/10.1016/j.eswa.2010.09.059.
Liaw, R., Liang, E., Nishihara, R., Moritz, P., Gonzalez, J.E., Stoica, I., 2018. Tune: A Research Platform for Distributed Model Selection and Training. Arxiv Preprint arxiv:1807.05118.
Lundberg, S.M., Lee, S.I., 2017a. A unified approach to interpreting model predictions. Adv. Neural Inf. Process. Syst. 30.
Lundberg, S.M., Lee, S.I., 2017b. Consistent Feature Attribution for Tree Ensembles. Arxiv Preprint arxiv:1706.06060.
Lundberg, S.M., Erion, G., Chen, H., Degrave, A., Prutkin, J.M., Nair, B., et al., 2020. From local explanations to global understanding with explainable AI for trees. Nat. Mach. Intell. 2 (1), 56–67.
Markowitz, H., 1952. The utility of wealth. J. Polit. Econ. 60 (2), 151–158.
McConville, R., Santos-Rodriguez, R., Piechocki, R.J., Craddock, I., 2021. N2d:(Not too) deep clustering via clustering the local manifold of an autoencoded embedding. In: 2020 25th International Conference on Pattern Recognition (Icpr), pp. 5145–5152. Ieee.
McInnes, L., Healy, J., Melville, J., 2018. Umap: Uniform Manifold Approximation and Projection for Dimension Reduction. Arxiv Preprint arxiv:1802.03426.
Micheaux, A., Bosio, B., 2019. Customer journey mapping as a new way to teach data-driven marketing as a service. J. Market. Educ. 41 (2), 127–140.
Nagaraj, P., Birunda, S.S., Venkatesh, R., Muneeswaran, V., Narayanan, S.K., Shree, U.D., Sunethra, B., 2022. Automatic and adaptive segmentation of customer in R framework using K-means clustering technique. In: 2022 International Conference on Computer Communication and Informatics (Iccci), pp. 1–5. Ieee.
Nambisan, S., Wright, M., Feldman, M., 2019. The digital transformation of innovation and entrepreneurship: progress, challenges and key themes. Res. Pol. 48 (8), 103773.
Narayana, v.L., Sirisha, S., Divya, G., Pooja, N.L.S., Nouf, S.A., 2022. Mall customer segmentation using machine learning. In: 2022 International Conference on Electronics and Renewable Systems (Icears), pp. 1280–1288. Ieee.
Nguyen, S., 2021. Deep customer segmentation with applications to a Vietnamese supermarkets' data. Soft Comput. 25 (12), 7785–7793. https://doi.org/10.1007/S00500-021-05796-0.
Park, J.K., Park, S.K., Lee, B.G., 2021. Priority of challenges for activation of mydata business: K-mydata case. Ksii Transactions on Internet & Information Systems 15 (10).
Peng, X., Li, Y., Tsang, I.W., Zhu, H., Lv, J., Zhou, J.T., 2022. Xai beyond classification: interpre_ neural clustering. J. Mach. Learn. Res. 23 (1), 227–254.
Prasch, R., Warin, T., 2016. Systemic risk and financial regulations: a theoretical perspective. J. Bank. Regul. 17 (3), 188–199.
Roweis, S.T., Saul, L.K., 2000. Nonlinear dimensionality reduction by locally linear embedding. Science 290 (5500), 2323–2326.
Saura, J.R., Ribeiro-Soriano, D., Palacios-Marqués, D., 2021. From user-generated data to data-driven innovation: a research agenda to understand user privacy in digital markets. Int. J. Inf. Manag. 60, 102331.
Seret, A., Bejinaru, A., Baesens, B., 2015. Domain knowledge based segmentation of online banking customers. Intell. Data Anal. 19 (S1), S163–S184. https://doi.org/10.3233/Ida-150776.
Shapley, L.S., 1953. Stochastic games. Proc. Natl. Acad. Sci. USA 39 (10), 1095–1100.
Shutaywi, M., Kachouie, N.N., 2021. Silhouette analysis for performance evaluation in machine learning with applications to clustering. Entropy 23 (6), 759.
Son, Y., Kwon, H.E., Tayi, G.K., Oh, W., 2019. Impact of customers' digital banking adoption on hidden defection: a combined analytical–empirical approach. J. Oper. Manag. 66 (4), 418–440. https://doi.org/10.1002/joom.1066.
Statman, M., 2004. The diversification puzzle. Financ. Anal. J. 60 (4), 44–53.
Tabianan, K., Velu, S., Ravi, v., 2022. K-means clustering approach for intelligent customer segmentation using customer purchase behavior data. Sustainability 14 (12), 7243.
Temelkov, Z., 2018. Fintech firms opportunity or threat for banks? Int. J. Inf. Bus. Manag. 10 (1), 137–143.
Tenenbaum, J.B., de Silva, V., Langford, J.C., 2000. A global geometric framework for nonlinear dimensionality reduction. Science 290 (5500), 2319–2323.
Tobin, J., 1958. Liquidity preference as behavior towards risk. Rev. Econ. Stud. 25 (2), 65–86.
Torgerson, W.S., 1952. Multidimensional scaling: I. Theory and method. Psychometrika 17 (4), 401–419.
Van Der Maaten, L., Hinton, G., 2008. Visualizing data using T-sne. J. Mach. Learn. Res. 9 (11).
Van Der Maaten, L., Postma, E., Van Den Herik, J., 2009. Dimensionality reduction: a comparative. J. Mach. Learn. Res. 10 (66–71).
Vial, G., 2019. Understanding digital transformation: a review and a research agenda. J. Strat. Inf. Syst. 28 (2), 118–144.
Wang, F., Sun, J., 2015. Survey on distance metric learning and dimensionality reduction in data mining. Data Min. Knowl. Discov. 29 (2), 534–564.
Williams, C., 2000. On a connection between kernel PCA and metric multidimensional scaling. Adv. Neural Inf. Process. Syst. 13.
Wu, S., Zhou, X., Cao, G., Shi, N., Liu, Z., 2018. High-dimensional data-driven optimal design for hot strip rolling of microalloyed steel. Steel Res. Int. 89 (7), 1800015.
Yanik, S., Elmorsy, A., 2019. Som approach for clustering customers using credit card transactions. International Journal of Intelligent Computing and Cybernetics 12 (3), 372–388. https://doi.org/10.1108/Ijicc-11-2018-0157.
Zhang, Z., Chow, T.W., Zhao, M., 2012. M-isomap: orthogonal constrained marginal Isomap for nonlinear dimensionality reduction. IEEE Trans. Cybern. 43 (1), 180–191.
Zhang, T., Moro, S., Ramos, R.F., 2022. A data-driven approach to improve customer churn prediction based on telecom customer segmentation. Future Internet 14 (3), 94.

Further reading

Patterson, J., Gibson, A., 2017. Deep Learning: A Practitioner's Approach. O'Reilly Media, Inc.".
Aggarwal, C.C., 2018. Neural networks and deep learning. Spring 10 (978), 3.
Sze, V., Chen, Y.H., Yang, T.J., Emer, J.S., 2017. Efficient processing of deep neural networks: a tutorial and survey. Proc. IEEE 105 (12), 2295–2329.
Ivezić, ž., Connolly, A.J., Vanderplas, J.T., Gray, A., 2020. Statistics, Data Mining, and Machine Learning in Astronomy: A Practical Python Guide for the Analysis of Survey Data. Princeton University Press.
Jolliffe, I.T., 2002. Principal Component Analysis for Special Types of Data. Springer, New York, pp. 338–372.
Borg, I., Groenen, P.J., 2005. Modern Multidimensional Scaling: Theory and Applications. Springer Science & Business Media.