Erasmus School of Economics

Thesis Business Analytics & Quantitative Marketing

A Review of Dynamic Customer Segmentation Methods in E-commerce Business Contexts

Author:

Veerle van Oostenbruggen

Student ID number: 456447

Supervisor:

dr. C. Cavicchia

Second Assessor:

prof. dr. P.J.F. Groenen

Abstract:

This paper aims to explore relevant dynamic customer segmentation methods across E-commerce business contexts. Website traffic data is collected from four E-commerce companies and segments are extracted from an online tool for customer segmentation. This paper examines the tool's validity, as well as whether its resulting segments can be outperformed by other dynamic customer segmentation methods. The following components for dynamic customer segmentation methods are selected: existing distance measurements, clustering methods, and frequently used numbers of segments. This paper sets itself apart from existing literature by comparing three selected features for customer distinction and applying the methods to various business contexts. An interesting finding of this thesis is that the Silhouette Index (SI) appears to be sensitive to the distance measurement applied in its calculation, whereas the Davies-Bouldin Index appears to be less sensitive to this matter. In order for the SI to provide meaningful results, the Dynamic Time Warping distance is used for the SI calculation. The resulting clusters seem highly influenced by the distance metrics used, where some distance metrics are not accurate in defining similar customer behaviours. Moreover, the number of frequent buyers within a dataset seems to affect which segmentation method and which cluster size is preferred. The Customer Lifetime Value is determined as the preferred feature selection for dynamic customer segmentation. Concluding, the validity of the online tool appears to be weak and can likely be outperformed by executing one of the methods provided in this thesis.

June 28, 2022

Table of Contents

1 Introduction

Online customer loyalty is challenging, a competitor's website is just one click away. In order to remain profitable, it is essential for organisations to build and maintain engagement with new, but mainly existing customers (Guerola-Navarro et al., 2020). Customers who have previously purchased on this website are likely to make more purchases and to spend a higher value in future orders compared to one-time-buyers. Moreover, it is five times more expensive to gain new customers than maintaining existing ones (Gupta, Lehmann, & Stuart, 2004). In addition, it is stated that an increase in customer retention rates by just 5% results in a profit increase of 25% to 95% (Gallo, 2014). Nonetheless, research shows that almost half of retail businesses continue to aim for new customer acquisition, while only 18% have developed strategies to boost customer retention.

Therefore, customer relationship management (CRM) must be seen as indispensable and includes all activities that improve relationships with customers. CRM assists in customer retention and loyalty, as well as the acquisition of new customers. However, a one-size-fits-all CRM strategy for all customers does not work, considering that customers who visit a website once a week should be treated differently compared to customers who only visit once a year (Zhou, Wei, & Xu, 2021). Distinctive types of customers must be classified to ensure that appropriate strategies can be assigned to every category. Often, this results in the primary focus being on customers who provide the highest profitability while letting non-profitable customers go.

Customer segmentation is an approach for grouping customers with similar customer behaviours. Customised CRM strategies based on the customer behaviours of each segment can be used to optimise marketing approaches. In the literature, many methods for segmentation have been discussed. Most studies implement a static segmentation method; however it is discussed that dynamic segmentation approaches are better suited methods due to the dynamics of the customer behaviours itself (Abbasimehr & Bahrini, 2021). This allows for the detection and inclusion of meaningful behaviour trends and seasonal patterns in segmentation models. A recent trend that confirms the dynamics of customer behaviour has come to light during the COVID-19 crisis. The pandemic highly accelerated the transition to E-commerce and according to McKinsey's customer-sentiment surveys (2021), customers also intend to shop more online even after COVID-19. Through the use of time series to capture customer shopping activities, dynamic segmentation recognises such significant and critical changes in behaviour.

Customer segmentation consists of multiple components, each with a variety of alternatives to choose from. Considering that dynamic structure is a relatively new approach to customer segmentation, a comprehensive overview is required in order to investigate the performance of different component sets. Components such as potential clustering methods and various distance measurements must be analysed to compare diverse dynamic segmentation performances with each other. Distance measurements are essential in dynamic clustering to quantify the similarity between two time series (Lines & Bagnall, 2015). Outperforming combinations of components will be explored using a sequential approach for selecting the dynamic segmentation's components. A sequential approach has been found to substantiate winners and failures without excluding any sequence of components (Chen et al., 2018). However, a drawback is the amount of time it takes to complete the analysis compared to a continuous approach.

Another component of customer segmentation is variable selection and it is the first to be considered in the sequential approach. Dynamic segmentation is attained with website data mining methods by tracking individual customer data. Customer characteristics are identified in a time series format such as website interaction, purchasing data, and many other variables. However, with the use of high-dimensional data comes the curse of dimensionality that arises according to the ratio between the number of variables and the number of samples (Fop & Murphy, 2018). To overcome this problem, feature selection must be conducted by selecting a limited subset of relevant features from all accessible data variables. For a long time, literature has concluded that not all variables contribute equally to establish cluster structure, such that no significant information loss will occur if correct features are chosen. In this paper, feature selection is chosen over feature extraction to keep segmentation models readable and interpretable. In feature extraction, the original meaning of the original features cannot be easily retrieved after the transformation of the high dimensional space into a lower dimensional space (Hancer, Xue, & Zhang, 2018). Popular features selected for customer distinction are Recency, Frequency, Monetary (RFM) variables and Customer Lifetime Values (Gupta et al., 2006). These purchase-related variables are well-documented and should undoubtedly be considered when characterising customers. However, these features have only been addressed separately and have never been examined simultaneously. Research must be done in order to evaluate their performances and decide whether or not other variables, such as website interaction, should be included.

Several online platforms already provide customer segmentation (D'Antonio, 2019). However, most of their underlying methods are black boxes. No documentation discusses whether a static or dynamic segmentation approach is executed, if feature selection has been performed for dimensionality reduction, nor if data preparation has been done. The disadvantage to this automated segmentation process is the lack of user control over the development process. Furthermore, results from a black box are frequently not acknowledged by one's organisation's management due to the absence of transparency. This management's lack of reliance is rooted in the black box algorithms' unknown capacity to interpret important structures such as trends and seasonal patterns (Heinrich et al., 2019). Research is needed to examine the validity of such an online tool and to provide alternatives applicable to the needs and circumstances of different E-commerce businesses. Moreover, the black box tools often maintain a fixed number of clusters, despite the business context such as customer characteristics and customer size. Literature shows that segmentation performances depend on the number of clusters that the customers are grouped in (Zhou et al., 2021). Research should highlight whether a fixed number of clusters is the way to go or if an adjustable cluster size is more desirable.

In the existing literature, the components of dynamic segmentation performances are mostly stressed out individually. However, the resources are very scattered, making it is difficult to recognise all the components together of dynamic customer segmentation. As a result, this thesis aims to provide a review of dynamic customer segmentation methods used in different E-commerce business contexts. To provide a sufficient review, several research questions need to be answered. First of all:

What different (components of) dynamic customer segmentation methods currently exist?

This needs to be investigated to give an extensive and contemporary overview of methods currently available and the components they consist of. This research question considers CRM, data mining, feature selection, customer segmentation, distance measurements, time series clustering methods, and centroid extraction.

After answering the first research question, different dynamic customer segmentation methods must be performed. In order to do so, the first step is to gather customer time series data using data mining methods. In this thesis, data is extracted from four different websites of E-commerce companies. The second step to consider is providing different feature selections to identify which variables are effective to cluster on such that the dimensionality is reduced. The next step is to execute various customer clustering methods combined with diverse distance measurements and for different numbers of clusters. Afterwards, the performance of the obtained clusters and of the black box clusters of the online tool will be calculated. After obtaining those performances, the second research question of interest is:

What are the best performing different dynamic customer segmentation methods and how do they compare with the black box customer segments of an online tool, considering four different companies' customers?

The answer to this research question will highlight which dynamic segmentation methods, consisting of which components, are best performing. Furthermore, it will clarify if the online tool can be outperformed by non-black box customer segmentation methods. The performances are calculated based on the compactness within the customer segments and the separation between the different segments.

Importantly, this paper adds to the literature as it analyses the segment performances of multiple companies, setting itself apart from previous studies that mainly focus on one individual company. Analysing and comparing multiple data sets from various business contexts with diverse customer sizes provide a broader insight into dynamic customer segmentation performances. Methods that work well in one business context may be sub-optimal in different business contexts. Therefore, the final sub-question in this study is proposed as:

To what extent are best performing variable selections, preferred number of clusters, distance measurements, and clustering methods correlated across different business contexts in E-commerce?

This question should be investigated to work out recommendations on which method(s) fit best in different business contexts. Business contexts can be characterised by the number of customers a company serves, the ratio of one-time-buyers, the average transaction value, and the average number of visits to the company's website. This thesis is highly relevant for organisations since insights in which method(s) should be preferred in their specific business contexts help them to stand out in the highly competitive business environment. Considering that marketing margins are getting tighter, cost-effective marketing approaches are crucial. The acquired insights of this paper will help companies to increase sales and use the data to understand their customers' behaviour. In addition, it opens up opportunities for personalised marketing and increases their customer engagement and retention.

This thesis is structured as follows: in Section 2, the existing methods and relevant information of scientific literature are reviewed. The used components of dynamic segmentation methods will be in-depth explained in Section 3, as well as performance measurements to uncover outperforming methods. The segmentation methods will be applied to four distinctive datasets described in Section

4. The results are stated in Section 5, and the conclusion of this thesis will be provided in Section 6. The paper will finish with a discussion in Section 7, along with possible future work.

2 Literature study

Customer Relationship Management (CRM) was introduced in the 1970s and presented as a unique technique for managing various automations of sales operations inside organisations (Buttle, 2004). Since then it has become a well-known tool for enterprise information management, not purely for sales and marketing applications (King & Burgess, 2008), but for knowledge management as well. CRM was designed to acquire the best understanding of a company's customers (Chen & Popovich, 2003). There are two types of CRM frameworks; operational and analytical (Berson & Thearling, 1999; He et al., 2004; Teo et al., 2006). Operational CRM includes sales force automations within organisations, whereas analytical CRM aims to gain insight into a company's customers (Ngai, Xiu, & Chau, 2009).

Data mining methods that facilitate customer analysis serve as the cornerstone of analytical CRM frameworks. Definitions for data mining are given amongst others by Berson and Thearling (1999), Lejeune (2001), Ahmed (2004) and Berry and Linoff (2004), by defining data mining as a tool for extracting and identifying useful online customer information, including demographic and purchasing data, and to uncover valuable hidden patterns in enormous amounts of data. Exposing and analysing customer information and characteristics enables to group customers with corresponding online shopping behaviours.

Customer segmentation is a popular technique for identifying various customer groupings (Manjunath & Kashef, 2021; Parvaneh, Tarokh, & Abbasimehr, 2014). In the literature, numerous methods for customer segmentation, also known as market segmentation, are available. In general, the methods can be classified into static and dynamic segmentation (Akhondzadeh-Noughabi & Albadvi, 2015). Abbasimehr and Bahrini (2021) state that dynamic segmentation approaches, compared to static approaches, are more suitable methods because of the dynamism of customer behaviours itself. Meaningful behaviour trends and patterns can be detected and included in the segmentation models. Static segmentation approaches assume that customers within segments behave consistently, but in reality, customer behaviour is evolving over time and the segments evolve accordingly. To overcome missing behaviour trends and patterns by implementing static approaches, dynamic methods have been proposed that consider the dynamics of customer behaviour in the analysis.

Dynamic segmentation approaches can be classified into two sections. The first section focuses on identifying changes in customer segments. In this case, at each point in time, customer information is collected and customers are then divided into segments. Following that, changes in segments over time are diagnosed and structural and/or content changes are analysed. Researchers Böttcher et al. (2009), Chen et al. (2005), and Wang and Lei (2010) implemented frequent item sets and association rule mining to detect segment changes, where Blocker and Flint (2007) proposed the creation of techniques with the ability to forecast the direction of the segments' changes. The aim of the second section of dynamic segmentation approaches is to track customer shifts between segments over time. This is mainly accomplished in the literature through the use of two distinct methods. The first method assumes customer transitions across segments to follow a first-order Markov process (Brangule-Vlagsma, Pieters, & Wedel, 2002). Hidden Markov models have been employed to model customer movements by considering different transition probabilities (Lemmens et al., 2012; Mark et al., 2013; Netzer et al., 2008). The second method employs association rule mining to uncover changes in customer behaviours. In this case, the top rules characterises the primary patterns of customer members within segments and distinguish them from one another (Akhondzadeh-Noughabi & Albadvi, 2015; Mosaddegh, Albadvi, Sepehri, & Teimourpour, 2021).

The dynamic segmentation methods stated above have been criticised by Abbasimehr and Bahrini (2021). They state that the time dimension of individual customer behaviours has been neglected. The methods focus mainly on the segment level rather than on the customer level. Furthermore, association rule mining is said to be an exploratory method, resulting in difficulties in measuring model performance. Abbasimehr and Bahrini are the first to represent dynamic customer behaviours in the format of time series, with customer observations being time-ordered sequences. The key advantage of time series data is the ability to predict customer behaviour, which is not achievable with exploratory methods (Yanovitzky & VanLear, 2008). Dynamic customer behaviour tracking in time series format is also adopted in this paper, whereas predictions will not be presented. However, this paper will serve as a basis for extending predictive customer analysis.

Regardless of whether a static or dynamic segmentation method is chosen, a key part is defining attributes to perform segmentation on, where attributes can either be discrete or continuous. Not all gathered data is suitable as variable inclusion into segmentation models to avoid over-parametrization (Fop & Murphy, 2018). Dimensionality reduction, a well-known approach in data preparation, attempts to eliminate unnecessary and redundant features that decrease the efficiency of an algorithm. Dimensionality reduction methods are classified into two types: feature extraction and feature selection. The dimensionality of the data is lowered through feature extraction by obtaining new features from the available original features. Principal Component Analysis (Jolliffe, 2002), Linear Discriminant Analysis (Ye, n.d.), and Singular Value Decomposition (Golub & Reinsch, 1971) are some examples of feature extraction methodologies. In contrast, feature selection attempts to choose a limited set of important original variables from all available variables using a predetermined criterion. Feature selection is often chosen over feature extraction in order to keep segmentation models readable and interpretable. In feature extraction, the meaning of the original variables cannot be easily retrieved after the transformation of a high dimensional space into a lower dimensional space (Hancer et al., 2018; Hancer, Xue, & Zhang, 2020).

The literature shows different feature selections as inputs for dynamic customer segmentation. One frequently used feature selection is clustering based on Recency, Frequency, and Monetary (RFM) variables, which are also considered to be the most important purchase-related variables. The RFM model was first proposed by Hughes (1994). Expanding the existing RFM model by a new dimension, interpurchase time, is introduced by Zhou et al. (2021). Interpurchase time keeps track of the average time between two consecutive purchases by a customer. Zhou et al. concluded that this new dimension adds value when considering customers' online purchase behaviour in a long-term period.

Another widely used approach is segmentation based on Customer Lifetime Value (CLV). This metric is defined by multiple definitions, but in general, it is identified as the total profit that can be expected from a customer as long as it remains active and is appropriate for developing group-specific marketing strategies (Liu et al., 2009; Hosseni et al., 2011). Marketers find this an important metric to determine how much can be invested in a customer while being profitable. The CLV is calculated as a function of customer acquisition, retention, and expansion (Gupta et al., 2006). When the CLV reaches zero, a customer is labelled as lost.

Purchase-related variables should undoubtedly be taken into account when characterising customers, and RFM variables and CLV are well-documented in the scientific literature. However, as far as we know, these issues have only been addressed separately, and have never been examined simultaneously. This study adds to the literature by reviewing what works best for variable selection and whether other variables should be included.

The next phase to consider after feature selection is clustering methods. Time series clustering methods, like non-time series segmentation approaches, can be categorised into two steps

(Abbasimehr & Bahrini, 2021). The first step consists of selecting a proper distance measurement. The second step is selecting a clustering method. A proper distance measurement is essential to quantify the similarity between two time series. These distance measurements must be computed for each pair of time series and captured in distance matrices. Popular distance measurements include Euclidean distance, (weighted) dynamic time warping, and shape-based distances (Lines & Bagnall, 2015; Wang et al., 2013). For the second step, Batista et al. (2014) and Paparrizos and Gravano (2015) state that the most widely applied time series clustering methods are hierarchical, spectral, and k-shape clustering. The distance measurements and clustering methods mentioned above are further elaborated in the next section.

3 Methodology

Website traffic data can be transformed into complete cross-device customer representation by applying data mapping methods. Cross-device data tracking is becoming more important as customers increasingly access companies' websites from multiple devices. Whereas original tracking methods identify one single device as a customer, cross-devices tracking methods are able to identify multiple used devices to represent one customer. Cross-device customer journeys can be realised by matching corresponding user identifiers. The next step is to convert every customer journey into time series of different website interactions. Time series are needed as input for dynamic customer segmentation, resulting in customer clusters with similar behaviours.

3.1 Feature selection for segmentation

Because of high-dimensional data, not all website traffic data can be included in clustering models. Feature selection must be performed choosing from all accessible data variables. No significant information loss will occur if correct features are chosen, because not all variables contribute equally to the cluster structure. A popular feature selected for customer distinguishment is clustering based on RFM variables, which consists of the most important purchase-related variables. In RFM the R refers to recency, the time interval between the first and most recent purchase of a customer. The F stands for frequency, the number of purchases of a customer between a certain start and end date. The M for monetary value, the amount of money spent by a customer between a certain start and end date. When considering customers' online purchase behaviour in a long-term period, Zhou et al. (2021) concluded that the additional dimension of interpurchase time adds value. The variable of interpurchase time (T) keeps track of the average time between two consecutive purchases by a customer. RFM(T) analysis frequently supports the Pareto Principle, also known as the 80/20 rule (Christy et al., 2021). A well-known principle among marketeers is that 80% of a business's revenue comes from 20% of its customers.

In contrast to non-dynamic segmentation methods, which allow multiple features to be included as input for clustering methods, clustering in time series format only allows for the selection of one feature. As a result, only one variable from the RFMT variables can be chosen for customer differentiation. Due to the time series format, the recency and interpurchase time are already taken into account by their position in an ordered transaction time series. As a result, the R and T attributes are omitted from this study as they have no additional ability for customer distinction. The frequency value of each customer indicates the number of transactions a customer has completed in a certain week. It is obvious that the monetary value involved in a transaction can be of any number, in a way that a large number of transactions does not necessarily reflect a high monetary value. As a result, for studying customer behaviour, the monetary value variable is chosen as a representation of customer behaviour because the overall objective of any organisation is to achieve high profitability (Kumar & Shah, 2004). Furthermore, Abbasimehr and Bahrini (2021) identifies the monetary variable as the most relevant of the RFMT variables as well.

Another popular feature selection, also implemented in this thesis, is segmentation based on Customer Lifetime Value (CLV). This feature identifies the total profit that can be expected from a customer within a year as long as it remains active. This allows a company to estimate its overall profitability, determine customer acquisition marketing budgets, and define growth and improvement targets. The CLV is calculated weekly for each customer as a function of customer acquisition, retention, and expansion (Gupta et al., 2006). The RFMT variables are also fed to the CLV method. Modeling future transaction variables, including purchase frequency and customer churn, is performed by the Beta Geometric/Negative Binomial Distribution (BG/NBD) model (Fader, Hardie, & Lee, 2005). The BG/NBD model states some necessary assumptions:

- Transactions of active customers are described by Poisson distribution with rate λ.
- Variations of transaction behaviour across customers are described by Gamma distribution with shape parameter r and scale parameter a.
- The probability of becoming an inactive customer after any transaction is defined by p and the dropout point between transactions has a Geometric distribution.

- The dropout probabilities are described by the Beta distribution with shape parameter α and scale parameter β.
- The transactions and dropout probability are independent and identically distributed across customers.

Modeling the monetary variables, including average order value, is performed by the Gamma-Gamma (GG) model. This model also states some assumptions:

- Monetary values of transactions are randomly distributed across the average transaction value for every customer.
- Average transaction values are independent across customers.
- Average transaction values are not related to customers' transaction frequency values.

The Pearson correlation coefficient can be used to test the latter assumption, and the parameters λ, r, a, α, and β are the fitted coefficients of the BG/NBD and GG model. After validating the assumptions, the CLV measurement is implemented by using the `customer_lifetime_value` function from the `lifetime` Python package. The weekly CLV is determined for each customer for the time frame of all weeks between the most recent date of website data collection and one year before. The weeks preceding this period are omitted since the calculating model is learning from all customers who have made two or more purchases. To train the model sufficiently with a reasonably large enough number of customers who made these transactions, the CLV computation starts at a later point in time than the first date of website data collection.

Besides the monetary value and the CLV time series features, this thesis also analyses the effect of including one other time series feature belonging to customers. The additional time series contains the number of pageviews for each website visit. A pageview is the occurrence of a company's pages being loaded. The total number of pages visited in a session is defined as pageviews. This variable is not yet included in other studies. However, it intuitively reflects the shopping behaviour of customers as well. A user who regularly visits the website and sees various website pages in a session, whether they place an order or not, has a higher probability of making a purchase than a user who visits the website only once. As a consequence, pageview time series can be used to also make distinctions between customers who did not place an order, something that the two previous mentioned features cannot do. When customers have not placed an order, the monetary value and

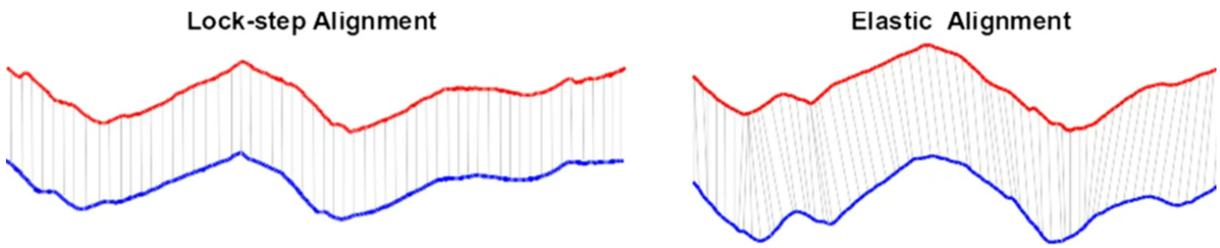


Figure 1: Alignment of a lock-step measure versus the alignment of an elastic measure

CLV time series will all be zero. In conclusion, this thesis analyses three different potential feature selections as inputs for the clustering methods.

For each feature selection, several clustering methods are performed. Similar to non-time series clustering, time series clustering methods can be categorised into two steps (Abbasimehr & Bahrini, 2021). The first step consists of selecting a proper distance measurements. The second step is selecting a cluster method.

3.2 Distance measurements

For the first step, let y_1 and y_2 be two time series. A distance measurement is chosen to quantify the similarity between y_1 and y_2. These distance measurements are calculated for every pair of time series and are captured in a distance matrix. To provide a sufficient review, multiple distance metrics are considered, each having its own approach of detecting similarity. The application highlights what distance metric is suitable in what business context.

The well-known Euclidean distance (ED) compares $y_{1,t}$ and $y_{2,t}$ at each time unit t. The ED is seen as a lock-step measure where fixed one-to-one pairs of elements are compared. The calculation of the ED is very time sensitive and is calculated as

$$ED(y_1, y_2) = \sqrt{\sum_{t=1}^{T} (\, y_{1,t} - y_{2,t}\,)^2}. \tag{1}$$

The ED is intuitive, simple and performs well when $y_{1,t}$ and $y_{2,t}$ are relatively similar. However, problems arise if one of the time series is not properly aligned with the other.

A more sophisticated similarity measurement is required if time series must be grouped based on their shapes and be less sensitive to the index t. Elastic measures attempt to generate a non-linear scaling that will align the time series and allow one-to-many point comparisons. Figure 1 displays the difference in approaches compared for lock-step and elastic measurements. Dynamic Time Warping (DTW) is such an elastic measure. DTW compresses or expands sections of one time series in an iterative manner to find the best match with the other time series. Every observation of y_1 is matched with one or more observations of y_2. The first and last observations of y_1 are at least matched with the first and last observations of y_2. Moreover, no cross-matches are allowed, such that $y_{1,i}$ cannot be matched with $y_{2,j}$ if $y_{1,h}$ is matched with $y_{2,k}$ where $h < i$ and $j < k$. In this way, the method aligns the peaks and valleys of y_1 and y_2, overcoming the disadvantages of the Euclidean distance. It detects similar shapes, even if the series are shifted or re-scaled (Cassisi et al., 2012). The DTW score quantifies the similarity between y_1 and y_2 as the minimised sum of absolute differences of all possible alignment paths between y_1 and y_2. Each alignment path is defined by $\pi \in M$, where M represents the set of all possible alignment paths. As a consequence, DTW distance can be calculated as

$$DTW(y_1, y_2) = \min_{\pi \in M} \Big(\sum_{(i,j) \in \pi} \mid y_{1,i} - y_{2,j} \mid \Big). \tag{2}$$

The lower the DTW score, the more similar the shapes of y_1 and y_2 are.

An elastic distance measure, both incorporating time-sensitive correspondence across time series as well as similarity in time series behaviour, is introduced by Chouakria and Nagabhushan (2007). The distance measurement is defined as the temporal correlation coefficient (CORT) and is a variant of the Pearson correlation involving first-order differences. It determines whether inaccuracy at a specific time index is correlated across its direct neighbours. The CORT coefficient is calculated as

$$CORT(y_1, y_2) = \frac{\sum_{t=1}^{T-1} (\, y_{1,t+1} - y_{1,t} \,) (\, y_{2,t+1} - y_{2,t} \,)}{\sqrt{\sum_{t=1}^{T-1} (\, y_{1,t+1} - y_{1,t} \,)^2} \sqrt{\sum_{t=1}^{T-1} (\, y_{2,t+1} - y_{2,t} \,)^2}}. \tag{3}$$

The value of CORT ranges between the interval [-1,1], where CORT$= 1$ represents two time series of similar behaviour with corresponding growth direction and rate. CORT$= -1$ means a similar growth rate but in the opposite direction, and CORT$= 0$ implies no similarity between the two series (Montero & Vilar, 2015). Afterwards, the CORT coefficient can be used to translate the value in a distance measurement by the use of an existing distance measurement, $d(y_1, y_2)$. The CORT distance can be calculated as

$$d_{CORT}(y_1, y_2) = \frac{2 \cdot d(y_1, y_2)}{1 + exp(k \cdot CORT(y_1, y_2))} \tag{4}$$

This thesis executes the CORT distance using the Euclidean distance as the existing distance measurement and $k = 2$. The lower the CORT distance, the better the fit.

The Shape-based distance (SBD) is another distance metric that takes correlations into account. The normalised cross-correlation (NCC) is incorporated to compare time series observations, irrespective of the index t. Where DTW compares sections of the time series, denoted as local alignment, SBD compares the entire time series, denoted as global alignment (Paparrizos & Gravano, 2017). To correct for misalignments, time series are shifted over time where y_2 remains static and y_1 slides over y_2 for each shift s of y_1. A shift of a time series is denoted as follows:

$$y_{1(s)} = \begin{cases} (\overbrace{0, \ldots, 0}^{|s|}, y_{1_1}, y_{1_2}, \ldots, y_{1_{m-s}}), & if\ s \geq 0 \\ (y_{1_{1-s}}, \ldots, y_{1_{m-1}}, y_{1_m}, \underbrace{0, \ldots, 0}_{|s|}), & if\ s < 0. \end{cases} \tag{5}$$

Cross-correlations (CC) are calculated to determine how much one time series must be shifted over time in order to match the other time series as closely as possible. The optimal shift is determined by the position w at which $CC_w(y_1, y_2)$ is maximised, where $s = w - m$. Afterwards, normalisation of the obtained maximum cross-correlation value is required such that the SBD can be calculated as in the following equation:

$$SBD(y_1, y_2) = 1 - \max_{w} \left(NCC_w \left(y_1, y_2 \right) \right). \tag{6}$$

The value of SBD ranges between the interval [0,2], where SBD = 0 represents two time series of perfect similar behaviour and SBD = 2 implies no similarity between two series. For efficient computation of SBD, a Fast Fourier Transformation is performed and normalised by the use of the geometric mean of the time series autocorrelation. Afterwards, the index w is located where the NCC is maximised (Abbasimehr & Bahrini, 2021).

The last distance metric considered by this thesis is the complexity-invariant dissimilarity measure (CID). Batista, Wang, and Keogh (2011) proposed this metric as they argued that complex time series were often incorrectly classified. The CID distance adds a correction factor to an existing distance measurement for complexity difference across two time series (Lines & Bagnall, 2015). Let $d(y_1, y_2)$ be an existing distance measure, the calculation of CID distance is given as

$$CID(y_1, y_2) = CF(y_1, y_2) \cdot d(y_1, y_2). \tag{7}$$

The complexity correction factor $CF(y_1, y_2)$ is a function of complexity estimators of y_1 and y_2, given by

$$CF(y_1, y_2) = \frac{\max\left\{ CE(y_1), CE(y_2) \right\}}{\min\left\{ CE(y_1), CE(y_2) \right\}}. \tag{8}$$

The complexity estimator proposed by Batista et al. (2011) is calculated as

$$CE(y_i) = \sqrt{\sum_{t=1}^{T-1} \left(y_{i,t} - y_{i,t+1} \right)^2}. \tag{9}$$

This thesis executes the CID distance using the Euclidean distance as existing distance measure. The lower the CID, the better the fit. The CID distance is a relatively unknown metric; nonetheless, it has outperforming results in the study of Abbasimehr and Bahrini (2021). It is in this paper's interest to implement and compare the results with other distance measurements.

3.3 Clustering methods

For the second step, the most widely applied time series clustering methods are used to perform clustering, consisting of hierarchical clustering, spectral clustering, and k-shape clustering (Paparrizos & Gravano, 2017). Each method is executed for multiple values of k, ranging from 4 to 8 clusters, representing the most frequently used numbers for k in previous studies.

In hierarchical clustering, a tree diagram is built and can be visualised by a dendrogram. Hierarchical clustering is created by either the agglomerative clustering algorithm or the divisive clustering algorithm (Bunge & Judson, 2005; Zhou et al., 2021). In agglomerative clustering, all the time series are initially seen as separate clusters. In every next iteration, two clusters are merged together that are most similar to each other until only one cluster is left or, in the application of this paper, until a set of k clusters is reached. An example of a dendrogram for agglomerative clustering is visualised in Figure 2. The individual data points can be seen on the x-axis, all initially represented as separate clusters, and the height on the y-axis indicates the number of iterations.

In divisive clustering, the algorithm begins the other way around. Every time series is initially placed in one cluster, and all clusters are divided into two clusters in each iteration. The algorithm terminates until all time series represent their own cluster. As divisive methods are computationally

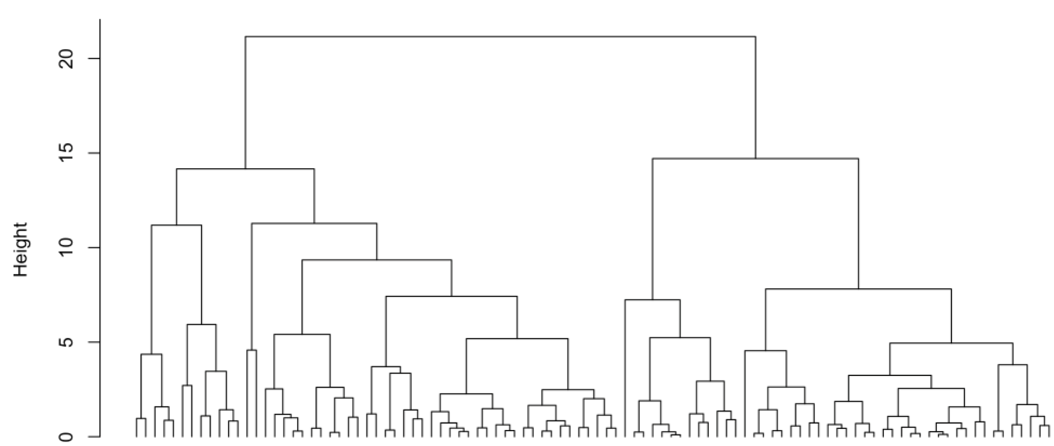


FIGURE 2: Dendrogram for agglomerative hierarchical clustering

costly and applications in the literature are limited (Bunge & Judson, 2005), this thesis implements the agglomerative clustering algorithm.

In every iteration of the agglomerative clustering algorithm, pairs of clusters are merged. To define the cluster similarities, the distance matrix must be updated after each merge of cluster c_i and c_j for all $i \neq j$. The distance values belonging to c_i and c_j must be removed from the distance matrix and replaced by values of the new cluster $c_i \cup c_j$. The Lance-Williams dissimilarity update formula, provided by Murtagh and Contreras (2012), computes the new distances between cluster $c_i \cup c_j$ and all other clusters. The formula calculating the distance between cluster $c_i \cup c_j$ and any cluster c_k is defined by

$$d(i \cup j, k) = \alpha_i \, d(i,k) + \alpha_j \, d(j,k) + \beta \, d(i,j) + \gamma \, |d(i,k) - d(j,k)|. \tag{10}$$

The parameters are set by $\alpha_i = \dfrac{|i| + |k|}{|i| + |k| + |k|}$, $\beta = \dfrac{|k|}{|i| + |k| + |k|}$ and $\gamma = 0$, where $|i|$ represents the number of time series in cluster i. These parameter settings are known as the Ward linkage criterion. The Ward's method (Ward, 1963) merges the two clusters for which the total within-cluster variance increases the least after merging. This criterion is chosen because it takes into account all time series in the clusters, whereas other criteria only take into account some specific time series in the clusters or are not suited for this thesis. Single linkage only considers the time series that obtain the minimum distance between time series in one cluster and time series in the other cluster. Complete linkage is the opposite of single linkage in that this method only considers the time series that obtain the maximum distance between time series in one cluster and time series in the other. A linkage criterion that takes into account all time series in the clusters, in a different manner than in the Ward linkage criterion, is Average linkage. Average linkage is calculated by averaging all distances between time series in one cluster and time series in the other. It works effectively on well-separated clusters but decreases in performance otherwise. Average linkage is not selected in this thesis since the clusters will not be clearly separated. At last, the centroid linkage approach calculates the distance between the two cluster centroids. This approach is not employed in this thesis as it could possibly fail to meet the condition of monotonicity of merge. This condition implies that clusters are merged together at a later stage than any of their components in the agglomerative clustering algorithm. When this assumption is not met, an inversion in the dendrogram can occur, which undermines the interpretation of the dendrogram (Fisher & van Ness, 1971).

One main advantage of hierarchical clustering is that a predefined number of clusters (k) is not requested. A cut from the dendrogram can be made for every value of k without having to rebuild the dendrogram again with the hierarchical clustering method for every k. The only request is that k must be smaller than the total number of time series. Another advantage is the allowance for every distance measurement, no requirements are needed such that hierarchical clustering can be considered in many cases. Furthermore, convergence towards local minima and initialisation issues do not apply to this algorithm. In contrast, a disadvantage of hierarchical clustering is that updating the distance matrix in every iteration is computationally expensive when the number of time series grows. Moreover, the algorithm is a greedy approach, so only the best merging act in the current iteration is considered. There is no possibility to look forward and to see whether another merge is preferred in the long run. The greedy algorithm also has no room for flexibility, as the dendrogram cannot be modified after clusters are merged in the early stages. Errors made in the lower part cannot be undone (Sardá-Espinosa, 2017).

In spectral clustering, input data is seen as a graph and graph theory is applied. Time series are seen as nodes and the similarity values are used as edges in the applied case of this paper. The aim of spectral clustering is to find different communities of nodes, resulting in the division of time series into clusters. First, the distance matrix is converted into a similarity matrix $\boldsymbol{S}$ as

$$s_{ij} = \begin{cases} \exp(-\gamma\, d(y_i, y_j)) & \text{for } i \neq j. \\ 0 & \text{for } i = j. \end{cases} \tag{11}$$

with γ as scaling factor and $d(y_i, y_j)$ as the distance measurement between time series y_i and y_j. The next step is to use the similarity matrix and the diagonal matrix $\boldsymbol{D}$, where $d_i = \sum_{j=1}^{n} s_{ij}$, to obtain the normalised Laplacian matrix $\boldsymbol{L}$ following

$$\boldsymbol{L} = \boldsymbol{D}^{-1/2}\boldsymbol{S}\boldsymbol{D}^{-1/2}. \tag{12}$$

The eigenvectors corresponding to the largest k eigenvalues of $\boldsymbol{L}$ are then extracted and used as columns in the newly created $n \times k$ matrix $\boldsymbol{V}$. This matrix must be normalised, resulting in a $n \times k$ matrix $\boldsymbol{U}$, calculated as

$$u_{ij} = \frac{v_{ij}}{\sqrt{\sum_j v_{ij}^2}}. \tag{13}$$

Every row in $\boldsymbol{U}$ can be interpreted as a k-dimensional data point, $x_i \in \mathbb{R}^k$. The k-means clustering algorithm can now be used to cluster (x_1,..., x_n).

Spectral clustering has the advantage of flexibility in terms of incorporating no restrictions on types of data. Flexibility is also reflected in the acceptance of every cluster form, where other cluster approaches often assume data to be centred around the cluster mean. Spectral clustering solutions have grown in popularity by outperforming traditional clustering methods as k-means. The disadvantage, however, is that the number of clusters k must be predefined and is computationally expensive for high-dimensional data (Baek & Kim, 2021).

k-Shape clustering is an algorithm specially developed for time series data clustering, having its roots in the k-means algorithm (Paparrizos & Gravano, 2015). In k-shape clustering, time series are iteratively matched together based on how similar their shapes are. The algorithm consists of two steps. In the first step, the assignment step, the current centroid of each cluster is defined, and every time series is assigned to the cluster with the closest centroid. The distance metric used in this technique is the previously mentioned SBD with the incorporated normalised cross-correlation. The index where the NCC is maximised reflects the cluster centroid. In the second step, the refinement step, new cluster centroids are calculated due to shifts in cluster memberships of the time series in the assignment step. This algorithm is terminated until no more membership shifts arise or if the number of iterations reaches its maximum. The maximum number of iterations is set to 100.

k-Shape is claimed to run fast and return compatible and well-separated clusters. Moreover, time series of different lengths can be incorporated without any transformation needed. This is beneficial when considering customer behaviour time series, as customer relationships start at various moments in time. A drawback of this algorithm is that it can only be combined with the SBD measurement, whereas the other clustering algorithms enable to vary with different distance metrics. If future research uncovers a new outperforming distance metric, it cannot easily be integrated into the k-shape clustering method. Another disadvantage is the fact that a predefined number of k is required.

3.4 Performance metrics

After obtaining the partitioning of customers into clusters from the different methods, the performances of the clustering models as well as the online tool are calculated. The literature shows that no superior single performance metric exists for cluster validation. Therefore, two commonly-used performance measurements are considered, both having a different calculation on the goodness of the cluster division. The two metrics include the Davies-Bouldin Index (DBI) and the Silhouette

Index (SI) in which k represents the number of clusters. A better-defined cluster partitioning holds a lower Davies-Bouldin index score and a higher Silhouette index score (Zhou et al., 2021).

The Davies-Bouldin index considers the compactness within the clusters and the separation between the clusters. A lower DBI score is given when the compactness within a cluster is minimised and the separation between different clusters is maximised. In terms of customer segmentation, this means that customers with similar shopping behaviours are clustered together, whereas customers with different shopping behaviours are not. Let z_i correspond to the centroid of cluster i. The compactness of cluster i can then be defined as

$$S_i = \frac{1}{|C_i|} \sum_{x_j \in C_i} d(z_i, x_j), \tag{14}$$

and the separation of two clusters, C_i and C_j, is calculated as the distance between their centroids for every $i \neq j$. As a result, the Davies-Bouldin index can be obtained as

$$DBI_k = \frac{1}{k} \sum_{i=1}^{k} \max_{j, j \neq i} \{ \frac{S_i + S_j}{d(z_i, z_j)} \}. \tag{15}$$

The Silhouette index differs from many other methods as it is based on the calculation of a performance measurement known as the Silhouette width for each individual sample. The Silhouette width is a confidence indication for the ith sample's allocation in cluster C_j. In this thesis, a sample is equivalent to a customer's time series. The SI is calculated by taking the average of all computed Silhouette widths, where the Silhouette width is calculated as follows. First, the average distance between customer i's time series and all other customers' time series within its assigned cluster is calculated as in the following equation:

$$a(x_i) = \frac{1}{|C_i| - 1} \sum_{x_j \in C_i, j \neq i} d(x_i, x_j). \tag{16}$$

The average distance to all of the customers' time series in the other cluster is after that calculated separately for each cluster m, where $m \leq k$ and m is not the same cluster as the cluster where customer i is assigned to. The following equation shows that, for further calculation, the minimum value of the $k - 1$ average distances is used:

$$b(x_i) = \min_{m \neq i} \frac{1}{|C_m|} \sum_{j \in C_m} d(x_i, x_j). \tag{17}$$

For the Silhouette width, a ratio of the two equations above is composed, and the average of all Silhouette widths generates the Silhouette index:

$$SI_k = \frac{1}{N} \sum_{i=1}^{N} \frac{b(x_i) - a(x_i)}{\max \{b(x_i), a(x_i)\}}. \tag{18}$$

The Silhouette width for each customer varies from -1 to 1, with a customer receiving a value of -1 actually being closer to another cluster than the one it has been assigned to.

The distance between two samples, $d(x_i, x_j)$, is included in the DBI and SI. It is calculated using all the distance measurements described in Section 3.2. Previous research has shown that results of performance metrics are generally unaffected by the used calculation of distance (Azuaje & Bolshakova, 2003). Because these studies used non-time series data, it is of interest in this thesis to validate whether the same conclusion applies when using time series data.

3.5 Cluster centroids

With the resulting DBI and SI scores, the best performing dynamic segmentation method and a suitable number of clusters can be determined. For each cluster of the best performing method, the centroid time series will be extracted. It will display the patterns and trends of the customer behaviours included in a cluster. This is essential information for marketeers who will be building cluster-specific CRM strategies. Each centroid is calculated by the DTW barycenter averaging (DBA) algorithm of all time series belonging to a certain cluster. DBA is a global averaging strategy that includes iteratively optimising the average sequence. The objective is to minimise the sum of squared DTW distances between the average sequence and the group of series to be averaged. In each iteration, the DBA algorithm calculates the DTW between each individual sequence and the temporary average sequence. This provides for every coordinate of the temporary average sequence a set of associated coordinates that belong to the group of series. These sets are in turn used to calculate its barycenter. After that, the temporary average sequence coordinates are updated as the calculated barycenters. Therefore, every coordinate of the average sequence S_i can be updated as

$$S_i' = barycenter(\, assoc(S_i)\,), \tag{19}$$

$$barycenter(y_1, y_2, ..., y_n) = \frac{1}{n}(y_1 + y_2 + ... + y_n).$$

The algorithm continues until the optimal average sequence is identified (Petitjean, Ketterlin, & Gançarski, 2011). The algorithm will always converge since the sum of squared DTW distances decreases between every two iterations. The proof of the convergence of the DBA algorithm is provided in Appendix A.

4 Data

4.1 Website traffic data of four E-commerce companies

All customer data collected from the websites of four different E-commerce companies is used for this thesis. None of the companies have contractual relationships with their customers, meaning no contractual agreement or membership applies in their relationships. In contractual relationships, companies know immediately whenever customers are leaving, whereas in non-contractual relationships it cannot be as easily observed. Another distinction can be made in the opportunity for transactions. Goods can be offered limited to certain periods in time, referred to as discrete-time transactions. This includes transactions such as charity donations and event ticket orders. The opposite of discrete-time transactions are continuous-time transactions, where purchases are not dependent on time periods and can take place at any moment. This thesis only includes companies with non-contractual relationships having transactions continuous in time. The main retail products of the distinctive E-commerce companies are sportswear, office furniture, home accessories, and lamps. For the previous 26 months, anonymous data of every interaction on all four websites was collected. This includes dates, sessions, pageviews, the number of items that are added to the cart, transaction data, transaction identifiers, client identifiers, the sources where the website's traffic comes from, and by which medium a visitor landed on the website.

The website traffic data is stored in different tables in a database which can be called by SQL queries. The basic predefined query sorts the website's traffic data by date. This must be converted into time series formats for every individual user. To acquire cross-device representations, the different tables must first be joined on user level. When website visits have occurred via multiple devices, the same user has overlapping types of identifiers in the tables. Grouping all these identifiers into a mapping table results in a table that stores all the data of every single user in one row. All these rows reflect the customer journeys. For this thesis, we only consider customer journeys with at least one transaction included. This results in 1,965 unique customer journeys for the sportswear retailer; 2,291 for the office furniture supplier; 3,138 customer journeys for the company selling home accessories; and 3,719 customer journeys for the retailer of lamps. The different companies are in this thesis further referred to as company 1, 2, 3, and 4, respectively. The characteristics of the companies' customers are summarised in Table 1 where customers are categorised by the number of completed orders.

The next step is to convert every customer journey into a time series of different interactions

	Company 1			Company 2			Company 3			Company 4		
# Orders	%	€	#	%	€	#	%	€	#	%	€	#
1	75.9%	€78.80	2.6 (56.5)	95.4%	€550.03	2.5 (32.6)	21.2%	€65.15	3.0 (85.6)	93.5%	€152.64	2.4 (23.1)
2	19.4%	€102.43	5.6 (138.6)	3.4%	€583.73	5.6 (83.1)	62.8%	€103.10	7.4 (122.8)	5.6%	€147.00	4.6 (50.0)
3	3.6%	€90.80	10.7 (271.9)	0.8%	€768.85	7.2 (102.7)	10.2%	€99.54	15.6 (241.0)	0.7%	€170.87	7.6 (88.7)
≥ 4	1.1%	€111.48	14.9 (385.6)	0.4%	€1121.80	13.7 (245.0)	5.8%	€102.20	29.3 (527.3)	0.2%	€121.46	10.0 (128.5)

Table 1: Characteristics of company's customers where % represents the percentage of the total number of customers, € the average transaction value and # representing the average number of visits to the website with the average number of pageviews per customer in parentheses

aggregated into weekly bins. When working with time series data, there are two forms of clustering. When the original data is one long time series that needs to be split down into sections in order to execute clustering on those parts, this is referred to as sequence clustering. When you have numerous individual time series that you wish to compare, you must use whole clustering. The latter is the case in this research. The time series of selected features are normalised such that their mean over the whole time period is zero and their standard deviation equals one. Data normalisation is an important step before feeding data to any model to ensure the quality of the data. As a consequence, normalisation helps clustering algorithms work more efficiently, which in turn leads to a substantial influence on their performance (Panigrahi, Karali, & Behera, 2013).

4.2 Segmentation results of the online tool

To be able to compare the segmentation results of this thesis with the segments obtained by the black boxes of an online tool, Datatrics will be used as a tool to gather data from one organisation. Customer segmentation is one of the services provided by this paid customer data platform. A customer's user-id can be used to locate them in one of the clusters. Datatrics considers a fixed number of five clusters that are based on the customer type. Regardless of what type of retail product a company offers or what number of customers a company has, the number of clusters is always five. The customer types are labelled and described as follows:

- Explorer: This customer type is known for the fact that he or she has no specific need or desire. He or she possibly wanders around looking for an enjoyable and pleasurable experience. Characteristics: high number of pageviews and a medium number of transactions.

- Single-minded: This customer possibly knows what he or she wants, has a specific intention and is focused on achieving a particular goal.

Characteristics: visits from a search engine, low activity, short website sessions.

- Passionate: This customer visits the store on a whim and does not have the intention of buying something initially. They will possibly make purchase decisions on impulse, acting on what seems good at that time.

  Characteristics: visits from a non-search engine and long average stays on the website.

- Economical: This customer shops frequently, however, buying decisions depend on economic and personal financial situations.

  Characteristics: pays attention to deals and may also conduct price comparisons before purchasing.

- Believer: This customer is possibly the most loyal one, frequently visiting the shop and buying products.

  Characteristics: frequent visits, new products, more than average lifetime value, often subscribed to newsletters.

The descriptions and characteristics of the customer types provide some insight into the clusters' classification. In the descriptions above, the following variables are mentioned: CLV, transactions, sessions, and pageviews. It is, however, unknown whether other variables are used or how CLV is determined. Furthermore, the manner in which the variables are being used is not mentioned. The disadvantage of these black boxes is that it is unclear whether the approaches are set in stone or can be adapted to the type of retail goods offered by a company and the quantity of people they serve. Another issue is that the defined clusters by the tool include both known and unknown customers. It is not possible to re-target these undefined customers as no customer information is provided. In Table 2, an overview can be found of how many customers belong to each segment and how known and unknown customers are divided for company 2. Except for the Explorer segment, all clusters contain a large proportion of unknown customers. Also, the total number of clustered known customers is summed up to 1,176; whereas, as stated earlier, 2,291 known customer journeys exist. The reason for this gap is undocumented by the tool.

	Explorer	Single-minded	Passionate	Economical	Believer
Known customers	380	757	16	2	21
Unknown customers	18	14.711	82	66	50
Total customers	398	15.468	98	68	71

TABLE 2: Segmentation of customers by Datatrics for company 2

5 Results

5.1 Time series computation

The gathered data from the four different E-commerce companies must be converted into a time series format for each individual user. To achieve this output, different storage tables are joined by overlapping customer and transaction identifiers. Table 3 shows a sub-table of this resulting joined table, in which the interaction with one of the websites of an anonymised customer is sorted by date. The exposed customer landed on the website via numerous sources that direct traffic to the website and completed three transactions throughout this time period. The data is converted into a time series by taking the preceding 26 months and aggregating the website interactions of interest into weekly bins. The time series of the monetary value of the completed transaction and the number of visited pages for the same customer outlined in Table 3 are displayed in Figure 3 and Figure 4, respectively.

Date	Customer ID	Source / Medium	Sessions	Pageviews	Quantity of ordered goods	Transactions	Transaction revenue	Transaction ID
17-09-2021	123.456	Newsletter / email	1	16	0	0	0	
26-10-2021	123.456	Google / organic	4	30	5	1	€111.94	000110
07-11-2021	123.456	Criteo / display	2	8	0	0	0	
27-11-2021	123.456	Abandoned AddToCart / email	2	35	0	0	0	
20-12-2021	123.456	Newsletter / email	1	21	1	1	€47.43	000150
03-01-2022	123.456	Product Review / email	1	3	0	0	0	
21-01-2022	123.456	Newsletter / email	1	45	3	1	€59.99	000250
24-02-2022	123.456	Google / cpc	1	2	0	0	0	
28-03-2022	123.456	Newsletter / email	1	8	0	0	0	

TABLE 3: Subset of website interaction of one customer

Figure 3: Monetary value time series

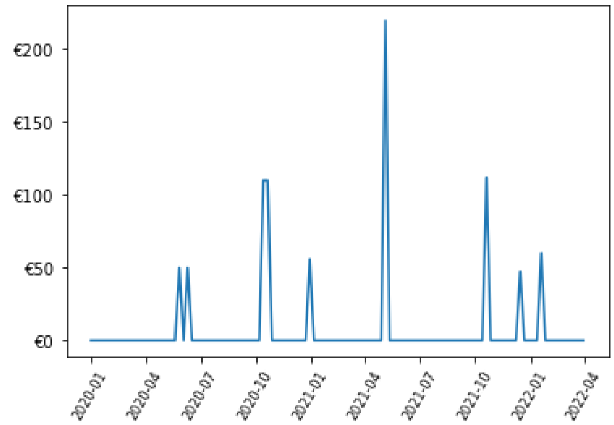
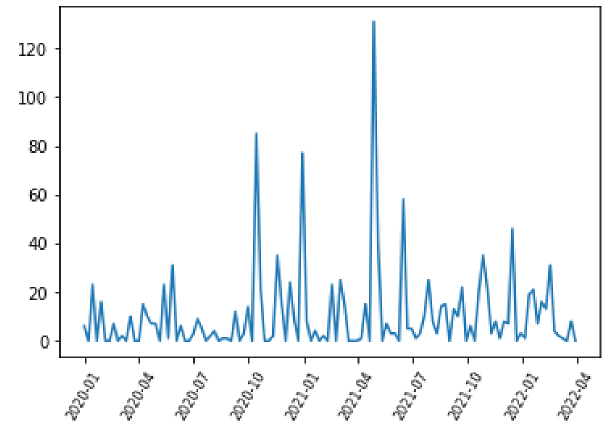


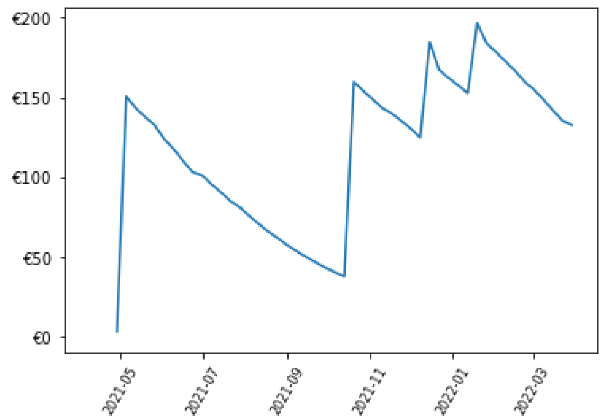
Figure 4: Pageview time series



5.1.1 CLV calculations

The Customer Lifetime Value time series requires more attention, as the values have to be estimated first. As previously stated in Section 3.1, the time series will be considered for all weeks between the most recent date of website data collection and one year before. The preceding weeks are used to train the model. The values of recency, frequency, monetary, and interpurchase time for each customer are calculated at each time point in the time series, since this is required as input for the CLV calculation. Within this calculation, the BG/NBD model is used to forecast the number of expected future transactions for each customer during the following year. Next, the Gamma-Gamma Model estimates the average monetary value of the expected transactions. It is necessary for the latter model to first verify that no linkage exists between the frequency and monetary values. The Pearson correlation is used to check the relationship between the two variables. The average correlation value throughout all time steps is 0.0474, with a maximum value of 0.0868 found for one company. These correlation values all appear to be weak. The correlations of the other companies are all similar and do not exceed the value of 0.01. As a result, we can assume that the assumption has been met and fit the Gamma-Gamma model to the data of this thesis. The GG model initially calculates the conditional expectation of the average monetary value per transaction for each customer. This can afterwards be used to estimate a customer's lifetime value. It should be noted that the customer lifetime value is the predicted order value, not the customer profit. The time series of the CLV of the previous exposed customer is illustrated in Figure 5. Peaks in the time series can be found at the time stamps when transactions have occurred.

FIGURE 5: Customer Lifetime Value time series



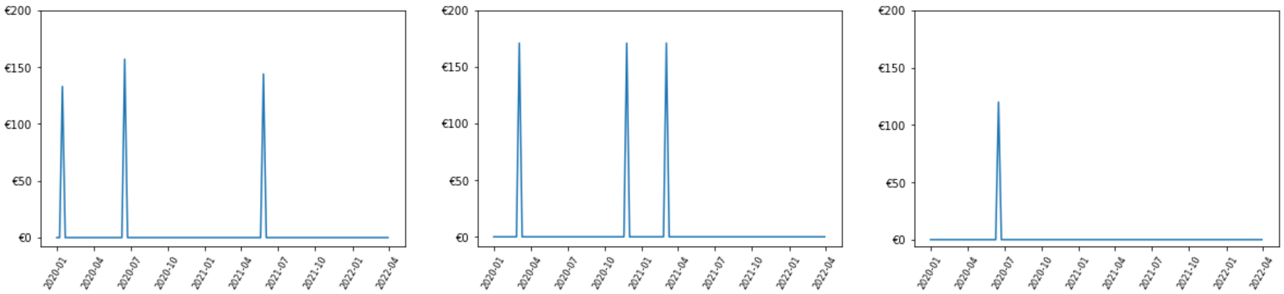
5.2 Distance metrics

To quantify the similarity between two time series, distance measurements are calculated. For each pair of time series, several distance matrices are considered, each having its own approach of detecting similarity. Whereas the Euclidean distance compares the ith time stamp of one series to the ith time stamp of another, are other distance measurements defining similarity based on shape.

5.2.1 Detection of time series similarity

When we compare three monetary value time series as shown in Figure 6, we can see that the first two customers made three transactions with a company during their active lifetime, while the third customer made one transaction. The transactions of the first two customers did not occur in the same weeks, but overall, their shopping behaviour appear to be quite similar. In contrast, if we compare the last time series to the first two, we may conclude that the last customer has a substantially different pattern. This is especially true given that the last customer only made one transaction in comparison to the other customers who made three transactions. When the various distance metrics described in Section 3.2 are calculated for these three time series, the results do not always correspond to what we would expect. Table 4 summarises the values of the five distance measurements considered in this thesis. As the table shows, the Euclidean distance does not consider the behaviour of the two first time series to be very similar. It even shows that time series 1 and 3 are the most comparable of the set. This is entirely due to the fact that the transactions in the first two time series do not occur in the same week. When slight misalignments along the time axis exist between two time series, the Euclidean distance can be substantially affected. As a result, the

FIGURE 6: Monetary value time series of three different customers



	ED	DTW	CORT	CID	SBD
[1,2]	15.56	0.74	15.56	15.58	0.65
[2,3]	15.48	10.05	15.48	20.46	0.43
[1,3]	9.46	9.46	4.21	12.49	0.38

TABLE 4: Distance measurements between the three time series illustrated in Figure 6

ED imposes a penalty for any misalignment. Customer 3's transaction, however, occurred within the same week as customer 1's second transaction. As a consequence, the distance between the two time series at that point in time is substantially smaller than when the transaction does not occur within the same week. Because the Euclidean distance is incorporated into the CID distance, the CID distance concludes, by the same reasoning as above, that the most comparable time series in the set are those of customers 1 and 3.

One possible solution to the problem of misalignments is to use Dynamic Time Warping. By matching peaks and valleys, DTW compresses or expands sections of one time series to better fit the other. In other words, DTW allows for non-linear alignment of observations and is hence insensitive to misaligned time series. As a result, the Dynamic Time Warping distance shows a much higher rate of similarity between customers 1 and 2 and a much lower rate between customers 1 and 3, and between customers 2 and 3.

The CORT distance is based on the temporal correlation coefficient computation. This coefficient determines whether the inaccuracy at a specific time stamp is correlated across its neighbours. As a result, the alignment of two time series is not as strict as considered by the Euclidean distance. However, there is only searched for matching peaks and/or valleys at the direct neighbour time stamps, rather than across the entire time period. As the transactions of customers 1 and 2 did not occur on the same time stamp or at their direct neighbours, the CORT coefficient recognises no similarity between the first two time series and equals zero. As a result, the CORT distance between the two time series is equal to the Euclidean distance by computation. Time series 1 and 3 are considered to contain more similar behaviour, again due to the fact that customer 3's transaction occurred within the same week as customer 1's second transaction.

The shape-based distance appears to be quite accurate in indicating two time series as being similar when only one transaction occurred in the time period, regardless whether the two order dates are misaligned or not. It thereby performs much better than the Euclidean distance. However, when more peaks/transactions arise, the performance of recognising two similar shapes of time series by SBD decreases. The reason behind this is that SBD examines global alignment. Cross-correlations are used to determine how much one time series must be shifted over time in order to match the other time series as closely as possible. Considering the fact that the time series is shifted as a whole, one peak will be properly aligned, but this does not have to be true for the other peaks, if any. This applies only if the total numbers of weeks between the matched transaction and the other transactions are equal. In practise, this is often not the case, resulting in the fact that the non-matched peaks get penalties for dissimilarity. In summary, when only one transaction occurs in both compared time series, the model can match peaks and concludes similarity quite accurately. With more than one transaction, the SBD loses its accuracy in defining corresponding time series.

5.2.2 Running times

The running time of the calculation is an important factor to consider when choosing a distance metric. Empirical observations led to the running times in minutes of the five distance metrics studied in this thesis, which are described in Table 5. These values are obtained by calculating distance matrices of the pageview time series of the four companies under consideration. Table 5 reveals that the running times are increasing in the same direction as the number of customers served per company. This is to be expected since every additional customer requires the calculation of the distances between the new customer and all the other existing customers. The total number of calculated distances per company is $N(N+1)/2$, where N represents the number of customers served by the company, and it is growing exponentially. For the first three companies, the running times of the Euclidean distance, the Dynamic Time Warping distance, and the shape-based distance are quite comparable. Yet, when we look at the company with the most customers served, we observe some differences. When there is a relatively high number of customers, the SBD appears to be the most time-efficient. The CORT and CID distances are measurements that require a substantial

	Company 1 (n = 1965)	Company 2 (n = 2291)	Company 3 (n = 3138)	Company 4 (n = 3719)
ED	11.9	17.9	31.7	107.9
DTW	11.7	18.1	36.7	88.7
CORT	53.7	117.9	293.7	815.1
CID	55.8	81.3	259.3	514.3
SBD	11.7	16.6	36.0	63.3

Table 5: Running times* in minutes of various distance metrics of pageview time series

*all calculations were run on an Intel Core i5 (2.4 GHz) processor with 16 RAM

amount of time to calculate. They can fairly increase as the number of customers served grows. This is an important consideration before incorporating one of these two distances into your customer segmentation algorithm. While these four companies have a relatively large number of customers, these numbers are only small in comparison to certain corporations that serve millions of customers.

5.3 Clustering methods

The resulting distance matrices of the five considered distance measurements for each dataset are, as the next step, given as input for the clustering algorithms. The three proposed clustering methods; hierarchical, spectral, and k-shape clustering, are used to group customers based on their behaviour similarities. For each combination of a distance matrix and a clustering method, the customers are divided into k numbers of clusters, where $k = [4, 5, 6, 7, 8]$ in this study. The dendrogram of the hierarchical clustering algorithm needs to be calculated once, from which the clusters belonging to the different numbers of k can be extracted. For spectral and k-shape clustering, the algorithms must be executed separately for every value of k.

5.3.1 Cluster partitions

To discover the effect of the different distance measurements on the three cluster algorithms, the total number of customers within each cluster is calculated for every dynamic customer segmentation combination. An overview of all the possible combinations of the distance metrics, cluster algorithms, and values of k for the four different companies can be found in Appendix A, Table 26 through Table 37. To give an interpretation of these tables, a sub-table can be found in Table 6. In this table, customers from company 3 are partitioned into eight clusters based on the pageview feature selection. The partitions of the different customer segmentation methods, shown in this table, reflect the overall pattern that can be found for all different feature selections for this company. The partitions obtained by the monetary value and pageview features can be found quite similar, where the CLV feature follows the same pattern, but in a more extreme way. For example, whereas one cluster in the pageview feature is rather large, the same cluster is even larger in the CLV feature selection. The same is true for small clusters in the reversed direction. The spectral clustering method, however, seems not suitable for the CLV feature, no matter what distance matrix is given as input for the clustering method. It results in almost all customers being grouped together in one cluster.

Table 6 shows that the resulting partitions are influenced by both the distance measurement and the clustering method. The combination of the CORT distance together with the hierarchical clustering method and the combination of the ED and CID distance together with the spectral clustering method, in general, seems to result in one large cluster of about 80% of the customers. The other customers are divided into smaller remaining clusters. The opposite holds for the three distances mentioned above when combined with the other clustering method. The combination of the CORT distance, now combined with the spectral clustering method, and the ED and CID distance together with the hierarchical clustering method, generates more evenly separated clusters in comparison with before, all of reasonable size. Despite the fact that one large cluster continues to exist, the percentage of customers within this cluster has decreased to about 60%. The shape-based distance combined with the k-shape and the hierarchical clustering methods, generates evenly separated clusters as well. It results in one rather large cluster and the remaining customers are grouped into smaller clusters. The DTW distance results in nearly all cases in the most even division of the customers into clusters for both the hierarchical and the spectral clustering method. Only for the CLV feature does this not apply when spectral clustering is combined with DTW. For the combination of spectral clustering with DTW for the other two feature selections, it seems that eight clusters are not necessary for cluster partition. The clusters from 6 until 8 are so small that they would only contain a handful of customers, such that a lower number for k would be more suitable. However, the performance metrics of the next section must prove these empirical observations.

The partitions for the other three companies show, in general, the same results as previously described. The most evenly spaced clusters are generated by the k-shape clustering method combined with the SBD, the hierarchical clustering method with the CID distance, and both the hierarchical and spectral clustering methods combined with the DTW distance. However, this does not hold when evaluating the monetary value feature for companies 2 and 4. It shows that the DTW and

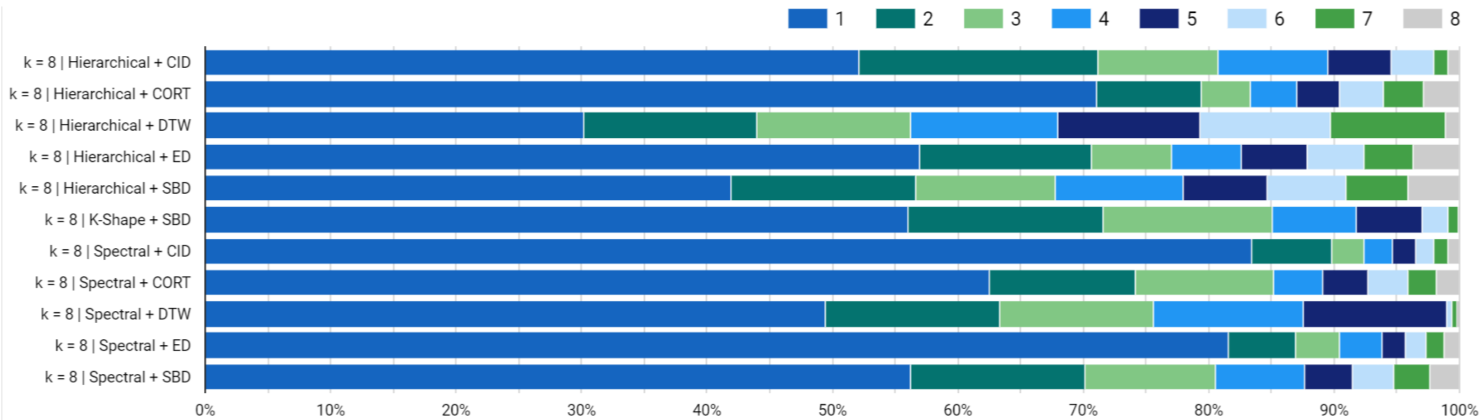


TABLE 6: Company 3's cluster partition considering the pageview feature for $k = 8$

SBD distances with both hierarchical and spectral clustering methods generate non-evenly separated clusters for this feature. These customer segmentation methods provide one fairly large group of around 90% of the customers. Overall, it can be observed that the CLV feature selection gives for all companies the most evenly separated clusters.

It is important to keep in mind is that more evenly separated clusters do not necessarily be more desirable. This is dependent on the considered dataset. If a large proportion of customers are one-time-buyers and made their purchases in about the same time period, it is likely that these customers are grouped together and that this cluster is larger compared to others.

5.4 Clustering performance

In this step, we compute the Silhouette and Davies-Bouldin indices to determine the best performing dynamic customer segmentation method. The indices are calculated of the various customer cluster techniques combined with diverse distance measurements and executed for different numbers of clusters. The performance indices of the online tool, Datatrics, are calculated as well and included in the comparison analysis. The distance between two time series in the SI and DBI is calculated by the use of all the distance measurements described in Section 3.2, except for the CORT distance. The CORT coefficients are observed to be zero very often, such that their corresponding CORT distances equal the ED as a consequence. To avoid overlapping conclusions, the CORT is not included as a distance measurement in the SI and DBI calculations. In this section, an in-depth analysis will be provided for company 2. All the results for the four different companies can be found in Appendix A, Tables 18 through 25, whereas the results of company 2 will be explained more extensively in this section. The overall patterns of the other companies, as well as their overlaps and differences with the results of company 2, will also be highlighted.

5.4.1 Performance results of company 2

The three different feature selections are performed on the data of company 2. The resulting Silhouette and Davies-Bouldin indices of the CLV time series can be found in Table 7 and Table 8, respectively. The highest number in the Silhouette table per value of k is stated in bold, and the overall maximum value for every distinctive distance calculation method used in the Silhouette indices is highlighted in red. The same holds for the DBI table but then applied to the minimum values. As the ED is incorporated in the CID distance, and the results show the same patterns and preferred cluster partitions for these two distances, only the ED is shown in the tables. Furthermore, the values of $k = 6$ and $k = 7$ are also omitted as they are not the preferred number of clusters in general. Since the online tool processed divisions into five clusters, only the performance indices of Datatrics for $k = 5$ are given.

By looking at the Silhouette indices of all three feature selections in Table 7, 9 and 11, we observe that not the same conclusions are made by the different calculations of distance. It seems likely that there is a relationship between the highest performance indices and the distance calculation method used in the Silhouette index calculation. This remains true for all three considered selected features. For $d(x_i, x_j) =$ DTW and SBD, the highest indices are obtained by combining the hierarchical and spectral clustering methods with the DTW and SBD distances, respectively. The hierarchical and spectral clustering methods, combined with either the ED or the CORT distance, yield the highest values for $d(x_i, x_j) =$ ED. The CID distance gives values close to the highest, or is sometimes even the highest. It is worth noting that the values for the three different feature selections where $d(x_i, x_j)$ is equal to DTW and SBD are very similar for all combinations of the hierarchical and spectral clustering methods in combination with the DTW and SBD distances.

When evaluating the number of clusters, the value of $k = 4$ is preferred when considering CLV time series. For the other time series, the preference is higher by $k = 8$. When using the $d(x_i, x_j) =$ CID, which is not included in the tables, the value of $k = 4$ is preferred, which is the only substantial difference with the Silhouette indices when $d(x_i, x_j)$ is equal to ED.

Another thing to notice is that the performance indices of the CLV time series are all relatively close to each other, disregarding the indices of Datatrics. The Silhouette indices for the monetary value time series, however, vary highly between the range of -1 and 1. While some values of $d(x_i, x_j)$ equal to DTW and SBD are very close to 1, with a maximum value of 0.96, other values are extremely

	$d(x_i, x_j)$ = ED			$d(x_i, x_j)$ = DTW			$d(x_i, x_j)$ = SBD		
	k=4	k=5	k=8	k=4	k=5	k=8	k=4	k=5	k=8
Hierarchical + ED	0.40	0.43	0.42	0.33	0.36	0.37	0.55	0.53	0.52
Hierarchical + DTW	0.29	0.35	0.34	0.51	0.46	0.42	0.55	0.53	0.49
Hierarchical + CORT	**0.61**	**0.60**	**0.59**	0.48	0.44	0.41	0.55	0.52	0.41
Hierarchical + CID	0.57	0.42	0.40	0.39	0.40	0.42	0.48	0.50	0.48
Hierarchical + SBD	0.15	0.30	0.32	0.47	0.39	0.42	0.54	0.57	**0.55**
Spectral + ED	0.37	0.36	0.44	0.51	0.47	0.36	0.55	0.54	0.54
Spectral + DTW	0.35	0.29	0.27	**0.55**	**0.51**	**0.45**	0.47	0.43	0.34
Spectral + CORT	0.59	0.59	0.58	0.30	0.24	0.22	0.42	0.38	0.29
Spectral + CID	0.37	0.36	0.45	0.51	0.46	0.37	0.55	0.54	0.55
Spectral + SBD	0.40	0.37	0.32	0.50	0.47	0.37	0.57	0.55	0.54
K-Shape + SBD	0.41	0.33	0.30	0.42	0.41	0.41	**0.61**	**0.58**	0.49
Datatrics		-0.18			-0.42			-0.19	

TABLE 7: Silhouette indices considering the CLV feature selection of company 2

	$d(x_i, x_j)$ = ED			$d(x_i, x_j)$ = DTW			$d(x_i, x_j)$ = SBD		
	k=4	k=5	k=8	k=4	k=5	k=8	k=4	k=5	k=8
Hierarchical + ED	1.34	1.40	1.25	2.01	4.94	3.80	##	##	##
Hierarchical + DTW	1.55	1.72	1.58	1.81	**2.02**	4.40	##	##	##
Hierarchical + CORT	**1.23**	**1.27**	1.40	**1.48**	2.03	**2.24**	##	##	##
Hierarchical + CID	1.53	1.38	1.25	2.53	2.19	3.89	##	##	##
Hierarchical + SBD	3.51	3.33	2.73	3.51	3.36	3.86	##	3.59	**2.33**
Spectral + ED	1.36	1.29	1.29	1.69	2.69	5.26	**1.55**	**1.97**	##
Spectral + DTW	1.50	1.55	1.53	1.52	2.04	2.55	##	##	##
Spectral + CORT	2.77	2.43	2.30	4.56	5.15	5.99	##	##	##
Spectral + CID	1.42	1.46	**1.24**	3.09	3.78	5.91	2.28	##	##
Spectral + SBD	1.50	1.67	1.79	2.00	2.72	4.83	##	##	##
K-Shape + SBD	1.48	1.58	1.57	2.32	2.18	4.45	1.60	##	##
Datatrics		12.15			6.53			23.30	

TABLE 8: Davies-Bouldin indices considering the CLV feature selection of company 2

low, with a minimum value of -0.92. Low values are frequently obtained when using the distance measurements of ED, CID, and CORT. The indices for the pageview time series follow the same pattern as the monetary indices but in a less extreme way, having a minimum value of -0.61 and a maximum value of 0.72.

If we evaluate the different clustering methods, we can observe that for the CLV and monetary

	$d(x_i, x_j)$ = ED			$d(x_i, x_j)$ = DTW			$d(x_i, x_j)$ = SBD		
	k=4	k=5	k=8	k=4	k=5	k=8	k=4	k=5	k=8
Hierarchical + ED	**0.07**	**0.08**	**0.13**	-0.90	-0.89	-0.87	-0.70	-0.73	-0.66
Hierarchical + DTW	-0.01	-0.02	-0.03	**0.96**	**0.96**	0.96	0.90	0.83	0.83
Hierarchical + CORT	0.06	0.08	0.13	-0.92	-0.92	-0.92	-0.76	-0.76	-0.72
Hierarchical + CID	0.04	0.06	0.11	-0.91	-0.90	-0.87	-0.33	-0.34	-0.48
Hierarchical + SBD	-0.01	-0.02	-0.03	0.96	0.96	0.96	0.93	0.93	0.86
Spectral + ED	**0.07**	**0.08**	**0.13**	-0.92	-0.91	-0.89	-0.74	-0.76	-0.73
Spectral + DTW	-0.01	-0.02	-0.04	0.95	0.96	**0.96**	0.94	0.93	0.92
Spectral + CORT	0.00	0.00	-0.01	-0.12	-0.18	-0.35	-0.18	-0.19	-0.28
Spectral + CID	**0.07**	**0.08**	**0.13**	-0.92	-0.91	-0.89	-0.74	-0.76	-0.73
Spectral + SBD	-0.02	-0.03	-0.05	0.92	0.92	0.91	**0.95**	**0.95**	**0.94**
K-Shape + SBD	0.02	0.03	0.04	0.09	-0.14	-0.27	0.26	0.07	-0.07
Datatrics		-0.05			-0.91			-0.84	

TABLE 9: Silhouette indices considering the monetary value feature selection of company 2

	$d(x_i, x_j)$ = ED			$d(x_i, x_j)$ = DTW			$d(x_i, x_j)$ = SBD		
	k=4	k=5	k=8	k=4	k=5	k=8	k=4	k=5	k=8
Hierarchical + ED	**0.99**	**0.99**	**0.99**	56.26	46.22	43.11	1.71	**1.57**	2.06
Hierarchical + DTW	11.52	10.16	7.43	15.20	13.59	11.11	1.93	1.89	**1.84**
Hierarchical + CORT	2.58	2.28	1.71	12.96	**10.92**	**10.10**	##	##	##
Hierarchical + CID	9.98	7.99	5.37	16.45	54.78	48.57	**1.66**	1.99	2.34
Hierarchical + SBD	10.99	9.27	7.17	14.97	12.78	11.25	2.13	1.99	1.88
Spectral + ED	1.00	1.00	1.00	53.67	43.63	28.76	12.69	##	13.76
Spectral + DTW	8.00	8.03	6.73	**11.55**	12.93	11.17	##	##	##
Spectral + CORT	1.69	1.55	1.35	54.15	44.08	29.28	##	##	##
Spectral + CID	1.00	1.00	1.01	14.56	17.24	20.95	12.69	##	##
Spectral + SBD	2.86	2.74	2.46	29.95	111.45	75.64	##	##	##
K-Shape + SBD	6.28	6.01	5.60	23.32	23.14	27.56	##	##	##
Datatrics		12.63			2.00			3.77	

TABLE 10: Davies-Bouldin indices considering the monetary value feature selection of company 2

value time series, the overall outperforming method is the hierarchical clustering method. The spectral clustering method is preferred more by the Silhouette index in the other feature selections. However, the differences between the two clustering methods are very small, considering that both are with the same distance matrix obtained. Using the k-shape clustering algorithm did not show convincing results. The performance indices were (one of) the highest only in the case of the CLV

	$d(x_i, x_j)$ = ED			$d(x_i, x_j)$ = DTW			$d(x_i, x_j)$ = SBD		
	k=4	k=5	k=8	k=4	k=5	k=8	k=4	k=5	k=8
Hierarchical + ED	**0.05**	**0.06**	**0.10**	-0.09	-0.23	-0.24	-0.12	-0.31	-0.30
Hierarchical + DTW	0.00	0.00	-0.01	0.59	0.60	0.60	0.45	0.42	0.38
Hierarchical + CORT	0.05	0.06	0.09	-0.18	-0.19	-0.28	-0.23	-0.24	-0.36
Hierarchical + CID	0.00	0.01	0.04	0.63	0.39	-0.21	0.61	0.38	-0.38
Hierarchical + SBD	0.00	0.00	-0.01	0.63	0.64	0.63	0.60	0.63	0.61
Spectral + ED	**0.05**	**0.06**	0.09	-0.46	-0.45	-0.45	-0.61	-0.61	-0.61
Spectral + DTW	0.00	0.00	-0.01	**0.67**	**0.69**	**0.69**	0.64	0.67	0.64
Spectral + CORT	0.04	0.06	0.08	-0.08	-0.08	-0.20	-0.15	-0.13	-0.17
Spectral + CID	0.05	0.06	0.09	-0.34	-0.34	-0.45	-0.45	-0.45	-0.60
Spectral + SBD	-0.01	-0.01	-0.02	0.57	0.58	0.54	**0.69**	**0.69**	**0.72**
K-Shape + SBD	0.00	0.00	-0.01	0.06	-0.01	0.00	0.18	0.08	0.09
Datatrics		-0.04			-0.65			-0.71	

TABLE 11: Silhouette indices considering the pageview feature selection of company 2

	$d(x_i, x_j)$ = ED			$d(x_i, x_j)$ = DTW			$d(x_i, x_j)$ = SBD		
	k=4	k=5	k=8	k=4	k=5	k=8	k=4	k=5	k=8
Hierarchical + ED	2.34	2.11	2.12	9.28	9.31	18.27	7.47	11.59	##
Hierarchical + DTW	31.00	26.97	20.83	52.07	43.88	32.40	##	##	##
Hierarchical + CORT	2.21	1.76	1.32	5.20	4.59	6.88	##	##	##
Hierarchical + CID	23.22	19.82	10.78	36.54	30.19	20.61	##	##	##
Hierarchical + SBD	24.44	22.37	17.75	37.65	33.88	26.76	##	##	##
Spectral + ED	2.03	2.09	2.26	**3.44**	**3.99**	**5.84**	##	##	##
Spectral + DTW	25.44	22.42	18.64	36.34	32.15	29.02	4.34	##	##
Spectral + CORT	2.50	2.58	2.53	4.03	8.32	6.43	##	##	##
Spectral + CID	**1.17**	**1.17**	**1.15**	7.02	7.72	7.96	##	##	##
Spectral + SBD	20.12	18.20	11.98	27.71	25.36	17.53	##	**3.46**	##
K-Shape + SBD	16.45	16.46	13.00	31.27	34.29	31.25	**2.97**	##	##
Datatrics		12.04			4.95			##	

TABLE 12: Davies-Bouldin indices considering the pageview feature selection of company 2

feature selection and $d(x_i, x_j)$ = SBD.

By looking at the Davies-Bouldin indices for all the three feature selections in Table 8, 10 and 12, we observe that, in contrast with before, the same conclusions can be drawn by executing different calculations of distances. It seems that the DBI scores are overall unaffected by the distance calculation used. The performance indices of the CLV time series are all relatively low for the hierarchical clustering method combined with the CORT distance and the spectral clustering method combined with the Euclidean distance. The hierarchical clustering method combined with the CORT distance also results in low values for the monetary and pageview time series. This holds even more for the spectral clustering method combined with the CID distance. After observing the accurate ability of the DTW distance to determine similarity between two misaligned time series in Section 5.2.1, it was expected to result in good cluster partitions as well. However, the results are not outperforming the other distances. This is not even the case when the DTW was incorporated in the performance measurements. Another contrast with before is that, when evaluating the number of clusters, the value of $k = 4$ is now the overall preferred number of clusters. Corresponding with the Silhouette indices, the Davies-Bouldin performance indices of the CLV time series are all relatively close to each other, disregarding the indices of Datatrics. The values for the monetary value and pageview time series, however, have a much wider range of appearance.

As can be observed in the DBI performance tables with the SBD as the incorporated distance calculation, a large ratio of cells are equal to ##. This is substituted with the true indices, because they are of very high value. As the value of the SBD ranges between the interval [0,2], it could occur that the SBD is not yet equal, but very close to zero. Since the DBI is a sum of ratios, where the dividend can range between zero and four, and the divisor is equal to the distance between two time series, it could happen that the ratio is exploding. This happens when the divisor is very close to zero, resulting in a very high value for the DBI. Because more values are exploded as non-exploded values, it seems not meaningful to take the results of the DBI with SBD as its distance calculation into consideration when determining outperforming methods.

Even if it is stated that a better-defined cluster has a lower DBI and a higher SI, comparing all the indices of both performance measurements does not appear to be meaningful. The Silhouette index provides evidence to believe that the type of distance calculations has a substantial impact on the results, such that we could not achieve an unambiguous conclusion per feature selection. In contrast, the DBI performance measurements appear to be unaffected by the distance measurement utilised in their calculation. As a result, the observed outperforming method using this performance metric is more substantiated. As previously reported, the DBI scores reveal that the hierarchical clustering algorithm combined with the CORT distance is the best method based on the CLV feature selection, and the spectral clustering method combined with the CID distance is the best performing for the monetary value and pageview feature selection. The recommended number of clusters for the CLV and pageview time series with the outperforming methods equals four, whereas a number of eight clusters is preferable when evaluating monetary value time series. This is likely caused by the fact that the CLV and pageview time series contain far more non-zero time stamps than the monetary value time series.

The observed sensitivity of the Silhouette index can be explained by its own computation and the strategies of the clustering methods. Both algorithms use the same distances between all pairs of time series. The resulting partitions in clustering methods are obtained by grouping customers with the shortest distances between their time series. Even though the various clustering methods utilise different approaches, the objective is roughly the same. As distances can be determined in a variety of ways, the resulting cluster divisions differ depending on the method of distance calculation used. This was also observed in Section 5.3.1. In summary, clusters are formed by minimising the distances within the clusters and are influenced by the distance measurement. For the SI, partitions where the distances between customers' time series within its assigned cluster are low and the distances to all of the customers' time series in the other cluster are high, are given a higher SI. This can be translated to the same objective as the clustering methods. That is compactness within the clusters and the separation between the clusters. Therefore, it is expected that the SI to be minimised by using the same distance measurement in its calculation as the distance measurement used by the clustering method. In contrast, the DBI does not include distances between all pairs of customers' time series. It considers the centroid diameters of all the clusters as well as the distances between the k centroids. The centroid time series were formed within the DBI calculation and were not used in the clustering algorithms. Although the k-shape takes centroids into account, its calculation differs from that of the DBI because it does not use the DTW barycenter averaging algorithm. As a result, new distances must be calculated in the DBI calculations that were not used to generate clusters in the clustering methods. Therefore, the DBI appears to be less sensitive for the distance measurement in its computation, and it can be seen that, overall, the same conclusions can be drawn from the indices by performing different distance calculations.

Despite the fact that the Silhouette index appears to generate distance-sensitive results, the SI can nevertheless provide useful information. In order for the SI to provide meaningful results, a proper distance measurement must be selected to incorporate into the SI calculation. While a distance selection is not required for static customer segmentation performance evaluation, because earlier studies indicated that the used distance calculation had generally no effect on SI performance results, it appears to be required for dynamic customer segmentation performance evaluation. With this distance selection, you indicate the type of similarity detection you consider relevant for distinguishing samples. We discovered in this thesis that the DTW distance worked best in finding similarity between customers' time series. The exact timing of a customer's website engagement is regarded as less relevant than the general pattern and quantity of website engagement. As a result, we chose to evaluate the SI performance results with only the DTW distance utilised in their calculation. With these results, in combination with the DBI results, we can now detect which dynamic customer segmentation methods have a higher SI and a lower DBI. Similar to the SI, only the performance results of the DBI with the DTW distance utilised in its computation are considered. As a result, the outperforming methods for company 2 are the hierarchical clustering method combined with the CORT distance and $k = 4$ for the CLV feature, the spectral clustering method combined with the DTW distance and $k = 4$ for the monetary value feature, and the spectral clustering method combined with the SBD distance and $k = 8$ for the pageview feature.

5.4.2 Performance results of the online tool

As no documentation is provided about the customer segmentation method of Datatrics, we can only guess what variables are included as input for their clustering methods. Although it cannot be guaranteed, it seems likely that they used any of the features mentioned in this thesis because these are the most important purchase-related variables. As a result, the performance measurements of the online tool's customer segmentation method are calculated as if they used one of the three features chosen for this thesis. Although the performance results for Datatrics in Tables 7 through 12 cannot be guaranteed to be factual, they do provide evidence that the clusters are not properly divided. Regardless what distance calculation method is used in the SI and DBI, the performance measurements are relatively far away from the highlighted best value in each table for all three feature selections. Furthermore, when evaluating the various component combinations of customer segmentation, the division of customers into five clusters is rarely the preferred cluster number. This also holds for the other evaluated companies. Although we cannot be certain, Datatrics' customer segmentation is likely a poor performing customer segmentation solution and can be outperformed by executing one of methods provided in this thesis.

5.4.3 Performance results of all companies

The patterns discovered for company 2, in general, can also be found for the other companies. The Silhouette index appears to be sensitive to the distance measurement applied in its calculation, whereas the Davies-Bouldin index appears to be less sensitive to this matter. As a result,

	CLV	Monetary Value	Pageview
Company 1	Hierarchical + DTW ($k = 4$)	Spectral + CID ($k = 8$)	Spectral + CID ($k = 8$)
Company 2	Hierarchical + CORT ($k = 4$)	Spectral + CID ($k = 4$)	Spectral + CID ($k = 4$)
Company 3	Hierarchical + DTW / k-shape ($k = 4$)	Spectral + CID ($k = 8$)	Spectral + CID ($k = 8$)
Company 4	Hierarchical + CORT ($k = 4$)	Spectral + CID ($k = 4$)	Spectral + CID ($k = 4$)

TABLE 13: Davies-Bouldin's outperforming methods

the attention shifts to the Davies-Bouldin results which provide more meaningful results. The DBI obtained with the SBD are disregarded in the evaluation, since most of its values are exploded for all companies. Table 13 summarises the outperforming methods based on the DBI scores of the four companies for the three feature selections. It is noticeable that the hierarchical clustering algorithm is preferred when selecting the CLV feature, while the spectral clustering mixed with the CID distance is preferred when selecting monetary value and pageview features. The outperforming results of the CID distance are in line with the results obtained in the study of Abbasimehr and Bahrini (2021). The CORT distance outperforms in the CLV feature selection for companies with a high ratio of one-time-buyers, whereas the DTW distance outperforms for datasets with substantially more frequent buyers. This disparity in distance selection can be motivated by the fact that the CORT distance can determine relatively accurate similarities between time series with only one peak/transaction. With several transactions, the CORT distance loses its precision in defining related time series and is replaced by the better performing DTW distance.

Another difference is observed when selecting the monetary value and pageview features. Considering these feature selections, customers of companies with a high proportion of one-time-buyers can be better divided into four clusters, whereas eight clusters are preferred for companies with relatively more frequent buyers. This is likely because customers who buy more than once from companies with a high ratio of one-time-buyers are all grouped together since their behaviour differ substantially from that of the average customer. The remaining customers, the one-time-buyers, are then divided into groups based on the timing and value of their transaction. This differentiation appears to require no more than three categories. In contrast, for companies with a lower percentage of one-time-buyers, more clusters are necessary. Aside from distinguishing transaction timing and value from one-time-buyers, customers who made multiple purchases can also be distinguished. As this is now a larger group, a bigger number of k is required. The k-shape clustering method, which is always executed with the SBD, outperforms for company 3 in the CLV feature selection.

	CLV	Monetary Value	Pageview
Company 1	Hierarchical + DTW ($k = 4$)	Spectral + SBD ($k = 8$)	Spectral + SBD ($k = 8$)
Company 2	Hierarchical + CORT ($k = 4$)	Spectral + DTW ($k = 4$)	Spectral + SBD ($k = 8$)
Company 3	Hierarchical + DTW ($k = 4$)	Spectral + DTW ($k = 4$)	Spectral + DTW ($k = 4$)
Company 4	Hierarchical + CORT ($k = 4$)	Spectral + SBD ($k = 8$)	Spectral + SBD ($k = 8$)

Table 14: The DTW incorporated Davies-Bouldin and Silhouette's outperforming methods

This clustering algorithm appears to function best when considering a dataset consisting mainly of frequent buyers and non-zero data points.

As stated previously, in order for the SI to provide meaningful results for dynamic customer segmentation, a proper distance measurement must be selected to be incorporated into the SI calculation. In this thesis, we chose to evaluate the two performance measurements with the DTW distance incorporated in their calculation. Table 14 summarises the outperforming methods with a lower DBI and a higher SI. Except for the k-shape clustering method, the results for the CLV feature selection in this table are the same as in Table 13. The spectral clustering method is still preferred for the monetary value and pageview feature selections, but the distance metrics have been modified to either DTW or SBD. The DTW distance prefers $k = 4$, whereas the SBD distance prefers $k = 8$. As mentioned in Section 5.3.1 and shown in the tables in Appendix A, the spectral clustering approach generates one fairly large group and several smaller groups. An exception applies in the case when combined with the CID distance and, in some scenarios, in combination with the SBD. The larger the number of k, the smaller the clusters besides the large cluster become. This is also true for the DTW distance. For a k larger than four, it seems less likely that certain small clusters must exist next to each other.

One result worth mentioning in the pageview feature of Table 14 is that for every company, the spectral clustering method combined with the SBD is outperforming other customer segmentation methods, except for company 3. Customers of company 3 are visiting the website more often, and with a higher average number of pageviews as well, compared to customers of the other companies. The DTW distance outperforms for datasets with substantially more frequent buyers.

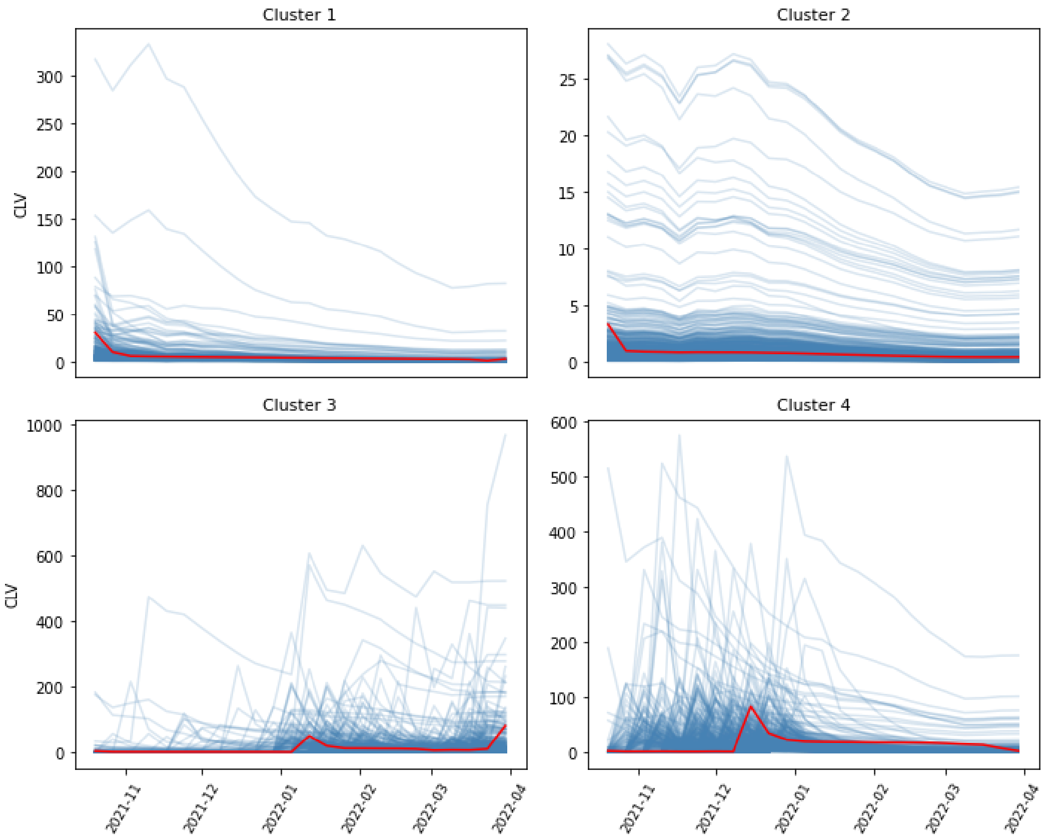
Overall, it does not seem that the number of customers per company is affecting the preferred customer segmentation approach. In addition, the average value of the transactions does not seem to be of influence, as company 2 has a much higher average order value, but no notable results for this matter can be found.

A pattern can be discovered for all four companies and for both the SI and DBI scores. This pattern relates to the fact that the performance scores of the CLV time series are all relatively close to each other and of reasonable value. The values for the monetary value and pageview time series, however, have a much wider range of appearance and fluctuate highly between the least and best performing segmentation methods. It is preferable to avoid having a poor cluster partition as a result of subconsciously selecting a less suitable distance matrix. This could occur when selecting monetary value or pageview time series for customer distinction, as the least performing dynamic segmentation is performing quite poorly. In contrast, the least performing method considering CLV time series for customer distinction is not performing so badly. Furthermore, the CLV time series contains more customer information than the monetary value time series. It is calculated as a function that includes monetary value in addition to other RFMT variables. Another reason to select the CLV feature over the other two features is that CLV is an estimation. The CLV estimates the total profit that can be expected from a customer within a year. It would be of more interest to develop CRM strategies based on future customer behaviours to be able to respond to the needs of customers. As a result, this thesis concludes that the CLV time series is the better overall choice for dynamic customer segmentation.

5.5 Cluster characteristics

After determining the best performing dynamic segmentation methods and the number of clusters, the final step is to translate the generated clusters into useful information for marketers. They can use this information to develop CRM strategies. The centroid of each cluster is determined to acquire the characteristics of its members. Each centroid is computed using the DTW barycenter averaging algorithm. When observing company 1's CLV time series, the best performing segmentation method of Table 13 is the hierarchical clustering method combined with the DTW distance by dividing customers into four clusters. The non-normalized original time series of the members of these four clusters are shown in Figure 7, where the red line denotes the centroid of each cluster. The RFM variables are listed in Table 15 providing additional information about the clusters. We can observe from these two cluster representations that the first cluster contains mainly customers that made one transaction at the beginning of the time frame, with a mean transaction value of €50. The second cluster includes clients who performed multiple low-value transactions and were more engaged at the start of the time frame than at the end. The third cluster, in contrast, consists of customers who have been lately active. The average transaction value is roughly €85, although

FIGURE 7: Customer Lifetime Value time series partition with hierarchical clustering method combined with the DTW distance for company 1



Cluster	Recency	[min, max]	Frequency	[min, max]	Monetary	[min, max]
1	11.0	[0,248]	1.2	[1,3]	€45.37	[€8.10, €440.59]
2	29.2	[0,342]	2.3	[1,8]	€11.76	[€1.90, €75.58]
3	22.9	[0,168]	1.8	[1,4]	€85.76	[€10.95, €885.58]
4	4.8	[0,202]	1.4	[1,3]	€89.46	[€15.76, €624.73]

TABLE 15: Average RFM variables of cluster partition obtained by the hierarchical clustering method combined with the DTW distance for company 1

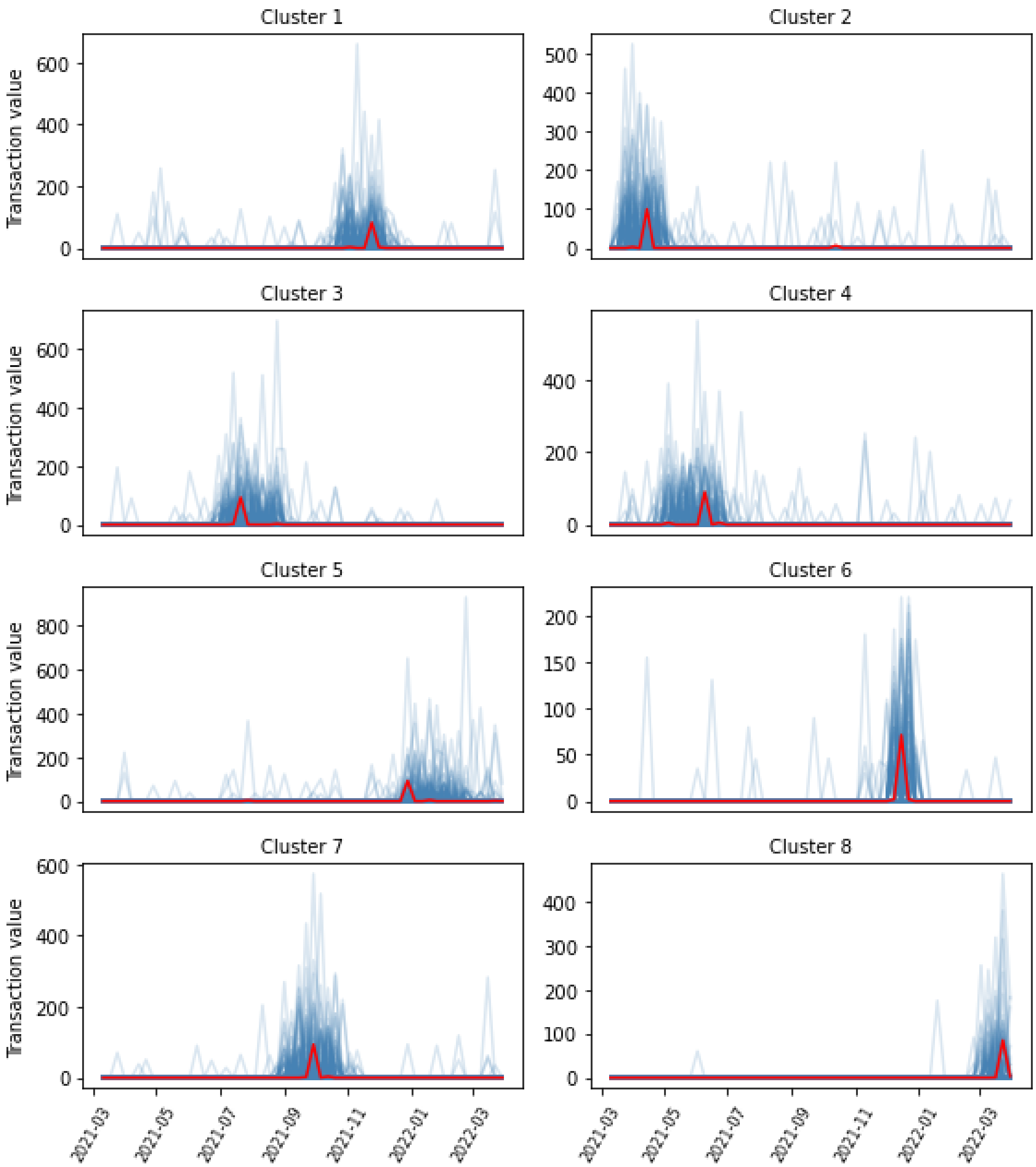
some customers spend much more, with a maximum of €885.58. The fourth cluster contains a combination of one-time-buyers and repeat customers. Their activity is largely concentrated in the middle of the time frame, and the average transaction value is higher than the total average, with some outliers reaching €600. In conclusion, the resulting clusters are all distinctive from each other considering all the RFM variables.

To display the best-performing segmentation method of Table 14, where the recommended number of clusters is eight, we consider the monetary value time series, yet again from company 1. This cluster partition is obtained using the spectral clustering method in combination with the CID distance, as shown in Figure 8. The related RFM variables are listed in Table 16. In this case, we can observe that the timing of transactions was the most important element in assigning customers to different clusters. Another considered element is the monetary value of each transaction. The number of transactions is not properly regarded in this partition, since the average value of frequency is very much the same for every cluster. As previously stated, this is due to the inability of the CID distance to detect similarities between non-aligned time series with multiple peaks.

When we evaluate the best performing segmentation method in Table 14 for company 1's monetary value feature, we obtain the partition shown in Figure 9 and the corresponding RFM variables in Table 17. This partition is achieved by combining the spectral clustering method with the SBD. In this case, the frequency is regarded as more essential than before. As a result, practically all one-time-buyers are grouped together, while the frequent buyers are separated into the other seven groups based on the number of transactions, transaction timing, and transaction values. Although it may be argued to merge certain clusters, since they are rather small, the performance measurements show that a partition of eight clusters is preferred over a smaller number.

In comparison to the partition in Figure 8, the cluster partition in Figure 9 are more useful to develop cluster-specific marketing strategies. When transaction frequency is considered as relevant as transaction timing and transaction value, the cluster characteristics are more distinguishable from one another. As a result, it enables to relate different clusters to customer types with the latter partition. Cluster 5's customers could, for example, be defined as loyal customers. This is due to the fact that they purchased the most and also on a regular basis. Customer types are a common phenomenon that marketers take into account when targeting specific customer groups. Different sets of names for customer types are used, but in all cases, groups are related to customer types using distinctive group characteristics. By using the DTW and SBD distances, all RFM variables are considered, rather than giving frequency a lower priority as the partition with the CID distance does.

Figure 8: Monetary value time series partition with spectral clustering method combined with the CID distance for company 1



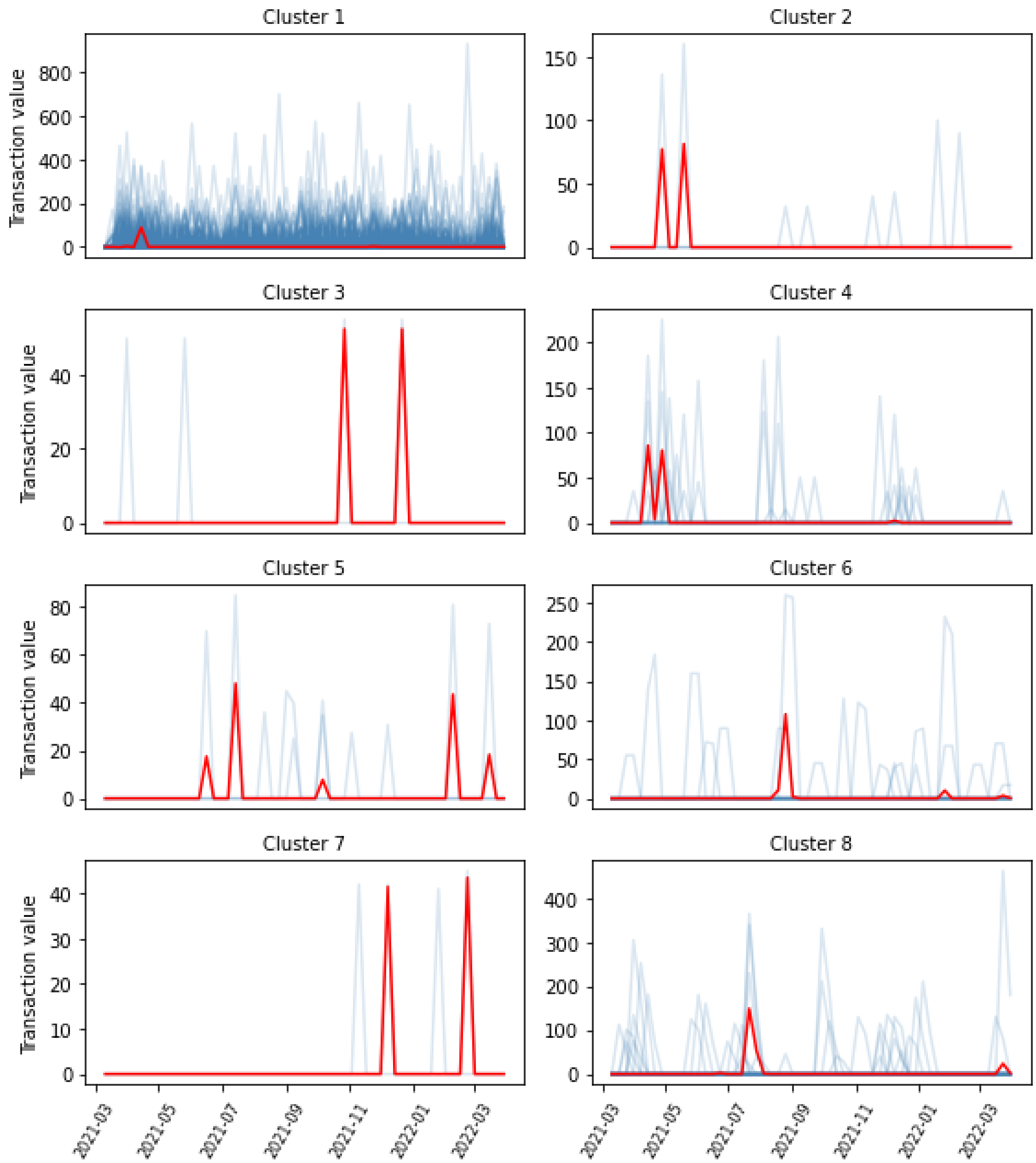
Cluster	Recency	[min, max]	Frequency	[min, max]	Monetary	[min, max]
1	9.3	[0,281]	1.18	[1,4]	€75.28	[€12.10, €619.60]
2	15.5	[0,342]	1.20	[1,4]	€92.72	[€11.90, €514.30]
3	6.0	[0,184]	1.16	[1,4]	€85.67	[€3.80, €640.59]
4	11.0	[0,304]	1.23	[1,8]	€81.69	[€5.90, €490.60]
5	10.1	[0,291]	1.24	[1,8]	€79.85	[€11.90, €885.58]
6	6.5	[0,192]	1.15	[1,8]	€66.70	[€14.90, €219.85]
7	8.1	[0,242]	1.16	[1,4]	€87.59	[€1.95, €573.30]
8	3.3	[0,76]	1.10	[1,2]	€80.12	[€14.90, €520.78]

Table 16: Average RFM variables of cluster partition obtained by the spectral clustering method combined with the CID distance for company 1

Cluster	Recency	[min, max]	Frequency	[min, max]	Monetary	[min, max]
1	9.7	[0,342]	1.20	[1,3]	€81.58	[€1.95, €885.58]
2	21.3	[17,22]	1.20	[2,2]	€79.04	[€31.95, €167.83]
3	56.5	[55,56]	1.19	[2,2]	€52.43	[€49.9, €54.95]
4	28.9	[10,192]	1.21	[2,4]	€79.92	[€14.9, €222.76]
5	92.5	[28,271]	1.15	[2,8]	€42.86	[€30.43, €87.92]
6	14.5	[1,127]	1.17	[2,4]	€89.78	[€16.90, €260.96]
7	30.5	[27,32]	1.17	[2,2]	€42.40	[€41.90, €47.90]
8	16.5	[1,220]	1.17	[2,5]	€117.11	[€29.95, €480.76]

Table 17: Average RFM variables of cluster partition obtained by the spectral clustering method combined with the SBD distance for company 1

Figure 9: Monetary value time series partition with spectral clustering method combined with the SBD distance for company 1



6 Conclusion

The objective of this study was to explore relevant dynamic customer segmentation methods across E-commerce business contexts. Customer segmentation is an approach for grouping customers with similar customer behaviours such that CRM strategies can be built. For this thesis, customer data has been collected from four different E-commerce websites. Data has also been collected from an online tool providing customer segmentation, whose underlying methods are black boxes. The disadvantages of black box segmentation processes include the lack of user control over the development process, and the absence of model transparency. Research was needed to analyse the online tool's validity, as well as whether its resulting segments could be outperformed by other dynamic customer segmentation methods.

Several research questions needed to be answered in order to provide a sufficient review of dynamic customer segmentation methods. To answer the first research question, a literature study was conducted to identify the various components of dynamic customer segmentation methods. We found that the first component includes feature selection. In contrast to non-dynamic segmentation methods, clustering in time series format only allows for the selection of one feature for customer differentiation. This can be seen as a limitation. In this thesis, three features for customer differentiation were selected. The second component is to choose a distance measurement to quantify the similarity between time series of different customers. Multiple distance metrics were considered, each having its own approach of detecting similarity. Dynamic customer segmentation's third component consists of defining the number of segments you want to obtain, ranging in this thesis from four to eight clusters. The last component is applying clustering methods such that different customer segments are obtained. This thesis compared three widely applied clustering methods. The clusters' centroids were afterwards extracted to give insights into the behaviour of customers inside each cluster. CRM strategies can be built from this information.

To answer the second research question, a sequential approach is executed for selecting the dynamic segmentation's components to discover outperforming combinations. For cluster performance evaluation, the main attention is shifted to the Davies-Bouldin Index (DBI), as the Silhouette Index (SI) appears to be sensitive to the distance measurement applied in its calculation. However, the results obtained with the SBD in the DBI calculation were concluded to be non-usable. For all companies, the DBI showed that the hierarchical clustering approach with $k = 4$ for the CLV feature selection is preferred, while the spectral clustering method combined with the CID distance is preferred when selecting the monetary value and pageview features. In order for the SI to provide meaningful results for dynamic customer segmentation, a proper distance measurement must be selected to incorporate in the SI calculation. With this distance selection, you indicate the type of similarity detection you consider relevant for distinguishing your samples. The inclusion of the results of the SI made us to evaluate the two performance measurements with the DTW distance incorporated. The outperforming clustering methods all remain the same as when considering the DBI scores only, however, the distance metrics for the monetary value and pageview features have been modified to either DTW or SBD. The DTW distance prefers customer division into four clusters, whereas the SBD distance prefers eight clusters.

After extracting the centroids of the clusters obtained by clustering methods combined with the CID, DTW, and SBD distances, we concluded that the clusters obtained with the latter two distances are more useful for CRM strategies. For obtaining these clusters with the latter two distances, all RFM variables have been considered, where the CID distance is given the variable frequency a lower importance.

As no documentation is provided for the online tool's customer segmentation method, its performance measurements are calculated as if one of the features chosen for this thesis were selected. The performance measurements are relatively far away from this thesis' outperforming methods. Moreover, the preferred number of clusters is rarely the cluster number that the online tool provides. Therefore, it is likely that the online tool is a poor performing customer segmentation solution and can be outperformed by executing one of the methods provided in this thesis.

The segmentation results of the CLV feature selection for both performance measurements are relatively close to each other and of reasonable value for all four companies. Furthermore, the CLV time series contains more customer information than the other two features and has predicting abilities. As a result, this thesis concludes that the CLV time series is overall the preferred feature selection for dynamic customer segmentation.

The characteristics of the different datasets must be considered to answer the third research question. For datasets containing customers with several transactions, the CORT distance loses its precision in defining similar time series for the CLV feature selection and is replaced by the better performing DTW distance. For the same feature, the k-shape clustering algorithm performs even better for datasets consisting mainly of frequent buyers. For the monetary value and pageview feature selection, the spectral clustering method is preferred combined with the CID distance when considering the DBI results. For companies with a high proportion of one-time-buyers, the segmentation appears to require no more than four clusters, whereas datasets with relatively more frequent buyers prefer eight clusters. Companies serving customers who visit the website relatively often with a high average number of pageviews per visit, must choose the spectral clustering method combined with the DTW distance instead of the SBD for the pageview feature selection. This would be the case when both the SI and DBI scores are considered. Overall, it does not seem that the number of customers per company or the average value of the transactions is affecting the preferred dynamic customer segmentation approach. It appears that the number of customers is only affecting the calculation time. In particular, the CORT and CID distance measurements require a substantial amount of time to calculate as the number of customers grows.

7 Discussion

This research has several limitations and suggestions for further research. At first, our research focuses on analysing the clusters based on four different companies, where only one company makes use of the considered online segmentation tool. The conclusions that have been drawn from this study would reach more validity if more companies were considered, as well as more companies using Datatrics, in order to compare if the same conclusions were made. In addition, datasets of companies containing customers with a higher average number of transactions could provide insights into the ability to recognise seasonality and trends. Customers with corresponding seasonality patterns and/or trend characteristics can then be observed to see if they are well clustered together. The datasets of this thesis mainly contains customers with a low average number of transactions, such that seasonality and trends where not properly detectable.

The fact that time series clustering only allows for the selection of one feature is a second limitation of this research. Feature selection was preferred over feature extraction to keep segmentation models readable and interpretable. It is only arguable whether a single feature can capture a customer's entire shopping behaviour. For example, the time series representing the monetary value of transactions cannot reflect the number of visits to a company's website, and vice versa. Because several variables are used in its calculation, the CLV feature selection in this thesis already contains more information than the other two selected features. This is one of the reasons why the CLV time series was chosen as the preferable feature selection in this thesis. However, it is still questionable if this feature truly describes a customer's entire shopping behaviour. For example, the number of website visits is not taken into account in the CLV calculation. Customers may be incorrectly clustered together as a result of poor feature selection, even though their entire customer behaviours do not match. In short, clustering customers in time series format works well, but because only one feature can be selected, feature selection must be done very carefully. Another drawback of dynamic clustering is that it does not consider static data, like demographic and geographic factors. In some circumstances, static variables may be useful for customer differentiation and are informative for developing CRM strategies. In conclusion, dynamic customer segmentation provides great potential to detect seasonality and trends, as well as the ability to predict customer behaviour. It does, however, raise several challenges.

A third limitation to consider is that only time series of customers who placed at least one order were included. According to this thesis, an advantage of the pageview feature is that it allows for all customers to be segmented. The pageview feature does not require customers to have placed an order, which is the case with the other two considered features. However, an analysis was not provided in this thesis on whether it would be preferred to be able to cluster customers who did not place an order. Therefore, no complete review of the pageview feature selection was presented.

The first suggestion for future research concerns the CORT distance considered in this thesis. This distance metric determines whether inaccuracy at a certain time index is correlated across its direct neighbours. In order to not just look for correlation between direct neighbours but also between other neighbours, we suggest considering the COR distance as well. The COR distance could allow for more accurate similarity rates of misaligned time series. The results of the COR distance can be compared with the other distance measurements addressed in this thesis.

The second suggestion relates to the fact that the DTW distance provides the most accurate similarity rates in this paper. As there exist several variants of the DTW distance, it would be interesting to examine their results as well. This thesis could be further analysed by the addition of the CID and CORT distances with the DTW distance incorporated rather than the ED. It can then be concluded whether the inclusion of these distances could improve the performance of the dynamic customer segmentation methods that are currently included.

The Silhouette and Davies-Bouldin indices are used in this thesis to evaluate the performance of the generated clusters. The Silhouette index appears to be affected by the distance measurement used in its calculation. As a result, it may be useful to study whether additional performance measurements, such as the Caliński–Harabasz cluster validity index, are sensitive as well. If that does not happen to be the situation, it would be interesting to observe whether the same conclusions are derived as they are using the Davies-Bouldin index.

As a final suggestion, customer behaviour tracking in time series format allows the ability to predict customer behaviour, which is not achievable with exploratory methods. In this research, predictions are only incorporated for the CLV time series. Extending the customer segmentation analysis with more feature selections based on customer behaviour predictions would give more insights.

References

Abbasimehr, H., & Bahrini, A. (2021). An analytical framework based on the recency, frequency, and monetary model and time series clustering techniques for dynamic segmentation. *Expert Systems with Applications*, *192*, 116373.

Ahmed, S. R. (2004). Applications of data mining in retail business. In *International conference on information technology: Coding and computing* (pp. 455–459). IEEE.

Akhondzadeh-Noughabi, E., & Albadvi, A. (2015). Mining the dominant patterns of customer shifts between segments by using top-k and distinguishing sequential rules. *Management Decision*, *53*(9), 1976–2003.

Azuaje, F., & Bolshakova, N. (2003). Clustering genomic expression data: design and evaluation principles. In *A practical approach to microarray data analysis* (pp. 230–245). Springer.

Baek, M., & Kim, C. (2021). A review on spectral clustering and stochastic block models. *Journal of the Korean Statistical Society*, *50*(3), 818–831.

Batista, G. E., Keogh, E. J., Tataw, O. M., & de Souza, V. (2014). Cid: an efficient complexity-invariant distance for time series. *Data Mining and Knowledge Discovery*, *28*(3), 634–669.

Batista, G. E., Wang, X., & Keogh, E. J. (2011). A complexity-invariant distance measure for time series. In *Proceedings of the 2011 siam international conference on data mining* (pp. 699–710). SIAM.

Berry, M. J., & Linoff, G. S. (2004). *Data mining techniques: for marketing, sales, and customer relationship management.* John Wiley & Sons.

Berson, A., & Thearling, K. (1999). *Building data mining applications for crm.* McGraw-Hill, Inc.

Blocker, C. P., & Flint, D. J. (2007). Customer segments as moving targets: integrating customer value dynamism into segment instability logic. *Industrial Marketing Management*, *36*(6), 810–822.

Böttcher, M., Spott, M., Nauck, D., & Kruse, R. (2009). Mining changing customer segments in dynamic markets. *Expert Systems with Applications*, *36*(1), 155–164.

Brangule-Vlagsma, K., Pieters, R. G., & Wedel, M. (2002). The dynamics of value segments: modeling framework and empirical illustration. *International Journal of Research in Marketing*, *19*(3), 267–285.

Bunge, J. A., & Judson, D. H. (2005). Data mining. In *Encyclopedia of social measurement* (p. 617-624). Elsevier.

Buttle, F. (2004). *Customer relationship management: Concepts and tools.* Elsevier Butterworth-Heinemann.

Cassisi, C., Montalto, P., Aliotta, M., Cannata, A., & Pulvirenti, A. (2012). Similarity measures and dimensionality reduction techniques for time series data mining. *Advances in Data Mining Knowledge Discovery and Applications*, 71–96.

Chouakria, A. D., & Nagabhushan, P. N. (2007). Adaptive dissimilarity index for measuring time series proximity. *Advances in Data Analysis and Classification*, *1*(1), 5–21.

Christy, A. J., Umamakeswari, A., Priyatharsini, L., & Neyaa, A. (2021). Rfm ranking–an effective approach to customer segmentation. *Journal of King Saud University-Computer and Information Sciences*, *33*(10), 1251–1257.

D'Antonio, M. (2019). Ovum market radar: Customer data platforms.

Fader, P. S., Hardie, B. G., & Lee, K. L. (2005). "counting your customers" the easy way: An alternative to the pareto/nbd model. *Marketing Science*, *24*(2), 275–284.

Fisher, L., & van Ness, J. W. (1971). Admissible clustering procedures. *Biometrika*, *58*(1), 91–104.

Fop, M., & Murphy, T. B. (2018). Variable selection methods for model-based clustering. *Statistics Surveys*, *12*, 18–65.

Gallo, A. (2014). The value of keeping the right customers. *Harvard Business Review*, *29*(10), 304–309.

Golub, G. H., & Reinsch, C. (1971). Singular value decomposition and least squares solutions. In *Linear algebra* (pp. 134–151). Springer.

Guerola-Navarro, V., Oltra-Badenes, R., Gil-Gomez, H., & Gil-Gomez, J.-A. (2020). Customer relationship management (crm): a bibliometric analysis. *International Journal of Services Operations and Informatics*, *10*(3), 242–268.

Gupta, S., Hanssens, D., Hardie, B., Kahn, W., Kumar, V., Lin, N., … Sriram, S. (2006). Modeling customer lifetime value. *Journal of Service Research*, *9*(2), 139–155.

Gupta, S., Lehmann, D. R., & Stuart, J. A. (2004). Valuing customers. *Journal of Marketing Research*, *41*(1), 7–18.

Hancer, E., Xue, B., & Zhang, M. (2018). Differential evolution for filter feature selection based on information theory and feature ranking. *Knowledge-Based Systems*, *140*, 103–119.

Hancer, E., Xue, B., & Zhang, M. (2020). A survey on feature selection approaches for clustering. *Artificial Intelligence Review*, *53*(6), 4519–4545.

He, Z., Xu, X., Huang, J. Z., & Deng, S. (2004). Mining class outliers: concepts, algorithms and applications in crm. *Expert Systems with Applications*, *27*(4), 681–697.

Heinrich, K., Zschech, P., Skouti, T., Griebenow, J., & Riechert, S. (2019). Demystifying the black box: A classification scheme for interpretation and visualization of deep intelligent systems. In *25th americas conference on information systems.* Association for Information Systems.

Hosseni, M. B., Tarokh, M. J., et al. (2011). Customer segmentation using clv elements. *Journal of Service Science and Management*, *4*(03), 284.

Hughes, A. M. (1994). *Strategic database marketing.* Probus Pub. Co.

Jolliffe, I. T. (2002). *Principal component analysis for special types of data.* Springer.

King, S. F., & Burgess, T. F. (2008). Understanding success and failure in customer relationship management. *Industrial Marketing Management*, *37*(4), 421–431.

Kumar, V., & Shah, D. (2004). Building and sustaining profitable customer loyalty for the 21st century. *Journal of Retailing*, *80*(4), 317–329.

Lejeune, M. A. (2001). Measuring the impact of data mining on churn management. *Internet Research: Electronic Networking Applications and Policy*, *11*(5), 375–387.

Lemmens, A., Croux, C., & Stremersch, S. (2012). Dynamics in the international market segmentation of new product growth. *International Journal of Research in Marketing*, *29*(1), 81–92.

Lines, J., & Bagnall, A. (2015). Time series classification with ensembles of elastic distance measures. *Data Mining and Knowledge Discovery*, *29*(3), 565–592.

Liu, D.-R., Lai, C.-H., & Lee, W.-J. (2009). A hybrid of sequential rules and collaborative filtering for product recommendation. *Information Sciences*, *179*(20), 3505–3519.

Manjunath, Y. S. K., & Kashef, R. F. (2021). Distributed clustering using multi-tier hierarchical overlay super-peer peer-to-peer network architecture for efficient customer segmentation. *Electronic Commerce Research and Applications*, *47*, 101040.

Mark, T., Lemon, K. N., Vandenbosch, M., Bulla, J., & Maruotti, A. (2013). Capturing the evolution of customer–firm relationships: How customers become more (or less) valuable over time. *Journal of Retailing*, *89*(3), 231–245.

Montero, P., & Vilar, J. A. (2015). Tsclust: An r package for time series clustering. *Journal of Statistical Software*, *62*, 1–43.

Mosaddegh, A., Albadvi, A., Sepehri, M. M., & Teimourpour, B. (2021). Dynamics of customer segments: A predictor of customer lifetime value. *Expert Systems with Applications*, *172*, 114606.

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

Netzer, O., Lattin, J. M., & Srinivasan, V. (2008). A hidden markov model of customer relationship dynamics. *Marketing Science*, *27*(2), 185–204.

Ngai, E., Xiu, L., & Chau, D. (2009). Application of data mining techniques in customer relationship management: A literature review and classification. *Expert Systems with Applications*, *36*(2), 2592-2602.

Panigrahi, S., Karali, Y., & Behera, H. (2013). Normalize time series and forecast using evolutionary neural network. *International Journal of Engineering Research & Technology*, *2*(9), 2518–2522.

Paparrizos, J., & Gravano, L. (2015). k-shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 acm sigmod international conference on management of data* (pp. 1855–1870).

Paparrizos, J., & Gravano, L. (2017). Fast and accurate time-series clustering. *ACM Transactions on Database Systems (TODS)*, *42*(2), 1–49.

Parvaneh, A., Tarokh, M., & Abbasimehr, H. (2014). Combining data mining and group decision making in retailer segmentation based on lrfmp variables. *International Journal of Industrial Engineering & Production Research*, *25*(3), 197–206.

Petitjean, F., Ketterlin, A., & Gançarski, P. (2011). A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, *44*(3), 678–693.

Sardá-Espinosa, A. (2017). Comparing time-series clustering algorithms in r using the dtwclust package. *R Package Vignette*, *12*, 41.

Teo, T. S., Devadoss, P., & Pan, S. L. (2006). Towards a holistic perspective of customer relationship management (crm) implementation: A case study of the housing and development board, singapore. *Decision Support Systems*, *42*(3), 1613–1627.

Wang, X., Mueen, A., Ding, H., Trajcevski, G., Scheuermann, P., & Keogh, E. (2013). Experimental comparison of representation methods and distance measures for time series data. *Data Mining and Knowledge Discovery*, *26*(2), 275–309.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, *58*(301), 236–244.

Yanovitzky, I., & VanLear, A. (2008). Time series analysis: Traditional and contemporary approaches. In *The sage sourcebook of advanced data analysis methods for communication re-*

search (pp. 89–124). Sage Publications.

Ye, J. (n.d.). Least squares linear discriminant analysis. In *Proceedings of the 24th international conference on machine learning* (pp. 1087–1093). Association for Computing Machinery.

Zhou, J., Wei, J., & Xu, B. (2021). Customer segmentation by web content mining. *Journal of Retailing and Consumer Services*, *61*(C), 102588.

Appendix A

A.1 Proof of convergence of DBA

A proof is given which states that DBA produces a better average sequence ($\bar{T}$) at each iteration. This is done by providing a proof that the sum of squares is decreasing at each iteration. DTW ensures that the minimum alignment between two sequences is found. This proofs that the first step of the DBA is optimal. To proof convergence of the algorithm involves demonstrating that the calculated $\bar{T}$ is optimal for a given multiple alignment M.

Let us denote M as the multiple alignment between the current average sequence ($\bar{T}$), and the set of sequences to be averaged (D), and M_j as the index j of M. To begin, we rewrite the objective function, the sum of squares (SS), as:

$$\mathrm{SS}(\bar{T}, D) = \sum_{i=0}^{N} \mathrm{DTW}\left(\bar{T}, T_i\right)^2 = \sum_{j=1}^{L} \sum_{e \in M_j} (\bar{T}(j) - e)^2$$

Where e is an element of a sequence in D that Dynamic Time Warping has associated to the j^{th} element of $\bar{T}$. Given that there is no maximum for this function, it is minimised when its partial derivative is zero:

$$\frac{\partial \mathrm{SS}(\bar{T}, D)}{\partial \bar{T}(j)} = 0$$

$$\sum_{e \in M_j} 2 \cdot (\bar{T}(j) - e) = 0$$

$$\bar{T}(j) = \frac{1}{|M_j|} \sum_{e \in M_j} e$$

As a result, the $\mathrm{SS}(\bar{T}, D)$ is minimised when every element j of $\bar{T}$ is positioned as the mean of $|M_j|$.

A.2 Tables

	$d(x_i, x_j)$= ED					$d(x_i, x_j)$= DTW					$d(x_i, x_j)$= CID					$d(x_i, x_j)$= SBD				
	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8
CLV Time Series																				
Hierarchical + ED	**0.35**	0.33	0.35	0.38	0.41	0.39	0.38	0.35	0.30	0.24	0.38	0.36	0.39	0.41	0.43	0.51	0.51	0.49	0.45	0.37
Hierarchical + DTW	0.33	0.33	0.29	0.29	0.30	**0.63**	0.56	0.51	0.50	0.51	**0.40**	0.40	0.35	0.35	0.35	0.54	0.56	0.51	0.52	0.51
Hierarchical + CORT	0.31	0.34	0.34	0.38	0.38	0.38	0.26	0.27	0.24	0.23	0.34	0.35	0.37	0.39	0.40	0.43	0.39	0.42	0.40	0.39
Hierarchical + CID	0.34	0.32	0.33	0.37	0.37	0.47	0.44	0.42	0.35	0.35	0.38	0.36	0.39	0.42	0.42	0.53	0.51	0.46	0.46	0.47
Hierarchical + SBD	0.23	0.27	0.27	0.30	0.27	0.43	0.33	0.34	0.28	0.26	0.28	0.31	0.32	0.34	0.31	**0.67**	0.54	**0.58**	0.54	0.52
Spectral + ED	0.31	0.33	0.33	**0.43**	0.45	0.49	0.14	0.07	0.28	0.27	0.35	0.32	0.31	0.46	0.47	0.44	0.19	0.09	0.28	0.27
Spectral + DTW	0.32	0.32	0.30	0.29	0.26	0.62	**0.61**	**0.60**	**0.60**	**0.57**	0.39	0.39	0.35	0.36	0.33	0.49	0.49	0.46	0.52	0.45
Spectral + CORT	0.36	0.37	0.34	0.37	0.39	0.15	0.18	0.08	0.15	0.14	0.35	0.38	0.33	0.37	0.39	0.18	0.11	0.07	0.13	0.13
Spectral + CID	0.32	**0.36**	**0.37**	0.43	**0.45**	0.28	0.35	0.21	0.29	0.30	0.40	**0.41**	**0.40**	**0.46**	**0.48**	0.54	0.44	0.27	0.29	0.31
Spectral + SBD	0.29	0.23	0.28	0.28	0.30	0.36	0.41	0.41	0.38	0.29	0.32	0.27	0.32	0.34	0.35	0.53	**0.62**	0.56	**0.56**	**0.54**
K-Shape + SBD	0.27	0.25	0.26	0.30	0.21	0.45	0.43	0.28	0.35	0.30	0.31	0.28	0.28	0.36	0.25	0.54	0.57	0.44	0.56	0.52
Monetary Time Series																				
Hierarchical + ED	**0.11**	**0.13**	**0.15**	**0.18**	**0.20**	-0.35	-0.35	-0.35	-0.35	-0.35	0.10	**0.13**	**0.15**	**0.17**	**0.19**	-0.31	-0.30	-0.29	-0.29	-0.28
Hierarchical + DTW	0.00	-0.01	-0.01	-0.01	-0.02	**0.88**	0.89	0.89	0.91	0.91	0.11	0.07	0.07	0.07	0.07	0.74	0.71	0.71	0.73	0.73
Hierarchical + CORT	0.11	0.13	0.15	0.17	0.19	-0.51	-0.51	-0.51	-0.51	-0.51	0.10	0.12	0.14	0.16	0.18	-0.40	-0.39	-0.39	-0.38	-0.38
Hierarchical + CID	0.07	0.09	0.11	0.11	0.13	-0.80	-0.80	-0.80	-0.80	-0.80	0.08	0.10	0.13	0.13	0.15	-0.30	-0.50	-0.50	-0.50	-0.48
Hierarchical + SBD	-0.01	-0.01	-0.02	-0.02	-0.02	0.87	0.87	0.87	0.87	0.86	0.10	0.10	0.10	0.10	0.08	0.80	0.80	0.80	0.80	0.77
Spectral + ED	0.11	0.13	0.15	0.17	0.20	-0.52	-0.52	-0.51	-0.60	-0.60	0.10	0.12	0.14	0.16	0.18	-0.45	-0.50	-0.50	-0.55	-0.54
Spectral + DTW	0.00	0.00	-0.01	-0.01	-0.02	0.88	**0.90**	**0.91**	**0.91**	**0.91**	**0.14**	0.12	0.09	0.08	0.05	-0.28	0.09	0.10	0.00	0.01
Spectral + CORT	0.11	0.13	0.15	0.18	0.20	-0.18	-0.17	-0.17	-0.23	-0.23	0.11	0.13	0.14	0.16	0.17	-0.09	-0.08	-0.07	-0.18	-0.17
Spectral + CID	0.09	0.11	0.13	0.16	0.18	-0.48	-0.49	-0.49	-0.49	-0.59	0.08	0.10	0.13	0.15	0.17	-0.57	-0.58	-0.57	-0.57	-0.61
Spectral + SBD	-0.02	-0.03	-0.04	-0.05	-0.07	0.77	0.77	0.77	0.79	0.79	0.10	0.09	0.08	0.03	0.02	**0.86**	**0.86**	**0.87**	**0.82**	**0.82**
K-Shape + SBD	0.01	0.00	0.02	0.04	0.03	-0.01	0.00	-0.08	-0.10	-0.06	0.03	0.03	0.04	0.06	0.06	0.28	0.31	0.23	0.16	0.25
Pageview Time Series																				
Hierarchical + ED	**0.09**	**0.11**	**0.13**	**0.15**	**0.17**	-0.25	-0.25	-0.25	-0.25	-0.25	0.08	0.10	0.12	**0.14**	**0.16**	-0.34	-0.33	-0.32	-0.32	-0.34
Hierarchical + DTW	0.00	0.00	-0.01	-0.01	-0.01	0.62	0.63	0.64	0.64	0.61	0.03	0.03	0.03	0.03	0.00	0.47	0.47	0.49	0.49	0.22
Hierarchical + CORT	0.09	0.11	0.13	0.14	0.16	-0.14	-0.14	-0.14	-0.14	-0.15	**0.09**	**0.11**	**0.13**	0.14	0.16	-0.15	-0.14	-0.14	-0.16	-0.16
Hierarchical + CID	0.03	0.05	0.06	0.08	0.08	0.13	-0.14	-0.01	0.00	0.01	0.06	0.07	0.08	0.10	0.11	-0.03	-0.33	-0.17	-0.16	-0.15
Hierarchical + SBD	0.00	-0.01	-0.01	-0.01	-0.02	0.65	0.64	0.64	0.64	0.65	0.07	0.05	0.05	0.04	0.04	0.62	0.59	0.60	0.60	0.60
Spectral + ED	0.09	0.11	0.13	0.14	0.16	-0.27	-0.27	-0.27	-0.27	-0.41	0.08	0.10	0.12	0.13	0.15	-0.30	-0.31	-0.34	-0.34	-0.42
Spectral + DTW	0.00	0.00	0.00	-0.01	-0.01	**0.70**	**0.70**	**0.71**	**0.67**	**0.71**	0.06	0.04	0.03	0.03	0.02	-0.21	-0.19	-0.16	-0.03	-0.08
Spectral + CORT	**0.09**	**0.11**	**0.13**	**0.15**	**0.17**	-0.19	-0.19	-0.19	-0.27	-0.27	0.09	0.10	0.12	0.13	0.16	-0.22	-0.21	-0.22	-0.31	-0.30
Spectral + CID	0.07	0.09	0.11	0.13	0.15	-0.20	-0.28	-0.28	-0.31	-0.31	0.06	0.08	0.10	0.11	0.13	-0.64	-0.64	-0.64	-0.63	-0.63
Spectral + SBD	-0.01	-0.01	-0.02	-0.02	-0.03	0.60	0.60	0.61	0.61	0.60	0.03	0.03	0.03	0.03	0.02	**0.68**	**0.68**	**0.68**	**0.69**	**0.68**
K-Shape + SBD	-0.01	0.00	0.02	0.03	0.03	0.56	0.17	-0.05	-0.02	-0.06	0.03	0.03	0.03	0.04	0.04	0.56	0.27	0.09	0.08	0.07

Table 18: Silhouette index scores of company 1

	$d(x_i,x_j)$= ED					$d(x_i,x_j)$= DTW					$d(x_i,x_j)$= CID					$d(x_i,x_j)$= SBD				
	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8
CLV Time Series																				
Hierarchical + ED	1.87	1.73	1.67	1.57	1.43	2.52	2.26	3.83	4.07	3.30	2.43	2.25	2.36	2.04	1.83	1.67	1.73	##	##	##
Hierarchical + DTW	1.51	1.92	1.79	2.38	2.19	1.38	**1.38**	**1.31**	2.57	**2.24**	2.28	2.86	2.63	3.08	2.84	##	##	##	##	##
Hierarchical + CORT	2.03	1.84	1.72	1.65	1.91	2.56	2.21	2.04	**2.02**	3.03	2.48	2.03	1.91	1.64	2.16	##	##	##	##	##
Hierarchical + CID	1.89	1.75	1.77	1.61	1.58	2.38	2.13	3.26	4.02	3.86	2.71	2.48	2.72	2.30	2.07	##	##	##	##	##
Hierarchical + SBD	2.76	3.93	3.48	3.32	3.22	3.07	4.22	3.76	3.74	3.61	4.12	6.05	5.30	4.86	4.66	##	##	##	##	##
Spectral + ED	**1.47**	**1.32**	**1.23**	1.52	1.42	**1.37**	1.43	2.71	3.16	2.73	**2.15**	**1.81**	**1.63**	2.02	1.77	##	##	##	##	##
Spectral + DTW	1.48	1.39	1.33	2.06	1.85	1.47	2.69	2.43	2.73	2.49	2.17	1.74	1.67	2.93	2.59	##	##	##	##	##
Spectral + CORT	2.23	1.38	1.24	**1.32**	**1.29**	2.67	1.47	2.13	2.11	2.76	2.67	2.24	2.17	**1.48**	**1.45**	##	**1.36**	##	##	##
Spectral + CID	1.53	1.66	1.48	1.55	1.46	1.45	2.61	2.50	3.95	3.54	2.25	2.24	1.98	2.00	1.87	##	##	##	##	##
Spectral + SBD	1.99	2.55	2.28	2.29	2.03	2.31	3.15	2.65	2.63	2.83	2.76	3.91	3.39	3.44	2.95	##	##	##	##	##
K-Shape + SBD	1.91	1.76	1.90	2.43	2.51	2.14	1.97	2.50	3.07	2.78	2.52	2.25	2.47	3.61	3.78	##	##	##	##	##
Monetary Time Series																				
Hierarchical + ED	1.09	1.09	1.52	1.11	1.11	5.56	5.79	5.46	7.88	8.06	1.04	1.05	1.61	1.06	1.06	##	##	##	##	##
Hierarchical + DTW	17.85	15.70	13.17	12.45	11.01	22.94	20.82	17.05	17.09	15.16	27.31	22.40	16.16	14.74	12.62	##	##	##	##	##
Hierarchical + CORT	1.73	1.61	3.16	2.86	2.76	8.39	9.75	12.02	11.13	17.15	1.44	1.37	3.13	2.83	2.73	##	##	##	##	##
Hierarchical + CID	10.72	8.42	6.76	6.18	5.35	15.75	48.43	40.57	35.52	42.57	18.58	14.84	11.80	9.51	8.26	**1.48**	##	**2.99**	##	##
Hierarchical + SBD	17.30	14.67	13.22	12.19	11.26	20.89	18.27	18.18	16.58	16.09	21.00	17.25	15.13	13.89	11.96	##	##	##	##	##
Spectral + ED	1.06	1.06	1.05	1.05	1.04	4.70	5.29	6.74	6.85	6.72	**1.03**	**1.03**	**1.03**	**1.02**	**1.03**	9.53	16.07	##	**15.23**	##
Spectral + DTW	14.56	13.41	12.46	10.98	10.24	18.72	16.39	15.68	14.03	13.11	18.86	16.55	14.61	11.50	11.24	##	##	##	##	##
Spectral + CORT	1.11	1.61	1.53	1.46	1.42	7.78	6.98	7.13	7.43	7.83	1.05	1.38	1.33	1.29	1.26	##	##	##	##	##
Spectral + CID	**1.06**	**1.05**	**1.05**	**1.05**	**1.04**	**5.69**	**5.93**	**4.33**	**4.02**	**3.02**	1.22	1.18	1.15	1.13	1.12	11.41	9.71	15.84	##	**15.20**
Spectral + SBD	8.06	7.24	6.21	6.79	6.25	10.04	9.34	8.26	8.97	8.42	7.72	6.78	5.61	6.52	5.95	##	##	##	##	##
K-Shape + SBD	9.71	8.74	7.57	6.97	7.02	26.17	17.14	16.49	18.60	14.53	12.86	10.30	9.86	9.37	8.27	##	##	##	##	##
Pageview Time Series																				
Hierarchical + ED	1.25	1.66	1.76	1.69	1.49	**5.15**	**4.82**	10.35	11.03	7.93	1.16	1.73	1.88	1.75	1.50	##	##	##	##	##
Hierarchical + DTW	25.60	21.12	19.42	17.76	18.54	35.39	29.37	28.29	25.22	26.25	47.11	38.42	32.15	28.82	30.05	##	##	##	##	##
Hierarchical + CORT	1.33	1.35	1.33	1.32	1.31	5.10	7.02	7.79	8.41	8.32	1.23	1.24	1.21	1.21	1.20	##	##	##	##	##
Hierarchical + CID	10.94	8.85	8.18	7.17	7.10	15.54	14.83	12.71	11.50	13.37	17.32	14.17	13.03	11.44	11.35	**2.14**	##	##	**9.43**	8.42
Hierarchical + SBD	25.74	22.77	20.28	18.20	16.46	34.58	30.72	27.70	25.57	23.06	48.71	38.68	32.49	27.18	24.01	##	##	##	##	##
Spectral + ED	1.77	1.19	1.19	1.18	1.17	5.44	9.39	10.31	10.08	10.03	1.84	**1.12**	**1.11**	**1.11**	**1.10**	##	##	##	##	##
Spectral + DTW	18.56	18.11	17.17	19.16	15.49	30.40	26.28	24.17	27.10	21.44	34.58	30.81	26.95	31.89	22.81	##	##	##	##	**2.42**
Spectral + CORT	1.22	1.70	1.62	1.21	1.49	7.45	6.70	8.11	10.11	9.30	**1.15**	1.47	1.40	1.13	1.51	30.90	##	##	##	##
Spectral + CID	**1.13**	**1.12**	**1.13**	**1.13**	**1.14**	5.97	6.15	**7.16**	**7.24**	**7.73**	1.26	1.22	1.20	1.18	1.17	##	##	##	##	##
Spectral + SBD	16.95	15.14	13.21	12.07	10.84	25.05	21.35	19.39	18.25	16.20	24.93	20.69	17.92	16.01	14.07	##	##	##	##	##
K-Shape + SBD	18.43	12.89	10.99	10.30	10.67	26.52	23.33	19.58	20.64	22.71	30.07	20.99	17.08	16.74	16.87	##	##	##	##	##

Table 19: Davies-Bouldin index scores of company 1

	$d(x_i, x_j)$= ED					$d(x_i, x_j)$= DTW					$d(x_i, x_j)$= CID					$d(x_i, x_j)$= SBD				
	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8
CLV Time Series																				
Hierarchical + ED	0.40	0.43	0.43	0.40	0.42	0.33	0.36	0.36	0.40	0.37	0.42	0.45	0.45	0.41	0.43	0.55	0.53	0.51	0.52	0.52
Hierarchical + DTW	0.29	0.35	0.37	0.37	0.34	0.51	0.46	0.46	**0.44**	0.42	0.32	0.37	0.38	0.38	0.35	0.55	0.53	0.53	0.51	0.49
Hierarchical + CORT	**0.61**	**0.60**	0.57	0.58	**0.59**	0.48	0.44	0.42	0.42	0.41	**0.59**	**0.59**	**0.59**	**0.59**	**0.59**	0.55	0.52	0.50	0.51	0.41
Hierarchical + CID	0.57	0.42	0.44	0.41	0.40	0.39	0.40	0.38	0.41	0.42	**0.59**	0.45	0.45	0.42	0.41	0.48	0.50	0.49	0.50	0.48
Hierarchical + SBD	0.15	0.30	0.29	0.32	0.32	0.47	0.39	0.43	0.41	0.42	0.20	0.33	0.32	0.35	0.34	0.54	0.57	**0.61**	**0.54**	**0.55**
Spectral + ED	0.37	0.36	0.40	0.41	0.44	0.51	0.47	0.43	0.40	0.36	0.40	0.39	0.42	0.42	0.46	0.55	0.54	0.55	0.52	0.54
Spectral + DTW	0.35	0.29	0.29	0.20	0.27	**0.55**	**0.51**	**0.50**	0.43	**0.45**	0.37	0.30	0.30	0.21	0.28	0.47	0.43	0.39	0.36	0.34
Spectral + CORT	0.59	0.59	**0.59**	**0.58**	0.58	0.30	0.24	0.20	0.41	0.22	0.42	0.38	0.37	0.34	0.29	0.11	0.10	0.12	0.06	0.06
Spectral + CID	0.37	0.36	0.40	0.41	0.45	0.51	0.46	0.43	0.40	0.37	0.40	0.39	0.42	0.43	0.46	0.55	0.54	0.53	0.53	0.55
Spectral + SBD	0.40	0.37	0.33	0.28	0.32	0.50	0.47	0.48	0.38	0.37	0.43	0.39	0.37	0.34	0.36	0.57	0.55	0.56	0.52	0.54
K-Shape + SBD	0.41	0.33	0.42	0.18	0.30	0.42	0.41	0.34	0.33	0.41	0.43	0.34	0.44	0.21	0.31	**0.61**	**0.58**	0.53	0.43	0.49
Datatrics		-0.18					-0.42					-0.21					-0.19			
Monetary Time Series																				
Hierarchical + ED	**0.07**	**0.08**	**0.10**	**0.12**	**0.13**	-0.90	-0.89	-0.88	-0.88	-0.87	0.06	0.08	0.09	0.11	0.13	-0.70	-0.73	-0.69	-0.69	-0.66
Hierarchical + DTW	-0.01	-0.02	-0.02	-0.03	-0.03	**0.96**	**0.96**	**0.96**	0.96	0.96	0.10	0.07	0.07	0.07	0.06	0.90	0.83	0.83	0.83	0.83
Hierarchical + CORT	0.06	0.08	0.10	0.11	0.13	-0.92	-0.92	-0.92	-0.92	-0.92	0.06	0.07	0.09	0.11	0.12	-0.76	-0.76	-0.76	-0.74	-0.72
Hierarchical + CID	0.04	0.06	0.08	0.09	0.11	-0.91	-0.90	-0.88	-0.87	-0.87	0.05	0.07	0.09	0.10	0.12	-0.33	-0.34	-0.50	-0.48	-0.48
Hierarchical + SBD	-0.01	-0.02	-0.02	-0.03	-0.03	0.96	0.96	0.96	0.96	0.96	0.14	0.14	0.13	0.08	0.08	0.93	0.93	0.92	0.86	0.86
Spectral + ED	**0.07**	**0.08**	**0.10**	**0.12**	**0.13**	-0.92	-0.91	-0.91	-0.89	-0.89	0.06	0.08	0.09	0.11	0.13	-0.74	-0.76	-0.76	-0.73	-0.73
Spectral + DTW	-0.01	-0.02	-0.03	-0.03	-0.04	0.95	0.96	0.95	**0.96**	**0.96**	**0.15**	**0.14**	**0.13**	**0.13**	**0.13**	0.94	0.93	0.93	0.92	0.92
Spectral + CORT	0.00	0.00	-0.01	-0.01	-0.01	-0.12	-0.18	-0.15	-0.35	-0.35	0.06	0.07	0.09	0.11	0.10	-0.18	-0.19	-0.19	-0.19	-0.28
Spectral + CID	**0.07**	**0.08**	0.10	**0.12**	**0.13**	-0.92	-0.91	-0.92	-0.91	-0.89	0.00	-0.01	-0.01	-0.01	-0.01	-0.74	-0.76	-0.74	-0.76	-0.73
Spectral + SBD	-0.02	-0.03	-0.03	-0.04	-0.05	0.92	0.92	0.92	0.92	0.91	0.12	0.11	0.11	0.10	0.09	**0.95**	**0.95**	**0.95**	**0.94**	**0.94**
K-Shape + SBD	0.02	0.03	0.02	0.03	0.04	0.09	-0.14	-0.13	-0.40	-0.27	0.02	0.03	0.03	0.04	0.05	0.26	0.07	0.13	-0.18	-0.07
Datatrics		-0.05					-0.91					-0.03					-0.84			
Pageview Time Series																				
Hierarchical + ED	**0.05**	**0.06**	**0.07**	**0.09**	**0.10**	-0.09	-0.23	-0.24	-0.24	-0.24	0.04	**0.05**	**0.06**	**0.07**	**0.08**	-0.12	-0.31	-0.30	-0.30	-0.30
Hierarchical + DTW	0.00	0.00	-0.01	-0.01	-0.01	0.59	0.60	0.61	0.61	0.60	0.02	0.01	0.02	0.02	0.01	0.45	0.42	0.46	0.46	0.38
Hierarchical + CORT	0.05	0.06	0.07	0.08	0.09	-0.18	-0.19	-0.19	-0.19	-0.28	0.03	0.03	0.04	0.06	0.07	-0.23	-0.24	-0.23	-0.23	-0.36
Hierarchical + CID	0.00	0.01	0.02	0.03	0.04	0.63	0.39	0.11	-0.21	-0.21	**0.09**	0.05	0.05	0.06	0.06	0.61	0.38	0.03	-0.38	-0.38
Hierarchical + SBD	0.00	0.00	-0.01	-0.01	-0.01	0.63	0.64	0.62	0.62	0.63	0.04	0.04	0.04	0.04	0.02	0.60	0.63	0.61	0.61	0.61
Spectral + ED	**0.05**	**0.06**	**0.07**	**0.08**	0.09	-0.46	-0.45	-0.45	-0.46	-0.45	0.02	0.03	0.04	0.06	0.07	-0.61	-0.61	-0.61	-0.61	-0.61
Spectral + DTW	0.00	0.00	-0.01	-0.01	-0.01	**0.67**	**0.69**	**0.65**	**0.69**	**0.69**	0.05	0.03	0.03	0.04	0.02	0.64	0.67	0.53	0.68	0.64
Spectral + CORT	0.04	0.06	0.06	0.07	0.08	-0.08	-0.08	-0.07	-0.20	-0.20	0.04	0.05	0.06	0.07	0.07	-0.15	-0.13	-0.11	-0.16	-0.17
Spectral + CID	0.05	0.06	0.07	0.08	0.09	-0.34	-0.34	-0.34	-0.41	-0.45	0.03	0.04	0.05	0.06	0.06	-0.45	-0.45	-0.45	-0.49	-0.60
Spectral + SBD	-0.01	-0.01	-0.01	-0.02	-0.02	0.57	0.58	0.58	0.55	0.54	0.03	0.03	0.03	0.04	0.04	**0.69**	**0.69**	**0.70**	**0.72**	**0.72**
K-Shape + SBD	0.00	0.00	-0.01	0.00	-0.01	0.06	-0.01	0.57	0.07	0.00	0.01	0.00	0.01	0.00	0.00	0.18	0.08	0.61	0.17	0.09
Datatrics		-0.04					-0.65					-0.05					-0.71			

Table 20: Silhouette index scores of company 2

	$d(x_i, x_j)$= ED					$d(x_i, x_j)$= DTW					$d(x_i, x_j)$= CID					$d(x_i, x_j)$= SBD				
	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8
CLV Time Series																				
Hierarchical + ED	1.34	1.40	1.41	1.33	**1.25**	2.01	4.94	5.49	4.85	3.80	1.57	1.89	1.95	1.79	**1.61**	##	##	##	##	##
Hierarchical + DTW	1.55	1.72	1.71	1.62	1.58	1.81	2.02	4.91	4.89	4.40	1.78	1.95	2.06	1.90	1.81	##	##	##	##	##
Hierarchical + CID	**1.08**	1.45	1.34	2.08	1.89	**1.10**	**1.63**	**1.62**	**2.24**	**2.03**	**1.12**	1.48	**1.38**	2.17	1.99	##	##	##	##	##
Hierarchical + CORT	1.53	1.38	1.39	1.31	1.25	2.53	2.19	4.85	4.31	3.89	1.95	1.70	1.92	1.76	1.64	##	##	##	##	##
Hierarchical + SBD	3.51	3.33	3.01	2.44	2.73	3.51	3.36	2.98	3.62	3.86	3.51	3.42	3.07	2.59	2.91	##	3.59	3.28	##	2.33
Spectral + ED	1.36	**1.29**	**1.35**	**1.30**	1.29	1.69	2.69	3.66	3.79	5.26	1.65	**1.43**	1.72	**1.56**	1.63	**1.55**	**1.97**	**2.42**	**2.78**	##
Spectral + DTW	1.50	1.55	1.46	1.71	1.53	1.52	2.04	1.89	2.14	2.55	1.78	1.89	1.71	1.79	1.78	##	##	##	##	##
Spectral + CID	2.77	2.43	2.24	2.33	2.30	4.56	5.15	6.50	6.09	5.99	3.33	3.07	2.63	2.70	2.70	##	##	##	##	##
Spectral + CORT	1.36	1.29	1.35	1.30	1.30	1.70	2.71	3.11	4.01	5.42	1.65	1.44	1.67	1.58	1.67	##	##	##	##	##
Spectral + SBD	1.50	1.67	1.92	1.98	1.79	2.00	2.72	2.55	4.41	4.83	2.01	1.98	2.09	2.20	1.97	##	##	##	##	##
K-Shape + SBD	1.48	1.58	1.57	2.43	1.57	2.32	2.18	4.93	2.73	4.45	1.94	1.88	1.88	2.57	1.89	1.60	##	##	##	##
Datatrics		12.15					6.53					22.80					23.30			
Hierarchical + ED	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	56.26	46.22	56.16	48.42	43.11	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	1.71	**1.57**	##	##	2.06
Hierarchical + DTW	11.52	10.16	8.96	8.03	7.43	15.20	13.59	12.54	**11.59**	**11.11**	10.16	8.98	7.38	6.63	6.24	1.93	1.89	**1.88**	1.84	**1.84**
Hierarchical + CID	2.57	2.27	1.95	2.00	1.70	13.57	**11.41**	**10.40**	12.53	58.67	1.96	1.79	1.58	1.61	1.41	##	##	##	##	##
Hierarchical + CORT	9.98	7.99	6.70	5.98	5.37	16.45	54.78	46.72	55.27	48.57	10.73	8.81	7.53	6.84	6.05	**1.66**	1.99	##	##	2.34
Hierarchical + SBD	10.99	9.27	8.10	7.86	7.17	14.97	12.78	11.45	11.43	11.25	9.19	7.81	6.59	6.52	5.86	2.13	1.99	1.89	**1.83**	1.88
Spectral + ED	1.00	1.00	1.00	1.00	1.00	53.67	43.63	36.98	32.19	28.76	1.00	1.00	1.00	1.00	1.00	12.69	##	##	##	13.76
Spectral + DTW	8.00	8.03	6.13	6.76	6.73	**11.55**	12.93	10.45	12.15	11.17	6.06	6.37	4.63	5.33	5.41	##	##	##	##	##
Spectral + CID	4.58	4.33	4.19	4.55	4.25	26.22	16.50	16.80	16.24	21.56	9.30	8.63	7.94	8.42	7.39	##	##	16.29	##	##
Spectral + CORT	1.00	1.00	1.02	1.01	1.01	14.56	17.24	15.51	17.15	20.95	1.00	1.00	1.01	1.01	1.00	12.69	##	13.50	##	##
Spectral + SBD	2.86	2.74	2.69	2.65	2.46	29.95	111.45	96.81	86.15	75.64	1.79	1.98	1.94	1.88	1.81	##	##	##	##	##
K-Shape + SBD	6.28	6.01	6.26	5.64	5.60	23.32	23.14	26.79	20.98	27.56	11.86	12.61	11.69	10.44	10.23	##	##	##	##	##
Datatrics		12.63					35.39					8.91					3.77			
Hierarchical + ED	2.34	2.11	2.24	2.34	2.12	9.28	9.31	9.21	16.39	18.27	3.35	2.90	3.01	3.20	2.63	7.47	11.59	##	**54.32**	##
Hierarchical + DTW	31.00	26.97	25.17	22.79	20.83	52.07	43.88	39.72	36.04	32.40	52.17	42.88	36.80	32.42	28.44	##	##	##	##	##
Hierarchical + CID	1.88	2.26	2.10	1.97	2.02	5.56	5.68	5.35	6.37	8.88	1.54	1.85	1.74	1.64	1.67	##	##	##	##	##
Hierarchical + CORT	23.22	19.82	16.33	13.38	10.78	36.54	30.19	26.45	22.70	20.61	36.03	31.24	25.98	21.23	17.57	##	##	##	##	##
Hierarchical + SBD	24.44	22.37	20.62	18.54	17.75	37.65	33.88	31.43	28.09	26.76	35.05	30.94	27.39	23.99	21.68	##	##	##	##	##
Spectral + ED	2.03	2.09	1.93	2.42	2.26	**3.44**	**3.99**	**4.68**	**4.91**	**5.84**	2.45	2.49	2.25	3.89	3.58	##	##	##	##	##
Spectral + DTW	25.44	22.42	24.55	18.99	18.64	36.34	32.15	38.61	28.40	29.02	36.16	29.87	36.84	23.92	22.73	4.34	##	##	##	##
Spectral + CID	4.31	3.92	4.41	4.31	3.98	19.54	14.53	14.03	14.28	19.36	8.69	7.26	8.99	8.53	7.05	22.85	##	##	##	##
Spectral + CORT	**1.17**	**1.17**	**1.18**	**1.16**	**1.15**	7.02	7.72	6.45	6.62	7.96	**1.11**	**1.11**	**1.12**	**1.11**	**1.11**	##	##	##	##	##
Spectral + SBD	20.12	18.20	16.38	12.95	11.98	27.71	25.36	23.48	17.97	17.53	25.55	22.65	20.09	13.81	12.72	##	**3.46**	**3.13**	##	##
K-Shape + SBD	16.45	16.46	18.64	16.37	13.00	31.27	34.29	26.66	33.32	31.25	24.90	25.28	23.15	21.97	18.52	**2.97**	##	3.47	##	##
Datatrics		12.04					27.18					7.81					##			

Table 21: Davies-Bouldin index scores of company 2

	$d(x_i, x_j)$= ED					$d(x_i, x_j)$= DTW					$d(x_i, x_j)$= CID					$d(x_i, x_j)$= SBD				
	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8
CLV Time Series																				
Hierarchical + ED	0.49	0.50	0.47	0.47	0.46	0.49	0.49	0.45	0.45	0.42	0.49	0.50	0.47	0.48	0.46	0.61	0.65	0.61	0.61	0.59
Hierarchical + DTW	0.40	0.40	0.39	0.44	0.38	0.55	0.51	0.50	0.43	0.45	0.40	0.39	0.39	0.44	0.38	0.47	0.50	0.50	0.57	0.46
Hierarchical + CORT	0.66	0.50	0.50	0.50	0.50	0.60	0.50	0.50	0.50	0.49	0.68	0.50	0.51	0.51	0.51	0.79	0.79	0.78	0.77	0.76
Hierarchical + CID	0.51	0.49	0.48	0.45	0.45	0.49	0.48	0.47	0.43	0.43	0.52	0.49	0.49	0.46	0.46	0.64	0.63	0.62	0.58	0.59
Hierarchical + SBD	0.51	0.41	0.40	0.36	0.41	0.49	0.39	0.38	0.33	0.39	0.51	0.41	0.40	0.36	0.41	0.64	0.51	0.51	0.43	0.53
Spectral + ED	0.44	0.78	**0.79**	0.42	0.78	0.46	0.58	0.58	0.42	0.52	0.46	0.79	0.80	0.44	0.79	0.56	**0.80**	**0.80**	0.53	0.79
Spectral + DTW	**0.78**	0.78	0.78	0.78	0.78	**0.79**	**0.77**	**0.79**	**0.78**	**0.77**	0.81	**0.81**	**0.80**	0.79	0.79	0.78	0.78	0.77	0.77	0.77
Spectral + CORT	0.78	0.78	0.77	0.76	0.76	0.58	0.57	0.51	0.43	0.43	0.79	0.79	0.78	0.77	0.76	**0.80**	0.80	0.78	0.76	0.76
Spectral + CID	0.78	**0.78**	0.78	**0.79**	**0.78**	0.77	0.75	0.68	0.64	0.63	**0.81**	0.80	0.79	**0.80**	**0.79**	0.77	0.78	0.79	**0.79**	0.79
Spectral + SBD	0.69	0.69	0.68	0.70	0.75	0.72	0.70	0.61	0.60	0.52	0.72	0.72	0.72	0.73	0.76	0.78	0.78	0.78	0.78	**0.79**
K-Shape + SBD	0.37	0.40	0.31	0.26	0.25	0.42	0.44	0.36	0.31	0.31	0.37	0.39	0.30	0.25	0.24	0.46	0.47	0.29	0.21	0.17
Monetary Time Series																				
Hierarchical + ED	**0.06**	**0.07**	**0.08**	**0.10**	**0.11**	-0.08	-0.08	-0.08	-0.10	-0.17	0.05	0.06	**0.08**	**0.09**	0.09	0.00	0.00	0.00	-0.08	-0.13
Hierarchical + DTW	0.00	0.00	0.00	0.00	0.00	0.51	0.52	0.52	0.52	0.54	0.06	0.06	0.06	0.03	0.03	0.27	0.40	0.40	0.33	0.33
Hierarchical + CORT	0.06	0.07	0.08	0.10	0.10	-0.07	-0.08	-0.08	-0.08	-0.08	0.05	0.06	0.07	0.09	**0.10**	0.00	0.00	0.00	-0.01	-0.02
Hierarchical + CID	0.00	0.00	0.01	0.02	0.02	0.52	0.51	0.29	0.01	0.00	0.10	**0.09**	0.06	0.07	0.07	0.30	0.30	0.15	-0.29	-0.29
Hierarchical + SBD	0.00	0.00	0.00	-0.01	-0.01	0.49	0.45	0.44	0.45	0.41	0.05	0.03	0.01	0.01	0.00	0.49	0.39	0.33	0.33	0.34
Spectral + ED	0.06	0.06	0.08	0.08	0.09	-0.21	-0.20	-0.21	-0.27	-0.26	0.03	0.04	0.06	0.06	0.06	-0.22	-0.22	-0.22	-0.38	-0.38
Spectral + DTW	0.00	0.00	0.00	0.00	0.00	**0.60**	**0.56**	**0.57**	**0.55**	**0.55**	**0.11**	0.06	0.05	0.03	0.02	-0.68	-0.51	-0.61	-0.52	-0.49
Spectral + CORT	0.04	0.05	0.06	0.06	0.07	-0.08	-0.09	-0.11	-0.12	-0.12	0.05	0.07	0.07	0.09	**0.10**	-0.02	-0.02	-0.07	-0.09	-0.09
Spectral + CID	0.04	0.05	0.06	0.06	0.07	-0.19	-0.20	-0.25	-0.38	-0.39	0.02	0.03	0.04	0.03	0.04	-0.31	-0.31	-0.42	-0.58	-0.58
Spectral + SBD	-0.01	-0.01	-0.01	-0.01	-0.01	0.24	0.26	0.27	0.28	0.28	0.02	-0.02	-0.01	-0.01	-0.01	0.58	0.47	**0.49**	**0.50**	**0.51**
K-Shape + SBD	-0.01	-0.01	0.00	-0.01	0.00	0.29	0.29	0.02	0.30	0.12	0.02	0.02	0.00	0.02	0.00	**0.60**	**0.62**	0.19	0.63	0.28
Pageview Time Series																				
Hierarchical + ED	**0.06**	**0.06**	**0.07**	**0.08**	**0.09**	-0.02	-0.04	-0.05	-0.05	-0.05	0.05	0.06	0.07	**0.07**	**0.08**	0.02	-0.02	-0.02	-0.04	-0.04
Hierarchical + DTW	0.00	0.00	0.00	0.00	0.00	0.35	0.36	0.34	0.28	0.31	0.05	0.04	0.03	0.01	0.01	0.21	0.21	0.22	0.11	0.14
Hierarchical + CORT	0.03	0.04	0.06	0.06	0.07	0.01	0.00	-0.01	-0.01	-0.04	0.04	0.05	0.06	0.07	0.08	0.07	0.06	0.05	0.05	0.00
Hierarchical + CID	0.00	0.00	-0.01	-0.01	-0.01	0.30	0.29	0.25	0.23	0.23	0.12	**0.11**	**0.08**	0.06	0.06	-0.07	-0.07	0.03	0.03	0.03
Hierarchical + SBD	0.00	0.00	0.00	-0.01	-0.01	0.31	0.29	0.29	0.27	0.28	0.02	0.01	0.01	0.00	0.00	0.29	0.22	0.22	0.23	0.23
Spectral + ED	0.04	0.05	0.05	0.06	0.06	-0.15	-0.16	-0.21	-0.21	-0.21	0.01	0.01	0.01	0.02	0.02	-0.20	-0.25	-0.33	-0.33	-0.33
Spectral + DTW	0.00	0.00	0.00	-0.01	-0.01	**0.40**	**0.35**	**0.35**	**0.34**	**0.35**	**0.14**	0.06	0.06	0.03	0.01	-0.56	-0.45	-0.50	-0.51	-0.46
Spectral + CORT	0.04	0.05	0.07	0.07	**0.09**	-0.06	-0.08	-0.09	-0.11	-0.12	0.04	0.05	0.06	0.06	0.07	-0.07	-0.09	-0.11	-0.15	-0.12
Spectral + CID	0.03	0.02	0.03	0.03	0.05	-0.22	-0.22	-0.21	-0.30	-0.30	-0.01	-0.01	0.00	0.00	0.01	-0.42	-0.41	-0.43	-0.53	-0.52
Spectral + SBD	0.00	-0.01	-0.01	-0.01	-0.01	0.24	0.23	0.25	0.24	0.24	-0.01	-0.02	-0.02	-0.02	-0.02	0.33	0.35	**0.38**	**0.39**	**0.38**
K-Shape + SBD	-0.01	-0.01	-0.01	-0.01	-0.02	0.22	0.22	0.23	0.23	0.23	0.00	0.00	-0.02	-0.02	-0.03	**0.42**	**0.42**	0.35	0.37	0.37

Table 22: Silhouette index scores of company 3

	$d(x_i,x_j)$= ED					$d(x_i,x_j)$= DTW					$d(x_i,x_j)$= CID					$d(x_i,x_j)$= SBD				
	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8
CLV Time Series																				
Hierarchical + ED	1.34	1.40	1.41	1.33	1.25	2.01	4.94	5.49	4.85	3.80	**1.57**	1.89	1.95	1.79	**1.61**	##	##	##	##	##
Hierarchical + DTW	1.55	1.72	1.71	1.62	1.58	1.81	**2.02**	4.91	4.89	4.40	1.78	1.95	2.06	1.90	1.81	##	##	##	##	##
Hierarchical + CORT	**1.23**	**1.27**	1.44	1.43	1.40	**1.48**	2.03	3.67	3.37	**2.24**	1.93	1.95	1.87	1.80	1.74	##	##	##	##	##
Hierarchical + CID	1.53	1.38	1.39	1.31	1.25	2.53	2.19	4.85	4.31	3.89	1.95	1.70	1.92	1.76	1.64	##	##	##	##	##
Hierarchical + SBD	3.51	3.33	3.01	2.44	2.73	3.51	3.36	2.98	3.62	3.86	3.51	3.42	3.07	2.59	2.91	##	3.59	3.28	##	**2.33**
Spectral + ED	1.36	1.29	**1.35**	**1.30**	1.29	1.69	2.69	3.66	3.79	5.26	1.65	**1.43**	1.72	**1.56**	1.63	**1.55**	**1.97**	**2.42**	**2.78**	##
Spectral + DTW	1.50	1.55	1.46	1.71	1.53	1.52	2.04	**1.89**	**2.14**	2.55	1.78	1.89	**1.71**	1.79	1.78	##	##	##	##	##
Spectral + CORT	2.77	2.43	2.24	2.33	2.30	4.56	5.15	6.50	6.09	5.99	3.33	3.07	2.63	2.70	2.70	##	##	##	##	##
Spectral + CID	1.42	1.46	1.37	1.32	**1.24**	3.09	3.78	3.80	5.85	5.91	1.62	1.93	1.72	1.72	1.58	2.28	##	##	##	##
Spectral + SBD	1.50	1.67	1.92	1.98	1.79	2.00	2.72	2.55	4.41	4.83	2.01	1.98	2.09	2.20	1.97	##	##	##	##	##
K-Shape + SBD	1.48	1.58	1.57	2.43	1.57	2.32	2.18	4.93	2.73	4.45	1.94	1.88	1.88	2.57	1.89	1.60	##	##	##	##
Datatrics		12.15					6.53					22.80					23.30			
Monetary Time Series																				
Hierarchical + ED	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	56.26	46.22	56.16	48.42	43.11	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	1.71	**1.57**	##	##	2.06
Hierarchical + DTW	11.52	10.16	8.96	8.03	7.43	15.20	13.59	12.54	**11.59**	**11.11**	10.16	8.98	7.38	6.63	6.24	1.93	1.89	**1.88**	1.84	**1.84**
Hierarchical + CORT	2.58	2.28	1.95	2.01	1.71	12.96	**10.92**	**4.71**	13.95	10.10	1.96	1.80	1.58	1.61	1.41	##	##	##	##	##
Hierarchical + CID	9.98	7.99	6.70	5.98	5.37	16.45	54.78	46.72	55.27	48.57	10.73	8.81	7.53	6.84	6.05	**1.66**	1.99	##	##	2.34
Hierarchical + SBD	10.99	9.27	8.10	7.86	7.17	14.97	12.78	11.45	11.43	11.25	9.19	7.81	6.59	6.52	5.86	2.13	1.99	1.89	**1.83**	1.88
Spectral + ED	1.00	1.00	1.00	1.00	1.00	53.67	43.63	36.98	32.19	28.76	1.00	1.00	1.00	1.00	1.00	12.69	##	##	##	13.76
Spectral + DTW	8.00	8.03	6.13	6.76	6.73	**11.55**	12.93	10.45	12.15	11.17	6.06	6.37	4.63	5.33	5.41	##	##	##	##	##
Spectral + CORT	1.69	1.55	1.68	1.39	1.35	54.15	44.08	42.46	32.68	29.28	1.41	1.33	1.43	1.24	1.22	##	##	16.29	##	##
Spectral + CID	1.00	1.00	1.02	1.01	1.01	14.56	17.24	15.51	17.15	20.95	1.00	1.00	1.01	1.01	1.00	12.69	##	13.50	##	##
Spectral + SBD	2.86	2.74	2.69	2.65	2.46	29.95	111.45	96.81	86.15	75.64	1.79	1.98	1.94	1.88	1.81	##	##	##	##	##
K-Shape + SBD	6.28	6.01	6.26	5.64	5.60	23.32	23.14	26.79	20.98	27.56	11.86	12.61	11.69	10.44	10.23	##	##	##	##	##
Datatrics		12.63					35.39					8.91					3.77			
Pageview Time Series																				
Hierarchical + ED	2.34	2.11	2.24	2.34	2.12	9.28	9.31	9.21	16.39	18.27	3.35	2.90	3.01	3.20	2.63	7.47	11.59	##	**54.32**	##
Hierarchical + DTW	31.00	26.97	25.17	22.79	20.83	52.07	43.88	39.72	36.04	32.40	52.17	42.88	36.80	32.42	28.44	##	##	##	##	##
Hierarchical + CORT	2.21	1.76	1.92	1.63	1.32	5.20	4.59	8.83	5.94	6.88	1.83	1.49	1.60	1.41	1.21	##	##	##	##	##
Hierarchical + CID	23.22	19.82	16.33	13.38	10.78	36.54	30.19	26.45	22.70	20.61	36.03	31.24	25.98	21.23	17.57	##	##	##	##	##
Hierarchical + SBD	24.44	22.37	20.62	18.54	17.75	37.65	33.88	31.43	28.09	26.76	35.05	30.94	27.39	23.99	21.68	##	##	##	##	##
Spectral + ED	2.03	2.09	1.93	2.42	2.26	**3.44**	**3.99**	**4.68**	**4.91**	**5.84**	2.45	2.49	2.25	3.89	3.58	##	##	##	##	##
Spectral + DTW	25.44	22.42	24.55	18.99	18.64	36.34	32.15	38.61	28.40	29.02	36.16	29.87	36.84	23.92	22.73	4.34	##	##	##	##
Spectral + CORT	2.50	2.58	2.97	2.77	2.53	4.03	8.32	9.52	8.25	6.43	2.89	4.93	5.15	3.28	2.99	##	##	##	##	##
Spectral + CID	**1.17**	**1.17**	**1.18**	**1.16**	**1.15**	7.02	7.72	6.45	6.62	7.96	**1.11**	**1.11**	**1.12**	**1.11**	**1.11**	##	##	##	##	##
Spectral + SBD	20.12	18.20	16.38	12.95	11.98	27.71	25.36	23.48	17.97	17.53	25.55	22.65	20.09	13.81	12.72	##	**3.46**	**3.13**	##	##
K-Shape + SBD	16.45	16.46	18.64	16.37	13.00	31.27	34.29	26.66	33.32	31.25	24.90	25.28	23.15	21.97	18.52	**2.97**	##	3.47	##	##
Datatrics		12.04					27.18					7.81					##			

Table 23: Davies-Bouldin index scores of company 3

	$d(x_i, x_j)$= ED					$d(x_i, x_j)$= DTW					$d(x_i, x_j)$= CID					$d(x_i, x_j)$= SBD				
	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8
CLV Time Series																				
Hierarchical + ED	0.43	0.47	0.48	0.44	0.45	0.44	0.48	0.45	0.41	0.40	0.44	0.49	0.49	0.45	0.46	0.53	0.65	0.62	0.58	0.56
Hierarchical + DTW	0.43	0.42	0.43	0.43	0.40	0.49	0.48	0.44	0.44	0.41	0.45	0.44	0.45	0.44	0.42	0.62	0.63	0.59	0.59	0.56
Hierarchical + CORT	**0.66**	**0.66**	**0.67**	**0.67**	**0.68**	0.55	**0.54**	**0.53**	**0.52**	**0.53**	**0.70**	**0.68**	**0.69**	**0.69**	**0.68**	0.61	0.60	0.60	0.60	0.60
Hierarchical + CID	0.49	0.43	0.44	0.43	0.43	0.50	0.45	0.41	0.40	0.40	0.52	0.45	0.45	0.45	0.45	0.67	0.59	0.56	0.59	0.59
Hierarchical + SBD	0.45	0.44	0.44	0.44	0.42	**0.59**	**0.54**	0.51	**0.52**	0.49	0.49	0.47	0.46	0.46	0.44	**0.74**	**0.69**	0.62	0.66	**0.63**
Spectral + ED	0.40	0.42	0.51	0.51	0.51	0.33	0.33	0.42	0.41	0.39	0.40	0.42	0.55	0.55	0.55	0.31	0.33	0.55	0.54	0.52
Spectral + DTW	0.39	0.39	0.39	0.40	0.41	0.52	0.50	0.51	**0.52**	**0.53**	0.41	0.40	0.40	0.42	0.43	0.55	0.54	0.55	0.59	0.58
Spectral + CORT	0.61	0.50	0.51	0.52	0.50	0.56	0.49	0.47	0.47	0.44	0.64	0.54	0.54	0.55	0.51	0.70	0.66	0.65	0.63	0.59
Spectral + CID	0.43	0.44	0.45	0.46	0.53	0.36	0.37	0.38	0.38	0.42	0.44	0.46	0.47	0.47	0.57	0.38	0.41	0.45	0.48	0.56
Spectral + SBD	0.44	0.43	0.43	0.43	0.44	0.48	0.50	0.51	0.49	0.47	0.46	0.45	0.45	0.46	0.46	0.60	0.61	0.63	0.63	**0.63**
K-Shape + SBD	0.41	0.45	0.44	0.45	0.33	0.46	0.49	0.50	0.48	0.41	0.42	0.50	0.47	0.48	0.36	0.58	0.68	**0.65**	**0.68**	0.54
Monetary Time Series																				
Hierarchical + ED	**0.07**	**0.09**	**0.10**	**0.12**	**0.13**	-0.90	-0.90	-0.90	-0.90	-0.90	0.06	0.08	0.09	**0.11**	0.12	-0.77	-0.80	-0.80	-0.80	-0.80
Hierarchical + DTW	0.01	0.00	0.00	-0.01	-0.01	**0.95**	**0.95**	0.95	0.95	0.95	0.11	0.08	0.07	0.07	0.07	0.17	0.17	0.17	0.17	0.17
Hierarchical + CORT	0.07	**0.09**	0.10	**0.12**	**0.13**	-0.53	-0.53	-0.53	-0.53	-0.53	0.06	0.08	0.09	**0.11**	**0.13**	-0.41	-0.40	-0.40	-0.40	-0.40
Hierarchical + CID	0.04	0.05	0.07	0.09	0.10	-0.87	-0.88	-0.88	-0.86	-0.85	0.04	0.06	0.08	0.09	0.11	-0.51	-0.50	-0.49	-0.54	-0.51
Hierarchical + SBD	-0.01	-0.01	-0.01	-0.02	-0.02	0.91	0.89	0.89	0.89	0.89	**0.12**	0.07	0.07	0.07	0.07	0.92	0.88	0.88	0.88	0.88
Spectral + ED	**0.07**	**0.09**	**0.10**	**0.12**	**0.13**	-0.73	-0.73	-0.73	-0.82	-0.81	0.06	0.08	0.09	**0.11**	0.12	-0.52	-0.52	-0.56	-0.68	-0.67
Spectral + DTW	0.01	0.00	0.00	-0.01	-0.01	0.94	**0.95**	**0.96**	**0.96**	**0.96**	0.15	**0.13**	**0.12**	0.07	0.07	-0.42	0.02	-0.01	-0.01	0.04
Spectral + CORT	0.05	0.06	0.07	0.08	0.10	-0.02	-0.25	-0.28	-0.37	-0.37	0.05	0.06	0.07	0.08	0.09	-0.61	-0.61	-0.60	-0.60	-0.59
Spectral + CID	0.06	0.08	0.10	0.11	**0.13**	-0.33	-0.33	-0.64	-0.64	-0.64	0.06	0.08	0.09	**0.11**	0.12	-0.62	-0.62	-0.62	-0.62	-0.64
Spectral + SBD	-0.01	-0.02	-0.03	-0.04	-0.04	0.85	0.85	0.86	0.86	0.86	0.11	0.11	0.10	0.09	0.07	**0.94**	**0.93**	**0.94**	**0.94**	**0.93**
K-Shape + SBD	0.01	0.01	0.02	0.03	0.03	0.24	-0.14	-0.21	-0.12	-0.23	0.02	0.02	0.03	0.04	0.04	0.27	0.16	0.10	0.08	0.04
Pageview Time Series																				
Hierarchical + ED	**0.05**	**0.06**	**0.08**	**0.09**	**0.10**	-0.25	-0.33	-0.33	-0.33	-0.33	0.05	0.06	**0.07**	**0.08**	**0.09**	-0.21	-0.46	-0.46	-0.46	-0.46
Hierarchical + DTW	0.00	0.00	0.00	-0.03	-0.03	0.70	0.66	0.65	0.66	0.67	0.03	0.01	0.01	0.01	0.01	0.63	0.43	0.43	0.43	0.45
Hierarchical + CORT	**0.05**	**0.06**	**0.08**	**0.09**	**0.10**	-0.21	-0.22	-0.22	-0.39	-0.40	0.05	0.06	0.07	**0.08**	**0.09**	-0.26	-0.27	-0.27	-0.43	-0.51
Hierarchical + CID	0.01	0.01	0.03	0.04	0.05	0.03	0.03	-0.51	-0.51	-0.51	0.03	0.03	0.04	0.06	0.07	0.02	0.03	-0.58	-0.58	-0.57
Hierarchical + SBD	0.00	-0.01	-0.01	-0.29	-0.29	0.72	0.72	0.69	0.69	0.69	**0.07**	**0.07**	0.03	-0.29	-0.28	**0.74**	0.74	0.63	0.63	0.62
Spectral + ED	**0.05**	**0.06**	**0.08**	**0.09**	**0.10**	-0.32	-0.35	-0.35	-0.37	-0.37	0.04	0.05	**0.07**	0.08	**0.09**	-0.28	-0.38	-0.39	-0.54	-0.53
Spectral + DTW	0.00	0.00	-0.01	-0.01	-0.01	**0.74**	**0.74**	**0.74**	**0.74**	**0.74**	0.04	0.04	0.02	0.02	0.04	0.74	0.69	0.71	0.72	**0.74**
Spectral + CORT	**0.05**	**0.06**	0.07	0.07	0.08	-0.08	-0.08	-0.08	-0.13	-0.10	0.04	0.05	0.06	0.07	0.08	-0.06	-0.06	-0.08	-0.15	-0.11
Spectral + CID	**0.05**	**0.07**	**0.08**	**0.09**	**0.10**	-0.46	-0.46	-0.47	-0.47	-0.47	0.04	0.05	0.06	0.08	**0.09**	-0.55	-0.54	-0.55	-0.55	-0.55
Spectral + SBD	0.00	-0.01	-0.01	-0.02	-0.02	0.65	0.65	0.64	0.62	0.64	0.04	0.07	0.06	0.06	0.03	**0.74**	**0.78**	**0.78**	**0.77**	0.71
K-Shape + SBD	0.00	0.00	0.00	0.01	0.00	0.69	0.11	0.08	0.02	0.05	0.03	0.01	0.01	0.01	0.01	0.73	0.19	0.18	0.12	0.17

Table 24: Silhouette index scores of company 4

	$d(x_i, x_j)$= ED					$d(x_i, x_j)$= DTW					$d(x_i, x_j)$= CID					$d(x_i, x_j)$= SBD				
	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8	k=4	k=5	k=6	k=7	k=8
CLV Time Series																				
Hierarchical + ED	2.09	1.91	1.91	1.83	1.76	7.13	5.94	5.69	5.07	9.06	3.96	3.43	3.06	2.86	2.47	##	##	##	##	##
Hierarchical + DTW	2.15	2.03	2.28	2.53	2.42	3.18	2.82	6.92	11.10	9.90	3.38	3.07	3.66	4.32	4.03	##	##	##	##	##
Hierarchical + CORT	1.16	**1.16**	**1.15**	**1.14**	**1.13**	3.76	4.24	4.34	5.60	6.10	**1.14**	**1.17**	**1.19**	**1.21**	**1.24**	9.33	9.09	8.64	9.21	##
Hierarchical + CID	2.18	1.99	2.04	1.96	1.91	8.57	7.09	7.88	6.96	6.29	4.16	3.65	3.32	3.11	2.94	##	##	**2.31**	**2.22**	**2.18**
Hierarchical + SBD	1.12	2.68	7.65	7.51	6.76	1.13	2.91	11.69	11.68	10.39	1.39	4.31	13.12	12.64	11.29	##	##	##	##	7.41
Spectral + ED	**1.09**	1.67	1.66	1.60	1.39	1.11	2.04	3.24	**4.72**	4.89	**1.25**	2.19	1.95	1.82	1.48	##	##	##	##	##
Spectral + DTW	6.50	9.87	3.45	3.53	5.29	7.28	14.54	4.70	4.65	8.71	13.13	16.95	6.11	6.01	9.45	##	##	##	##	4.54
Spectral + CORT	2.12	1.90	1.81	1.86	1.74	8.22	7.11	5.86	8.61	7.69	3.66	3.16	2.67	3.01	2.78	**2.53**	**2.88**	##	##	##
Spectral + CID	1.10	1.68	1.68	1.56	1.51	**1.12**	**2.02**	**2.84**	**4.26**	**4.14**	1.27	2.19	2.14	1.92	1.64	##	##	##	##	##
Spectral + SBD	2.38	4.95	4.04	3.63	3.40	6.65	8.33	6.28	6.53	7.59	4.54	8.13	6.85	6.12	5.40	##	##	##	##	##
K-Shape + SBD	2.12	3.54	3.42	3.30	3.11	3.67	6.34	7.32	7.75	5.36	3.40	5.39	5.34	4.98	4.73	##	4.35	##	5.23	##
Monetary Time Series																				
Hierarchical + ED	1.03	**1.02**	1.04	1.04	1.05	4.40	24.41	22.20	19.59	18.28	1.01	**1.00**	**1.01**	1.02	1.02	##	##	##	##	##
Hierarchical + DTW	16.24	14.42	12.73	11.65	10.65	22.47	19.81	17.74	16.04	14.53	19.71	15.65	12.69	11.51	10.08	##	##	##	##	##
Hierarchical + CORT	2.08	1.59	1.50	1.63	1.40	7.15	6.69	6.61	17.21	7.07	1.63	1.32	1.27	1.35	1.22	##	##	##	##	##
Hierarchical + CID	12.85	10.07	8.29	7.25	6.30	18.13	18.65	15.96	27.47	25.22	17.51	13.94	11.57	10.14	8.78	2.53	##	##	##	##
Hierarchical + SBD	15.25	14.54	12.66	11.67	10.59	20.80	20.40	19.15	17.77	17.17	14.46	14.21	11.95	10.65	9.73	##	##	##	##	##
Spectral + ED	1.02	1.03	1.03	1.02	1.02	4.19	4.45	7.81	7.23	7.30	**1.00**	1.01	**1.01**	**1.00**	**1.00**	##	##	##	##	##
Spectral + DTW	10.54	11.77	9.99	9.60	9.53	13.08	15.68	14.30	14.34	14.64	9.44	11.25	8.56	8.12	8.10	**2.43**	**2.27**	##	##	##
Spectral + CORT	1.06	1.04	1.04	1.04	1.04	4.45	4.60	4.48	4.56	5.96	1.02	1.01	1.02	1.01	1.02	##	##	##	##	##
Spectral + CID	**1.02**	1.02	**1.02**	**1.02**	**1.02**	**3.38**	**3.78**	**3.87**	**6.83**	**6.72**	1.21	1.17	1.14	1.12	1.11	14.99	##	##	##	##
Spectral + SBD	6.58	6.06	5.60	5.26	5.76	7.64	7.76	7.36	6.95	6.85	5.07	4.71	4.37	4.14	4.51	##	##	##	##	##
K-Shape + SBD	10.29	8.82	8.11	7.61	7.61	14.81	26.55	26.92	22.85	21.52	16.40	16.09	14.27	13.53	12.70	##	##	##	##	##
Pageview Time Series																				
Hierarchical + ED	1.33	1.28	1.62	1.90	1.81	4.28	4.23	**4.11**	**4.78**	**5.72**	1.20	1.17	1.62	2.19	2.06	##	##	##	##	##
Hierarchical + DTW	34.37	33.57	30.74	26.78	24.69	52.30	52.67	48.46	42.00	37.29	57.20	55.38	49.21	42.41	36.34	##	##	##	##	##
Hierarchical + CORT	1.29	1.77	1.65	**1.25**	1.53	3.57	**3.59**	4.39	8.28	7.69	1.18	1.46	1.39	1.15	1.31	##	##	##	##	##
Hierarchical + CID	20.79	19.14	15.50	13.24	11.13	31.94	27.50	23.47	20.96	20.79	35.01	28.80	23.49	20.38	17.13	##	##	##	##	##
Hierarchical + SBD	27.17	23.77	26.49	22.80	21.44	39.72	34.52	40.23	34.51	32.32	41.05	34.16	39.55	33.97	30.62	##	##	##	##	##
Spectral + ED	2.01	1.59	1.54	1.47	**1.16**	7.37	6.63	6.13	6.72	8.34	2.17	1.62	1.55	1.48	**1.10**	##	##	##	##	##
Spectral + DTW	30.68	28.53	24.79	22.30	18.89	46.44	42.80	37.59	33.59	28.82	48.30	45.86	35.85	30.12	23.76	##	**4.21**	**3.62**	##	##
Spectral + CORT	2.95	2.42	2.20	2.05	1.86	4.13	8.96	7.89	9.59	7.99	3.66	3.22	2.91	2.65	2.29	##	##	8.28	##	##
Spectral + CID	**1.14**	**1.13**	**1.12**	1.64	1.41	**3.82**	6.25	6.74	6.51	6.59	**1.08**	**1.07**	**1.07**	1.94	1.45	##	##	##	##	##
Spectral + SBD	23.52	18.27	15.99	13.87	15.05	37.48	25.20	22.46	21.87	19.32	31.89	25.89	21.65	17.10	18.75	##	##	##	##	##
K-Shape + SBD	30.24	19.81	18.04	16.89	18.45	44.39	39.70	35.93	32.08	32.22	47.10	33.54	28.92	26.52	26.40	##	##	##	##	##

Table 25: Davies-Bouldin index scores of company 4

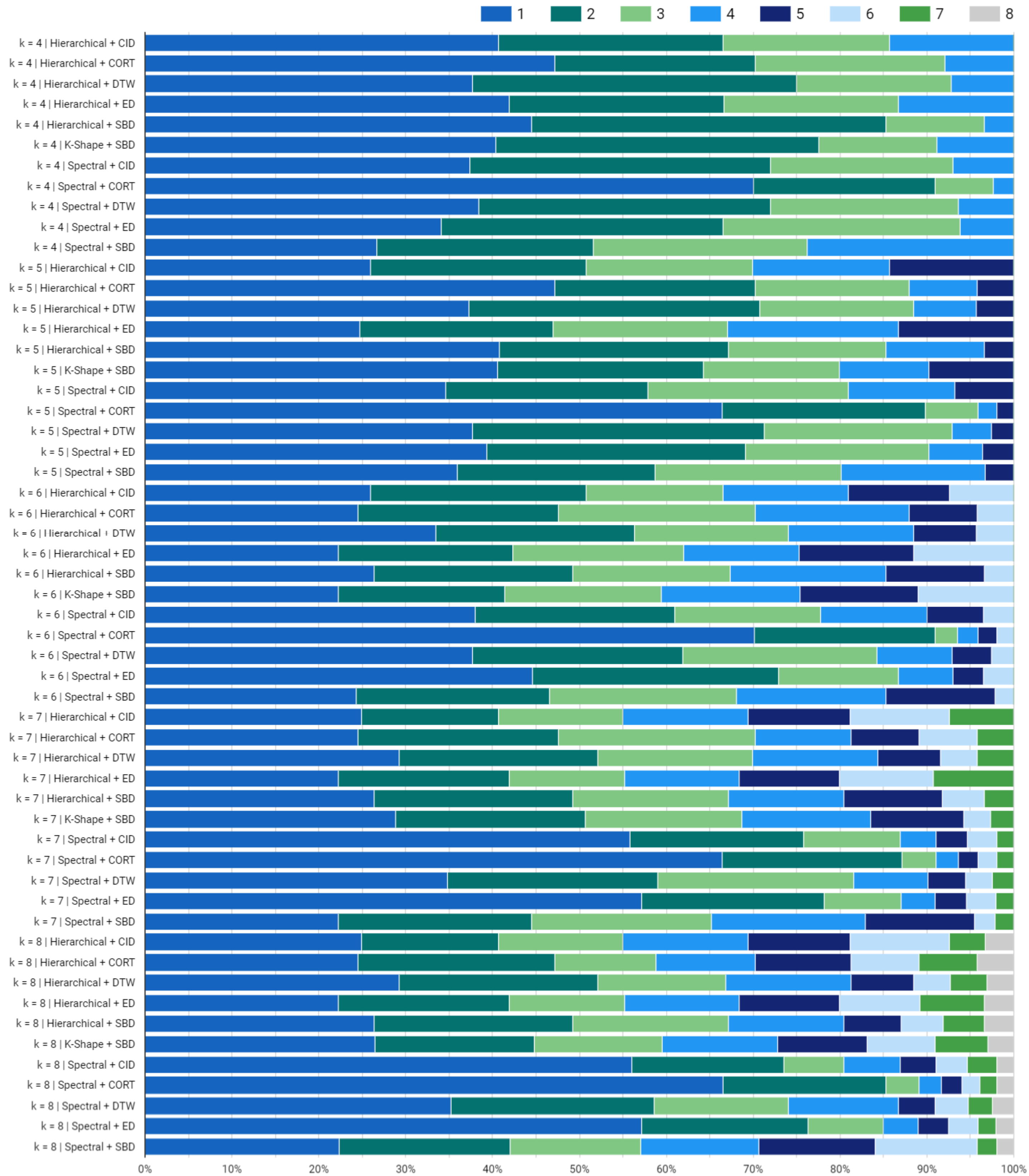


Table 26: Cluster partition based on the CLV feature for company 1

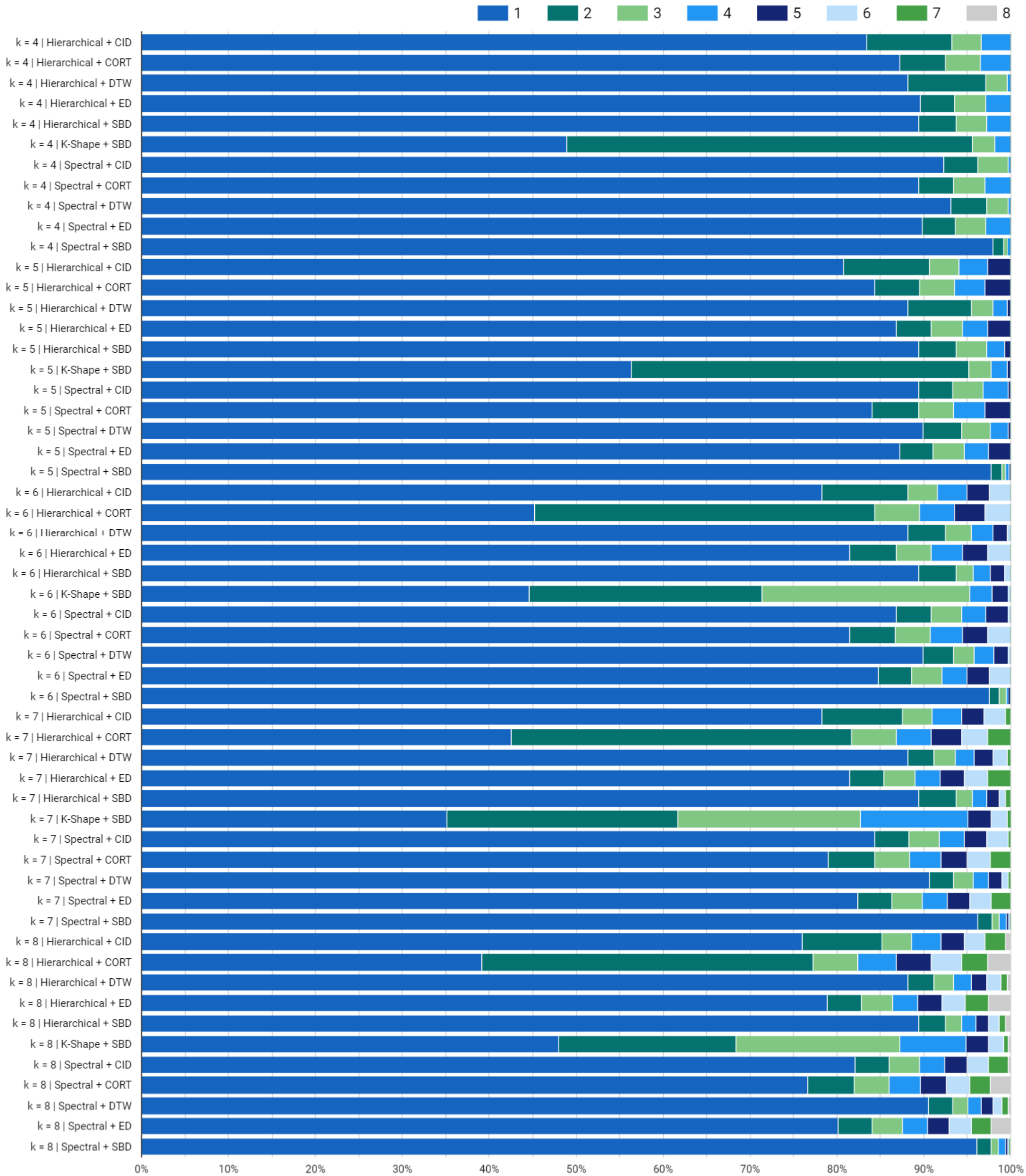


Table 27: Cluster partition based on the monetary value feature for company 1

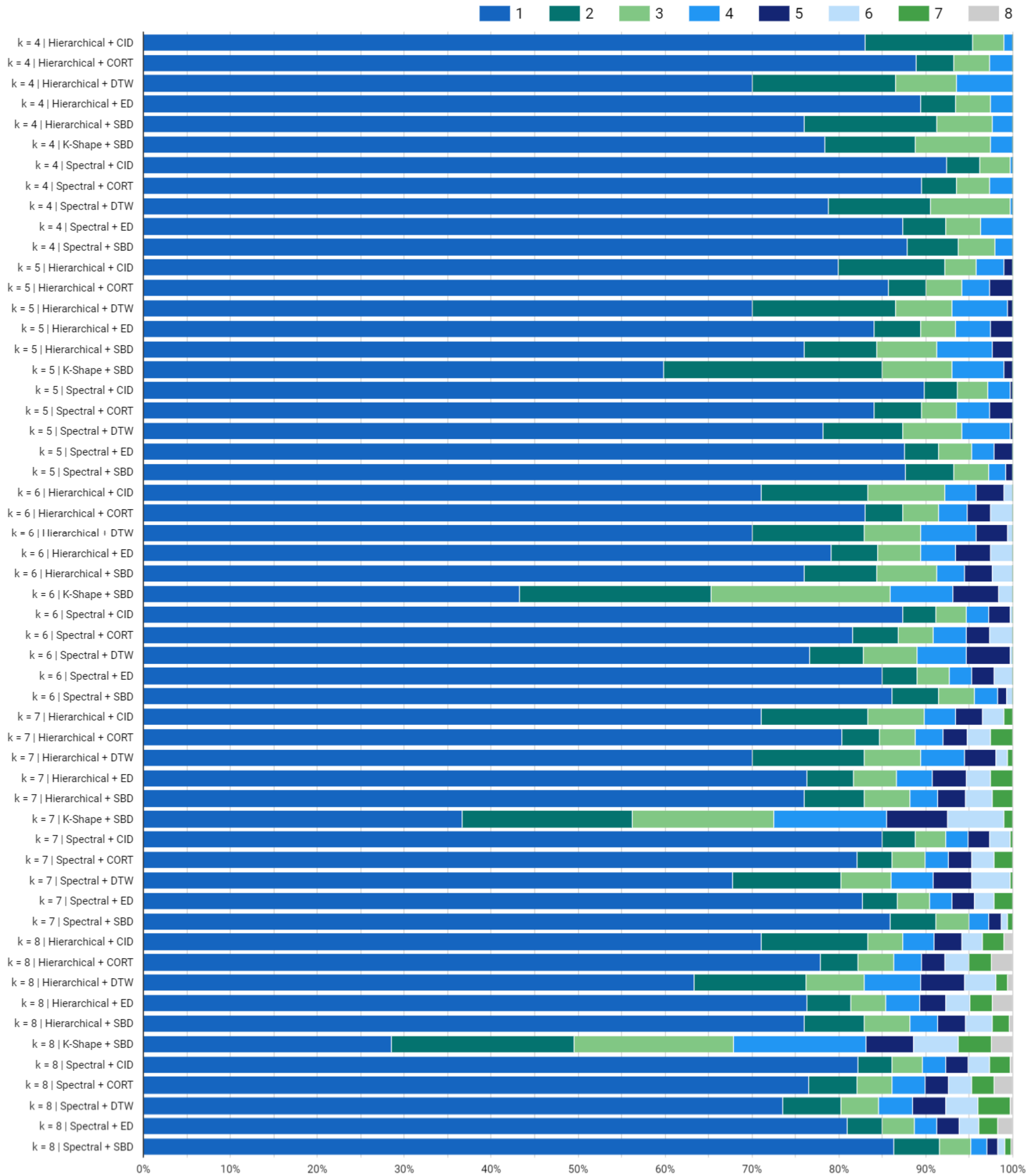


Table 28: Cluster partition based on the pageview feature for company 1

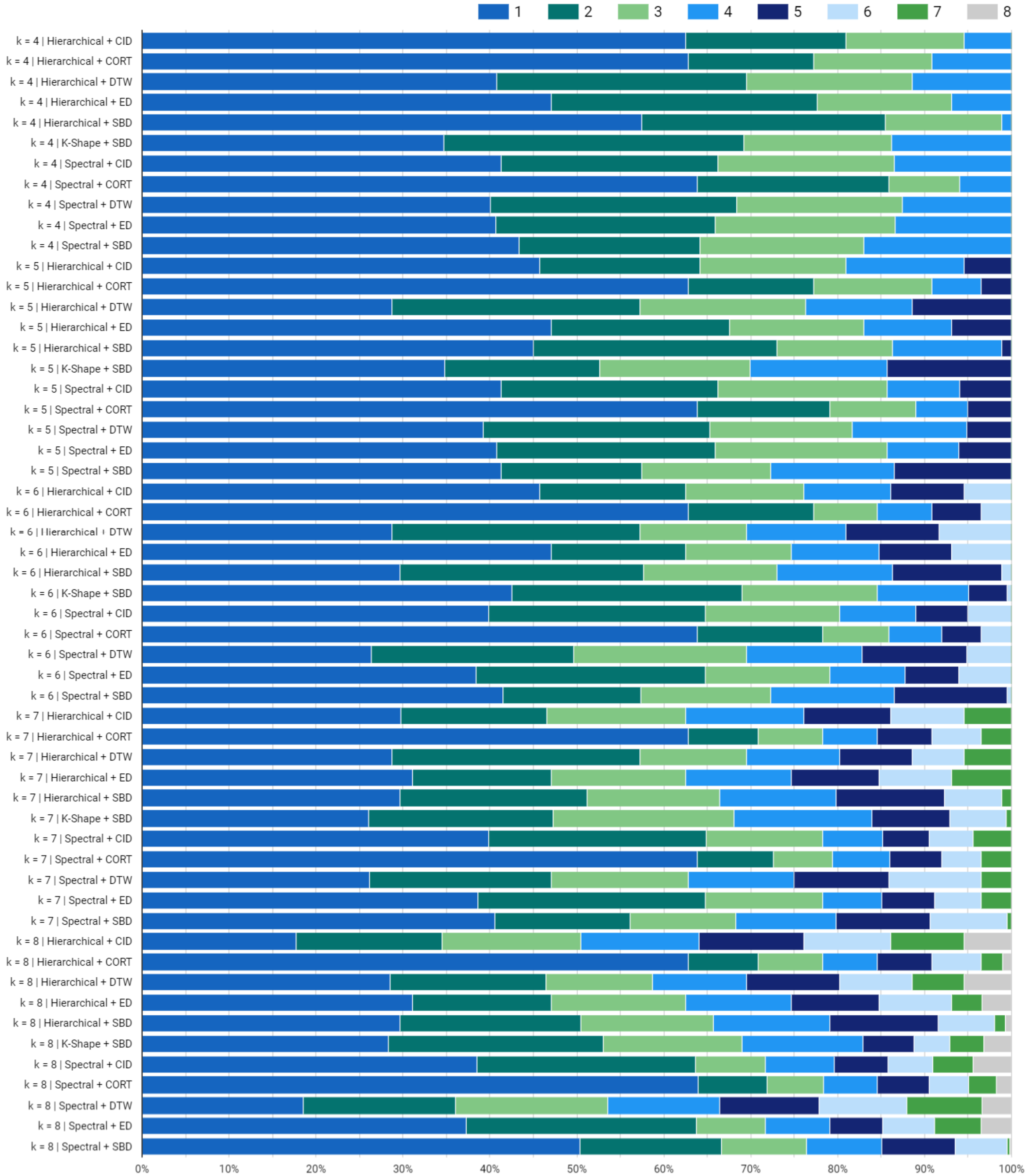


Table 29: Cluster partition based on the CLV feature for company 2

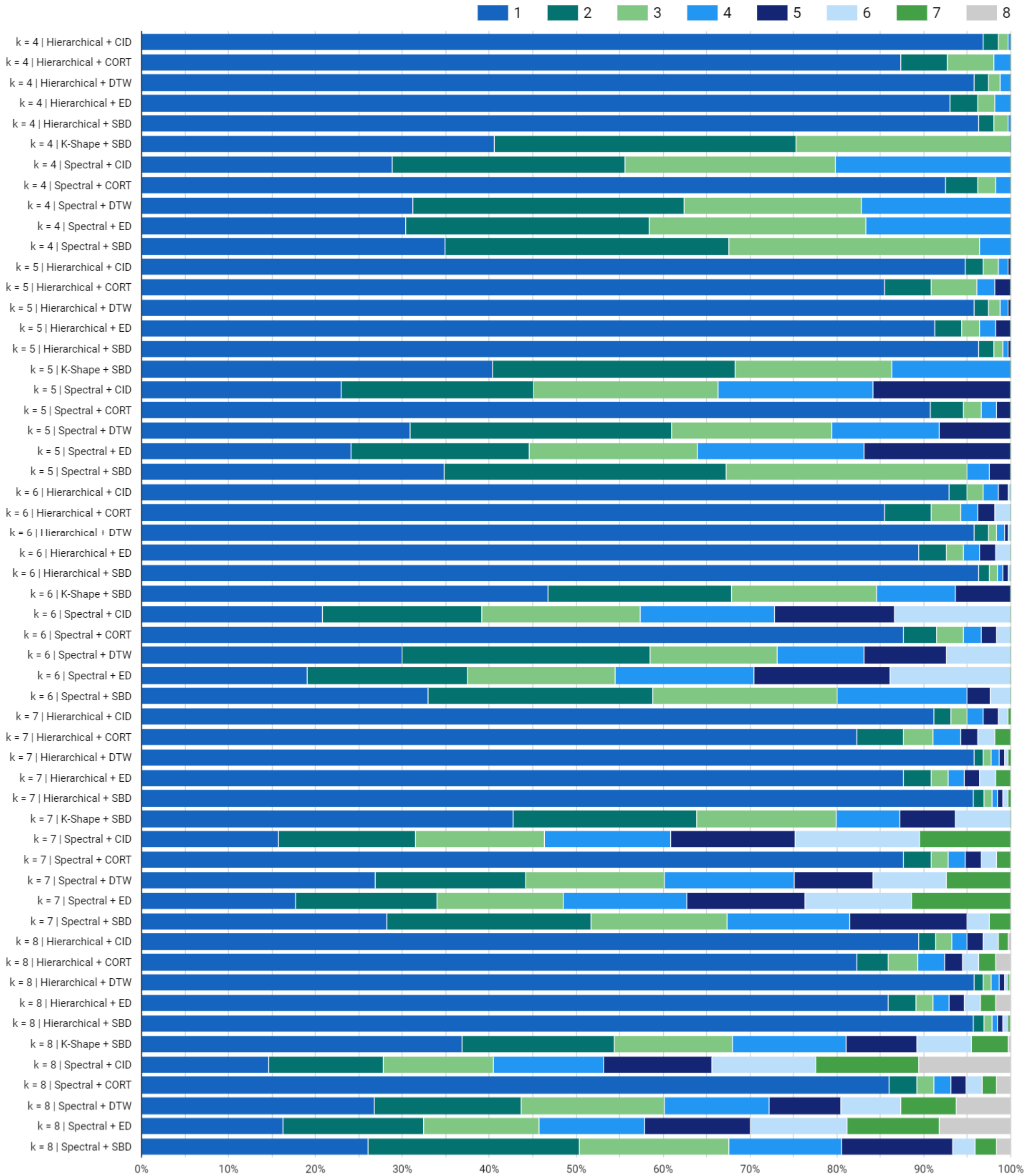


Table 30: Cluster partition based on the monetary value feature for company 2

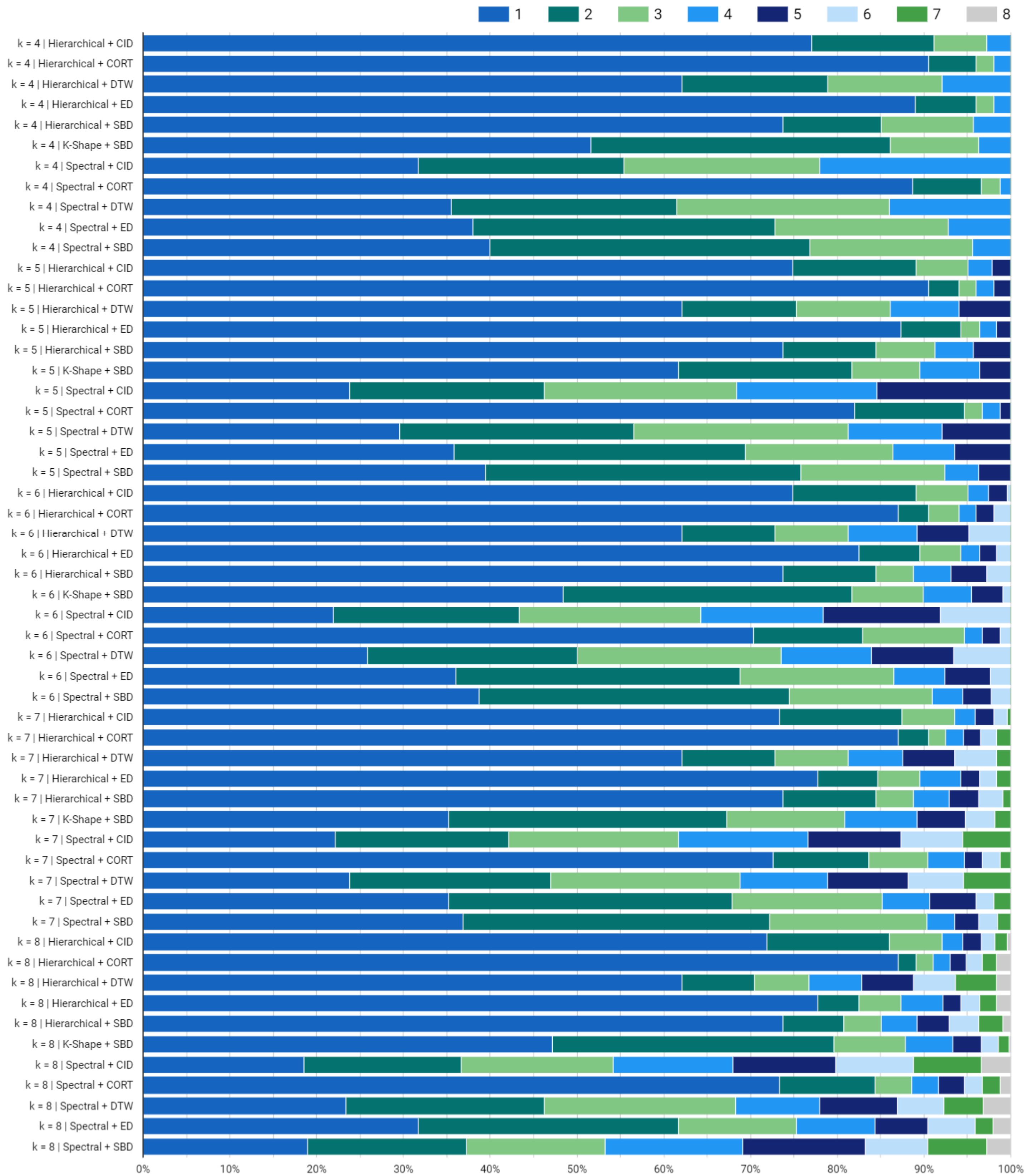


Table 31: Cluster partition based on the pageview feature for company 2

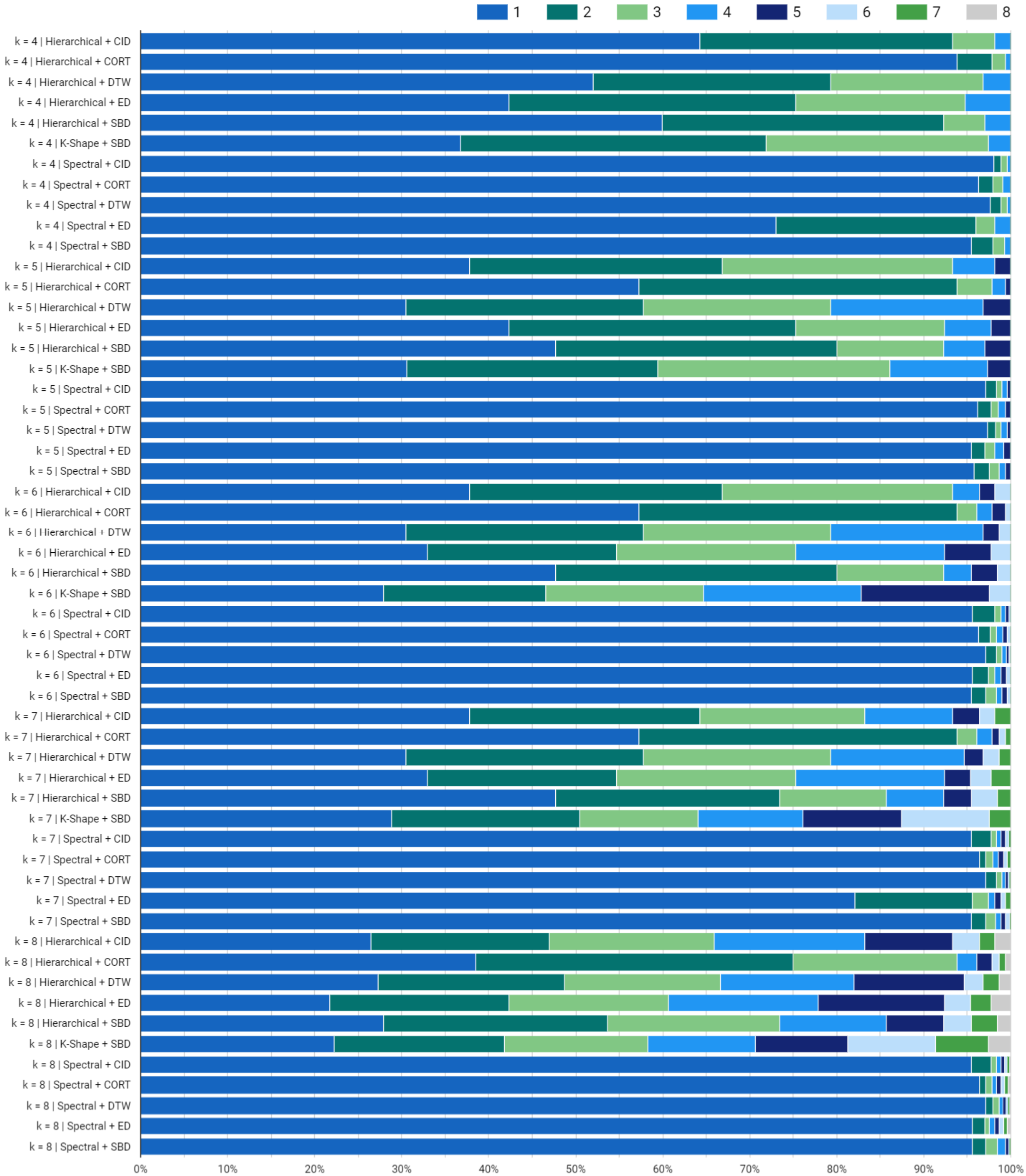


Table 32: Cluster partition based on the CLV feature for company 3

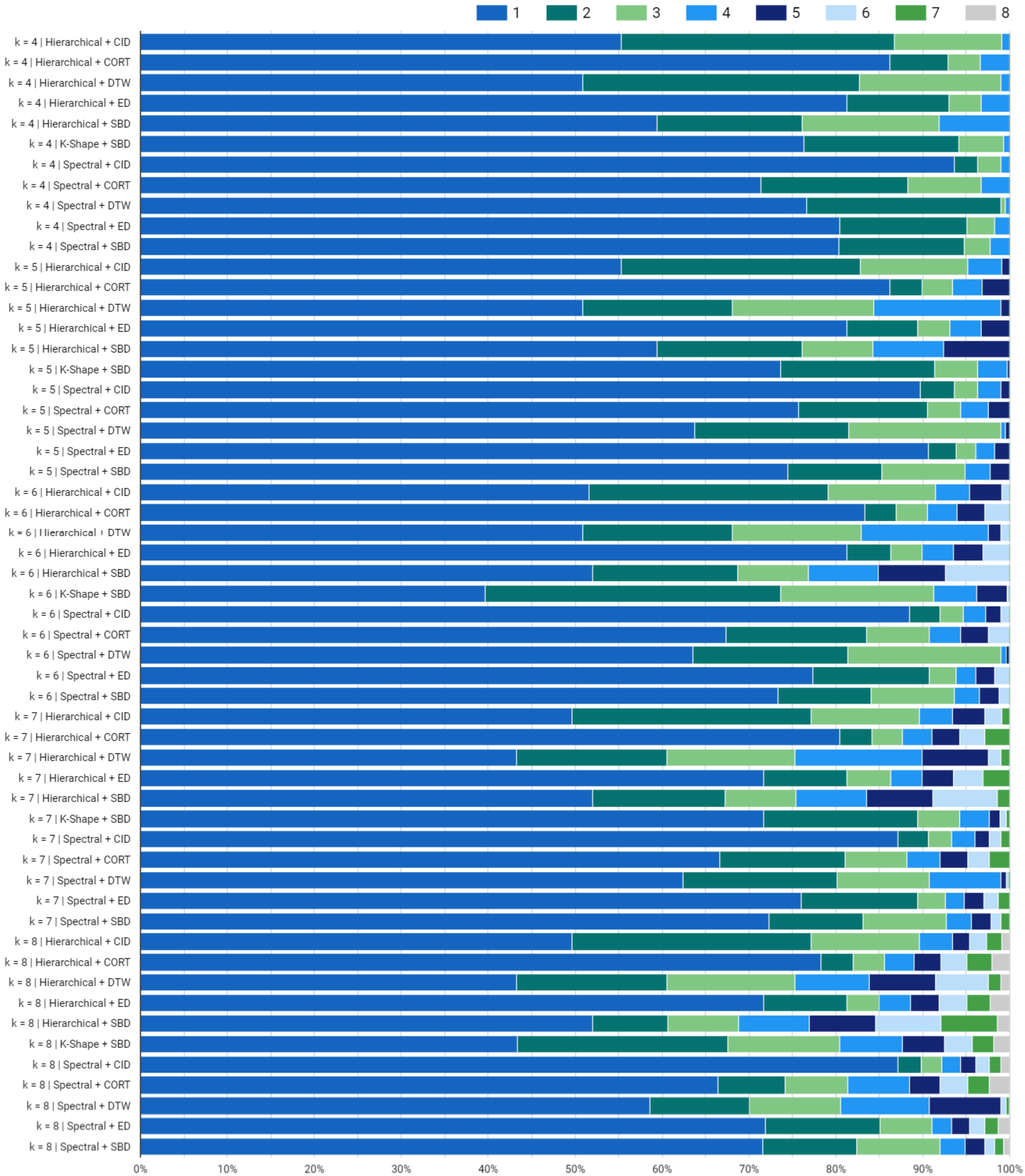


Table 33: Cluster partition based on the monetary value feature for company 3

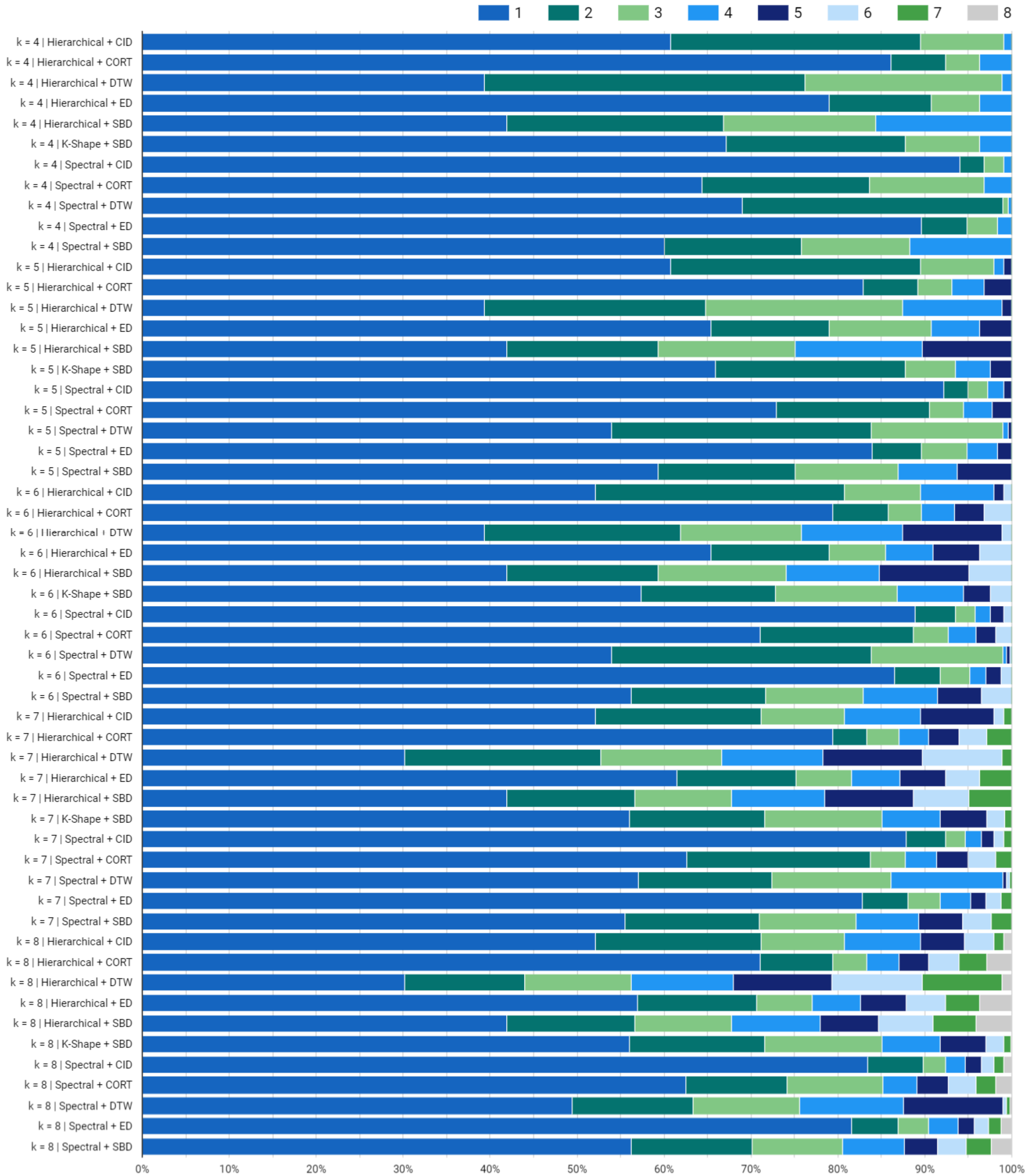


Table 34: Cluster partition based on the pageview feature for company 3

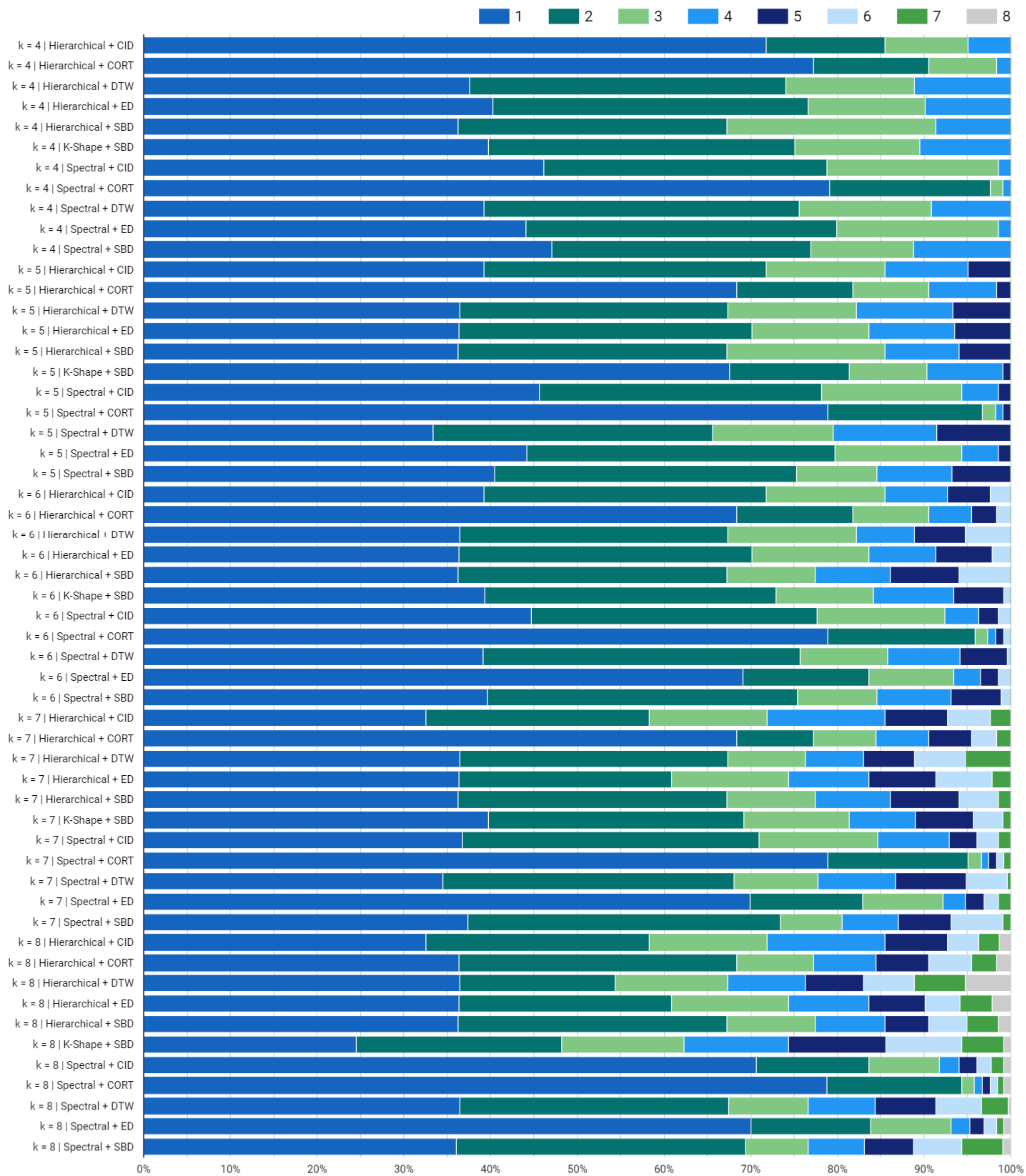


Table 35: Cluster partition based on the CLV feature for company 4

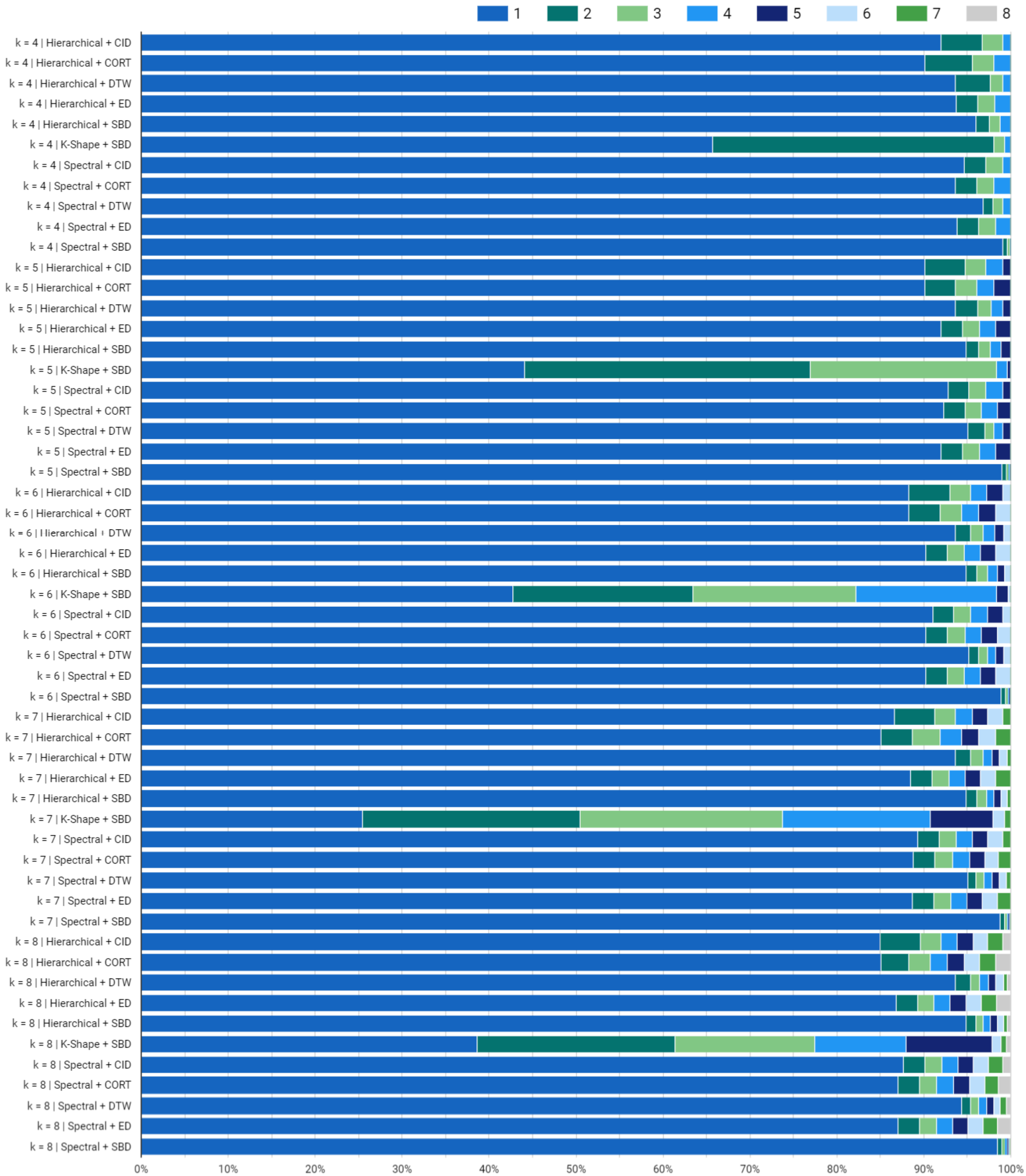


Table 36: Cluster partition based on the monetary value feature for company 4

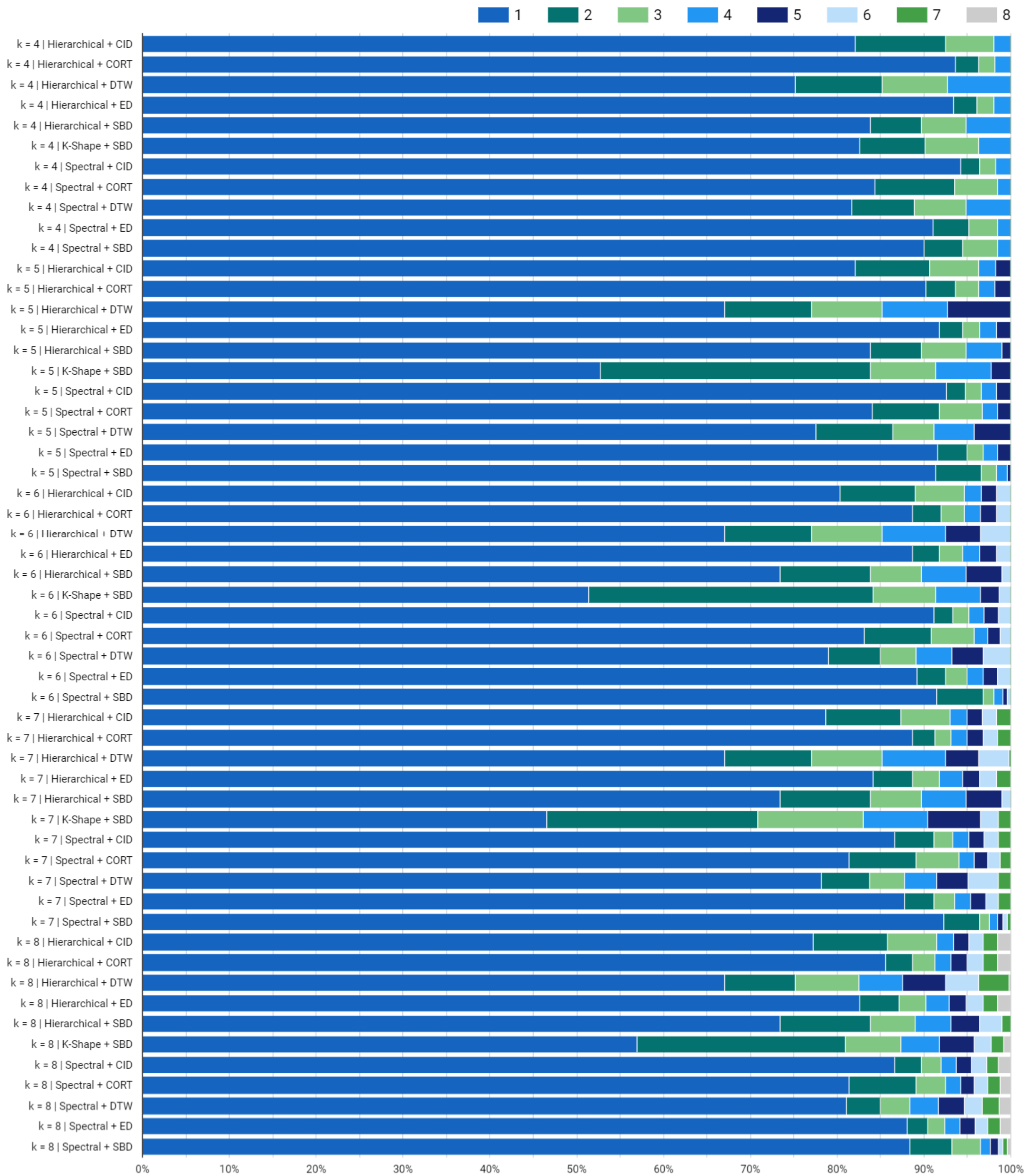


Table 37: Cluster partition based on the pageview feature for company 4