Erasmus University Rotterdam

Tracking Customer Segments in Alternative Finance using time-evolving Cluster Analysis

Master Thesis Business Analytics and Quantitative Marketing

Author: Lennert Aerts
Student ID number: 483376la

Supervisor: Dr. M. Zhelonkin
Second assessor: Prof. Dr. P.H.B.F. Franses

Abstract

In many applications such as in financial and e-commerce areas, it is valuable have a profound understanding of the type of customers one is dealing with. The focus of this work is on detecting customer segments in online peer-to-peer (P2P) lending data from Lending Club Club (2019) with cluster analysis and tracking these segments over time. We use the framework proposed by Sangam and Om (2018), where we cluster using K-means, K-prototypes, agglomerative clustering, DBSCAN and Gaussian Mixture Models (GMM) to identify customer segments and employ label strategies by extra trees classifiers and an Hybrid Data Labeling Algorithm (HDLA). In the application we conclude that by their incremental approach, K-means, K-prototypes and GMM allow for smooth cluster tracking while retaining the overall partition structure over time. DBSCAN is ineffective in detecting multiple segments in this setting whereas agglomerative clustering proves to be ineffective in tracking clusters by their significant shift in quarterly structure.

Keywords: cluster analysis; cluster tracing; customer segmentation; customer classification; alternative finance; peer-to-peer lending; K-means clustering; K-prototypes clustering; hierarchical clustering; DBSCAN; Gaussian Mixture Models

April 10, 2020

Contents

1 Introduction

In many applications such as in financial and e-commerce areas, it is valuable have a profound understanding of the type of customers one is dealing with. Companies are interested in understanding their customers and segment them into types which is helpful for business processes such as targeted marketing. The focus of this work is on detecting customer segments in online peer-to-peer (P2P) lending data from Lending Club Club (2019) using clustering techniques and tracking the discovered segments over time. We build on the framework proposed by Sangam and Om (2018), who identify a method of tracking clusters in time-evolving data streams which is independent of the clustering algorithm being used.

Within the financial industry, we observe significant growth of the consumer and business alternative finance market. Alternative finance refers to financial channels, processes, and instruments that have emerged outside of the traditional finance system such as regulated banks and capital markets (Cambridge Judge Business School, 2019). Consumers and businesses are searching for financing solutions outside of those provided by traditional banks and investors, rendering alternative finance less of an 'alternative'.

Market reports by Ziegler et al. (2019a,b) record an annual growth in online alternative finance of 88.5% and 79% for the USA and EU markets between 2013 and 2017, reaching a total market volume of US$42.8 billion and €10.44 billion. Two major models within this market are Balance Sheet Consumer Lending and P2P Consumer Lending which together accounted for 70% of the USA market (Ziegler et al., 2019a). P2P Consumer Lending allows individuals or institutional funders to provide a loan to a consumer.

Both traditional financial institutions and the online platforms providing alternative finance should be interested in gaining an understanding of the type of customers that use alternative financial products to identify why these customers prefer the alternatives over traditional methods. Furthermore, due to rapid growth of this market, it is also be interesting to capture market trends with respect to the type of customers. For example whether certain customers are most profitable at this moment in time but may shrink in size significantly in the future, making them less attractive to target. This knowledge allows interested parties to anticipate these market changes. By segmenting data from an alternative finance marketplace, where we focus on P2P loan data, we are able to distinguish customer segments, understand their lending behavior and track this behavior over time. The understanding of a changing market allows for better decision making and higher quality services and products tailored around the needs of the segments, further improving the customer service.

The proposed framework shown in Figure 1 contains two modules, one for clustering new observations and one that labels them. We define concept drift as the (dis)similarity between quarterly data, where we use the label module only when new observations are not significantly different from previously seen ones. These observations are then labeled to the most similar cluster from the previous timepoint. After identifying the segments and tracing them over time, the results are visualized by inspecting features that best characterize the segments. We test different clustering techniques as well as labeling strategies to identify different types of customers and investigate different methods to track segments.

The remainder of this thesis is organized as follows: Section 2 introduces the problem setting and provides an overview of the landscape of static and dynamic clustering while also mentioning related literature on tracing clusters over time. Afterwards, Section 3 elaborates

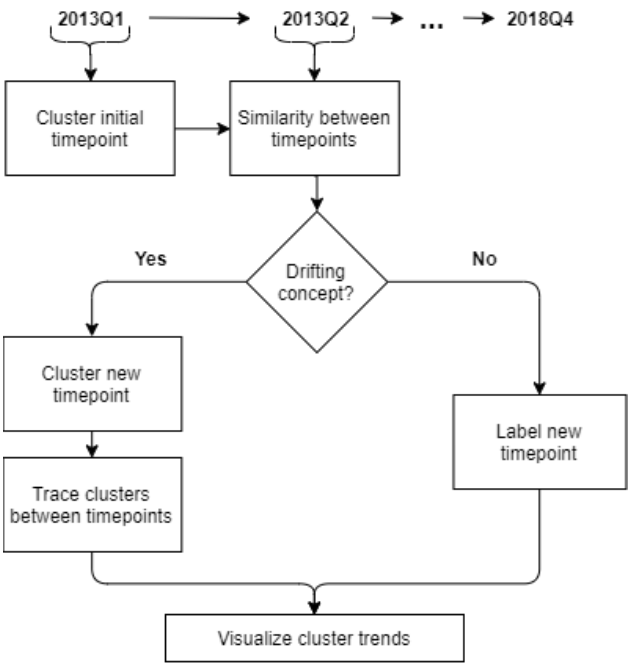

Figure 1: General outlook of the framework (Sangam and Om, 2018). Depending on the level on concept drift, observations are either clustered or labeled according to their similarity to clusters of the previous timepoint.

on the data set used to test the framework. It includes the preprocessing steps of the data and provides an exploratory data analysis. In Section 4 we explain our approach in detail and describe the set-up of the framework and methods used. Section 5 then presents the results obtained by applying our model to the aforementioned data. From the results, we draw our conclusions in Section 6.

2 Literature Review

Before reviewing literature related to cluster analysis and tracing clusters in time-dependent data, we give a brief description of the problem at hand and provide related works of this setting. Then in Sections 2.1 and 2.2, we describe clustering techniques applied to static and dynamic data. We refer to the clustering of static data as static clustering whereas dynamic clustering concerns the clustering of dynamic data. Finally, we mention related works on tracing clusters in time-dependent data in Section 2.3.

Our setting involves data from an online marketplace called Lending Club where P2P loans are provided between borrowers and investors. It contains quarterly data of P2P loans including background information on the borrowers. This type of data has been well studied as a classification problem, where researchers predict credit or default risk using advanced data mining techniques. Emekter et al. (2015) and Lin et al. (2017) study P2P data sets from online platforms with the purpose of detecting determinants of default through classification algorithms. Other aspects researched in P2P lending concern herding behavior (Herzenstein et al., 2011; Berkovich, 2011) and the influence of social networks on loan funding (Freedman and Jin, 2017; Herrero-Lopez, 2009).

However, research on the type of customers resorting to this type of finance has been left relatively untouched. Customer segmentation through cluster analysis improves the understanding of the customer types which is a core aspect of customer relationship management (Dibb, 1998). Therefore, customer segmentation is a powerful tool to apply on P2P lending for information retrieval. To the best of our knowledge, there is no literature capturing both the problem of customer segmentation and the tracing of discovered segments over time in one work.

2.1 Static Clustering

The process in which all observations are aggregated, separated from the dimension of time and then clustered, can also be referred to as static (data) clustering. We give a brief description of the general idea behind certain clustering types and provide their benefits and drawbacks. We also mention classic algorithms that belong to these types.

2.1.1 Partition-based clustering

Given a set of n observations, partitioning methods construct k partitions of the data, where each partition represents a cluster (Han et al., 2011). These clusters must contain at least one observation and each observation must belong to a exactly one group. However, this second requirement is allowed to be relaxed, for instance, in fuzzy clustering techniques where data points are allowed to belong to multiple clusters. Partitioning cluster algorithms aim to discover the groupings in the data by optimizing a specific objective function and iteratively improving the quality of the partitions. First the k cluster are initialized, creating an initial partitioning which is then improved by reallocating observations to other clusters based on the objective function.

Achieving a global optimal solution with partitioning techniques is often computationally exhaustive. Instead most methods employ greedy approaches like the K-means (MacQueen et al., 1967) and K-medoids (Kaufman and Rousseeuw, 1987) algorithms which approach local optima dependent on the position of the initial centroids (Han et al., 2011). Furthermore,

partitioning methods work well for spherical-shaped clusters but are usually not suitable for non-convex data. Other drawbacks include the sensitivity to outliers and the fact that the number of clusters need to be preset (Aggarwal and Reddy, 2014).

Next to K-means and K-medoids, other typical partition-based clustering algorithms include PAM (Kaufman and Rousseeuw, 1990), CLARA (Kaufman and Rousseeuw, 2008) and CLARANS (Ng and Han, 2002).

2.1.2 Hierarchy-based clustering

Hierarchical techniques create a hierarchical decomposition of the dataset which is either agglomerative or divisive (Johnson, 1967). Agglomerative refers to the bottom-up approach, which starts with each observation as an individual cluster and continuously merges the observations or clusters that are close to each other until one cluster is formed. The divisive approach works the other way around, starting with one cluster and then splitting it until each observation is seen as a separate cluster. Closeness of the observations can be measured by distance-based metrics with common strategies such as single and complete link (Aggarwal and Reddy, 2014). Once the hierarchical tree called a dendrogram is build, one is able to select any number clusters and deduct cluster membership from this tree. However, deciding the optimal number of clusters is not set by the algorithm and requires external input.

Drawbacks include the fact that after a split or merge operation is performed, it cannot be reversed and erroneous decisions can therefore not be corrected which may lead to local optima. However, it does reduce the computational cost as not all split or merge options need to be assessed. Another benefit of these techniques is that they are suitable for arbitrarily-shaped data and features of arbitrary type (Aggarwal and Reddy, 2014).

Typical algorithms that employ hierarchical clustering are BIRCH (Zhang et al., 1996), CURE (Guha et al., 1998), ROCK (Guha et al., 2000) and Chameleon (Karypis et al., 1999).

2.1.3 Density-based clustering

Many of the well-known clustering algorithms make the assumption that data are generated from a probability distribution. This is for example the case for EM (Expectation Maximization) clustering and K-means. Because of this assumption, algorithms produce spherical clusters and are not equipped to deal well with clusters of non-spherical shapes. The paradigm of density-based clustering has been proposed to address this challenge and the issues of dealing with noise and outliers (Aggarwal and Reddy, 2014). Observations which are in a region of high density are considered to belong to the same cluster which allows density-based methods to detect clusters of arbitrary shapes (Kriegel et al., 2011). The idea is for a cluster to keep growing as long as the density in its neighborhood exceeds a preset threshold. Besides clustering, density-based methods are able to filter out noise and outliers which occurs when points do not have a sufficiently dense neighborhood.

Drawbacks include the high sensitivity to input parameters and low quality results when dealing with clusters of varying density. Another major challenge is that density-based methods are defined on data in continuous space and cannot be implemented in discrete or non-Euclidean space without using an embedding approach (Aggarwal and Reddy, 2014).

Well-known density-based algorithms are DBSCAN (Ester et al., 1996), OPTICS (Ankerst et al., 1999), DBCLASD (Xu et al., 1998) and DENCLUE (Hinneburg et al., 1998).

2.1.4 Grid-based clustering

Grid-based clustering algorithms are closely linked to density-based ones. These algorithms partition the data space into a grid structure using a finite number of cells and then form clusters from these cells in the grid. Clusters correspond to regions of cells that contain a higher density of data points than their surrounding cells (Aggarwal and Reddy, 2014).

The great advantage of grid-based clustering is a significant reduction in time complexity, especially for very large data sets. Rather than clustering the data points directly, grid-based approaches cluster the neighborhood surrounding the data points represented by cells. In most applications since the number of cells is significantly smaller than the number of data points, the performance of grid-based approaches is significantly improved. Drawbacks are the sensitivity to granularity of the grid and reduced cluster quality in return for computational efficiency. Performance depends on the size of the grid structure and as this size increases significantly with more dimensions, grid-based clustering may not be scalable for very high-dimensional data (Aggarwal and Reddy, 2014)

STING (Wang et al., 1997), CLIQUE (Agrawal et al., 1998) and Wave-Cluster (Sheikholeslami et al., 1998) are classical algorithms of this type of clustering.

2.1.5 Model-based clustering

The idea behind model-based clustering algorithms is to optimize the fit between given data and a mathematical model. This is based on the assumption that the data is generated by a mixture of components each represented by a probability distribution (Aggarwal and Reddy, 2014). Each cluster can be represented using a parametric probability distribution which transforms the clustering problem into a parameter estimation problem. This way the data is modeled by a mixture of k component distributions.

Estimation of this type of models is often done through an iterative optimization algorithm called the Expectation-Maximization algorithm (EM) (McLachlan and Krishnan, 2007). Two limitations of this algorithm is that it converges to a local optimum and it is highly dependent the initial parameter guesses. Furthermore, selecting the number of components is an important issue. Selecting too many components may overfit the data, while too few may lack flexibility to approximate the true underlying model.

MCLUST is a well-known example of model-based clustering (Fraley and Raftery, 1999). It implements Gaussian hierarchical clustering and the EM algorithm for (Gaussian) mixture models. Other typical algorithms are COBWEB (Fisher, 1987) and ART (Carpenter and Grossberg, 1988).

2.1.6 Overlapping (or nonexclusive) techniques

Overlapping techniques produce partitions that can be soft or fuzzy. In soft partitions, observations can belong to one or more clusters whereas in fuzzy partitions each observation belongs to one or more clusters to different degrees (Aggarwal and Reddy, 2014). Methods that are soft or fuzzy are oftentimes an adopted version of an algorithm belonging to one of the former classes. For example Fuzzy-CMeans (FCM) by Bezdek et al. (1984), which is very similar to K-means clustering, can be classified as a partition-based method. Mixture models can also be seen as fuzzy techniques as they output probability estimates of cluster membership.

Advantages of overlapping methods are the relatively high accuracy of clustering as it is

more realistic to give a probability of belonging (to a cluster) instead of a hard clustering. One drawback is that such methods often end up in local optima.

2.1.7 Clustering Ensembles

Discovering multiple distinct partitions and combining them into a single consolidated partition, is also referred to as a clustering ensemble (Strehl and Ghosh, 2002). Typically, an effective clustering ensemble should be able to provide an improved performance at characteristics such as robustness, accuracy and stability (Ghaemi et al., 2009; Alqurashi and Wang, 2019). The main problem of clustering ensembles is finding a suitable consensus function to produce the final partition. The main reason for this is the fact that clustering is an unsupervised task with unlabeled patterns. Therefore, there is no direct link between the labels produced by different clusterings. Moreover, the shape and number of clusters is often not known a priori and individual solutions may differ based on the method used (Ghaemi et al., 2009).

Besides the described classes, many other classes for static clustering are mentioned in literature, for example subspace clustering methods or graph/network-based methods which are upcoming and are receiving more and more attention. More information regarding other classes can be found in Aggarwal and Reddy (2014) and Han et al. (2011). Furthermore, it should be noted that while clustering techniques can be split in the aforementioned classes, algorithms often include characteristics of multiple classes and fall not necessarily within one specific class. These methods combine characteristics in an attempt to achieve greater performance.

2.2 Dynamic Clustering

Considerable amount of research on clustering static data has been carried out. However in recent years, clustering dynamic data such as data streams has received more and more attention due the advances in hardware technology, allowing us automatically record information at a rapid pace (Aggarwal and Reddy, 2014). Examples of real-time applications include the exploration of the general trend of evolution in social networks (Kumar et al., 2006; Leskovec et al., 2008; Tantipathananandh et al., 2007), detection of communities in complex changing networks (Nan et al., 2018; He et al., 2019; Ma et al., 2019) or discovering customer interests in an e-commerce setting using click-stream data (Su and Chen, 2015).

Over the past two decades, literature describes the cluster analysis of dynamic data by data stream clustering (Aggarwal and Reddy, 2014), incremental clustering (Ackerman and Dasgupta, 2014) or evolutionary clustering (Chakrabarti et al., 2006) each focusing on a particular aspect of the challenges involved with dynamic data. One of the challenges is the massive volume that typically characterizes data streams. Therefore, it is usually not possible to store all data locally and requires the data processing in a single pass. Furthermore, the time required to process the data must be small as otherwise the model cannot keep up with the high pace of the stream. However, in our setting we process data in an offline environment where it still possible to store data locally, allowing continuous access. As we do not require online clustering components, we omit clustering algorithms aimed at solving online stream clustering scenarios.

2.2.1 Incremental Clustering

One set of techniques that attempt to tackle the problem of clustering dynamic data is called incremental clustering. It tackles the problem of sequentially clustering objects that are each observed only once, when storing all objects for static clustering is impractical (Xu et al., 2014). Two examples of the application of such techniques are trajectory clustering of moving objects (Li et al., 2010) and the detection of evolving (network) communities (Aggarwal and Yu, 2005; Ning et al., 2007).

Incremental clustering is attractive when observations are acquired over time and one intends to start analysing clusters before all data is observed. This is the case in online streams environments where new data is observed continuously and it is impractical to constantly consider all data for clustering once new data is seen (Ackerman and Dasgupta, 2014). However as previously mentioned, in our setting observations are not measured one-by-one but come in large groups at fixed points in time. In this situation, evolutionary clustering techniques are more suitable.

2.2.2 Evolutionary Clustering

The problem of processing time-series data in order to create a sequence of clusterings (at each timestep) is called evolutionary clustering. Evolutionary clustering has been first addressed by Chakrabarti et al. (2006) who propose a generic framework that includes temporal smoothness. This framework assumes that clustering at any timepoint should remain faithful to newly seen data but should also not shift too far from the previous timepoint. Hence it tries to smoothen cluster change over time whilst still allowing temporal drift. This change is measured through two qualities; the snapshot and history quality. The snapshot quality measures how well the clustering represents the data at the current timestep whereas the history quality represents the similarity of the current clustering with the clustering used during the previous timestep. The framework has been applied to a large evolving dataset of user-tags placed on images from flickr.com with two widely-used clustering algorithms, K-means and agglomerative hierarchical clustering, and results show high accuracy in capturing current clustering and high fidelity in retaining historical trends when applied.

Extending the evolutionary clustering framework by Chakrabarti et al. (2006), Chi et al. (2007) propose an evolutionary spectral clustering framework which also incorporates temporal smoothness. Their main extensions are the application of spectral clustering algorithms to evolutionary clustering and the flexibility of their framework to allow the number of clusters to change with time when new data is seen and old data may be removed. Both frameworks attempt to optimize a cost function that trades of the snapshot cost or quality and historic (or temporal) cost

$$Cost = \alpha \cdot CS + (1 - \alpha) \cdot CT,$$

where CS and CT describe the snapshot and temporal costs, respectively. α is a parameter that reflects the emphasis on historic data. As α increases, more and more weight is placed on historical partitions.

Both Chakrabarti et al. (2006) and Chi et al. (2007) use a fixed smoothing parameter α to control the weight between historic data and temporal drift. However, a shortcoming in these works is that they do not address how to choose this parameter. Xu et al. (2010) provide a

method to estimate the optimal smoothing parameter in what they call the forgetting factor. Their method uses a shrinkage approach which is inspired by the Ledoit-Wolf shrinkage estimator for covariance matrices (Ledoit and Wolf, 2003).

In Zhang et al. (2013), the authors extend the evolutionary framework with the DBSCAN (Ester et al., 1996) clustering algorithm. This allows them to deal with outliers distinguish clusters of arbitrary shapes during the evolution process.

2.3 Tracing Clusters in time-evolving Data

Cluster tracing or transition algorithms have been developed to map clusters at consecutive timepoints such that we can monitor cluster change over time. These algorithms often match clusters based on similarity in object sets, i.e. the number of shared objects between clusters.

Kalnis et al. (2005) propose algorithms to automatically discover moving clusters in spatio-temporal data. The contents of these moving clusters may change while its density is assumed to remain constant over time. They find moving clusters by tracing common observations between clusters of consecutive timepoints.

In Spiliopoulou et al. (2006), the authors propose MONIC, a framework for modeling and monitoring cluster transitions. MONIC allows for the tracking and detection of internal and external cluster transitions by capturing transition events, regardless of the type of clustering algorithm used. It assesses the overlap of clusters to capture whether a cluster at one timepoint survived, disappeared, merged or split at the next timepoint.

FINGERPRINT (Ntoutsi et al., 2012) proposes a graph representation of cluster changes and provide two algorithms to summarize the graph into a fingerprint. They use MONIC's approach to determine cluster succession and also adopt their transition definitions.

Another algorithm for tracking clusters is MEC (Oliveira and Gama, 2012), which traces the evolution of clusters over time by identifying temporal relationships between them. A bi-partite graph structure is used to visualize cluster evolution. MEC observes this evolution graph as a state-transition graph and applies Hidden Markov Models to explain the evolution. They use condition probabilities to represent the weights of each edge to evaluate the similarity of a cluster pair between consecutive timepoints. The use of conditional probabilities requires structurally identical datasets, which means the same objects need to be present at each timepoint.

These tracing methods map clusters if the corresponding object sets are similar. In contrast, our setting requires the mapping of clusters based on the similarity of their corresponding object values, independently of shared objects. Günnemann et al. (2011) tackle this limitation by introducing a novel approach to trace behavior types in temporal data using subspace clusterings. They map clusters at consecutive timepoints by combining object value similarity, through the Kullback-Leibler divergence measure, and subspace similarity, by measuring the feature space overlap. Like other transitions algorithms, they are able to detect emerging and disappearing behaviour and split and merge operations. Furthermore, the authors introduce a strategy using subspaces allowing them to capture other evolutions as well; segments can gain or lose relevant dimensions over time. The limitation of this tracing strategy is that it only works for metric data.

This issue has been tackled by Sangam and Om (2018), who propose Equi-Clustream, a framework that clusters time-evolving mixed data and also visualizes concept drift detected

during the cluster analysis. Based on the characteristics of features, they measure the dissimilarity between concepts of two sliding windows and set up a Hybrid Drifting Concept Detection Algorithm to detect concept drift. If the dissimilarity is greater then a predefined threshold, concept drift is observed and the current window is clustered. In contrast, if the detected concept is steady, they propose a Hybrid Data Labeling Algorithm to assign cluster labels to observations in the current window. This way they prevent the need of clustering large data sets at consecutive time-steps in case the overall population does not show significant change.

3 Data Preprocessing and Exploratory Data Analysis

The data contains complete P2P loan data for all loans issued from 2007Q1 to 2018Q4. It includes features describing the borrower's background such as credit scores, number of finance inquiries, zip codes and income statements. Zip codes, as well as, the identities of both the lender and borrower are (partially) censured such that their privacy is retained. Some features have been added or removed over the years as the platform developed and grew in size. A brief description of the features in the data set is given in Appendix A.1.

We provide a description of the data preparation process in Section 3.1, elaborating on feature engineering, missing values and outliers in the dataset. Then in Section 3.2, an exploratory data analysis is given, which provides initial insights of the data and serves as a starting point for clustering.

3.1 Data Preparation Process

Initially, the Lending Club data set contains over 2.2 million loans capturing 151 variables per loan. The data set contains a significant number of missing values and outliers. Furthermore, some features need to be adjusted first for them to become useful in the clustering task. First, we describe the adjustments made to a number of features, followed by the methods to deal with missing and outlying values.

3.1.1 Feature Engineering

With over 150 features, this dataset naturally contains redundant features and those that are not that informative. Features also contain values that are not sensible, such as negative debt-to-income ratios, or ones that are that uncommon, that they need to be changed or removed. The following adjustments to features have been made:

- *emp_title*, which describes the borrower's occupation, contains over half a million unique job titles. These titles are structured by dividing them among job categories retrieved from Recruiter (2019). The categories describe 12 different groups of careers such as Health Science, Information Technology or Marketing, Sales and Service.
- *emp_length*, describing the length of employment, contains values between 0 and 10 years of experience. We transform these values to three categories, where 0-3 years describe a 'Junior' employee, 4-8 years as 'Experienced' and 9-10 years as 'Senior'.
- $fico\_score$ is created by taking the mean of $fico\_range\_low$ and $fico\_range\_high$. It describes the borrower's FICO score and is often used to describe a customer's credit-worthiness.
- *homeownership* initially contains values 'Other', 'None' and 'Any' which are not straightforward to interpret and are very uncommon. Therefore, these are observations are removed.

3.1.2 Dealing with Missing values

Regarding missing values, 150 out of the 151 features contain at least one missing value, where 58 of them have more than 15% missingness. Methods that deal with missingness such as mean or multiple imputation are often not able to solve missingness at such a degree. Furthermore, by inspecting the distribution of these features and checking the information

they give on the customers, we decided that none exhibit great importance for customer segmentation. Hence it is decided to remove these features from the data set.

This leaves us with 92 features of which 87 include less than 4% missingness. These missing values often occur on the same observations and imputing all these features separately is a computationally intensive task. Therefore, we decide to remove the observations that contain missing values for the features with less than 4% missingness. This operation reduces the number of variables with missingness to 5. However, in our cluster analysis we do not use *mths_since_recent_inq*, *num_tl_120dpd_2m* and *mo_sin_old_il_acct* and leave their missingness as it is.

Table 1: Remaining features with missing values, including a feature description.

Feature name	**Pct.**	**Description**
mths_since_recent_inq	11.1%	Months since most recent inquiry
emp_title	7.2%	Job title supplied by the borrower
emp_length	6.4%	Employment length in years
num_tl_120dpd_2m	3.8%	Number of accounts currently 120 days past due
mo_sin_old_il_acct	3.1%	Months since oldest bank installment account opened

The type of missingness of these features is assumed to be missing completely at random (MCAR). Such type of missingness means that there is no relationship between the missingness of the data, observed or missing. The probability that a value is missing is independent of the feature itself and other features in the data set (García-Laencina et al., 2010). In order to cope with MCAR type of missingness, we decide to impute missing values of *emp_title* and *emp_length* using the **IterativeImputer** function from the scikit-learn python package (Pedregosa et al., 2011).

IterativeImputer

The **IterativeImputer** function is a multivariate imputer that estimates each feature from the others in the given data in a round-robin fashion. At each step, one feature is designated as output y and the remaining ones are treated as input X. Then a regression of y on X is fit which is used to predict the missing values of y. This step is repeated until the maximum number of iterations is reached. The estimator used in this function for metric and categorical features are the KNeighborsRegressor and KNeighborsClassifier (Pedregosa et al., 2011), respectively. The KNeighborsRegressor predict y by a local interpolation of the k nearest neighbors whereas KNeighborsClassifier takes the k nearest neighbors and predicts the category of y using a majority vote.

Instead of using the complete attribute set for imputation, only the most correlated features with the target feature are selected. This speeds up the process and imputation by regression is well suited in case the target feature is correlated with the available data (García-Laencina et al., 2010).

3.1.3 Outlier detection

Outliers are extreme values that deviate from other observations in the dataset. Usually they indicate a variability in a measurement, experimental errors or some novelty in the data. In

other words, an outlier is an observation that diverges from the overall pattern in the data. In the Lending Club data set outliers are seen as loans with extremely rare characteristics, loans that will not be straightforward to put in any customer segment and seem to deviate from the usual behavior. Furthermore, clustering algorithms such as K-means can be heavily affected by outlying observations. In this work we deal with univariate outliers by using the modified Z-score (Iglewicz and Hoaglin, 1993, Chapter 3) and multivariate outliers with the Local Outlier Factor (LOF) (Breunig et al., 2000).

Modified Z-score

The modified Z-score adepts the standard Z-score by using resistant estimators. The basic idea of using Z-scores is quite strong, but it is not satisfactory for labeling outliers as the sample mean and standard deviation are not resistant estimators. They are not resistant in the sense that they are already affected by only a small number of outlying observations. Instead, the modified Z-score method uses the sample median and mean absolute deviation (MAD) in determining the test score M_i. The mathematical expression for this method is given by:

$$M_i = \frac{0.6745\,(x_i - \tilde{x})}{MAD},$$

where $\tilde{x}$ denotes the median of x and MAD describes the median of the absolute deviations about the median, defined as follows:

$$MAD = median_i\{|x_i - \tilde{x}|\}.$$

An observation is labeled as an outlier when $|M_i| > D$, where Iglewicz and Hoaglin (1993) recommend using $D = 3.5$. Here, we follow this recommendation and remove the observations that obtain such a test score.

In order to use this outlier test, the data needs to approximately follow a normal distribution. In case this assumption is violated, labeling observations as outliers may be due the non-normality rather than the actual presence of outliers. The normality of the numerical features is assessed by generating normal probability plots. These plots visualize the similarity between the sample distribution and a standard normal distribution and provide an indication of normality of the sample. The resulting plots are shown in Appendix A.2.1.

The upper and lower bounds resulting from the modified Z-score method are visualized against the feature distribution in Appendix A.2.2. We compare this method to the outlier bounds computed by the standard Z-score and adjusted boxplot (Hubert and Vandervieren, 2008) methods. We observe that the modified Z-score method is more conservative compared to the other two, as it is less inclined to label observations as outliers. Therefore, this method is less likely to incorrectly label observations as outliers but may in return neglect true outliers. In order to reduce the probability missing true outliers, we extend the outlier detection process by investigating multivariate outliers using the Local Outlier Factor.

Local Outlier Factor

Regarding multivariate outliers, we employ the LOF detection method proposed by Breunig et al. (2000). This method computes an anomaly score (LOF score) for each observation in the data set by measuring the local deviation in density with respect to k neighbors. It

measures the degree of isolation using local density, where more isolated observations are more likely to be outliers. This local density is estimated using the distance to the k-nearest neighbors. The local density of a sample is compared to the densities of its neighbors to evaluate substantial differences in density. Observations with a significant lower density than its neighbours are considered outliers.

We perform the LOF method using a neighborhood size k of 100, where the distance between samples is measured using Euclidean distance. A resulting scatterplot of the numerical features denoting inliers and outliers in 2013Q1 is shown in Appendix A.2.2.

3.2 Exploratory Data Analysis

By performing an exploratory data analysis, we are able to gain an understanding of the features and possible relationships between them. These initial insights will provide a basis for the selection of features used in the cluster analysis. We want to focus on features that are straightforward to interpret and seem useful to include into the clustering algorithms. Here, we summarize the findings of the univariate and multivariate analyses on the core features in Table 2. The complete process can be found in Appendix A.3.

Table 2: Core set of metric (top) and categorical (bottom) features

Core Feature	**Description**
Numerical	
annual_inc	Annual income provided by the borrower during registration.
bc_util	Ratio of current credit balance to credit limit for all bankcards.
dti	Debt-to-income ratio.
fico_score	Borrower's FICO score at loan application.
int_rate	Interest rate on the loan.
loan_amnt	Listed amount of the loan applied for by the borrower.
revol_util	Revolving line utilization rate.
tot_cur_bal	Total current balance of all accounts.
Categorical	
emp_length	Employment length in years, possible values between 0 and 10.
emp_title	Job title supplied by the borrower.
grade	Lending Club assigned loan grade.
home_ownership	Home ownership status provided by the borrower.
purpose	Reason for applying for a loan.
term	Number of payments on the loan, either 3 or 5 years.

With the features *annual_inc*, *loan_amnt* and *tot_cur_bal*, we observe significant positive correlation. Borrowers with a higher income tend to have a higher savings and apply for larger loans. *bc_util* and *revol_util* are highly correlated, together describing the credit usage behavior of borrowers. Over time, we observe a gradual decrease in average utilization rate, possibly indicating that borrowers are becoming less dependent on their credit or due to stricter requirements for lending. This decrease over time has also a positive effect on the interest rate, as customers who use less credit get more favorable interest rates on their loan.

Furthermore, these two features have the largest negative influence on $fico\_score$ which is reasonable as it an indicator of financial stability. Surprisingly, there is no correlation between $annual\_inc$ and $fico\_score$, where it is expected that a higher income would lead to a higher FICO score.

The Lending Club provided grade seems to be moderately influenced by $fico\_score$, higher FICO scores result in a higher grade. In addition, $grade$ almost perfectly (negatively) correlates with int_rate. Borrowers with a higher grade get a better loan offer in terms of interest rate. Besides $grade$, $fico\_score$ also negatively influences the interest rate as higher grades and scores are the result of a better financial background. Regarding $term$, we observe a positive correlation with $loan_amnt$. Larger loans are generally paid off in 5 instead of 3 years. These larger loans will also often receive a worse interest rate, portrayed by the positive correlation between $term$ and int_rate. Larger loans paid over a longer period involves more risk.

Regarding emp_title and $purpose$ we are not able to detect interesting insights, as these categorical features are dominated by a single category and their distribution does not change over time. Therefore, we remove these features from the core set.

4 Methodology

In order to create a chain of partitions over time, we first need to define the cluster algorithms. These algorithms explore different type of clustering techniques in order to maximize the information gain on customer segments present the dataset. Afterwards, we define the process of tracking clusters. This process concerns matching the most similar clusters between consecutive timepoints. The main challenge in this process is the fact that each timepoint contains a distinct set of customers. Therefore, we are not able to apply strategies that make use of overlap criteria which measure the shared number of observations between clusters and match those with the highest overlap. Instead we resort to techniques that match clusters based on similarity in characteristics. This allows us to track segments over time and identify different types of behavior. Besides strategies to match clusters, we present methods which label new observations to the most similar segment of the previous timepoint. Such methods prevent the need for clustering repeatedly and are deemed less time-consuming at the cost of cluster accuracy.

Regarding the static clustering techniques, we opt for K-means and K-prototypes as partition-based algorithms. We implement hierarchy-based clustering through an agglomerative approach with Ward-linkage. DBSCAN is used as a density-based approach, searching for regions of high density that are split by those of low density. Finally, we apply Gaussian mixture models to model the data as a mixture of k component distributions which are solved using the EM algorithm. With respect to tracing clusters, we distinguish two methods; the KL-divergence measure for metric data as proposed by Günnemann et al. (2011) and the tracing strategy used in Equi-Clustream (Sangam and Om, 2018). Computing the similarity between the complete dataset of two timepoints is a similar problem as computing the similarity between clusters. Therefore, we opt again for the KL-divergence measure and test the Hybrid Drifting Concept Detection Algorithm, which is part of Equi-Clustream. In case the similarity between two timepoints is sufficiently high, we apply the labeling strategy HDLA from Equi-Clustream or implement an Extra-Trees classifier. We visualize the framework in Figure 2 and mention where each method is used.

First, we introduce the clustering techniques and measures for selecting the number of segments in Section 4.1. Then in Section 4.2, we present the strategies for tracing clusters over time. Finally in Section 4.3, we describe the classification methods used to assign cluster labels based on segments detected at previous timepoint.

4.1 Static Clustering Techniques

In this work, we apply several static clustering techniques that each take a different type of approach. This allows us to discover insights on customer segments from different perspectives and it illustrates which techniques are suitable in our setting. Here, we cover how the techniques function, what inputs are required and which measures we use to select the number of clusters. An outline of the algorithms is given in Appendix B.1.

Notation For each timepoint $t \in \{1, ..., T\}$, we have a set observations denoted by $S_t = \{x_1, ..., x_{n_t}\}$ with size n_t. A partition with k clusters made on the data at timepoint t is defined by $C_t = \{C_{t,1}, ..., C_{t,k}\}$, representing the individual clusters. $c_{t,j}$ defines the cluster label of data point j from the set S_t and μ_k defines the vectorized centroid of cluster k.

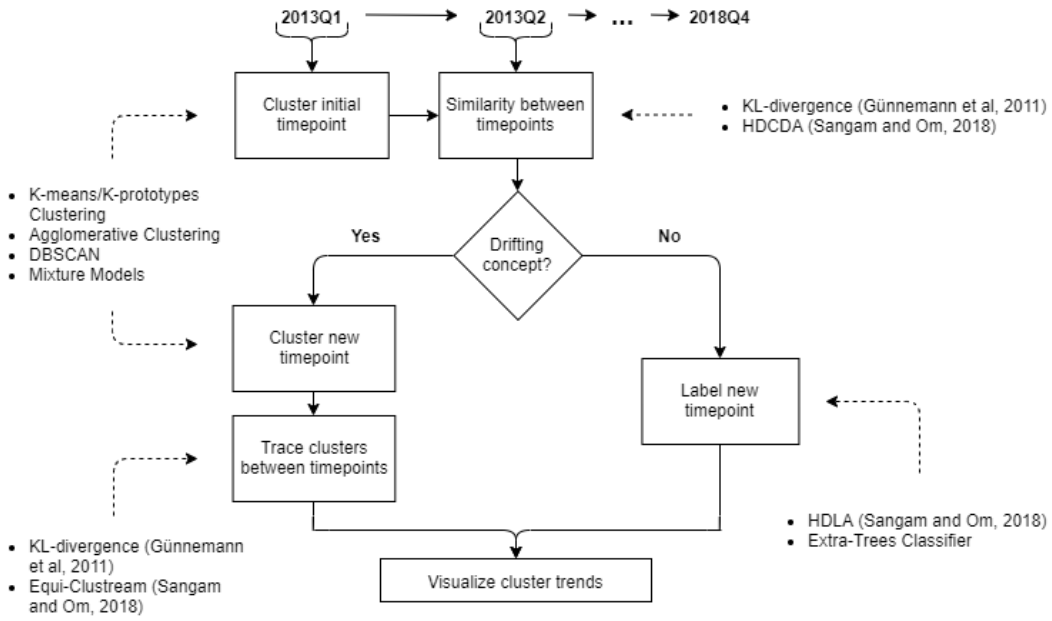

Figure 2: General outlook of the framework (Sangam and Om, 2018), including different strategies (dotted arrows) implemented in this work.

4.1.1 K-Means Clustering

The K-means algorithm (MacQueen et al., 1967) is a technique that segments data into k groups of equal variance. It initiates by selecting k points as initial cluster centers. Each point in the dataset is then appointed the cluster with the closest center based on the distance measure chosen. Once all data points are assigned, the cluster centers are updated by taking the centroid of the points within a cluster. These last two steps are iteratively repeated until the centers do not change or any other convergence criterion is met.

Two major factors that affect the performance of K-means are the choice of initial cluster centers and the estimation of the number of clusters. The method by MacQueen et al. (1967) selected k initial centriods at random. This is the most simple initialization and is widely used in literature. However, other popular initialization methods have been successfully introduced to improve the performance of K-means. We opt for the K-means++ initialization algorithm (Arthur and Vassilvitskii, 2007) for the first observation, 2013Q1. The remaining observations are initialized using the cluster centriods of the previous timepoint. It speeds up the clustering algorithm and aids in matching clusters as we do not expect significant change between quarters. Therefore, the partition of the new timepoint will not deviate too much from those of the previous timepoint and cluster matches become straightforward.

Estimating the correct number of clusters is a major challenges for K-means clustering. The K-means algorithm takes the number of clusters as an input and selecting an optimal number of clusters is seen as tedious job (Aggarwal and Reddy, 2014). The methods used for estimating the number of clusters are described in Section 4.1.6.

4.1.2 K-Prototypes Clustering

K-means is not equipped to deal with mixed data of both numerical and categorical features, as the Euclidean distance is not defined for such data. Furthermore, updating the cluster centers by averaging over categorical features is not practical and instead the mode should be used. The K-prototypes algorithm is proposed to solve this issue and combines both numerical and categorical features in a clustering algorithm (Huang, 1997a). This algorithm is practically more useful as real world data often includes both numerical and categorical features. K-prototypes combines the cost function of K-means and K-modes (Huang, 1997b) into a single function. The mathematical expression for this function is as follows:

$$\text{minimize} \quad \sum_{j=1}^{k} \sum_{x_i \in C_j} \left(\sum_{l=1}^{p} (x_{i,l} - \mu_{j,l})^2 + \gamma \sum_{l=p+1}^{m} \delta\left(x_{i,l}, \mu_{j,l}\right) \right), \tag{1}$$

where the first term describes the squared Euclidean distance of the numerical features with respect to the mean, which is the common distance measure used in K-means clustering. The second term defines the simple matching dissimilarity measure equals $\delta(p, q) = 0$ if $p = q$ and $\delta(p, q) = 1$ when $p \neq q$. Here, $\mu_{j,l}$ describes the mode of categorical feature l. The importance of categorical features with respect to numerical ones is set by the parameter γ.

We employ K-prototypes in quarter 2013Q1 with the initialization strategy proposed by Cao et al. (2009). The authors introduce a novel initialization method for categorical data, where both the distance between objects and the density of the object is considered. After 2013Q1 we initialize the algorithm similarly to K-means, using cluster centroids of the previous timepoint. The weighting parameter γ is estimated using the average standard deviation σ of numerical features (Huang, 1997a).

4.1.3 Agglomerative Clustering

Agglomerative methods build a bottom-up hierarchy of clusters, which starts by taking each data object as a unique cluster and continues merging two clusters until one cluster is formed at the top. The algorithm is initialized by constructing a dissimilarity matrix based on a proximity measure. All observations are represented at the root of the dendrogram, each as a unique cluster. Then closest set of clusters is merged at each level and the dissimilarity matrix is updated accordingly. This merging process carries on until one single cluster is formed.

The proximity measure we use to assess cluster similarity is the Ward's criterion with Euclidean distance which we compare with Complete linkage with Gower distance. Ward's criterion is not naturally applicable for categorical data whereas Gower distance combines both metric and categorical features. Ward's criterion uses the total within-cluster variance to determine cluster distance. At each step, the algorithm merges the pair of clusters that lead to a minimum increase of the within-cluster variance. For any two clusters C_x and C_y, this increase is measured by the squared distance between the cluster centroids, weighted by a factor proportional to the product of sizes of the respective clusters. Ward's criterion is given by:

$$W(C_x, C_y) = \frac{N_x N_y}{N_x + N_y} ||\mu_x - \mu_y||^2. \tag{2}$$

Gower distance measures the distance between two observations by combining metric and categorical data (Gower, 1971). It is calculated as follows:

$$S_{i,j} = \frac{1}{m}\left(\sum_{s=1}^{p}\left(1 - \frac{|x_{i,s} - x_{j,s}|}{R_s}\right) + \sum_{s=p+1}^{m}\delta_{i,j,s}\right), \tag{3}$$

where the first term describes the distance for quantitative features by taking the absolute distance, scaled by the feature range R_s. For qualitative features, the second term equals $\delta_{i,j,s} = 1$ if $x_{i,s} = x_{j,s}$ and 0 else.

4.1.4 Density-based spatial clustering of applications with noise (DBSCAN)

The DBSCAN algorithm (Ester et al., 1996) describes clusters as density regions, where clusters are regions of high density separated by sparser regions. Central to the DBSCAN algorithm, is the description of core points. An observation is called a core point if it contains at least $MinPts$ observations within a radius Eps. $minPts$ and eps are the two main input parameters for the algorithm which define the user's view on density in the data. A higher value of $minPts$ or lower eps sets a higher required density to form a cluster. A cluster is defined by a set of core points, which define the core of a cluster, and a set of border points which are close to a core point and define the edge of a cluster. Border points are points that belong to a cluster but whose neighborhood is not dense enough.

DBSCAN starts with by randomly selecting a data point p and retrieves its neighborhood with respect to Eps and $MinPts$. If p is considered as a core point, it is considered as a cluster and its surrounding points are evaluated to see whether they also belong to this cluster. If p is not a core point, it is classified as noise and the next data point is evaluated. If p is actually a border point, it is later reached when collecting all the points from a core point and is then assigned to the cluster of that core point. The algorithm terminates when all observations are assigned to either a cluster or the noise set.

As a rule of thumb, the input for $MinPts$ is set to the number of dimensions $D + 1$. For very large datasets that contain a lot of duplicates, it may be necessary to set higher values for $MinPts$. We use the rule of thumb as a starting point for $MinPts$ and assess higher values depending on the results. The value for Eps is estimated using a k-distance graph, which plots the distance to $k = MinPts - 1$ nearest neighbors ordered by size. We select Eps this plot shows an elbow-like shape.

4.1.5 Gaussian Mixture Model

The most well-known mixture model is the Gaussian mixture model (GMM), where each component or customer segment is represented by the parameters of a multivariate Gaussian distribution $p(x_k|\theta_k) = N(x_n|\mu_k, \Sigma_k)$. Formally, the mixture distribution of a Gaussian distribution is given by

$$p(x_n) = \sum_{k=1}^{K} \pi_k p(x_n|\mu_k, \Sigma_k), \tag{4}$$

where we need to estimate the set of parameters from the observations. This includes the mixing probabilities π_k and the parameters that describe the component distributions, μ_k and Σ_k. The number of components k is assumed to be fixed, but oftentimes this value is

unknown and has to be inferred from the available data. Overall, the set parameters in GMM is given by $\theta = \{\pi_1, ..., \pi_k, \mu_1, ..., \mu_k, \Sigma_1, ..., \Sigma_k\}$. Maximum likelihood estimation (MLE) is used for parameter estimation, which finds the estimators that maximize the probability of generating all the observations. To find maximum likelihood solutions, we compute the derivatives of $\log p(X|\pi, \mu, \Sigma)$ with respect to π_k, μ_k and Σ_k, respectively. The solutions for μ_k and Σ_k are as follows:

$$\begin{aligned} \mu_k &= \frac{\sum_{n=1}^{N} \gamma(z_{nk}) x_n}{\sum_{n=1}^{N} \gamma(z_{nk})}, \\ \Sigma_k &= \frac{\sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k)(x_n - \mu_k)^T}{\sum_{n=1}^{N} \gamma(z_{nk})}, \end{aligned} \tag{5}$$

where $\gamma(z_{nk})$ can be described as the responsibility that each component k takes for explaining observation x_n (Aggarwal and Reddy, 2014) and is defined by

$$\gamma(z_{nk}) = \frac{\pi_k N(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j N\left(x_n|\mu_j, \Sigma_j\right)}. \tag{6}$$

Finally, the solution for π_k is given by

$$\pi_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})}{N}. \tag{7}$$

The MLE solution lacks a closed-form expression and requires an iterative solution. Instead the Expectation-Maximization (EM) algorithm is used to solve this model. EM starts by initializing the parameters using guesses for the means, covariances and mixing probabilities. Then it alternates between two updating steps, the expectation and maximization steps. In the expectation or E-step, the current parameters are used to calculate the posterior probabilities as in Equation (6). In the maximization or M-step, the log-likelihood is maximized using the updated responsibilities in Equations (5) and (7). The algorithm converges once the change in log-likelihood is small enough.

As the EM algorithm converges to a local minimum, the final solution is highly dependent on the initial parameters. We can mitigate this risk by running the algorithm with different initial parameters chosen randomly and take the solution that results in the highest likelihood. This strategy is used for clustering 2013Q1, and afterwards we run the algorithm with model parameters from the previous fit.

4.1.6 Selecting Optimal number of Clusters

Determining the optimal number of clusters or segments is a fundamental step in cluster analysis. The methods in Section 4.1, except for DBSCAN, require the number of clusters k as an input to the algorithm. However, there is no definitive answer for this question and determining the optimal number of clusters is a subjective process dependent. From a statistical standpoint, we assess the number of clusters using the Elbow method, Silhouette score, Calinski-Harabasz and Davies-Bouldin indices and Akaike and Bayesian Information Criteria. However, the most efficient number of clusters does not necessarily align with the idea of interpretable customer segments. Therefore we also inspect different number

of clusters visually and assess the overall trends to select a value. We briefly discuss the evaluation measures.

Elbow Method

The within-cluster sum of squares ($WCSS$) is often used as a quality score of clusters. A smaller value denotes that points within a cluster are situated more densely together (Aggarwal and Reddy, 2014). As such, we can the $WCSS$ for a range of values of k and plot these scores against k. This plot shows that the marginal gain (or reduction in $WCSS$) drops off with increasing k, showing an elbow-like angle in the graph. The elbow criterion selects the optimal k where we find the point of inflection of the curve. It should be noted that the elbow can not always be identified unambiguously. The mathematical expression for the $WCSS$ is given by

$$WCSS = \sum_{j=1}^{k} \sum_{x_i \in C_j} ||x_i - \mu_j||^2. \tag{8}$$

Silhouette Statistic

The silhouette statistic validates the cluster performance based on the mean intra-cluster and the between-cluster distance for each sample (Aggarwal and Reddy, 2014). The score takes values between -1 and 1 where values near 0 indicate overlapping clusters and negative values indicate that the sample is assigned the wrong cluster. The optimal number of clusters is selected by measuring this score for a range of values of k and taking the partition with the maximum score. The expression for the silhouette score is given in Equations (9) and (10).

$$S = \frac{1}{k} \sum_{j=1}^{k} \left(\frac{1}{n_j} \sum_{x_i \in C_j} \frac{b(x_i) - a(x_i)}{max(a(x_i), b(x_i))} \right), \tag{9}$$

where

$$a(x_i) = \frac{1}{n_j - 1} \sum_{x_j \in C_j, j \neq i} ||x_i - x_j||^2, \quad b(x_i) = min_{k, k \neq i} \frac{1}{n_k} \sum_{x_j \in C_j} ||x_i - x_j||^2, \tag{10}$$

where $a(x_i)$ describes the average distance between x_i and all other data points in cluster C_j which is interpreted as a measure of how well x_i is assigned to its cluster. $b(x_i)$ is equal to the smallest average distance between x_i and all points in any of the other clusters. The cluster with smallest $b(x_i)$ is seen as the neighboring cluster of x_i and is the next best fit for this point.

Calinski-Harabasz Index

The Calinski-Harabasz index (CH) (Caliński and Harabasz, 1974) is based on the concept of dense and well-separated clusters. A higher CH score relates to a model with better defined clusters. The index, also known as the Variance Ratio Criterion, is the ratio of the sum of between-clusters dispersion and of inter-cluster dispersion for all clusters:

$$CH = \frac{\text{tr}(B_k)}{\text{tr}(W_k)} \left(\frac{n - k}{k - 1} \right), \tag{11}$$

where $\text{tr}(W_k)$ is the overall within-cluster variance and $\text{tr}(B_k)$ is the overall between-cluster variance defined by:

$$\begin{aligned} B_k &= \sum_{i=1}^{k} n(\mu_i - \mu_S)(\mu_i - \mu_S)^T, \\ W_k &= \sum_{i=1}^{k} \sum_{x \in C_i} (x - \mu_i)(x - \mu_i)^T. \end{aligned} \tag{12}$$

Davies-Bouldin Index

The Davies-Bouldin index (DB) (Davies and Bouldin, 1979) signifies the average distance between clusters, while taking the size of the clusters into account. The lowest possible score is zero and values closer to zero indicate a higher quality. For each cluster C in a partition, the similarity between C and the other clusters is calculated and the highest value is stored. The DB index is then obtained by averaging this highest value of all clusters. The mathematical expression for the DB index is as follows

$$DB = \frac{1}{k} \sum_{i=1}^{k} max_{j,j \neq i} \frac{\frac{1}{n_i} \sum_{x \in C_i} ||x - \mu_i||^2 + \frac{1}{n_j} \sum_{x \in C_j} ||x - \mu_j||^2}{||\mu_i - \mu_j||^2}. \tag{13}$$

Evaluating the aforementioned methods for a range of clusters k at every timepoint is too computationally expensive. Instead, we evaluate these methods at the first quarter of each year to investigate whether structural changes within a partition occur over time.

4.2 Cluster Tracing

We now describe the process of tracing clusters over time by outlining the methods for measuring similarity between clusters of consecutive partitions. As our data does not concern panel data, we are forced to match clusters based on their statistical properties. Two measures are presented based on methods proposed by Sangam and Om (2018) and Günnemann et al. (2011).

Equi-Clustream Tracing

Sangam and Om (2018) propose a framework called Equi-Clustream that analyses cluster relationships using dissimilarity functions for metric and categorical data. For categorical features, they sum the difference in frequency of each category in all categorical features between two clusters. This value is then scaled by the total number of categories present in two clusters. This way the dissimilarity with respect to categorical attributes takes values between 0 and 1. We subtract this value from 1 to transform in into a similarity measure, as show in Equation (14).

$$dist_{cat}\left(C_{t,i}, C_{t+1,j}\right) = 1 - \frac{1}{(m-p)\left(n_{t,i} + n_{t+1,j}\right)} \sum_{s=p+1}^{m} \sum_{r} |\lambda_{t,i}^{s,r} - \lambda_{t+1,j}^{s,r}|, \tag{14}$$

where $\lambda_{t,i}^{s,r}$ describes the total number of observations that contain category r of categorical feature s in cluster i at timepoint t. $n_{t,i}$ defines the number of observations in cluster i,

which multiplied by the number of categorical features gives the total number of categories $(m-p)$ for this cluster in the categorical attribute set.

Similarity in metric characteristics is detected using the mean and standard deviation. The authors min-max normalize the cosine similarity to map the values between 0 and 1, as shown in Equation (15).

$$\begin{aligned} dist_{met,\mu}\left(C_{t,i}, C_{t+1,j}\right) &= \frac{1}{2}\left(\frac{\sum_{s=1}^{p}\mu_{t,i}^{s}\mu_{t+1,j}^{s}}{\sqrt{\sum_{s=1}^{p}(\mu_{t,i}^{s})^{2}}\sqrt{\sum_{s=1}^{p}(\mu_{t+1,j}^{s})^{2}}}+1\right), \\ dist_{met,\sigma}\left(C_{t,i}, C_{t+1,j}\right) &= \frac{1}{2}\left(\frac{\sum_{s=1}^{p}\sigma_{t,i}^{s}\sigma_{t+1,j}^{s}}{\sqrt{\sum_{s=1}^{p}(\sigma_{t,i}^{s})^{2}}\sqrt{\sum_{s=1}^{p}(\sigma_{t+1,j}^{s})^{2}}}+1\right), \end{aligned} \tag{15}$$

where $\mu_{t,i}^{s}$ and $\sigma_{t,i}^{s}$ are the mean value the standard deviation of the s'th metric attribute in cluster i at timepoint t.

The total similarity between two clusters regarding the categorical and metric data is given by Equation (16). The authors opt to weight the separate similarities based on their relative dimensionality.

$$\begin{aligned} dist_{total}\left(C_{t,i}, C_{t+1,j}\right) = \frac{p}{2m}\Big(dist_{met,\mu}\left(C_{t,i}, C_{t+1,j}\right) + dist_{met,\sigma}\left(C_{t,i}, C_{t+1,j}\right)\Big) + \\ \frac{m-p}{m}\, dist_{cat}\left(C_{t,i}, C_{t+1,j}\right) \end{aligned} \tag{16}$$

Kullback-Leibler Divergence

Günnemann et al. (2011) propose to measure cluster similarity using the information theoretic Kullback-Leibler divergence (KL). For clusters $C_{t,i}$ and $C_{t+1,j}$, KL similarity is given by

$$KL(C_{t,i}, C_{t+1,j}, s) = ln\left(\frac{\sigma_{t,i}^{s}}{\sigma_{t+1,j}^{s}}\right) + \frac{(\sigma_{t+1,j}^{s})^{2} + (\mu_{t+1,j}^{s} - \mu_{t,i}^{s})^{2}}{2(\sigma_{t,i}^{s})^{2}} - \frac{1}{2}, \tag{17}$$

where $\mu_{t,i}^{s}$ and $\sigma_{t,i}^{s}$ describe the mean and standard deviation of metric feature s. Using this measure, the mean and variance are combined in defining the proximity between clusters. A high variance in one dimension allows for a high deviation in mean value for clusters to remain similar. Similarly, a low variance only permits a small deviation.

The overall similarity is obtained by averaging the KL over the p dimensions that the clusters have in common as shown in Equation (18). Smaller values of the average KL measure indicate a higher similarity between two clusters.

$$dist_{KL}(C_{t,i}, C_{t+1,j}) = \frac{\sum_{s=1}^{p} KL(C_{t,i}, C_{t+1,j}, s)}{p} \tag{18}$$

4.3 Cluster Labeling

In case the data does not show significant change over time, it may not be necessary to cluster new data. The concept of the data can be assumed to be non-drifting and the behavior of clusters is then considered steady. However, in order to detect concept drift, we require a strategy that measures the similarity between all observations of two timepoints.

Similarity between Quarterly Data

Sangam and Om (2018) define the Hybrid drifting concept detection algorithm (HDCDA) aimed at detecting concept drift. This algorithm adepts their similarity measure between clusters to assess the similarity between all observations of S_t and S_{t+1}. The equation for categorical similarity is given in Equation (19) whereas metric similarity is defined in Equation (20) for both the mean and standard deviation.

$$dist_{cat}\left(S_t, S_{t+1}\right) = \frac{1}{(m-p)(n_t + n_{t+1})} \sum_{s=p+1}^{m} \sum_{r} |\lambda_t^{s,r} - \lambda_{t+1}^{s,r}| \tag{19}$$

$$\begin{aligned} dist_{met,\mu}\left(S_t, S_{t+1}\right) &= \frac{1}{2}\left(\frac{\sum_{s=1}^{p} \mu_t^s \mu_{t+1}^s}{\sqrt{\sum_{s=1}^{p}(\mu_t^s)^2}\sqrt{\sum_{s=1}^{p}(\mu_{t+1}^s)^2}} + 1\right) \\ dist_{met,\sigma}\left(S_t, S_{t+1}\right) &= \frac{1}{2}\left(\frac{\sum_{s=1}^{p} \sigma_t^s \sigma_{t+1}^s}{\sqrt{\sum_{s=1}^{p}(\sigma_t^s)^2}\sqrt{\sum_{s=1}^{p}(\sigma_{t+1}^s)^2}} + 1\right) \end{aligned} \tag{20}$$

The total similarity between S_t and S_{t+1} is given by

$$\begin{aligned} dist_{total}\left(S_t, S_{t+1}\right) = \frac{p}{2m}\Big(dist_{met,\mu}\left(S_t, S_{t+1}\right) + dist_{met,\sigma}\left(S_t, S_{t+1}\right)\Big) + \\ \frac{m-p}{m}\, dist_{cat}\left(S_t, S_{t+1}\right), \end{aligned} \tag{21}$$

which scales the similarities by their dimensionality.

Similarly, we adopt the strategy by Günnemann et al. (2011) to measure similarity between two timepoints instead of clusters. The KL divergence measure is used to evaluate the similarity between S_t and S_{t+1} as shown in Equations (22) and (23).

$$KL(S_t, S_{t+1}, s) = ln\left(\frac{\sigma_t^s}{\sigma_{t+1}^s}\right) + \frac{(\sigma_{t+1}^s)^2 + (\mu_{t+1}^s - \mu_t^s)^2}{2(\sigma_t^s)^2} - \frac{1}{2} \tag{22}$$

$$dist_{KL}(S_t, S_{t+1}) = \frac{\sum_{s=1}^{p} KL(S_t, S_{t+1}, s)}{p} \tag{23}$$

In case the similarity between timepoints is sufficiently large, we assume the data does not experience concept drift and label new observations. Thresholds β_{equi} and β_{KL} are set to determine concept drift and are based on the average similarity between timepoints. When concept drift is observed, we label new observations using the data labeling algorithm HDLA proposed by Sangam and Om (2018) and a classifier based on Extremely Randomized Trees (Geurts et al., 2006). Classification accuracy is measured by the precision, recall and the f1-score, where the true labels follow from clustering new observations in S_{t+1}.

Hybrid Data Labeling Algorithm

By HDLA, the authors assess the similarity between an unlabeled observation x_j and a set of clusters C and label this observation to the closest cluster. The similarity with respect to categorical features is defined based on the importance of the categorical attributes within the cluster which is called the Importance of Nodule (IoN). A nodule is defined as the categorical value z along with its attribute name s in cluster i, represented by $T^i_{z,s}$. The IoN in a cluster depends on two factors, the relative frequency (RF) of a nodule in a cluster and the distribution of a nodule (DN) among all clusters. RF and DN are given in Equation (24) which together form the IoN, given in Equation (25).

$$RF(C_i, T^i_{z,s}) = \frac{|T^i_{z,s}|}{n_i}, \quad DN(C_i, T^i_{z,s}) = \frac{|T^i_{z,s}|}{\sum_{l=1}^{k} |T^l_{z,s}|}, \tag{24}$$

where RF is defined by the frequency of nodule $T^i_{z,s}$ in cluster C_i and DN is given by the frequency of nodule $T^i_{z,s}$ scaled by the overall frequency of the nodule.

$$IoN(C_i, T^i_{z,s}) = \sqrt{RF(C_i, T^i_{z,s}) \cdot DN(C_i, T^i_{z,s})}, \tag{25}$$

where $IoN(C_i, T^i_{z,s})$ describes the importance of the z'th nodule in cluster C_i whose values are set between 0 and 1. For an unlabeled observation x_j, the similarity with a cluster C_i with respect to all categorical features is given by

$$sim_{cat}(x_j, C_i) = \sum_{s=p+1}^{m} \sum_{z \in x_j} IoN(C_i, T^i_{z,s}), \tag{26}$$

which sums the IoN values for the all categorical values present in x_j.

Regarding the similarity for metric attributes, HDLA takes the squared Euclidean distance to the cluster centroid and scales it by the attribute range.

$$sim_{met}(x_j, C_i) = \sum_{s=1}^{p} \left(1 - \frac{(x_{j,s} - \mu_{i,s})^2}{\left(max_{x_i \in C_i}(x_{i,s}, x_{j,s}) - min_{x_i \in C_i}(x_{i,s}, x_{j,s})\right)^2}\right), \tag{27}$$

where $x_{j,s}$ describes the value of x_j for metric feature s and $max_{x_i \in C_i}(x_{i,s}, x_{j,s})$ and $min_{x_i \in C_i}(x_{i,s}, x_{j,s})$ represent the maximum and minimum values of the sth metric feature of cluster i and x_j.

The overall similarity between observation x_j and cluster C_i combines Equations (26) and (27), shown as follows:

$$sim_{total}(x_j, C_i) = \sum_{s=1}^{p} \left(1 - \frac{(x_{j,s} - \mu_{i,s})^2}{\left(max_{x_i \in C_i}(x_{i,s}, x_{j,s}) - min_{x_i \in C_i}(x_{i,s}, x_{j,s})\right)^2}\right) + \sum_{s=p+1}^{m} \sum_{z \in x_j} IoN(C_i, T^i_{z,s}). \tag{28}$$

Extra-Trees Classifier

The Extra-Trees classifier (Geurts et al., 2006) is an ensemble learning method based on decision trees. It is an estimator that fits a number of randomized decision trees on randomized sub-samples of the dataset and averages these trees to improve predictive performance and control overfitting of the data.

To form a node, the best split is determined by searching in a subset of randomly selected features. The split of each selected feature is chosen at random and assessed afterwards using the Gini impurity. Since these splits are random for each feature, the classifier is less computationally expensive than a random forest. We test the Extra-Trees classifier using the ExtraTreesClassifier implementation in scikit-learn (Pedregosa et al., 2011) with the number of trees set to 100.

5 Results and Discussion

An overview of the results of tracing clusters in time-evolving data using measures described in Section 4 is given here. Firstly, the static cluster results are discussed in Section 5.1. The cluster algorithms results are compared on first the timepoint in the dataset, 2013Q1. We evaluate the number of clusters and describe the characteristics of the resulting partitions, highlighting differences among algorithms. We select an initial cluster count k and present results of the proposed framework in Section 5.2. We discern between clustering or classifying new observations depending on the similarity between data of consecutive timepoints. We touch upon the trends observed and visualize the segment transitions.

5.1 Static Clustering Results

The core set of features used the cluster analysis is shown in Table 3, together with their description. We emphasize the different data types as the feature input varies per algorithm.

Table 3: Description of core features used in cluster analysis, separated between numerical and categorical features.

Core Feature	Description
Numerical	
log_annual_inc	Log-transform of the borrower's annual income.
bc_util	Ratio of current credit balance to credit limit for all bankcards.
dti	Debt-to-income ratio.
fico_score	Borrower's FICO score at loan application.
int_rate	Interest rate on the loan.
log_loan_amnt	Log-transform of the loan size applied for by the borrower.
revol_util	Revolving line utilization rate.
log_tot_cur_bal	Total current balance of all accounts.
Categorical	
emp_length	Employment length in years, possible values between 0 and 10.
grade	Lending Club assigned loan grade.
home_ownership	Home ownership status provided by the borrower.
term	Number of payments on the loan, either 3 or 5 years.

5.1.1 Evaluating Cluster Count

We evaluate the K-means, K-prototypes, Agglomerative clustering and DBSCAN algorithms using $WCSS$ values, silhouette scores and the CH and DB index scores to select the cluster count. AIC and BIC are used to evaluate GMM as the other evaluation methods promote dense and distinct clusters while this model produces overlapping clusters. The models are evaluated for $k \in \{2, 3, .., 20\}$ on the data of 2013Q1 and results are shown in Figures 3 to 5. We perform this analysis only for K-means every first quarter to assess cluster count transitions over time.

DBSCAN is omitted from the analysis as it is not able to distinguish multiple segments in the 2013Q1 dataset. Irrespective of input parameters $MinPts$ and Eps, DBSCAN either assigns observations to a single cluster or labels them as outliers, which is not useful from

a business perspective. This occurs due to the lack of separation among regions of high density, causing DBSCAN to detect only a single cluster.

Elbow Method Reflecting on the $WCSS$ scores and resulting Elbow plots in Figure 3, we do not observe a clear indication of an optimal value for k. The vertical dashed line indicates the location of the point of inflection, which is assumed as the optimal cluster count. The models approximate 6 to 8 segments as the optimum value.

Silhouette Score The silhouette score proves to be an approximately decreasing function with k. Each model suggests to use only 2 clusters as this cluster count produces maximum scores. The reason for this behavior may be similar to as why DBSCAN does not perform well, the data is not dispersed sufficiently. Cluster boundaries touch each other or even overlap, resulting in low between-cluster distances relative to the intra-cluster distances. In this scenario, the silhouette score is not equipped to select k.

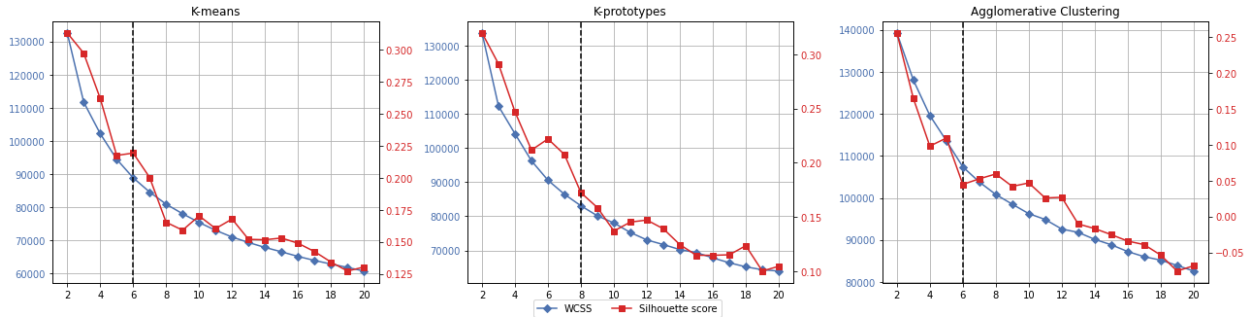

Figure 3: WCSS and Silhouette scores for K-means, K-prototypes and Agglomerative clustering models applied to 2 to 20 segments.

***CH* and *DB* Indices** Regarding the CH index score, we observe corresponding behavior to the silhouette score. The score steadily decreases with k, as shown in Figure 4. Like the silhouette score, the CH index promotes well-separated clusters and is therefore not equipped to select k in this situation. The DB index computes the average similarity between clusters which tends be low for well-separated clusters. We see significant volatility among the three models and neither model agree on a cluster count. Hence, we cannot conclude an optimal cluster count.

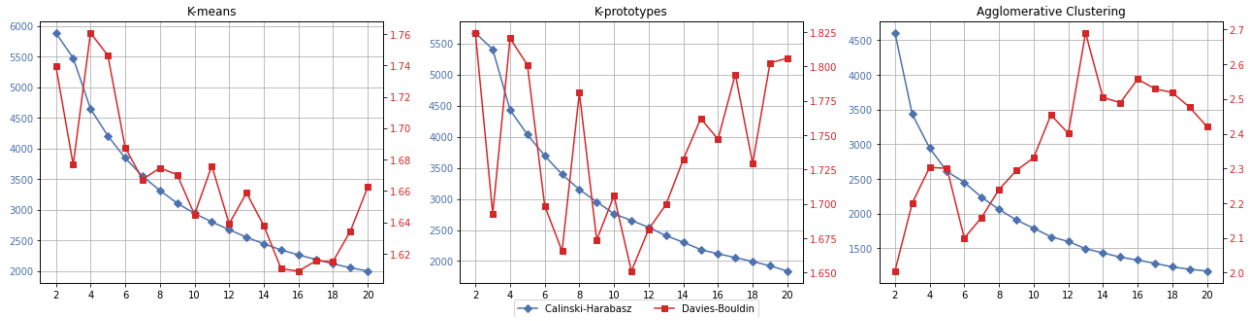

Figure 4: Calinski-Harabasz and Davies-Bouldin indices for K-means, K-prototypes and Agglomerative clustering models applied to 2 to 20 segments.

AIC* and *BIC The *AIC* and *BIC* scores for different values of k using the GMM model are shown in Figure 5. Using these criteria, a model with lower scores is perceived as a better model. As with previously assessed criteria, the scores steadily decrease with k and we are not able to confidently select an optimal cluster count.

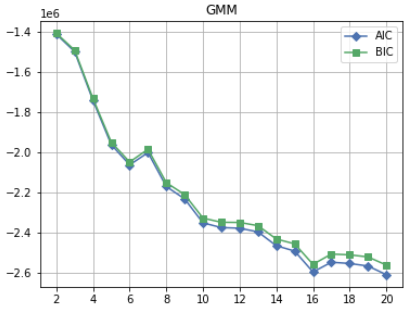

Figure 5: AIC and BIC scores for the GMM model applied to 2 to 20 segments.

For 2014Q1 and the remaining first quarters, we show the evaluation results of K-means in Appendix B.2. Corresponding behavior is detected compared to the results of 2013Q1. This prevents us from selecting an optimal cluster count but tells us that the overall structure in the data does not change significantly over time.

In order to set a final value for k, we examine the resulting partitions of each model performed on the data of 2013Q1. We select $k = 6$ and $k = 8$ to examine as these values are recommended through the elbow plots. We compare the observed behavior which allows us to set k before tracing segments over time. It should be noted that setting an optimal cluster count is dependent on the goals of the cluster analysis and in the case of customer segmentation it may be preferred to assess greater values of k.

5.1.2 Discussion of Observed Clusters

The cluster characteristics for each model in 2013Q1 are summarized in Tables 4 and 5. Deviations from the overall average results are highlighted, representing behavior by core feature characteristics. We split clusters with respect to their income level which allows us to provide a clear interpretation of the partitions. The complete characteristics are given

in Appendix B.5, distinguishing 6 and 8 cluster partitions. The mean values and standard deviations are given for each numerical variable and the mode for categorical ones. We briefly discuss the resulting model partitions.

Table 4: Cluster summaries for K-means, K-prototypes, Agglomerative clustering and GMM with 6 clusters on data from 2013Q1. Highlighting deviations from overall averages.

	Low-Income Clusters	**Mid-Income Clusters**	**High-Income Clusters**
K-means/ K-prototypes	**(1)** high credit usage, small loan size with high interest rates. Poor financial background. **(2)** high credit usage and high dti ratio, low available balance. **(3)** medium credit usage, low available balance. On average loans payed off in 36 months.	**(4)** high credit usage, large loan size on average with high interest rate payed off in 60 months. Poor financial background. **(5)** low credit usage and low dti ratio, very low interest rate. Loans payed off in 36 months. Strong financial background.	**(6)** high credit usage and high current balance, good interest rate for mid-to-high loan size. Loans payed off in 36 months on average. Stable financial background.
Agglomerative	**(1)** high credit usage and dti. **(2)** low-to-medium credit usage, high FICO score and Lending Club grade. Stable financial background. Both segments show small average loan size payed off in 36 months, and both rent a home on average.	**(3)** average loan size with slightly below average interest rate payed off in 36 months. **(4)** high average loan size with high interest rate payed off in 60 months. High dti ratio. Both show high credit usage behavior.	**(5)** very high average loan size. **(6)** low dti ratio, very high FICO score very favorable interest rate. Both show below average credit usage behavior and high current balance. Strong financial background.
GMM	**(1)** clear low-income cluster with high credit usage, overall rather average statistics. **(2)**, **(3)** and **(4)** very close in annual income (mid). **(3)** slightly better in terms of credit usage and dti whereas **(2)** shows worse characteristics compared to average level. Results in better FICO score for **(3)** with below average interest rate. **(4)** high average loan size with high interest rate. Very high dti compared to average figure. **(5)** and **(6)** close in income, leaning towards high-income region. Low-mid credit usage **(6)** against mid-high credit usage **(5)**. **(5)** contains extremely high balance.		

K-means/K-prototypes Because K-means and K-prototypes clustering are both partition-based clustering methods, their partitions look similar. Furthermore, both produce clusters of approximately the same size. K-prototypes deviates slightly through the influence of categorical features. In general, we observe three clusters that represent customers with a considerably low income (Clusters 1, 2 and 3), two that exhibit medium income (Clusters 4 and 5) and a single one representing customers with a high income (Cluster 6).

Agglomerative Regarding the agglomerative clustering algorithm, we portray cluster characteristics using Ward's criterion with Euclidean distance and Complete linkage with Gower distance in Appendix B.3. The partition with Gower distance is expected to put more focus on categorical data compared to Ward's criterion which is beneficial in identifying the type borrower. However, we do observe much different clusters regarding the categorical features. Furthermore, due to the extra focus it misses out on distinction in metric features. For example, using Ward's criterion we observe clear distinctions regarding credit usage behavior whereas this is not the case using Gower distance. This behavior captured using metric features is deemed more important and as such we implement agglomerative clustering with Ward's criterion. The resulting partition is evenly spread as opposed to K-means and K-prototypes, it includes two clusters for each income level. Similarly, clusters are separated through their financial background, in particular through credit usage behavior.

GMM By estimating a multivariate Gaussian distribution as a representation for each customer segment, Gaussian Mixture Models take a different approach. In Appendix B.4, we present cluster characteristics for GMM with K-means and random initialization while also in- and excluding categorical features. In terms of model efficiency, after running as few as 25 random initializations, K-means initialization produces only a marginally more efficient partition. Regarding segment characteristics, less focus is put on metric features when categorical features are included. This results in segments that become similar metric-wise and dissimilar on a categorical level. As with agglomerative clustering, describing segments through customer background or lending behavior by using metric features gives more interesting results. Hence, we opt for random initialization (number of init. = 50) with metric core features. For a 6-cluster partition, there is a reduced separation in annual income compared to the other models but differences in lending behavior and credit usage are still observed.

On the 8 cluster partitions, we observe a gradual range of segments with respect to annual income. K-means and K-prototypes still provide a partitioning with overall similar behavior. K-prototypes puts less focus on the high-income clusters, capturing their behavior in a single segment. The agglomerative partition puts more focus on the mid-income clusters, distinguishing their behavior in four clusters. On the high-income clusters, the model decides to differentiate their behavior through the loan size instead of lending behavior in K-means. With respect to the GMM partition, it is clear that including more clusters produces a better structured solution. The extra clusters allow GMM to further refine the data into distinct segments.

In general we describe segments through two characteristics, a customers' lending behavior and their financial background. The former through the loan size and its interest rate which is also linked to term length. The latter by the annual income and credit usage behavior through the bankcard credit and revolving credit utilization rates. When increasing k, the data is refined into more detailed clusters in terms of the mentioned characteristics. However, the question of how many segments to include comes then down to problem at hand and the goal to be achieved. For example, if the objective is to get a deep market understanding such that promotions can be target at certain segments, one would prefer a detailed segmentation. On the other hand, if the goal is exploratory it is not necessary to go for such a detailed segmentation. The focus of this work is to show exploratory results for customer segmentation by assessing different cluster models and tracking discovered segments over time. We observe slight differences between 6 and 8 segments which inform us about the general trend of increasing k. For this large data set, including only 6 segments is insufficient and test tracking methods with 8-cluster partitions.

5.2 Tracking Discovered Segments

Given a starting partition for each model using 8 segments, we are able to test the framework of Figure 2 and describe the cluster trends. First, we show the cluster transition results where only clustering algorithms are applied. Secondly, we show results where labeling techniques are implemented.

Table 5: Cluster summaries for K-means, K-prototypes, Agglomerative clustering and GMM with 8 clusters on data from 2013Q1. Highlighting deviations from overall averages.

	Low-Income Clusters	Mid-Income Clusters	High-Income Clusters
K-means	**(1)** high credit usage, very small loan size with high interest rates. Very low income and available balance. Poor financial background. **(2)** small average loan size, low credit usage but low available balance. Stable financial background **(3)** high credit usage with very high dti ratio. More frequently owning a mortgage compared to renting a home. On average loans payed off in 36 months for low-income segments.	**(4)** very high credit usage and low available balance but low dti ratio. Still renting a home. **(5)** very high credit usage and high dti ratio. Large average loan size with large interest rate payed off in 60 months. **(6)** very low credit usage and dti ratio resulting in very low interest rate. Strong financial background.	**(7)** medium credit usage. Strong financial background. **(8)** high credit usage. Stable financial background Similar characteristics, different in credit usage. Both above average loan size with below average interest rate, payed off in 36 months on average.
K-prototypes	**(1)** very high credit usage and dti, very low income. Small average loan size with relatively high interest rate. **(2)** low credit usage and dti. Small average loan size. Stable financial background. **(3)** medium credit usage and high dti. Average available balance giving below average interest rate. Stable financial background. **(1)** and **(2)** are similar segments except for their lending behavior. All segments pay off loans in 36 months on average.	**(4)** very high credit usage but very low dti, very low available balance. **(5)** very high credit usage and high dti ratio. Both loan size and interest rate of average size, payed off in 36 months. Stable financial background. **(6)** very high credit usage and dti ratio. On average large loan size payed off in 60 months which results in high interest rate. Poor financial situation. **(7)** very low credit usage and low dti ratio with available balance. Results in very favorable interest rate. Very strong financial background.	**(8)** medium credit usage, very high current balance. On average large loan size payed off in 36 months which results in a favorable interest rate. Strong financial background.
Agglomerative	**(1)** high credit usage and dti. **(2)** low-to-medium credit usage, high FICO score and Lending Club grade for low-income cluster. Stable financial background. Similar characteristics, different in credit usage. Both segments show small average loan size payed off in 36 months, and both rent a home on average.	**(3)** high dti ratio with mid to high credit usage. Owns on average a mortgage regarding home ownership. **(4)** low dti ratio with mid to high credit usage. Still renting a home. **(5)** very high credit usage and high dti ratio. Large average loan size payed off in 60 months, results in high interest rate. **(6)** small segment, very strong financially. Very low credit usage and dti ratio, high current balance. Resulting in very favorable interest rate.	**(7)** very high average loan size, payed off in 60 months resulting in a mediocre interest rate. **(8)** Slightly below average dti ratio. Slightly above average loan size, payed off in 36 months resulting in very low interest rate. Both show below average credit usage behavior and high current balance. Strong financial background
GMM	**(1)** high credit usage and high dti ratio. Slightly above average loan size with above average interest rate. Very low current balance. **(2)** medium credit usage and below average dti ratio. Slightly below average loan size and interest rate. **(3)** medium credit usage. Small average loan size and noticeably high interest rate. On average, term length equals 36 months and segments rent a home.	**(4)** medium credit usage. Below average loan size and favorable interest rate. Noticeably high current balance. **(5)** medium credit usage with below average dti ratio. High FICO score. Average loan size with below average interest rate. Stable financial background. **(6)** high credit usage with above average dti ratio. Below average loan size but above average interest rate. On average, home ownership type equals mortgage for all segments.	**(7)** high credit usage and dti ratio. Very high loan size with on average interest rate. **(8)** very low credit usage with low dti ratio. On average loan size with favorable interest rate. Signified by very high FICO score. Strong financial background. Both segments have a mortgage on average.

5.2.1 Framework: Cluster Analysis

We elaborate on the segments we discovered in Section 5.1 of each model and summarize our findings. Furthermore, we describe the different matching methods by their cluster similarity values and whether matched clusters are a clear match, or multiple matches are observed.

K-means and K-prototypes Clustering

By using the previous cluster centers as initial centroids for K-means and K-prototypes at the next timepoint, the resulting cluster centers do not deviate much from the initial ones. This is also because the quarterly change in customer behavior was not expected to differ much and instead long-term changes were suspected. Because of the lack in quarterly change, the

matching methods do not have trouble picking up the matching cluster. However, if we were to initialize the algorithms at the next timepoint randomly or using the K-means++ method, the tracking methods are inadequate. The partition shifts with respect to the previous one, causing a loss in structure. The differences in structure often do not match well, missing out on 1-to-1 matches for all clusters. Hence, new customer segments are continuously discovered which is not true what happens in reality.

Although both tracking methods succeed in selecting the best matching cluster, we notice that the KL divergence measure produces a clearer difference between matching and non-matching clusters. In Table 6, we display the average similarity of the matching cluster, compared to the average similarity of the next three closest clusters. By combining the mean and standard deviation in the KL divergence measure, instead of calculating cluster distance separately in EquiClustream, we allow a better interpretation of the distributions of metric features. We also saw in Section 5.1 that the influence of categorical features on cluster generation is only modest. Therefore, it is not an issue to exclude these features with respect to the challenge of matching clusters.

As both matching methods are both suitable in this setting, we track the segments initially discovered in 2013Q1 until 2018Q4 and capture their changes in behavior. The observed trends are briefly summarized in Table 7, highlighting changes in behavior. In Figures 6 and 7, we visualize the development over time of metric features using K-means clustering and illustrate changes in segment size. The cluster trends of K-prototypes are shown in Appendix C.1. Regarding segment size, we observe a gradual growth of all segments until 2015Q4 after which all segments experience a drop-off due to the Lending Club public scandal. Because K-means and K-prototypes are partition-based algorithms, the resulting clusters are spherical objects where all segments are and remain roughly the same size relative to each other.

Table 6: Similarity values of both cluster matching methods, distinguishing between the similarity of the matching clusters and the average similarity of the next three closest matches.

	EquiClustream		KL divergence	
	Matches	Non-matches	Matches	Non-matches
K-means	0.972	0.808	0.018	0.812
K-prototypes	0.973	0.805	0.016	0.803
GMM	0.974	0.802	0.093	1.057

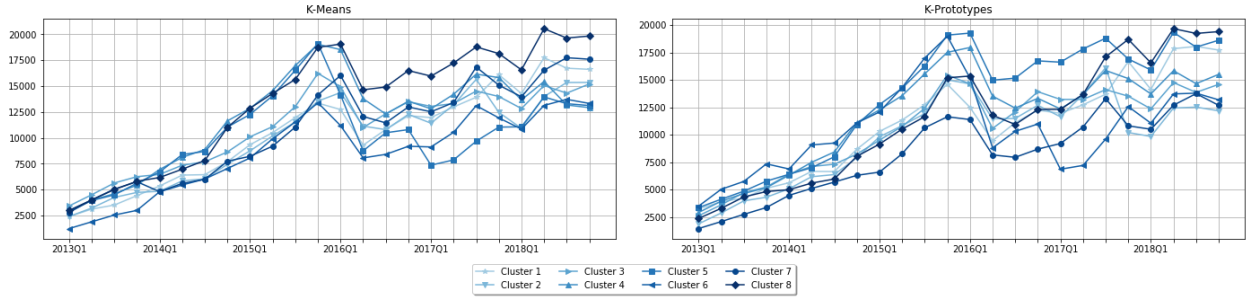

Figure 6: Development over time of segment sizes using K-means and K-prototypes.

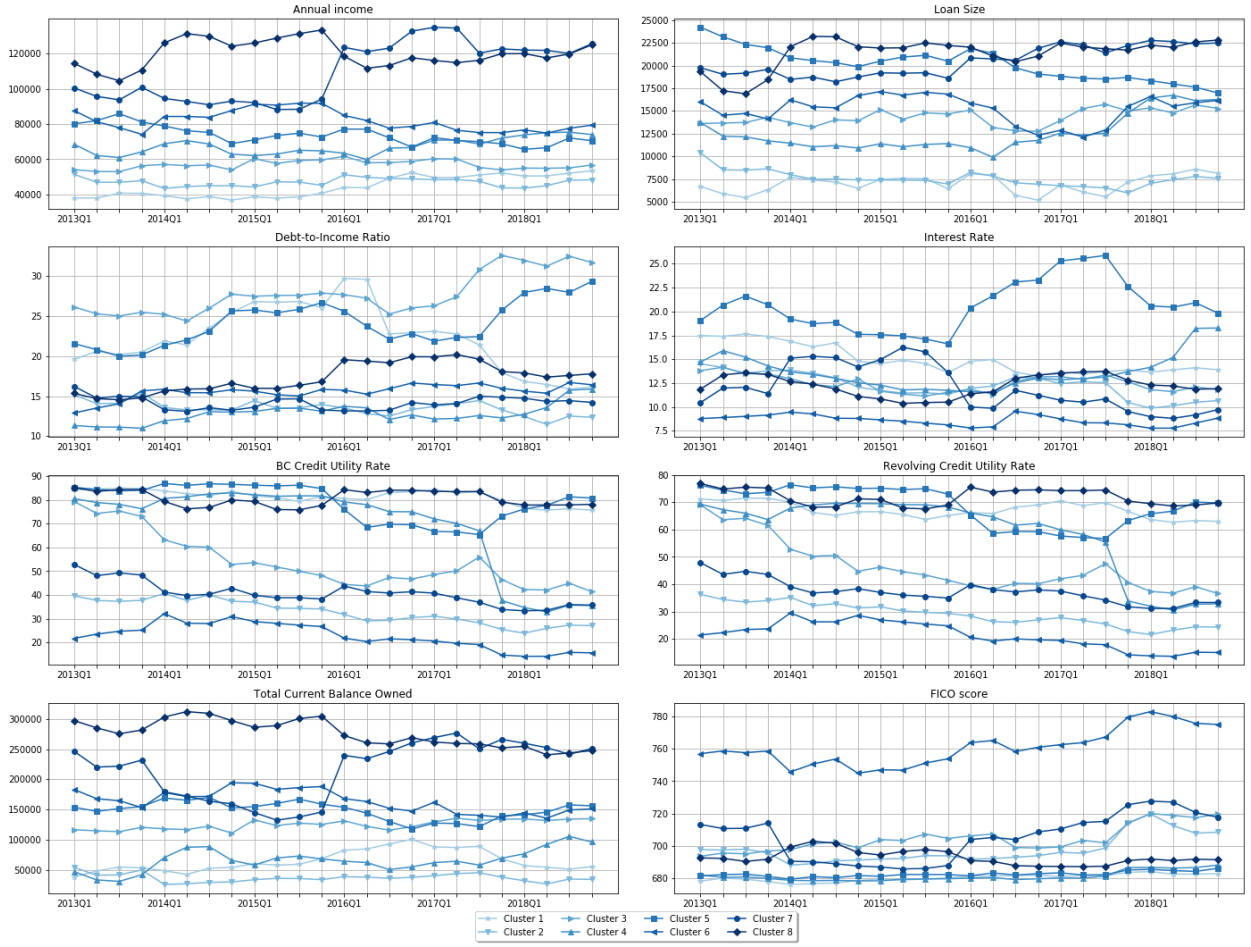

Figure 7: Customer segment trends of metric features between 2013Q1 and 2018Q4 using K-means clustering. The customer segments are ranked in terms of annual income as measured in 2013Q1.

Agglomerative Clustering

By using agglomerative clustering, a completely new model is initialized at each timepoint as opposed to the partition-based methods which use previous iteration results as initialization.

Table 7: Brief description of cluster trends observed for the discovered segments by K-means and K-prototypes. Each segment of the original partitions is used to describe its' trends.

	Low-Income Clusters	Mid-Income Clusters	High-Income Clusters
K-means	**(1)** Remains to be high credit user and increases in dti ratio until 2016Q1 after which it strongly improves an average dti ratio. Experiences significant rise in income which results in a decrease in interest rate. **(2)** Stable regarding income but sees a decrease in average loan size and dti ratio. Also steadily lowering its' credit usage, resulting in stronger FICO scores and lower interest rates. **(3)** Transitions from high to low/medium credit usage segment whilst significantly increasing its' dti ratio. Still sees a great improvement in FICO score and a better interest rate.	**(4)** Stable regarding income, but improves significantly in credit usage from 2017 onwards. Average loan size grows accompanied by an increase in interest rate. **(5)** Steadily decreases in loan size. Starts improving regarding credit usage and dti ratio between 2015Q4 and 2017Q2 but trend reverses afterwards resulting in high credit usage. Reversed behavior observed on interest rate, between 2015Q4 and 2017Q2 a steep climb followed by a drop-off. **(6)** Very steady segment over time. Becomes even better in terms of credit usage but slightly increases in dti ratio. On average extremely high FICO score.	**(7)** Slightly improves its' credit usage over time and we observer slight increase in loan size. Confirms its' status as a high-income cluster after 2015Q4 with strong increase in annual income. **(8)** Remains a very high credit usage segment. Slight increase in loan size but stable regarding interest rate. Even though its' very average current balance, still obtains a poor FICO score through its' risky lifestyle.
K-prototypes	**(1)** Slightly increases in income over time and improves its' dti ratio, which results in a lower interest rate over time. Still remains to be a high credit usage segment with below average loan sizes. **(2)** Stable on income and loan size over time. Significant decrease in credit usage, resulting in very high FICO scores on average and very favorable interest rates. **(3)** Slightly increases in loan size over time. Strong increase in dti ratio but slightly decreases its' dependence on credit. Because of relatively good credit usage, this segments receives decently high FICO scores. **(4)** Stable regarding annual income until 2017Q3, after which a steep decline is observed. Similar drop in average loan size is detected. Changes into a low credit usage cluster over time, with a notable decline after 2017Q3.	**(5)** Quickly becomes a high-income cluster accompanied by a very high loan size on average. Remains stable over time regarding other characteristics such as credit usage behavior. **(6)** Seems to slightly decrease in value regarding annual income, loan size and credit usage. Gradually increase in dti ratio and interest rate. **(7)** Stable regarding annual income until 2017Q3, after which a significant increase is observed, moving in the high-income region. Similar increase is noticed on the average loan size. Remains to obtain very favorable interest rates as this segment slightly lowers its' credit usage behavior. Best FICO scores of all segments observed.	**(8)** Fluctuates regarding annual income over time, ending with a worsened value. Gradually improves its' credit usage. Remains a stable cluster otherwise.

As we discussed earlier, this causes the algorithm to shift its partition with respect to the one of the previous timepoint, leading to a cluster mismatch. We display cluster results of 2013Q1 and 2013Q2 to illustrate this issue in Appendix C.2 in Table 19.

Some clusters still acquire an obvious match captured by either of the tracking methods. However, a mismatch is observed among mid-income clusters where multiple clusters in 2013Q1 match a single one in 2013Q2 because of high overlap. Not all segments in 2013Q2 are matched and new segments are initiated. This occurs frequently over time which implies continuous discovery of new segments. This is obviously not the case in our setting and we conclude that agglomerative clustering is not applicable here. Hence, the proposed tracking methods become ineffective.

GMM

Like K-means and K-prototypes clustering, GMM allows us to use previous model output as input for the next one timepoint. Similarly, it allows us to trace the segments discovered 2013Q1 and retain the structure devised in Section 5.1. Compared to K-means and K-prototypes clustering, the trends of GMM displays smoother results, shown in Figure 8. On the segment sizes in Figure 9, we observe similar trends as the partition-based algorithms. In Table 8 we briefly describe the changes we observe on the segments identified in 2013Q1.

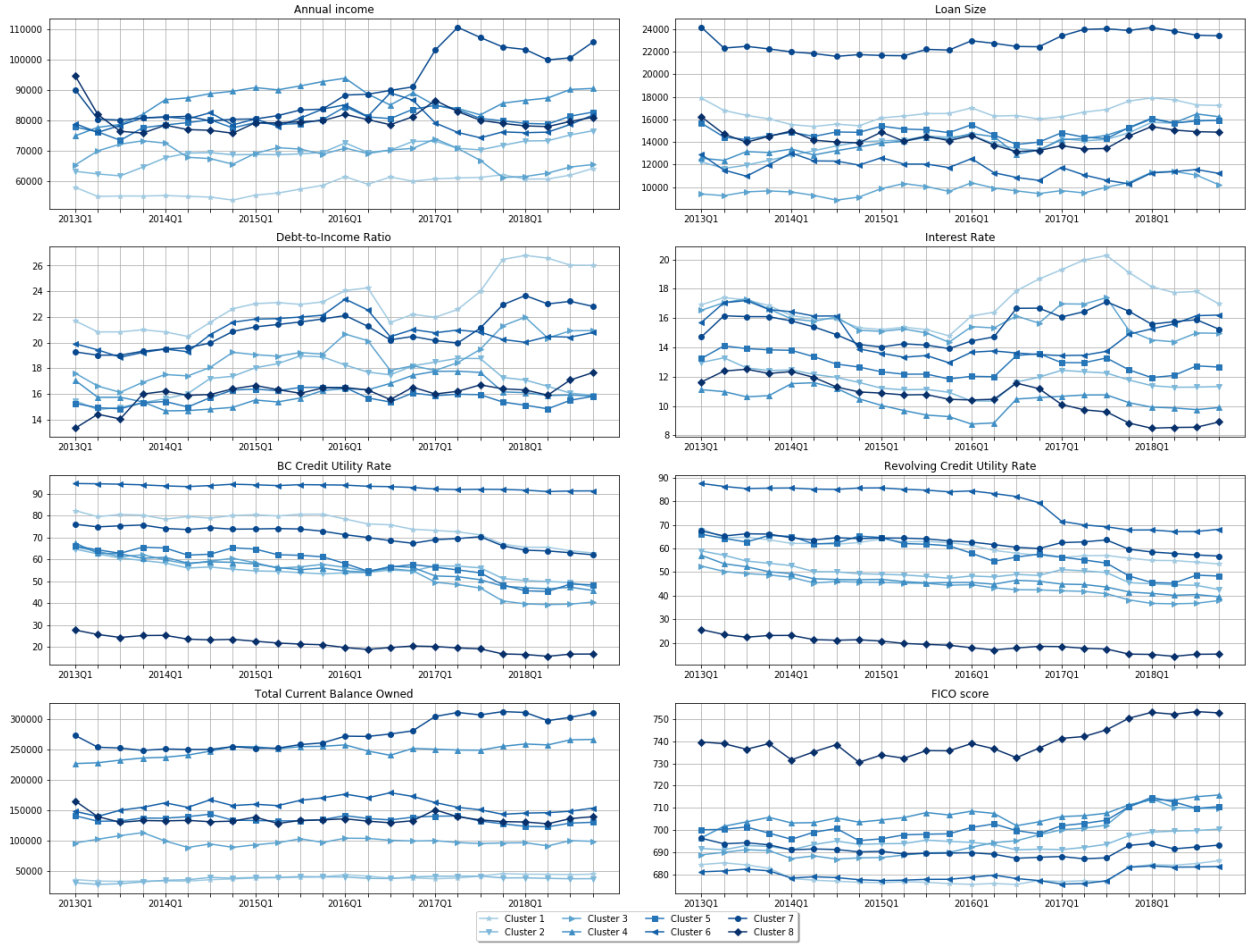

Figure 8: Customer segment trends of metric features between 2013Q1 and 2018Q4 by a Gaussian mixture model. The customer segments are ranked in terms of annual income as measured in 2013Q1.

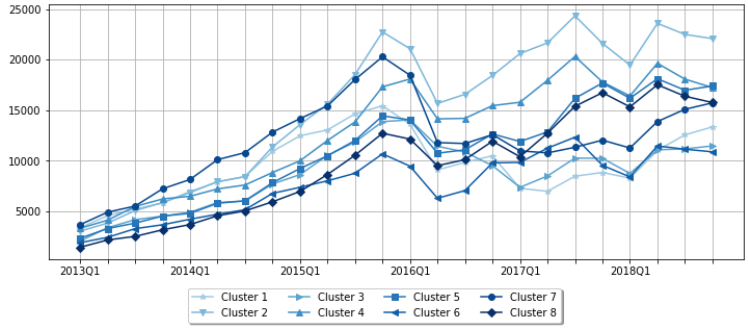

Figure 9: Development over time of segment sizes using GMM.

Table 8: Brief description of cluster trends observed for the discovered segments by GMM. Each segment of the original partitions is used to describe its' trends.

Low-Income Clusters	Mid-Income Clusters	High-Income Clusters
(1) Gradually improves on average income, while reducing its' dependence on credit over time. Significant increase in dti ratio from 2017 onwards. Stable regarding average loan size but fluctuating in terms of interest rate, ending a similar level at which it starts.	**(4)** Moves into the high-income region accompanied by an increase in average loan size. Decreases gradually on credit usage while remaining stable on the dti ratio. Sees a steady increase in FICO score, which also has a positive influence on the interest rate.	**(7)** Initially declines in terms of annual income but recovers from 2016Q4 onwards, moving into a surprisingly high average annual income. Gradually decreases its' credit usage whilst increase its' dti ratio. Remains the segment with notably high loan sizes accompanied by high interest rates.
(2) Gradually improves the average income, moving into the mid-income region. At the same time, increase its' average loan size steadily. Furthermore, experiences a gradual decrease in credit usage over time. This results in a slightly improvement in FICO scores.	**(5)** This segment remains surprisingly stable over time on almost all characteristics. Only noticeable change observed on credit usage, which gradually decreases over time. Which in turn causes the average FICO score to increase gradually.	**(8)** Moves into the mid-income region almost immediately where it remains. Stays stable regarding the annual income but significantly improves the interest rate as this segment gradually decreases its' credit usage. Regarding the dti ratio, we observe a significant increase over time. Nevertheless, this segment remains the strongest financially, portrayed by the huge FICO scores.
(3) Remains to be a segment with low loan sizes with relatively high interest rates on average. However, sees a significant increase in dti ratio and a gradual decrease in credit ratio. This results in improved FICO scores.	**(6)** Remains stable with respect to annual income and loan size. Bankcard credit usage remains stable whereas revolving credit usage slightly decreases. However still obtains poor FICO scores resulting in unfavorable interest rates.	

5.2.2 Framework: Clustering and Labeling

Sangam and Om (2018) argue that labeling new observations instead of clustering them saves computation time. This is useful when newly seen observations are comparable to the already clustered data. However, in our setting K-means clustering or GMM cluster and track segments from 2013Q1 to 2018Q4 in a matter of minutes. With over 2 million observations, this computation time not seen as an issue. Hence the argument of labeling instead of clustering to save computation time is not necessarily applicable in this case. Regarding K-prototypes, the framework completes clustering the loans of 2013Q1 to 2018Q4 within two hours, taking significantly longer compared to K-means because of the addition of categorical features. Nevertheless, we still deem this as sufficiently low, preventing the need for labeling.

On the quarterly similarity, in Table 9 we show the average similarity values for both methods as well as the average yearly similarity. As expected, the yearly similarity is somewhat smaller compared to the quarterly average. In general, a high quarterly and yearly similarity is observed which signifies the lack of substantial change in the landscape of P2P lending within the monitored timeframe.

In order to give an indication of the labeling performance of the HDLA algorithm by Sangam and Om (2018) and extra trees classifier, we test both methods with respect to partitions of K-means clustering. We compare the label strategies on precision, recall and accuracy on the timepoints on both the labeling timepoint as well as the next timepoint.

This allows us to analyze the labeling performance itself and the influence on following partitions. We illustrate the performance on quarterly data where we observe a similarity of over 0.95 which occurs at 9 out of 24 timepoints: {2014Q3, 2016Q1, 2016Q4, 2017Q1, 2017Q2, 2017Q4, 2018Q1, 2018Q3 and 2018Q4}. In Table 10, we display the classifying performance, where we see that the extra trees classifier achieves more than double the accuracy of HDLA.

Table 9: Quarterly and yearly data similarity values using EquiClustream's similarity method and the KL divergence measure.

EquiClustream		**KL divergence**	
Quarterly	Yearly	Quarterly	Yearly
0.969	0.906	0.0026	0.0135

Table 10: Multiclass classification evaluation statistics for extra trees classifiers and HDLA algorithm

	Extra Trees Class.	**HDLA**
Precision	0.856	0.281
Recall	0.845	0.312
F1-score	0.845	0.258
Accuracy	0.846	0.313

The extra trees classifier performs well with an average accuracy of almost 85% which is raised to 96% in the timepoint after labeling. A minimum accuracy of 70% observed as a series of 5 out of 6 quarters is labeled instead of clustered. If we are to impose the rule to force clustering the timepoint after labeling one, we get an average labeling accuracy of 86.3% which is raised to 99.3% in the successive timepoint due to the clustering. Hence it is not advised to label multiple timepoints in a row, but force in between clustering to retain performance and not dwell away from the cluster concepts.

The low scores for the HDLA algorithm are addressed to the fact that each time the algorithm labels a timepoint, it completely misses out on one or two of the segments and only classifies data to the remaining segments. Possibly because within an income-level, segments may show significant similarity and only differ on detailed aspects. These aspects may be too detailed for HDLA distinguish and as such it drops considerably in accuracy.

6 Conclusion

In this thesis we studied the performance of existing methods of tracking customer segments applied to P2P lending data. The purpose was primarily exploratory in presenting methods for obtaining interpretable customer segments that are useful for business processes such as targeted marketing. We focused on different types of cluster techniques and evaluated the effectiveness of tracking methods on these partitions.

We show that customer behavior can be adequately described by the financial background, the type of loans customers apply for, and credit usage behavior. Segments significantly differ in their utilization of available credit; bankcard and revolving credit, combined with the varying levels of annual income and available savings. On the P2P lending data set, we have seen that categorical features indicating customer characteristics such as home ownership and career type or career length have no significant influence on the cluster results.

On selecting the cluster count, we given a detailed analysis of partitions including six and eight clusters. We have seen that increasing cluster count leads to detailed segments in terms of aforementioned types of behavior. The criteria for selecting cluster count have shown to be inadequate in the setting of P2P lending data. Statistics that promote dense and well-separated clusters, such as the silhouette score and CH and DB indices, are ineffective in this setting.

We have shown that, without clear separation between clusters in terms of density, density-based clustering algorithms are unable to discover different customer segments. Other applied algorithms; K-means, K-prototypes, Agglomerative clustering and GMM show comparable results and discover clear distinct segment profiles.

On the ability of tracking segments over time, the overall partition structure has to remain unchanged whilst allowing for small deviations over time. When the structure deviates significantly both tracking methods become ineffective. This occurs with agglomerative clustering, where partitions are insightful but differ structurally from quarter to quarter. We have shown that incremental methods that include partition-based clustering algorithms perform well in this scenario. By continuously initializing K-means and K-prototypes with previous cluster centroids, we retain the initial structure which gradually develops over time. This allows for effortless segment tracking using EquiClustream tracing or the KL divergence measure. By continuously updating the GMM model over time, we maintain the structure in a similar fashion. Hence, the combination of an incremental cluster algorithm with a tracking method that utilises the mean and standard deviation work well in our setting.

By applying label strategies in case quarterly data is sufficiently similar, we prevent the need to cluster data at each timepoint which may be more time consuming. We have shown than an extra trees classifier produces satisfactory results whereas HDLA performs poorly in our setting.

Finally, this work is limited in the sense that we keep the number of clusters constant over time. The framework requires a method of selecting and updating the cluster count more efficiently, or one has to select a clustering algorithm which estimates the number of clusters along with the clustering. However, it also requires that this cluster method is applicable to a customer segmentation problem and allows for incremental updating.

References

Ackerman, M. and Dasgupta, S. (2014), "Incremental clustering: The case for extra clusters," in *Advances in Neural Information Processing Systems*, pp. 307–315.

Aggarwal, C. C. and Reddy, C. K. (2014), "Data clustering," *Algorithms and Application, Boca Raton: CRC Press*.

Aggarwal, C. C. and Yu, P. S. (2005), "Online analysis of community evolution in data streams," in *Proceedings of the 2005 SIAM International Conference on Data Mining*, SIAM, pp. 56–67.

Agrawal, R., Gehrke, J., Gunopulos, D., and Raghavan, P. (1998), *Automatic subspace clustering of high dimensional data for data mining applications*, vol. 27, ACM.

Alqurashi, T. and Wang, W. (2019), "Clustering ensemble method," *International Journal of Machine Learning and Cybernetics*, 10, 1227–1246.

Ankerst, M., Breunig, M. M., Kriegel, H.-P., and Sander, J. (1999), "OPTICS: ordering points to identify the clustering structure," in *ACM Sigmod record*, ACM, vol. 28, pp. 49–60.

Arthur, D. and Vassilvitskii, S. (2007), "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, Society for Industrial and Applied Mathematics, pp. 1027–1035.

Berkovich, E. (2011), "Search and herding effects in peer-to-peer lending: evidence from prosper. com," *Annals of Finance*, 7, 389–405.

Bezdek, J. C., Ehrlich, R., and Full, W. (1984), "FCM: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, 10, 191–203.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., and Sander, J. (2000), "LOF: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104.

Caliński, T. and Harabasz, J. (1974), "A dendrite method for cluster analysis," *Communications in Statistics-theory and Methods*, 3, 1–27.

Cambridge Judge Business School (2019), "Cambridge Centre for Alternative Finance," `https://www.jbs.cam.ac.uk/faculty-research/centres/alternative-finance/`.

Cao, F., Liang, J., and Bai, L. (2009), "A new initialization method for categorical data clustering," *Expert Systems with Applications*, 36, 10223–10228.

Carpenter, G. A. and Grossberg, S. (1988), "The ART of adaptive pattern recognition by a self-organizing neural network," *Computer*, 21, 77–88.

Chakrabarti, D., Kumar, R., and Tomkins, A. (2006), "Evolutionary clustering," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 554–560.

Chi, Y., Song, X., Zhou, D., Hino, K., and Tseng, B. L. (2007), "Evolutionary spectral clustering by incorporating temporal smoothness," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 153–162.

Club, L. (2019), "Peer-to-Peer loan data of loans up to $40,000," `https://www.lendingclub.com`, accessed: 2019-10-29.

Davies, D. L. and Bouldin, D. W. (1979), "A Cluster Separation Measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1, 224–227.

Dibb, S. (1998), "Market segmentation: Strategies for success," *Marketing Intelligence and Planning*, 16, 394–406.

Emekter, R., Tu, Y., Jirasakuldech, B., and Lu, M. (2015), "Evaluating credit risk and loan performance in online Peer-to-Peer (P2P) lending," *Applied Economics*, 47, 54–70.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996), "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Kdd*, vol. 96, pp. 226–231.

Fisher, D. H. (1987), "Knowledge acquisition via incremental conceptual clustering," *Machine learning*, 2, 139–172.

Fraley, C. and Raftery, A. E. (1999), "MCLUST: Software for model-based cluster analysis," *Journal of classification*, 16, 297–306.

Freedman, S. and Jin, G. Z. (2017), "The information value of online social networks: lessons from peer-to-peer lending," *International Journal of Industrial Organization*, 51, 185–222.

García-Laencina, P. J., Sancho-Gómez, J.-L., and Figueiras-Vidal, A. R. (2010), "Pattern classification with missing data: a review," *Neural Computing and Applications*, 19, 263–282.

Geurts, P., Ernst, D., and Wehenkel, L. (2006), "Extremely Randomized Trees," *Mach. Learn.*, 63, 3âĂŞ42.

Ghaemi, R., Sulaiman, M. N., Ibrahim, H., Mustapha, N., et al. (2009), "A survey: clustering ensembles techniques," *World Academy of Science, Engineering and Technology*, 50, 636–645.

Gower, J. C. (1971), "A general coefficient of similarity and some of its properties," *Biometrics*, 857–871.

Guha, S., Rastogi, R., and Shim, K. (1998), "CURE: an efficient clustering algorithm for large databases," in *ACM Sigmod Record*, ACM, vol. 27, pp. 73–84.

— (2000), "ROCK: A robust clustering algorithm for categorical attributes," *Information systems*, 25, 345–366.

Günnemann, S., Kremer, H., Laufkötter, C., and Seidl, T. (2011), "Tracing evolving clusters by subspace and value similarity," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Springer, pp. 444–456.

Han, J., Pei, J., and Kamber, M. (2011), *Data mining: concepts and techniques*, Elsevier.

He, C., Zhang, Q., Tang, Y., Liu, S., and Zheng, J. (2019), "Community detection method based on robust semi-supervised nonnegative matrix factorization," *Physica A: Statistical Mechanics and its Applications*, 523, 279–291.

Herrero-Lopez, S. (2009), "Social interactions in P2P lending," in *Proceedings of the 3rd Workshop on Social Network Mining and Analysis*, pp. 1–8.

Herzenstein, M., Dholakia, U. M., and Andrews, R. L. (2011), "Strategic herding behavior in peer-to-peer loan auctions," *Journal of Interactive Marketing*, 25, 27–36.

Hinneburg, A., Keim, D. A., et al. (1998), "An efficient approach to clustering in large multimedia databases with noise," in *KDD*, vol. 98, pp. 58–65.

Huang, Z. (1997a), "Clustering large data sets with mixed numeric and categorical values," in *Proceedings of the 1st pacific-asia conference on knowledge discovery and data mining,(PAKDD)*, Singapore, pp. 21–34.

— (1997b), "A fast clustering algorithm to cluster very large categorical data sets in data mining," *DMKD*, 3, 34–39.

Hubert, M. and Vandervieren, E. (2008), "An adjusted boxplot for skewed distributions," *Computational statistics & data analysis*, 52, 5186–5201.

Iglewicz, B. and Hoaglin, D. (1993), *How to Detect and Handle Outliers*, ASQC basic references in quality control, ASQC Quality Press.

Johnson, S. C. (1967), "Hierarchical clustering schemes," *Psychometrika*, 32, 241–254.

Kalnis, P., Mamoulis, N., and Bakiras, S. (2005), "On discovering moving clusters in spatio-temporal data," in *International Symposium on Spatial and Temporal Databases*, Springer, pp. 364–381.

Karypis, G., Han, E.-H. S., and Kumar, V. (1999), "Chameleon: Hierarchical clustering using dynamic modeling," *Computer*, 32, 68–75.

Kaufman, L. and Rousseeuw, P. J. (1987), "Clustering by means of medoids. Statistical Data Analysis based on the L1 Norm," *Y. Dodge, Ed*, 405–416.

— (1990), "Partitioning around medoids (program pam)," *Finding groups in data: an introduction to cluster analysis*, 344, 68–125.

— (2008), "Clustering large applications (Program CLARA)," *Finding groups in data: an introduction to cluster analysis*, 126–163.

Kriegel, H.-P., Kröger, P., Sander, J., and Zimek, A. (2011), "Density-based clustering," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1, 231–240.

Kumar, R., Novak, J., and Tomkins, A. (2006), "Structure and evolution of online social networks," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 611–617.

Ledoit, O. and Wolf, M. (2003), "Improved estimation of the covariance matrix of stock returns with an application to portfolio selection," *Journal of empirical finance*, 10, 603–621.

Leskovec, J., Backstrom, L., Kumar, R., and Tomkins, A. (2008), "Microscopic evolution of social networks," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 462–470.

Li, Z., Lee, J.-G., Li, X., and Han, J. (2010), "Incremental clustering for trajectories," in *International Conference on Database Systems for Advanced Applications*, Springer, pp. 32–46.

Lin, X., Li, X., and Zheng, Z. (2017), "Evaluating borrower's default risk in peer-to-peer lending: evidence from a lending platform in China," *Applied Economics*, 49, 3538–3545.

Ma, X., Li, D., Tan, S., and Huang, Z. (2019), "Detecting evolving communities in dynamic networks using graph regularized evolutionary nonnegative matrix factorization," *Physica A: Statistical Mechanics and its Applications*, 121279.

MacQueen, J. et al. (1967), "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Oakland, CA, USA, vol. 1, pp. 281–297.

McLachlan, G. J. and Krishnan, T. (2007), *The EM algorithm and extensions*, vol. 382, John Wiley & Sons.

Nan, D.-Y., Yu, W., Liu, X., Zhang, Y.-P., and Dai, W.-D. (2018), "A framework of community detection based on individual labels in attribute networks," *Physica A: Statistical Mechanics and its Applications*, 512, 523–536.

Ng, R. T. and Han, J. (2002), "CLARANS: A method for clustering objects for spatial data mining," *IEEE Transactions on Knowledge & Data Engineering*, 14, 1003–1016.

Ning, H., Xu, W., Chi, Y., Gong, Y., and Huang, T. (2007), "Incremental spectral clustering with application to monitoring of evolving blog communities," in *Proceedings of the 2007 SIAM International Conference on Data Mining*, SIAM, pp. 261–272.

Ntoutsi, E., Spiliopoulou, M., and Theodoridis, Y. (2012), "Fingerprint: Summarizing cluster evolution in dynamic environments," *International Journal of Data Warehousing and Mining (IJDWM)*, 8, 27–44.

Oliveira, M. and Gama, J. (2012), "A framework to monitor clusters evolution applied to economy and finance problems," *Intelligent Data Analysis*, 16, 93–111.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011), "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, 12, 2825–2830.

Recruiter (2019), "Careers and Occupations," `https://www.recruiter.com/careers/`, accessed: 2019-07-5.

Sangam, R. S. and Om, H. (2018), "Equi-Clustream: a framework for clustering time evolving mixed data," *Advances in Data Analysis and Classification*, 12, 973–995.

Sheikholeslami, G., Chatterjee, S., and Zhang, A. (1998), "Wavecluster: A multi-resolution clustering approach for very large spatial databases," in *VLDB*, vol. 98, pp. 428–439.

Spiliopoulou, M., Ntoutsi, I., Theodoridis, Y., and Schult, R. (2006), "Monic: modeling and monitoring cluster transitions," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 706–711.

Strehl, A. and Ghosh, J. (2002), "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *Journal of machine learning research*, 3, 583–617.

Su, Q. and Chen, L. (2015), "A method for discovering clusters of e-commerce interest patterns using click-stream data," *electronic commerce research and applications*, 14, 1–13.

Tantipathananandh, C., Berger-Wolf, T., and Kempe, D. (2007), "A framework for community identification in dynamic social networks," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 717–726.

Wang, W., Yang, J., Muntz, R., et al. (1997), "STING: A statistical information grid approach to spatial data mining," in *VLDB*, vol. 97, pp. 186–195.

Xu, K. S., Kliger, M., and Hero, A. O. (2010), "Evolutionary spectral clustering with adaptive forgetting factor," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, pp. 2174–2177.

Xu, K. S., Kliger, M., and Hero Iii, A. O. (2014), "Adaptive evolutionary clustering," *Data Mining and Knowledge Discovery*, 28, 304–336.

Xu, X., Ester, M., Kriegel, H.-P., and Sander, J. (1998), "A distribution-based clustering algorithm for mining in large spatial databases," in *Proceedings 14th International Conference on Data Engineering*, IEEE, pp. 324–331.

Zhang, T., Ramakrishnan, R., and Livny, M. (1996), "BIRCH: an efficient data clustering method for very large databases," in *ACM Sigmod Record*, ACM, vol. 25, pp. 103–114.

Zhang, Y., Liu, H., and Deng, B. (2013), "Evolutionary clustering with DBSCAN," in *2013 Ninth International Conference on Natural Computation (ICNC)*, IEEE, pp. 923–928.

Ziegler, T., Johanson, D., King, M., Zhang, B., Mammadova, L., Ferri, F., Trappe, R., Suresh, K., Hao, R., Ryll, L., and Yerolemou, N. (2019a), "Reaching New Heights: The 3rd Americas Alternative Finance Industry Report," `https://www.jbs.cam.ac.uk/faculty-research/centres/alternative-finance/publications/reaching-new-heights/#.XamAKPkzY2w`, note = Accessed: 2019-10-18.

Ziegler, T., Shneor, R., Wenzlaff, K., Odorovic, A., Johanson, D., Hao, R., and Ryll, L. (2019b), "Shifting Paradigms: The 4th European Alternative Finance Benchmarking Report," `https://www.jbs.cam.ac.uk/faculty-research/centres/alternative-finance/publications/shifting-paradigms/#.XaltwfkzY2w`, note = Accessed: 2019-10-18.

A Lending Club Data Set

A.1 Feature Description

Table 11: Complete attribute set of Lending Club data set.

Feature	Description
acc_now_delinq	Number of accounts on which the borrower is now delinquent.
acc_open_past_24mths	Number of trades opened in past 24 months.
addr_state	State provided by the borrower in the loan application
all_util	Balance to credit limit on all trades.
annual_inc	Self-reported annual income provided by the borrower.
annual_inc_joint	Combined self-reported annual income provided by the co-borrowers.
application_type	Individual application or a joint application with two co-borrowers.
avg_cur_bal	Average current balance of all accounts.
bc_open_to_buy	Total open to buy on revolving bankcards.
bc_util	Ratio of total current balance to credit limit for all bankcard accounts.
chargeoff_within_12_mths	Number of charge-offs within 12 months.
collection_recovery_fee	Post charge off collection fee.
collections_12_mths_ex_med	Number of collections in 12 months excluding medical collections.
delinq_2yrs	Number of 30+ days past-due incidences of delinquency in the past 2 years.
delinq_amnt	Past-due amount owed for delinquent accounts.
desc	Loan description provided by the borrower.
dti	Debt-to-income ratio.
dti_joint	Joint debt-to-income ratio.
earliest_cr_line	Month the borrower's earliest reported credit line was opened.
emp_length	Employment length in years. Possible values are between 0 and 10.
emp_title	Job title supplied by the borrower.
fico_range_high	Upper boundary of the borrower's FICO.
fico_range_low	Lower boundary of the borrower's FICO.
funded_amnt	Total amount committed to that loan at that point in time.
funded_amnt_inv	Total amount committed by investors at that point in time.
grade	LC assigned loan grade.
home_ownership	Home ownership status. Values are: RENT, OWN, MORTGAGE, OTHER.
id	Unique LC assigned ID for the loan listing.
il_util	Ratio of total current balance to credit limit on all install acct.
initial_list_status	Initial listing status of the loan. Possible values are: W, F.
inq_fi	Number of personal finance inquiries.
inq_last_12m	Number of credit inquiries in past 12 months.
inq_last_6mths	Number of inquiries in past 6 months (excl. auto and mortgage inquiries).
installment	Monthly payment owed by the borrower.
int_rate	Interest Rate on the loan.
issue_d	Date which the loan was funded.
last_credit_pull_d	Most recent month LC pulled credit for this loan.
last_fico_range_high	Upper boundary of the borrower's last FICO.
last_fico_range_low	Lower boundary of the borrower's last FICO.
last_pymnt_amnt	Last total payment amount received.
last_pymnt_d	Last month payment was received.
loan_amnt	Listed amount of the loan applied for by the borrower.
loan_status	Current status of the loan.
max_bal_bc	Maximum current balance owed on all revolving accounts.

member_id	Unique LC assigned Id for the borrower member.
mo_sin_old_il_acct	Months since oldest bank installment account opened.
mo_sin_old_rev_tl_op	Months since oldest revolving account opened.
mo_sin_rcnt_rev_tl_op	Months since most recent revolving account opened.
mo_sin_rcnt_tl	Months since most recent account opened.
mort_acc	Number of mortgage accounts.
mths_since_last_delinq	Months since the borrower's last delinquency.
mths_since_last_major_derog	Months since most recent 90-day or worse rating.
mths_since_last_record	The number of months since the last public record.
mths_since_rcnt_il	Months since most recent installment accounts opened.
mths_since_recent_bc	Months since most recent bankcard account opened.
mths_since_recent_bc_dlq	Months since most recent bankcard delinquency.
mths_since_recent_inq	Months since most recent inquiry.
mths_since_recent_revol_delinq	Months since most recent revolving delinquency.
next_pymnt_d	Next scheduled payment date.
num_accts_ever_120_pd	Number of accounts ever 120 or more days past due.
num_actv_bc_tl	Number of currently active bankcard accounts.
num_actv_rev_tl	Number of currently active revolving trades.
num_bc_sats	Number of satisfactory bankcard accounts.
num_bc_tl	Number of bankcard accounts.
num_il_tl	Number of installment accounts.
num_op_rev_tl	Number of open revolving accounts.
num_rev_accts	Number of revolving accounts.
num_rev_tl_bal_gt_0	Number of revolving trades with balance >0.
num_sats	Number of satisfactory accounts.
num_tl_120dpd_2m	Number of accounts currently 120 days past due.
num_tl_30dpd	Number of accounts currently 30 days past due.
num_tl_90g_dpd_24m	Number of accounts 90 or more past due in last 24 months.
num_tl_op_past_12m	Number of accounts opened in past 12 months.
open_acc	Number of open credit lines in the borrower's credit file.
open_acc_6m	Number of open trades in last 6 months.
open_il_12m	Number of installment accounts opened in past 12 months.
open_il_24m	Number of installment accounts opened in past 24 months.
open_act_il	Number of currently active installment trades.
open_rv_12m	Number of revolving trades opened in past 12 months.
open_rv_24m	Number of revolving trades opened in past 24 months.
out_prncp	Remaining outstanding principal for total amount funded.
out_prncp_inv	*out_prncp* funded by investors.
pct_tl_nvr_dlq	Percent of trades never delinquent.
percent_bc_gt_75	Percentage of all bankcard accounts >75% of limit.
pub_rec	Number of derogatory public records.
pub_rec_bankruptcies	Number of public record bankruptcies.
purpose	Purpose description for the loan request.
pymnt_plan	Indicates if a payment plan has been put in place for the loan.
recoveries	Post charge off gross recovery.
revol_bal	Total credit revolving balance.
revol_util	Revolving line utilization rate.
sub_grade	LC assigned loan subgrade.
tax_liens	Number of tax liens.
term	Number of payments on the loan. Values are: 36 or 60 months.
title	Loan title provided by the borrower.
tot_coll_amt	Total collection amounts ever owed.
tot_cur_bal	Total current balance of all accounts.
tot_hi_cred_lim	Total high credit/credit limit.

total_acc	Total number of credit lines in borrower's credit file.
total_bal_ex_mort	Total credit balance excluding mortgage.
total_bal_il	Total current balance of all installment accounts.
total_bc_limit	Total bankcard high credit/credit limit.
total_cu_tl	Number of finance trades.
total_il_high_credit_limit	Total installment high credit/credit limit.
total_pymnt	Payments received to date for total amount funded.
total_pymnt_inv	*total_pymnt* funded by investors.
total_rec_int	Interest received to date.
total_rec_late_fee	Late fees received to date.
total_rec_prncp	Principal received to date.
total_rev_hi_lim	Total revolving high credit/credit limit.
url	URL for the LC page with listing data.
verification_status	Indicates if income was verified by LC.
verified_status_joint	Indicates if the co-borrowers' joint income was verified by LC.
zip_code	First 3 numbers of the zip code provided by the borrower.
revol_bal_joint	Sum of revolving credit balance of the co-borrowers.
sec_app_fico_range_low	FICO range (high) for the sec. applicant.
sec_app_fico_range_high	FICO range (low) for the sec. applicant.
sec_app_earliest_cr_line	Earliest credit line for the sec. applicant.
sec_app_inq_last_6mths	Credit inquiries in the last 6 months for the sec. applicant.
sec_app_mort_acc	Number of mortgage accounts for the sec. applicant.
sec_app_open_acc	Number of open trades for sec. applicant.
sec_app_revol_util	*revol_util* for sec. applicant.
sec_app_open_act_il	No. currently active installment trades for the sec. applicant.
sec_app_num_rev_accts	No. revolving accounts for the sec.y applicant.
sec_app_chargeoff_within_12_mths	No. charge-offs within last 12 months for the sec. applicant.
sec_app_collections_12_mths_ex_med	No. collections within last 12 months for the sec. applicant.
sec_app_mths_since_last_major_derog	Months since major derog. for the sec. applicant.
hardship_flag	Flags whether or not the borrower is on a hardship plan.
hardship_type	Describes the hardship plan offering.
hardship_reason	Describes the reason the hardship plan was offered.
hardship_status	Describes hardship plan status.
deferral_term	Months with smaller payments due to a hardship plan.
hardship_amount	Interest payment due to hardship plan.
hardship_start_date	Start date of the hardship plan period.
hardship_end_date	End date of the hardship plan period.
payment_plan_start_date	Day the first hardship plan payment is due.
hardship_length	Length of hardship length
hardship_dpd	Account days past due as of the hardship plan start date.
hardship_loan_status	Loan Status as of the hardship plan start date.
orig_projected_additional_accrued_interest	Original additional interest accrued.
hardship_payoff_balance_amount	Payoff balance amount as of the hardship plan start date.
hardship_last_payment_amount	Last payment amount as of the hardship plan start date.
disbursement_method	Method by which the borrower receives their loan.
debt_settlement_flag	Flags if borrower is working with debt-settlement company.
debt_settlement_flag_date	Most recent date that the *debt_settlement_flag* has been set.
settlement_status	Status of the borrower's settlement plan.
settlement_date	Date that the borrower agrees to the settlement plan.
settlement_amount	Loan amount that the borrower has agreed to settle for.
settlement_percentage	Percentage of the loan settled for.
settlement_term	Months that the borrower will be on the settlement plan.

A.2 Outlier Analysis

A.2.1 Normal Probability Plots: QQ-plots

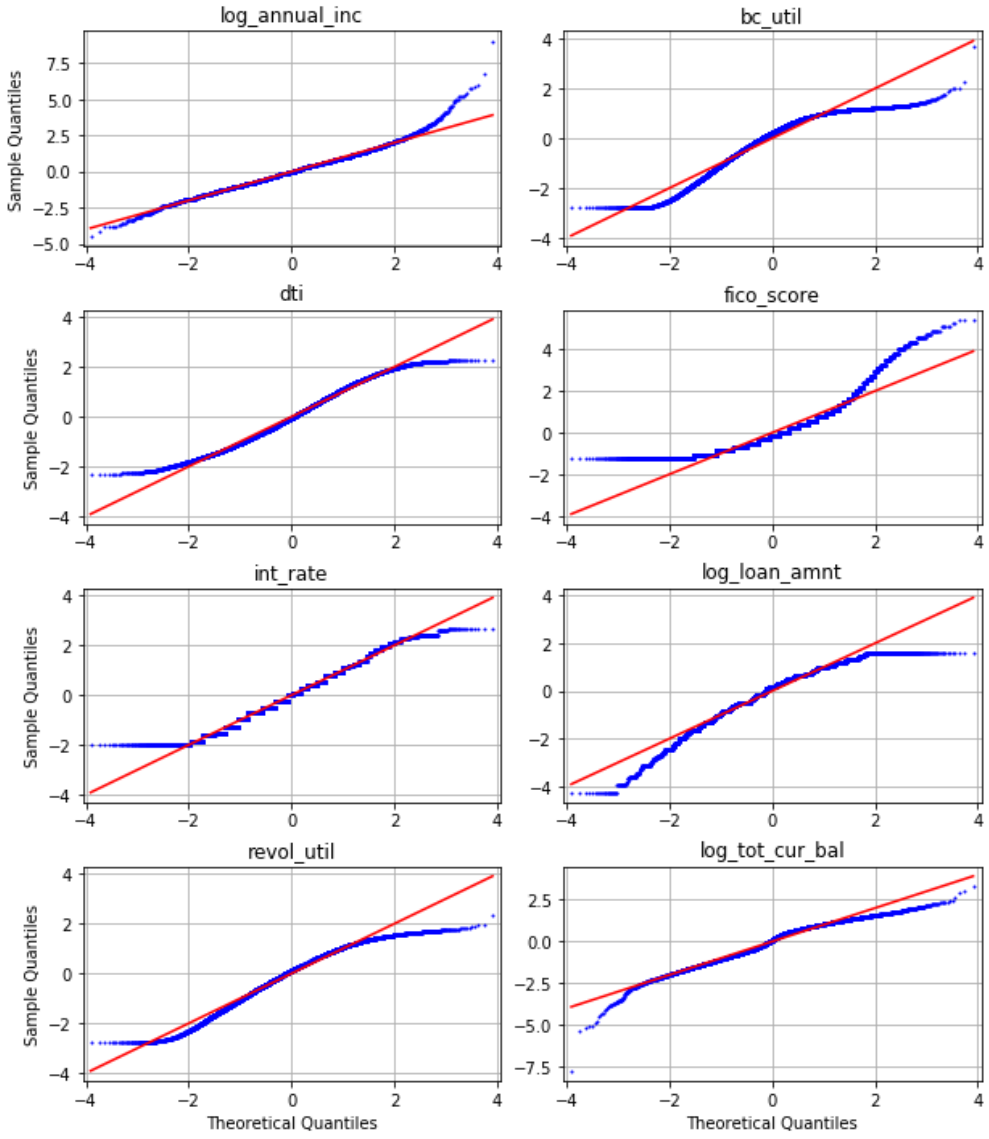

Figure 10: Normal Probability plots of metric features of 2013Q1 data, displaying the theoretical quantiles of a standard normal distribution against the quantiles of the sample distributions.

A.2.2 Outlier Labeling

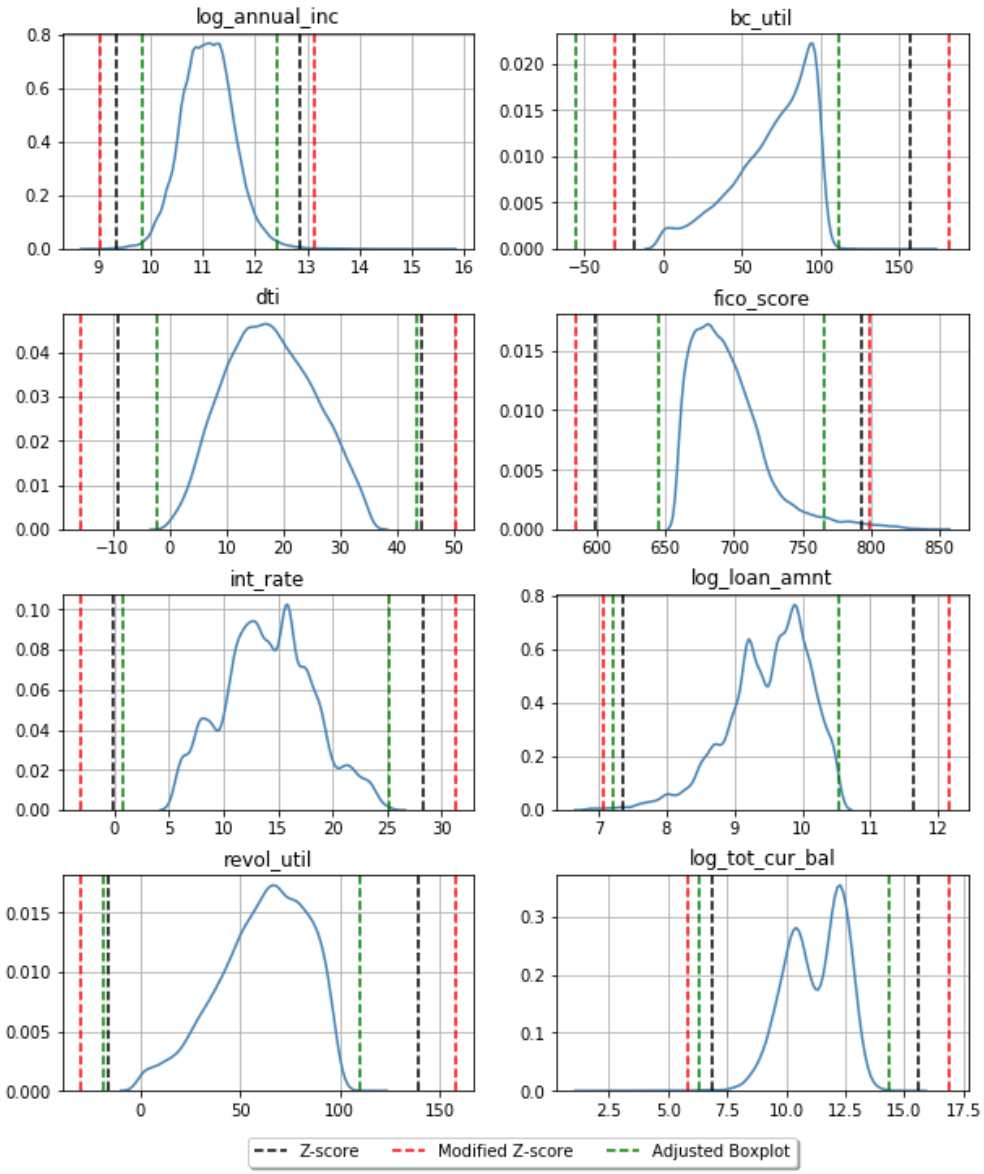

Figure 11: Feature distribution of 2013Q1 including upper and lower bound for outlier detection. Methods shown are the Z-score and Modified Z-score tests and Adjusted boxplot.

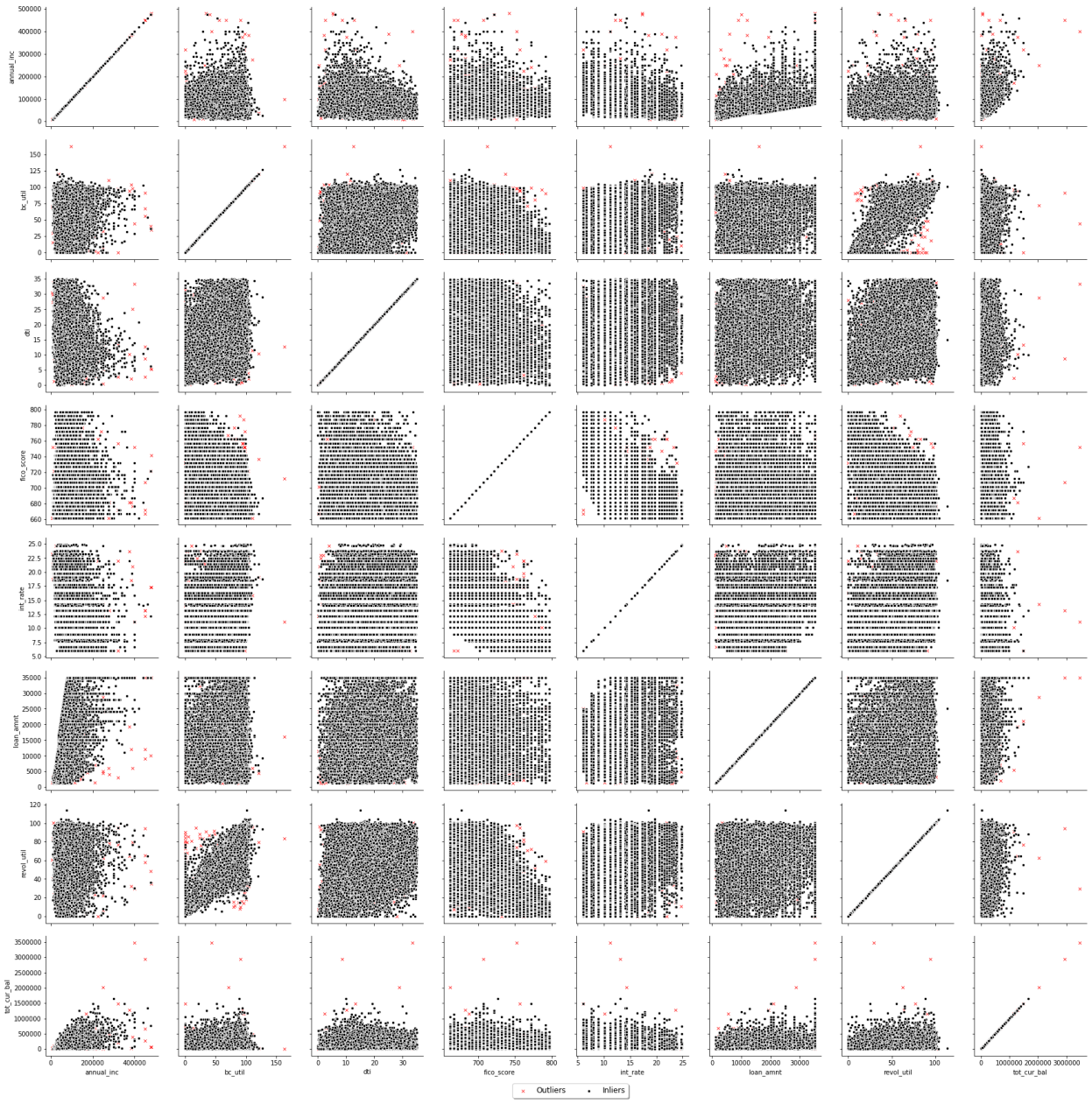

Figure 12: Multivariate outlier detection using Local-Outlier Factor scores. In- and outliers are highlighted on the scatter plots of 2013Q1.

A.3 Exploratory Data Analysis

In this section we elaborate on the exploratory data analysis performed to initial insights in the Lending Club loan data set. First investigate core features separately and then look for relationships between features.

A.3.1 Univariate Analysis

We start with *loan_amnt*, which is arguably the most important feature in the data set. It describes the listed amount of the loan applied for by the borrower. Figure 13 describes the distribution of loans on entire dataset, as well as quarterly and total loan size statistics.

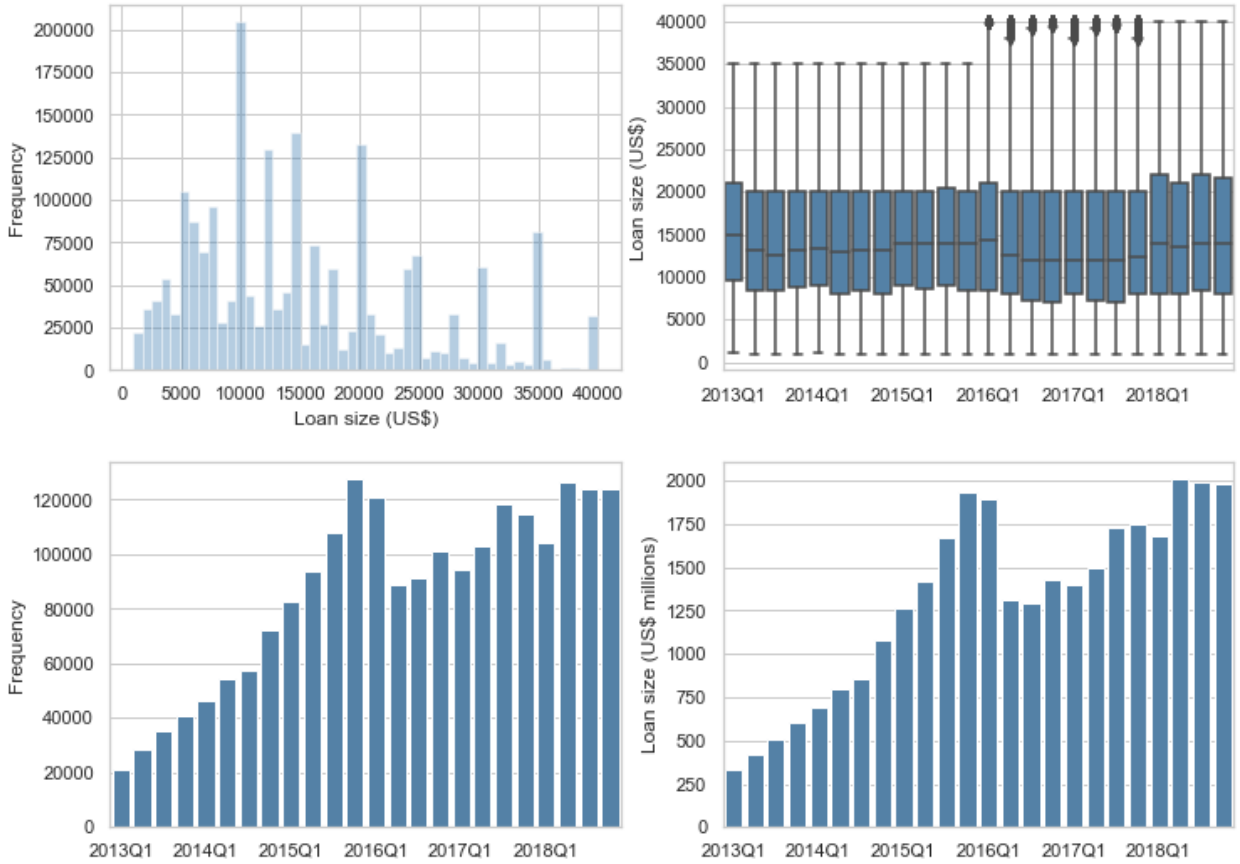

Figure 13: Distribution and statistics regarding the size of loans. Top left displays the distribution of loans in the complete data set, top right gives quarterly boxplots, bottom left shows the total number of loans per quarter and bottom right gives the total value of these loans.

We observe large peaks at every US $5000 as these are typical loan sizes for consumers. Furthermore, we see that the average loan size does fluctuate over time but does not differ significantly. We also see that only in the last 3 years loans of over US $35000 were issued. This is due to the fact that Lending Club raised their maximum loan size to US $40000. However this raise is not directly reflected into the average loan size, which remained relatively constant over time.

Regarding the total number of loans issued per quarter, we observe a steep growth of the Lending Club platform from 2013 to end of 2015. Then in the beginning of 2016, we see a drop off in loans issued which slowly recovers towards the previous level. As the average loan size does not change all that much over the years, the total sum of the loans per quarter follows the same trend as the number of loans in each quarter. The significant drop in loans in the beginning of 2016 followed due to a public scandal the company endured late 2015, resulting in difficulty of attracting investors.

Interest Rate

Next to loan size, it interesting to look at the interest rate (*int_rate*) its' development over the years. Figure 14 displays the distribution of the interest rate aggregated over the complete dataset. We see that the bulk of the loans receive an interest rate between 7 and 15% and a gradual drop off for rates above. With respect to quarterly rate, we discover a steady decrease up until the beginning of 2016 after which we observe a significant increase. The reason for this rise is most likely due to worse lending circumstances after the company scandal mentioned before.

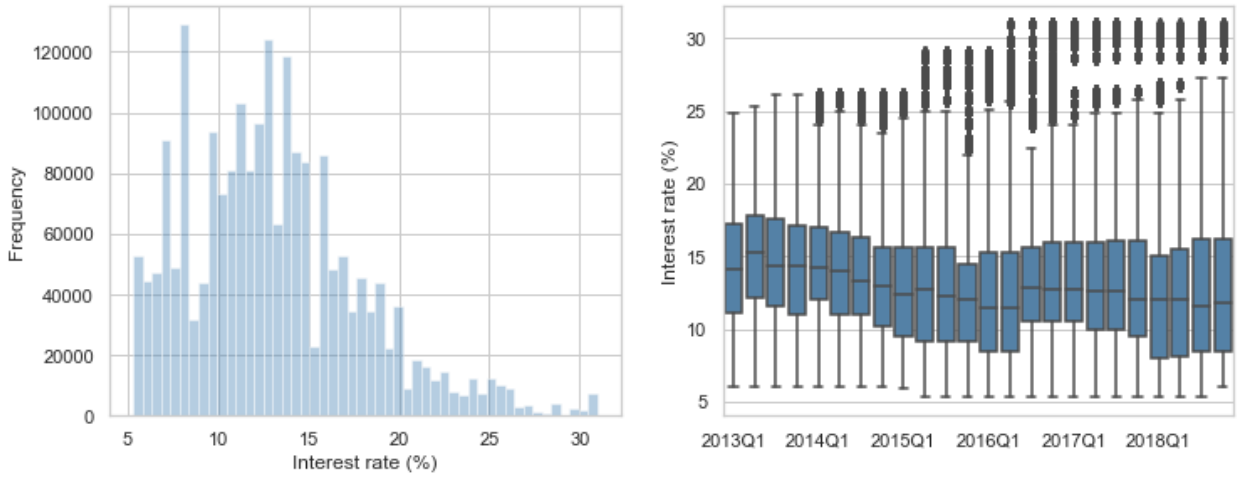

Figure 14: Distribution of interest rate on all loans in the complete data set (left) and boxplots of quarterly loan data (right).

Annual Income

Another important feature in this dataset is *annual_inc*, which describes the annual income of the borrower. We find the majority of the loans between US$40,000 and US$100,000 which is in line with the average income in the United States. As the platform grew over the years, we observe a slight increase in average income. However, it is more interesting to investigate relationships with other features such as *loan_amnt* and *int_rate*, as we do in Appendix A.3.2.

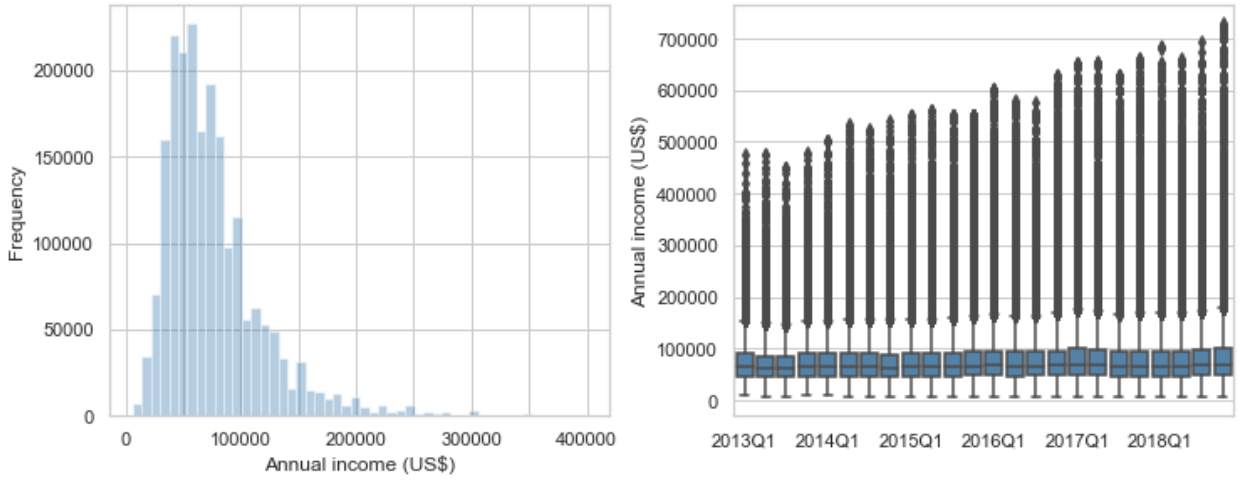

Figure 15: Distribution of borrower's annual income of the complete data set (left) and boxplots of quarterly loan data (right).

Type of Employment

The type of career the borrower is working in is described by *emp_title*. The distribution of the this feature is dominated by one career field called Business, Management and Administration (BMA), taking up approximately 30% of the observations. It is difficult to derive further insights on this feature as BMA remains to be the most dominant category over time and the distribution remains relatively the same. Furthermore, looking at other features conditional on one of the career categories does not reveal extra information. For example, the annual income given a type of career does not change noticeably over time and between the career categories we do not observe variation as in general all levels of income are present within each career and are approximately distributed the same.

Loan Purpose

Similar conclusions can be drawn on the feature *purpose*, which summarizes the reason for applying the loan. The main purpose for getting a loan is dominated by debt consolidation and credit card refinancing. The other categories are small in comparison and no significant changes in the distribution of *purpose* are observed over time. As *purpose* is dominated by a single category, it is most likely not very informative for cluster analysis.

Home Ownership

The feature *home_ownership* defines the type of home ownership status provided by the borrower. The three categories we distinguished and their frequency are shown in Figure 16. We observe that only a relatively small portion of the borrowers own a house and the majority is renting a home or still paying off a mortgage. Like the previously discussed features, its' distribution does not change significantly over time.

Length of Employment

The length of employment of the borrower is described by the feature *emp_length*. We may expect borrowers with different job experience to differ in other features such as annual income, and ultimately possess different lending habits. We could then possibly discover

clusters that are dominated by one type of employment length, such as junior or senior employees. Regarding its distribution in Figure 16, we observe quite an even spread of the categories. Furthermore, this distribution remains approximately constant over time and we look for more insights in the multivariate analysis.

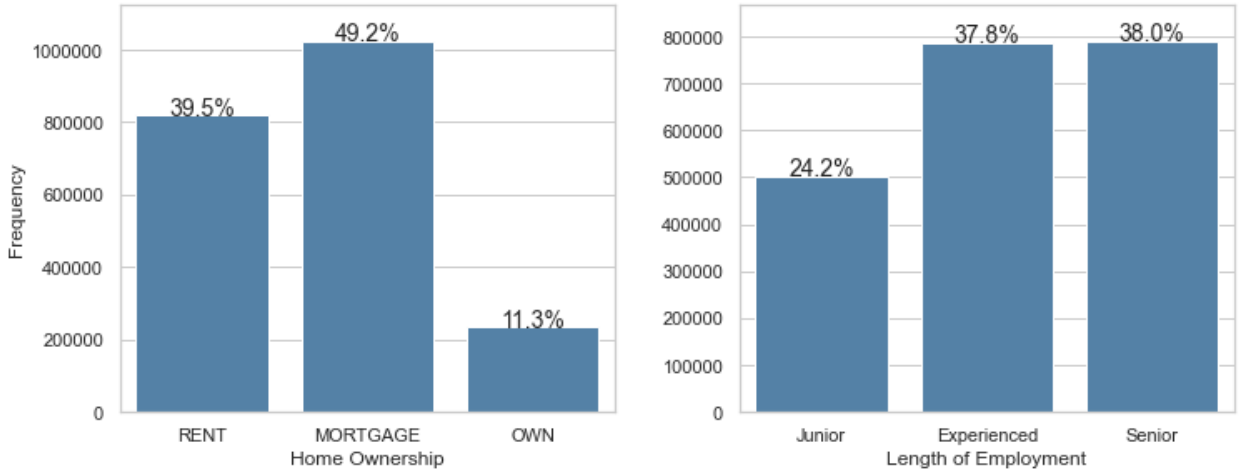

Figure 16: Distribution of categorical feature *home_ownership* (left) and *emp_length* (right).

Debt-to-Income

Debt-to-income (*dti*) is a ratio calculated using the borrower's total monthly debt payments on their total debt obligations, excluding mortgage and the requested loan, divided by the borrower's monthly income. This ratio is an indication to lenders of the ability of a borrower to repay the money being borrowed. The distribution of *dti* is shown in Figure 17 where we see that almost all mass is found between 0 and 40%. This is expected as the 43% debt-to-income ratio is seen as an important threshold. It is usually the highest *dti* a borrower can have and still get a qualified mortgage. As such most people try stay below it. Looking at the trend over time, we observe that from 2016 onwards borrowers with high *dti* ratios, above the 43% threshold, were able to apply for a loan. Interestingly, the average interest rate for loans with these high *dti* ratios does not significantly deviate from the average interest rate of all loans. This is unexpected, as one would think that a high debt-to-income ratio results in higher interest rates. Similarly regarding the annual income, we do not observe deviation from the overall average for these loans.

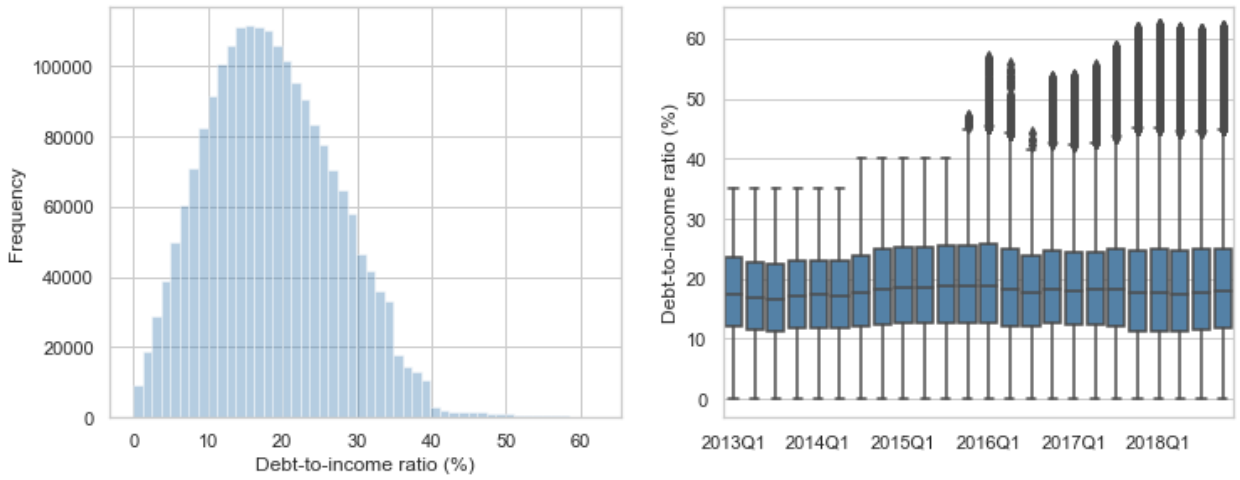

Figure 17: Borrower's debt-to-income ratio with the distribution of the complete data set (left) and boxplots of quarterly data (right).

Revolving Credit and Bankcard Utilization Rate

Revolving line utilization rate (*rev_util*) illustrates the amount of credit the borrower is using relative to all available revolving credit whereas *bc_util* describes the utilization rate of credit on bankcard accounts. We see that the revolving line rate is approximately normally distributed between 0 and 100% with a spike at 0%, which refers to the borrowers not using any revolving credit. Regarding its development over time, we observe a gradual decrease in average utilization rate, which may be an indication that borrowers are becoming less dependent on their credit. This observation is also in line the gradual decrease in interest rate we observed, as customers who use less credit would more favorable interest rates on their loan.

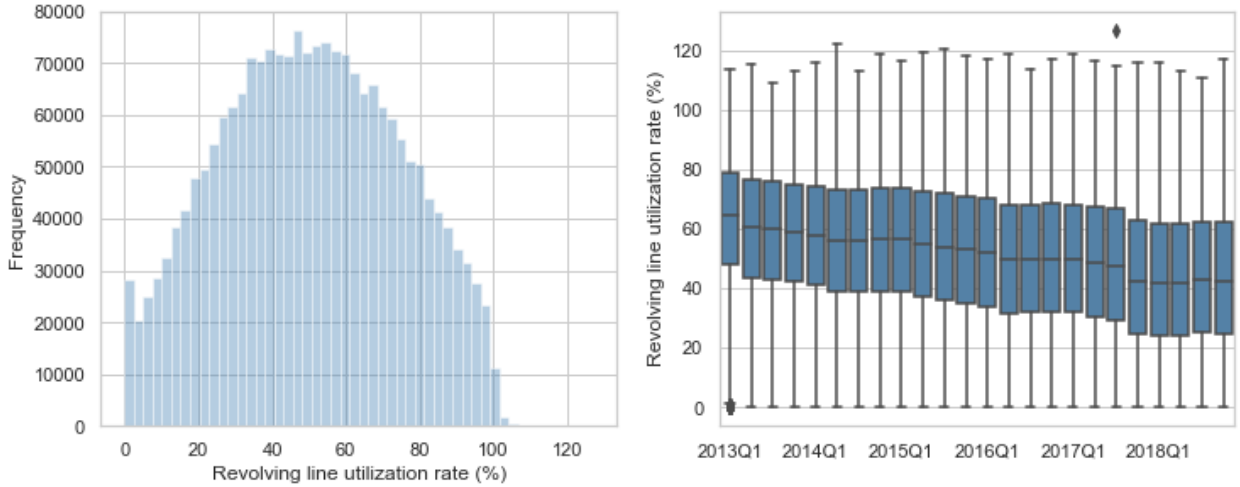

Figure 18: Distribution of revolving credit line utilization rate (left) and development over time through boxplot visualization (right).

For the bankcard utilization rate, we observe a large peaks at 0% and around 100% where

in between we see a s-curve like behavior. This way we could distinguish borrowers that do not use their bankcard credit, those use almost all of it and borrowers that are in the middle ground. As with *rev_util*, we observe a gradual decrease in the average bankcard usage of credit over time possibly signifying the reduced dependence on credit.

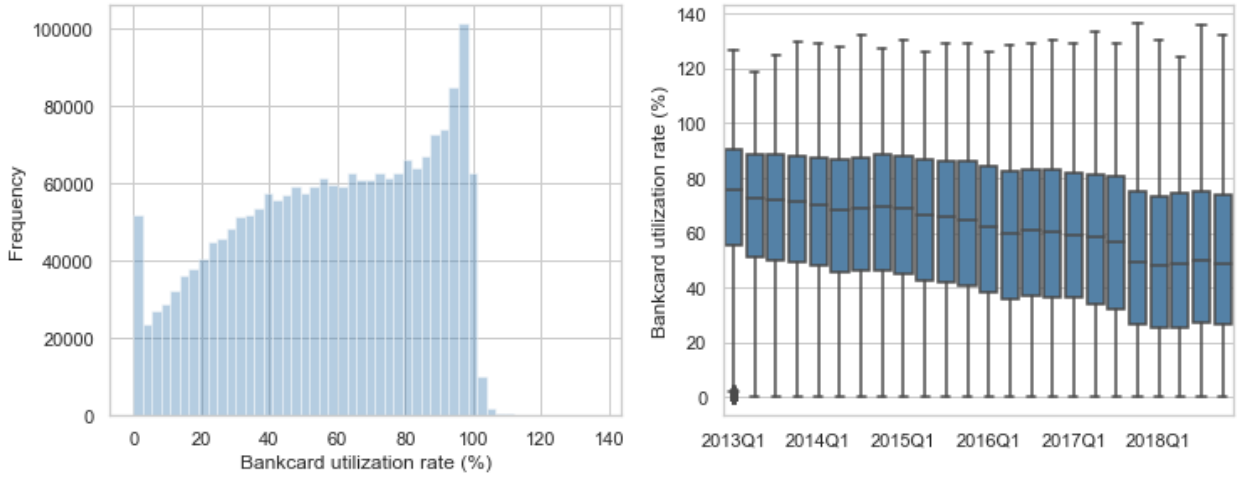

Figure 19: Distribution of bankcard utilization rate (left) and development over time through boxplot visualization (right).

FICO Credit Score

A FICO score is a particular brand of credit score. This score is used to predict how likely a borrower is to pay back a loan on time and is used in determining the interest rate of a loan. FICO scores take into account various factors in five areas to determine a customers' creditworthiness: payment history, current level of debt, types of credit used, length of credit history, and new credit accounts. In general, scores above 650 indicate a very good credit history. In contrast, individuals with scores below 620 often find it difficult to obtain financing at favorable rates. Therefore we do not observe such credit scores in the dataset, it only contains borrowers with scores above 662 . It is also interesting to note the increase in average score in Figure 20, which is in line with the observation made before regarding the improved financial situation of customers.

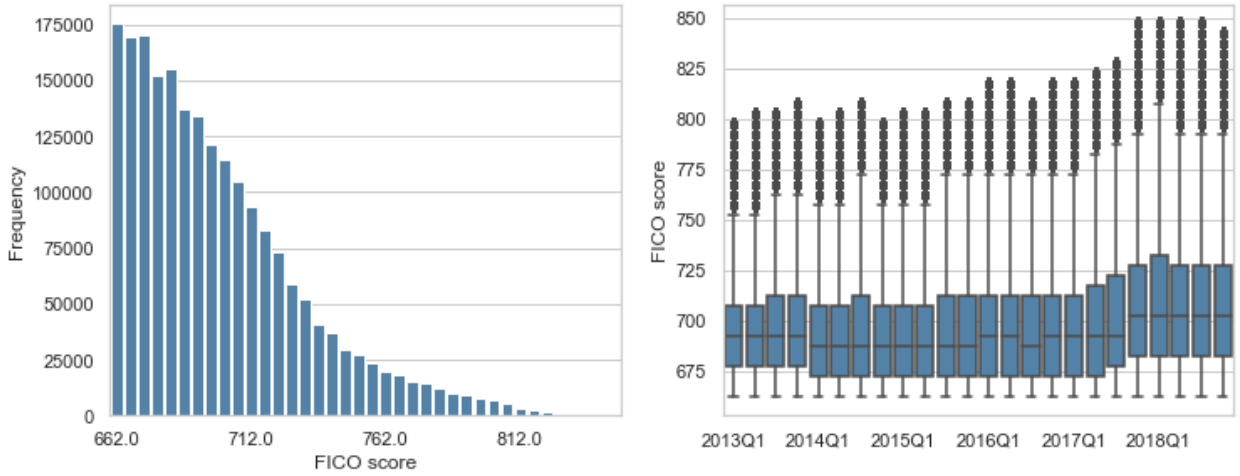

Figure 20: Distribution of FICO scores (left) and development over time through boxplot visualization (right).

Lending Club Loan Grade

Finally we will look at *grade*, which is a feature measured by Lending Club assessing the condition of the loan. In Figure 21 we show the distribution of the complete dataset. We only see a small portion of the loans in the lower grades (E, F and G) which may be due to the fact that loan applications with such grades are less likely to attract investors. *grade* itself does not reveal much information but by checking relationships with other variables we may detect interesting behavior which is what we do next in Appendix A.3.2.

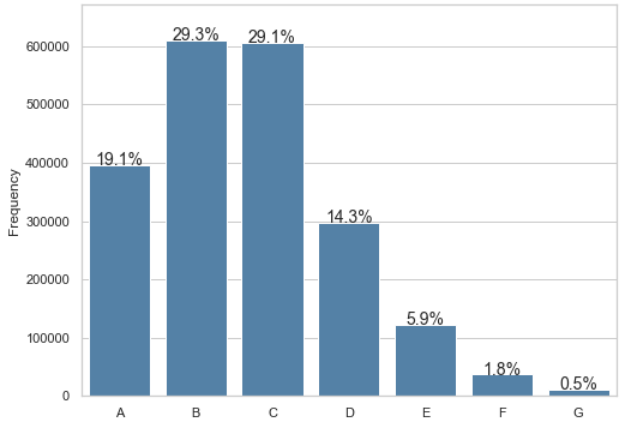

Figure 21: Distribution of *grade* based on the complete dataset.

A.3.2 Multivariate Analysis

Whereas the univariate analysis allows us to understand the distribution of single features and their behavior over the six years of data, the multivariate analysis investigates possible relationships between multiple features. Here, we go over some of the core features we mentioned before and present relationships and correlations to can gain insights on the data.

Lending Club Loan Grade

Starting with *grade*, we plot violin- and boxplots of the loan grades against other core features in Figure 22. Here we find expected behavior, people that receive a lower grade spend more available credit, possess lower FICO score and get higher interest rates. We also see that borrowers with a lower grade apply for larger loans, which may be because these type of borrowers are paying of other debt or refinancing their credit card more often. Regarding the annual income, we see that from top to mid grades the annual income decreases on average but increases from mid to low grades. This is possibly because borrowers may require a higher income for obtaining the bigger loans in this segment. Regarding the remaining features in Figure 22, we observe expected behavior as a higher loan grade is most likely obtained because of a strong financial situation of the borrower which results in better loan conditions.

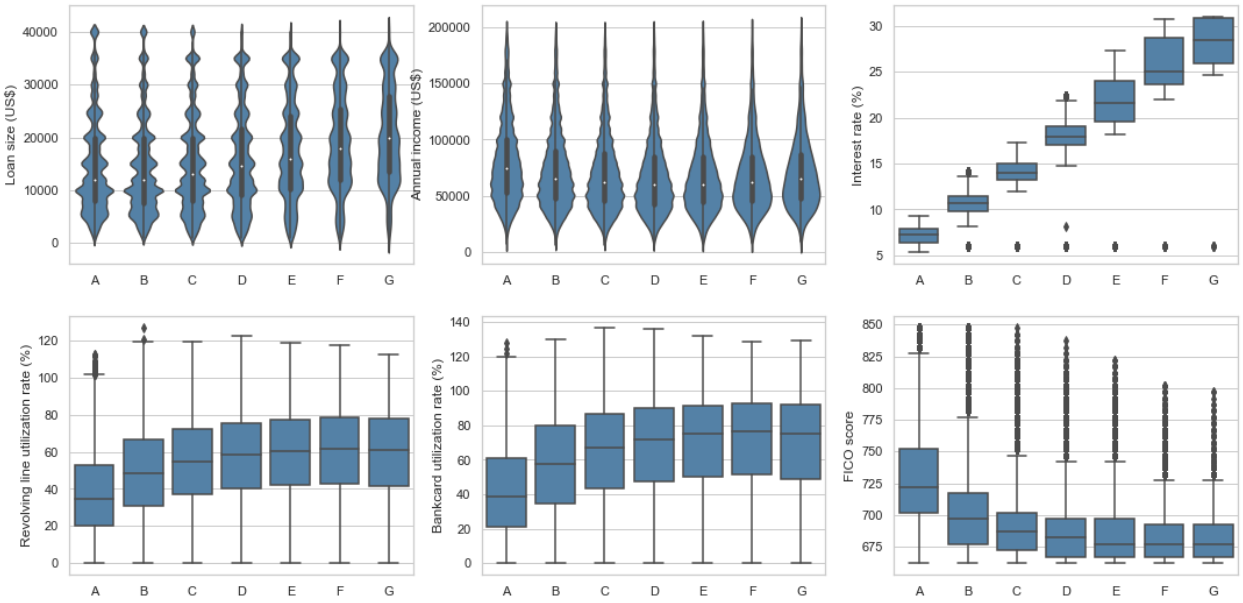

Figure 22: Categories of *grade* plotted against other core features

Income Level

To investigate the annual income, we separated the data in three levels; low, medium and high income. Low income describes an annual income up to US$25,000, medium from US$25,000 to US$100,000 and high for incomes over US$100,000. We generally observe expected behavior in Figure 23, low income levels generally apply for smaller loans and get on average higher interest rates whereas high income levels predominantly apply for larger loans and get on average the lowest interest rate. Borrowers in the high income level have on average a lower debt-to-income ratio but spend more of their available revolving credit. Regarding the FICO credit score, we observe higher values for higher income levels, which is expected.

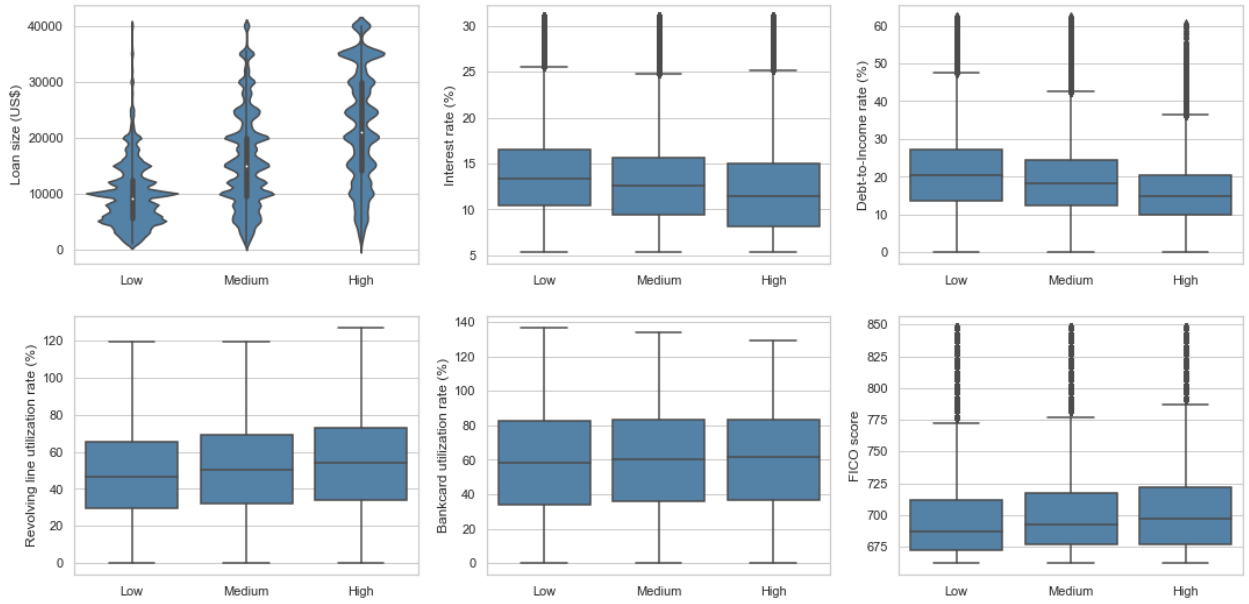

Figure 23: Categories of income level plotted against other core features

Loan term length

Looking at the term length in Figure 24, we see that loans payed off within three years are generally smaller than those payed off in five years. The loans with a three year term length also receive a lower interest rate on average, as these may involve less risk.

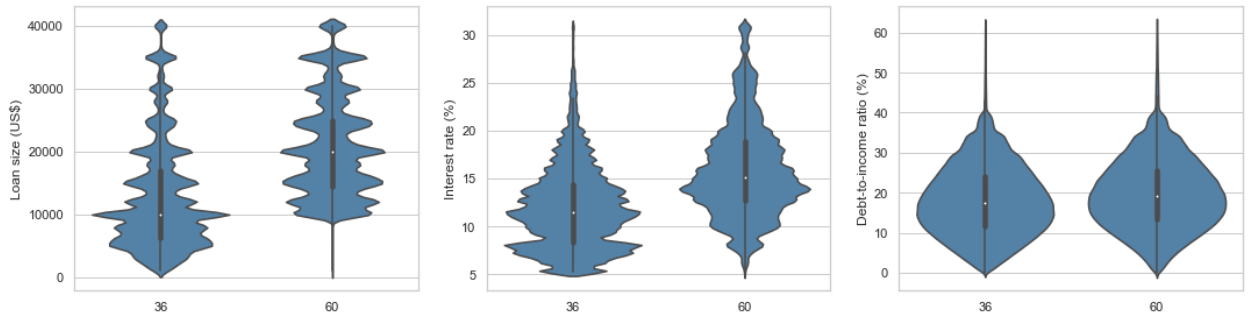

Figure 24: Influence of term length on (a) loan size and (b) interest rate

Correlation of Core Features

Investigating the correlations between core features gives a further look into the relationships between variables and may serve as a basis for feature selection. When features are highly correlated, some of the features may receive a higher weight in the cluster analysis. This is because highly correlated features effectively represent the same idea or concept, which is now represented twice and gets twice the weight compared to other features. This will most likely skew the final clustering in the direction of that concept, which can be an issue if not done intentionally. On the other hand, if we can find suitable correlated features that suggest certain cluster distribution resulting in interpretable and straightforward customer

segments, we actually want to include these features. Therefore, we present a correlation heatmap in Figure 25 of core features to discover cases of (multi-)collinearity.

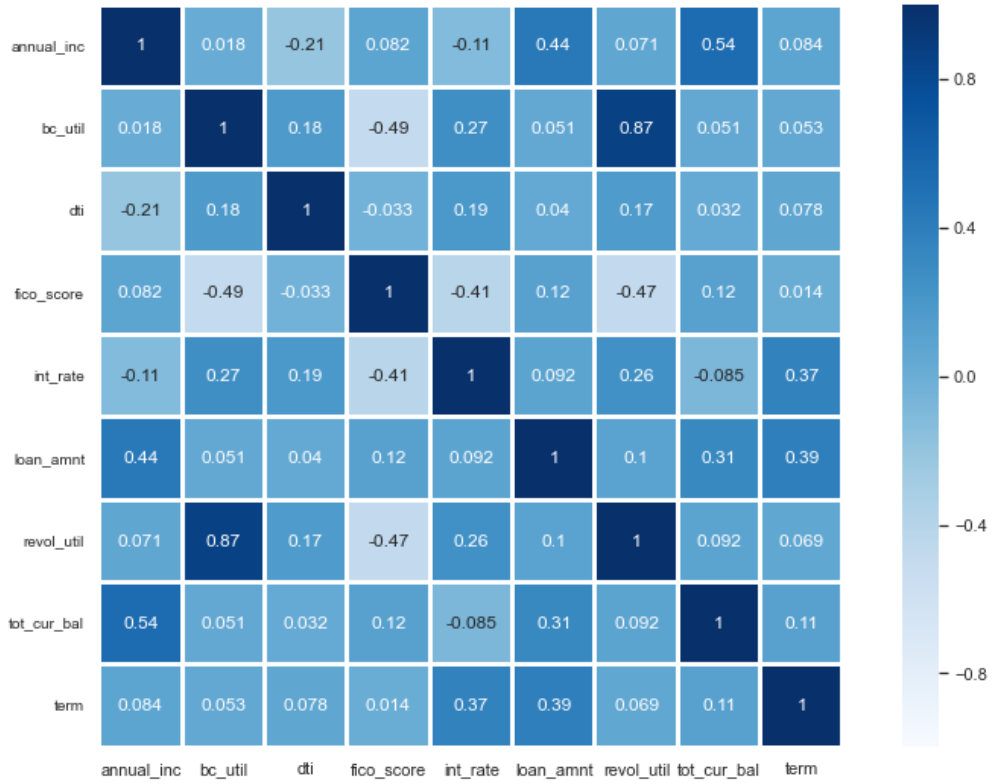

Figure 25: Correlation heatmap of core features

With the features *annual_inc*, *loan_amnt* and *tot_cur_bal*, we observe significant positive correlation. Borrowers with a higher income tend to have a higher balance and apply for bigger loans which confirms the observation made previously. *bc_util* and *revol_util* are highly correlated, together describing the credit usage behavior of borrowers. Furthermore, these two features have the largest negative influence on $fico\_score$ which is reasonable as it an indicator of financial stability. The Lending Club provided grade seems to be moderately influenced by $fico\_score$. Higher FICO scores result in a higher grade. In addition, *grade* almost perfectly (negatively) correlates with *int_rate* which we already understood from the paired plots between these features. Borrowers receiving a higher grade will get a better loan offer in terms of interest rate. Besides *grade*, $fico\_score$ also negatively influences the interest rate as better grades and scores are the result of a better financial situation. Regarding *term*, we observe a positive correlation with $loan_amnt$. Larger loans are generally paid off in 5 instead of 3 years. These larger loans will also often receive a worse interest rate, portrayed by the positive correlation between *term* and *int_rate*.

B Static Clustering Extra Results

B.1 Clustering Algorithms

Algorithm 1: K-means/K-prototypes Clustering

Initialization;
$\mu_1, ..., \mu_k \leftarrow$ Select k points as initial cluster centers;
while *Stopping criteria not met* **do**
    $c_1, ..., c_n \leftarrow$ Assign each point to its closest centroid;
    $\mu_1, ..., \mu_k \leftarrow$ Recompute the centroid of each cluster;
end

Algorithm 2: Agglomerative Clustering

Initialization;
Compute the dissimilarity matrix between all observations, given a set proximity measure;
while *Number of clusters > 1* **do**
    $C_{x \cup y} \leftarrow C_x \cup C_y$ Merge the closest pair of clusters;
    $N_{x \cup y} \leftarrow N_x + N_y$ Update cardinality of new cluster;
    Update dissimilarity matrix with a new row and column for $C_{x \cup y}$;
    Remove the old rows and columns of C_x and C_y;
end

Algorithm 3: DBSCAN

Initialization;
Set cluster counter C to 0;
for *each obs.* x_i *in* S_t **do**
    if *Label*(x_i) *is undefined* **then**
        Find neighbors N_i of x_i;
        if $|N_i| < MinPts$ **then**
            Label(x_i) $\leftarrow$ Noise;
        end
        $C \leftarrow C + 1$;
        for *each obs* x_j *in* N_i **do**
            Label(x_j) $\leftarrow C$ if Label(x_j) = Noise;
            Label(x_j) $\leftarrow C$ if Label(x_j) $\neq$ undefined;
            Find neighbors N_j of x_j;
            if $|N_j| > MinPts$ **then**
                $N_i \leftarrow N_i \cup N_j$;
            end
        end
    end
end

Algorithm 4: EM for Gaussian Mixture Model

Initialization;
μ_k^0, Σ_k^0, $\pi_k^0 \leftarrow$ Set initial parameters;
while *Convergence criterion not met* **do**
    E-step:
    $\gamma(z_{nk}) \leftarrow$ Calculate the responsibilities using Equation (6);
    M-step:
    μ_k, Σ_k, $\pi_k \leftarrow$ Update the parameters using the Equations (5) and (7);
end

B.2 Selecting Number of Segments

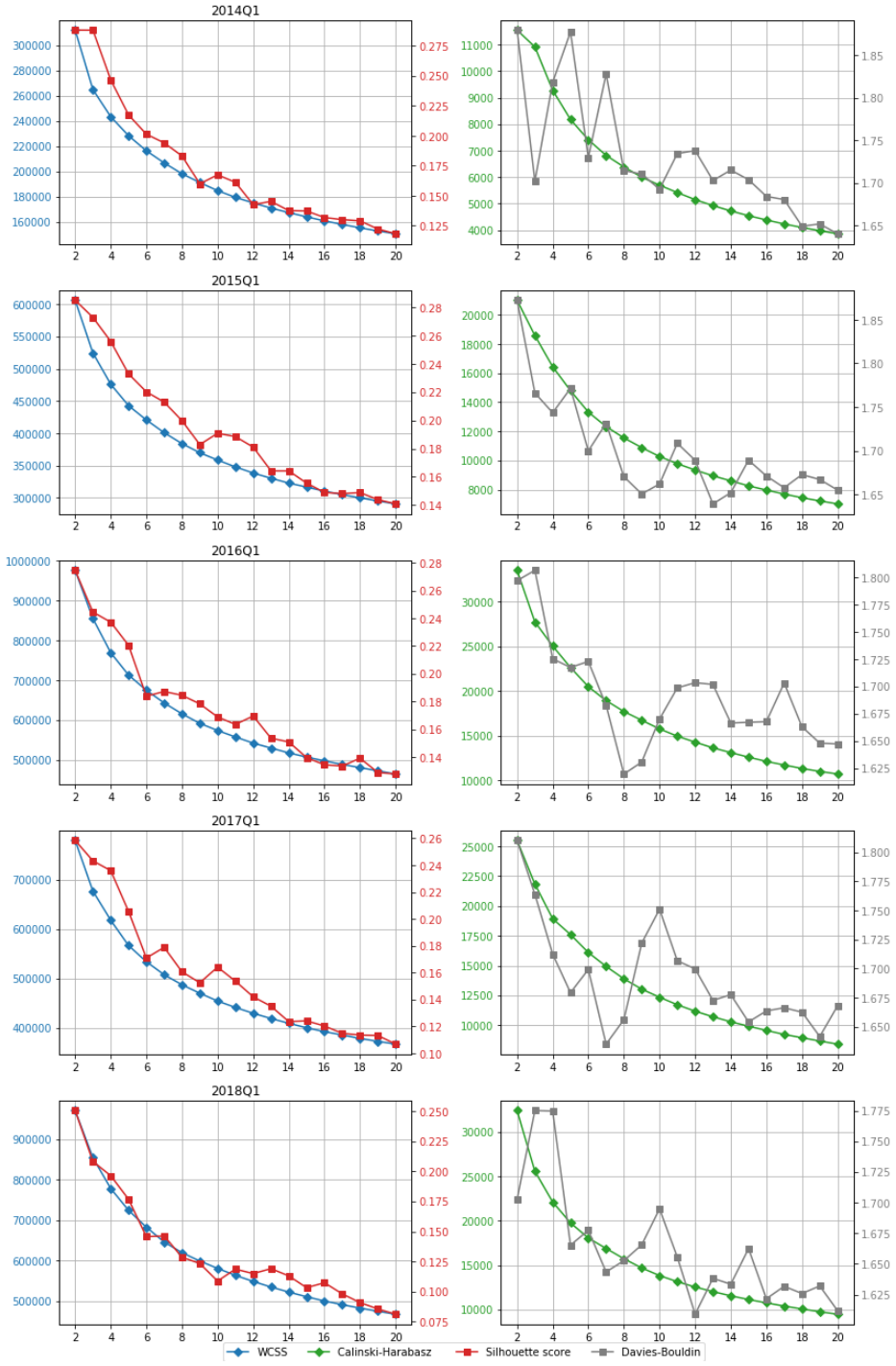

Figure 26: Elbow plot for WCSS, Silhouette scores, CH and DB indices scores used for selecting the number of clusters. Results shown using K-means clustering for the data of 2014-2018 Q1.

Table 12: Cluster statistics for K-means with 6 clusters on data of 2013-2015 Q1. Mean and standard deviation are given for numerical features and the mode for categorical features. *annual_inc*, *loan_amnt* and *tot_cur_bal* are given in US$1000.

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**	**Cluster 6**	**Overall**
K-means 2013Q1	17.2%	18.7%	14.7%	17.6%	10.3%	21.5%	21191
annual_inc	47.6 (19.7)	52.3 (15.6)	55.7 (21.7)	85.1 (33.6)	93.9 (44.5)	108.7 (45.3)	74.2 (40.0)
bc_util	84.75 (12.46)	79.67 (14.89)	43.3 (17.06)	85.52 (13.29)	30.39 (16.58)	75.59 (15.02)	70.26 (24.28)
dti	14.98 (6.25)	26.35 (4.59)	14.61 (6.81)	19.55 (6.27)	14.61 (7.35)	14.89 (5.91)	17.8 (7.58)
fico_score	678.38 (13.58)	691.4 (17.2)	696.58 (19.81)	681.64 (14.93)	742.98 (25.57)	698.54 (18.27)	695.08 (25.41)
int_rate	16.63 (2.7)	14.29 (2.54)	14.28 (3.01)	18.2 (2.94)	9.18 (2.97)	11.0 (2.53)	14.14 (4.03)
loan_amnt	8.6 (4.2)	13.2 (5.4)	11.4 (6.0)	23.9 (6.9)	17.8 (7.6)	18.7 (7.0)	15.7 (8.1)
revol_util	72.04 (14.74)	69.31 (14.71)	38.92 (13.04)	76.61 (13.97)	29.14 (13.97)	67.58 (14.51)	62.08 (21.54)
tot_cur_bal	36.7 (48.3)	109.0 (89.0)	59.8 (70.3)	163.5 (137.4)	214.3 (170.9)	261.2 (177.3)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Senior	Experienced	Senior	Experienced
grade	C	B	B	C	A	B	B
home_ownership	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	36	36	36
K-means 2014Q1	19.1%	13.7%	16.6%	17.4%	12.3%	20.9%	46110
annual_inc	47.4 (18.7)	54.0 (25.4)	57.4 (18.1)	77.7 (28.9)	87.0 (42.2)	116.4 (54.0)	74.5 (42.6)
bc_util	83.51 (12.27)	39.62 (18.26)	60.45 (17.93)	87.02 (12.07)	32.61 (16.9)	73.43 (16.84)	65.91 (25.14)
dti	16.88 (6.83)	12.5 (6.14)	24.78 (5.06)	21.45 (6.34)	15.61 (6.77)	13.54 (5.42)	17.54 (7.47)
fico_score	676.38 (14.05)	686.72 (20.1)	695.33 (19.48)	680.18 (15.42)	740.91 (24.44)	692.89 (20.0)	693.0 (26.92)
int_rate	15.37 (3.36)	14.13 (3.81)	13.45 (3.21)	18.8 (3.35)	10.04 (3.25)	13.07 (3.26)	14.34 (4.21)
loan_amnt	8.4 (3.9)	9.6 (5.2)	13.8 (6.1)	20.2 (7.7)	17.0 (8.1)	19.9 (8.3)	15.0 (8.3)
revol_util	69.82 (15.1)	35.3 (13.52)	50.91 (13.64)	76.76 (14.1)	30.19 (13.72)	64.47 (15.71)	57.16 (21.88)
tot_cur_bal	48.0 (58.9)	40.8 (57.7)	118.9 (93.3)	164.5 (126.7)	186.6 (152.5)	249.3 (182.0)	138.2 (145.1)
emp_length	Experienced	Experienced	Experienced	Senior	Senior	Senior	Senior
grade	C	B	B	D	A	B	C
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	36	36	36
K-means 2015Q1	18.7%	14.4%	14.2%	17.2%	12.5%	22.9%	82465
annual_inc	46.6 (18.8)	54.7 (19.8)	56.7 (26.6)	68.5 (26.0)	93.2 (48.4)	115.1 (56.0)	74.5 (45.0)
bc_util	83.99 (12.63)	50.23 (18.41)	38.19 (18.68)	85.88 (12.94)	30.85 (16.9)	76.41 (16.26)	64.55 (26.56)
dti	18.64 (7.51)	27.7 (5.73)	12.21 (5.73)	25.97 (7.05)	16.11 (7.15)	15.14 (6.03)	19.17 (8.57)
fico_score	678.31 (15.2)	696.29 (21.29)	688.31 (21.32)	681.24 (16.38)	740.47 (26.8)	690.4 (19.81)	693.38 (27.41)
int_rate	13.16 (3.21)	13.39 (3.85)	12.15 (3.86)	17.44 (3.76)	9.24 (3.22)	11.34 (3.3)	12.88 (4.28)
loan_amnt	8.2 (4.1)	13.9 (6.4)	9.8 (5.5)	19.6 (7.4)	18.1 (8.3)	20.6 (8.1)	15.3 (8.4)
revol_util	69.47 (16.04)	43.32 (13.43)	33.32 (14.02)	74.79 (14.88)	28.83 (13.99)	67.71 (15.94)	55.98 (23.19)
tot_cur_bal	54.0 (66.1)	110.6 (94.6)	47.0 (66.7)	145.9 (122.7)	199.5 (169.7)	236.4 (189.7)	137.0 (150.0)
emp_length	Experienced	Experienced	Experienced	Senior	Senior	Senior	Senior
grade	C	C	B	D	A	B	C
home_ownership	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	36	36	36

Table 13: Cluster statistics for K-means with 6 clusters on data of 2016-2018 Q1. Mean and standard deviation are given for numerical features and the mode for categorical features. *annual_inc*, *loan_amnt* and *tot_cur_bal* are given in US$1000.

K-means 2016Q1	15.9%	19.1%	16.7%	10.6%	19.3%	18.3%	120977
annual_inc	51.4 (22.1)	54.3 (21.5)	61.6 (24.0)	84.0 (47.3)	106.1 (54.5)	118.5 (60.0)	80.0 (49.5)
bc_util	33.19 (17.19)	81.03 (13.68)	72.53 (20.53)	22.49 (14.9)	46.91 (16.66)	84.52 (11.98)	59.83 (27.83)
dti	16.4 (8.41)	18.31 (7.51)	30.35 (7.42)	16.6 (8.44)	16.88 (7.36)	18.04 (6.95)	19.51 (9.07)
fico_score	691.83 (21.92)	681.14 (17.34)	684.76 (19.0)	760.63 (24.51)	704.63 (21.48)	686.75 (18.94)	697.46 (30.84)
int_rate	12.17 (4.18)	11.9 (3.27)	18.58 (4.13)	8.09 (3.12)	10.13 (3.52)	12.56 (3.81)	12.43 (4.82)
loan_amnt	9.0 (5.0)	9.0 (4.6)	17.1 (7.9)	15.8 (8.4)	20.0 (8.0)	22.3 (8.2)	15.6 (8.9)
revol_util	29.64 (13.24)	67.14 (16.05)	61.48 (18.41)	21.11 (12.37)	42.11 (13.18)	75.58 (13.55)	52.04 (24.04)
tot_cur_bal	47.3 (59.7)	69.1 (78.4)	128.1 (108.9)	165.9 (158.0)	217.1 (174.7)	251.5 (199.7)	147.7 (159.2)
emp_length	Experienced	Experienced	Senior	Senior	Senior	Senior	Senior
grade	B	B	D	A	B	C	B
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	60	36	36	36	36
K-means 2017Q1	17.2%	22.4%	9.8%	11.0%	19.8%	19.8%	94191
annual_inc	49.5 (21.3)	53.6 (21.0)	69.6 (33.0)	85.5 (47.7)	110.8 (60.8)	114.2 (58.9)	81.3 (51.9)
bc_util	35.07 (17.43)	79.78 (14.3)	65.54 (23.63)	21.46 (14.62)	43.89 (16.16)	82.85 (12.26)	57.78 (27.72)
dti	15.71 (7.99)	20.3 (7.88)	22.77 (8.08)	16.45 (7.95)	16.96 (7.56)	18.94 (7.56)	18.4 (8.11)
fico_score	694.33 (22.84)	681.41 (17.29)	683.71 (19.19)	760.42 (24.9)	704.83 (21.84)	686.51 (18.86)	698.18 (31.2)
int_rate	12.63 (3.61)	13.17 (3.0)	23.97 (4.13)	8.84 (3.33)	11.43 (3.43)	13.38 (3.14)	13.36 (5.06)
loan_amnt	7.7 (4.3)	8.6 (4.5)	18.4 (8.2)	13.6 (8.7)	20.2 (9.0)	21.7 (8.7)	14.9 (9.4)
revol_util	30.92 (13.33)	66.16 (16.03)	56.46 (20.34)	20.14 (12.11)	39.56 (12.96)	73.3 (13.61)	50.24 (23.82)
tot_cur_bal	46.1 (60.4)	79.7 (85.9)	123.7 (125.7)	171.1 (161.2)	224.1 (185.6)	239.3 (188.1)	148.4 (162.0)
emp_length	Experienced	Experienced	Senior	Senior	Senior	Senior	Senior
grade	C	C	E	A	C	C	C
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	60	36	36	36	36
K-means 2018Q1	15.5%	17.1%	14.4%	14.7%	17.9%	20.4%	104126
annual_inc	48.3 (22.4)	52.7 (22.1)	53.4 (21.0)	79.1 (45.8)	109.9 (58.7)	112.3 (60.2)	78.4 (51.4)
bc_util	25.76 (16.42)	74.36 (16.97)	51.75 (22.2)	16.92 (13.31)	81.26 (14.06)	39.14 (17.96)	49.19 (29.01)
dti	12.7 (7.4)	16.47 (7.21)	33.28 (8.18)	16.8 (9.05)	19.88 (7.52)	14.84 (6.6)	18.63 (9.94)
fico_score	707.65 (27.73)	682.37 (17.64)	706.2 (24.9)	774.75 (25.11)	687.67 (19.17)	710.85 (23.4)	710.06 (37.04)
int_rate	11.11 (3.88)	14.69 (4.46)	14.99 (5.16)	7.93 (2.56)	15.12 (5.03)	10.36 (3.51)	12.38 (4.98)
loan_amnt	9.0 (6.3)	9.3 (5.4)	16.2 (8.6)	16.7 (10.4)	22.0 (9.3)	21.7 (10.2)	16.1 (10.1)
revol_util	23.44 (13.14)	62.28 (17.18)	44.84 (17.78)	16.16 (11.37)	72.52 (14.76)	35.97 (14.77)	43.44 (24.91)
tot_cur_bal	34.9 (52.6)	56.7 (71.3)	130.2 (115.8)	154.9 (157.3)	233.5 (193.4)	220.7 (189.9)	143.4 (163.5)
emp_length	Experienced	Experienced	Experienced	Senior	Senior	Senior	Experienced
grade	B	C	C	A	C	B	B
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	36	36	36

B.3 Agglomerative Clustering Model Selection

Table 14: Cluster statistics for Agglomerative clustering with 6 clusters on data of 2013Q1. Mean and standard deviation are given for numerical features and the mode for categorical features. *annual_inc*, *loan_amnt* and *tot_cur_bal* are given in US$1000.

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**	**Cluster 6**	**Overall**
Gower distance, Complete linkage	25.4%	5.6%	36.4%	16.7%	3.6%	12.2%	21191
annual_inc	58.1 (31.5)	62.4 (36.6)	72.3 (38.0)	79.2 (37.6)	98.9 (38.1)	104.5 (45.2)	74.2 (40.0)
bc_util	72.54 (24.37)	79.0 (19.64)	72.39 (21.47)	76.44 (21.1)	51.59 (26.2)	52.26 (26.21)	70.26 (24.28)
dti	17.87 (7.73)	18.8 (7.58)	17.86 (7.51)	19.62 (7.29)	16.61 (7.15)	14.91 (7.03)	17.8 (7.58)
fico_score	686.08 (21.49)	680.83 (16.37)	691.95 (18.83)	690.18 (19.41)	726.9 (26.52)	726.88 (27.96)	695.08 (25.41)
int_rate	15.82 (3.2)	17.89 (1.52)	12.76 (1.8)	18.45 (3.01)	12.18 (2.11)	7.71 (1.02)	14.14 (4.03)
loan_amnt	11.8 (7.2)	13.6 (9.2)	14.3 (7.3)	22.7 (6.4)	22.4 (5.1)	17.3 (6.5)	15.7 (8.1)
revol_util	62.95 (21.7)	67.44 (18.78)	63.96 (19.61)	68.25 (18.85)	47.64 (23.01)	48.05 (23.01)	62.08 (21.54)
tot_cur_bal	48.6 (67.0)	121.6 (125.0)	150.1 (138.8)	157.6 (142.7)	273.8 (158.9)	265.9 (188.2)	142.6 (150.0)
emp_length	Senior	Junior	Experienced	Senior	Senior	Experienced	Experienced
grade	C	D	B	C	B	A	B
home_ownership	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	60	36	36
Agglomerative, Ward's Criterion	22.8%	9.4%	26.9%	17.7%	8.9%	14.4%	21191
annual_inc	49.2 (21.6)	49.3 (19.0)	75.4 (33.4)	81.1 (33.7)	100.2 (56.9)	103.2 (44.5)	74.2 (40.0)
bc_util	83.12 (15.19)	40.62 (17.86)	76.49 (17.29)	83.78 (14.01)	53.48 (24.39)	51.27 (26.3)	70.26 (24.28)
dti	19.23 (7.68)	16.92 (7.99)	17.21 (7.41)	19.98 (6.8)	17.41 (7.55)	14.77 (7.06)	17.8 (7.58)
fico_score	678.22 (13.37)	702.27 (21.69)	690.48 (15.84)	685.05 (16.32)	711.02 (25.85)	728.17 (28.09)	695.08 (25.41)
int_rate	16.94 (2.0)	13.76 (2.89)	12.27 (1.35)	18.16 (2.79)	15.12 (3.58)	7.9 (1.3)	14.14 (4.03)
loan_amnt	9.6 (5.1)	9.7 (4.8)	14.3 (6.1)	23.4 (6.6)	24.1 (6.5)	17.2 (6.5)	15.7 (8.1)
revol_util	70.55 (16.43)	36.81 (13.56)	67.8 (16.78)	74.61 (14.27)	48.91 (20.06)	47.14 (23.19)	62.08 (21.54)
tot_cur_bal	65.3 (81.1)	76.1 (85.4)	138.9 (134.6)	152.5 (132.3)	232.8 (185.9)	247.2 (189.6)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Senior	Senior	Experienced	Experienced
grade	C	B	B	C	B	A	B
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	60	36	36

Table 15: Cluster statistics for Agglomerative clustering with 8 clusters on data of 2013Q1. Mean and standard deviation are given for numerical features and the mode for categorical features. *annual_inc*, *loan_amnt* and *tot_cur_bal* are given in US$1000.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Cluster 8	Overall
Gower distance, Complete linkage	5.5%	19.9%	5.6%	7.1%	29.2%	16.7%	3.6%	12.2%	21191
annual_inc	48.7 (26.9)	60.8 (32.2)	62.4 (36.6)	69.7 (43.0)	72.9 (36.6)	79.2 (37.6)	98.9 (38.1)	104.5 (45.2)	74.2 (40.0)
bc_util	74.07 (25.39)	72.11 (24.06)	79.0 (19.64)	79.47 (19.41)	70.65 (21.59)	76.44 (21.1)	51.59 (26.2)	52.26 (26.21)	70.26 (24.28)
dti	18.56 (8.01)	17.68 (7.64)	18.8 (7.58)	18.83 (7.47)	17.62 (7.5)	19.62 (7.29)	16.61 (7.15)	14.91 (7.03)	17.8 (7.58)
fico_score	680.3 (19.14)	687.69 (21.83)	680.83 (16.37)	680.97 (15.49)	694.64 (18.6)	690.18 (19.41)	726.9 (26.52)	726.88 (27.96)	695.08 (25.41)
int_rate	18.46 (2.02)	15.09 (3.08)	17.89 (1.52)	15.52 (1.02)	12.09 (1.21)	18.45 (3.01)	12.18 (2.11)	7.71 (1.02)	14.14 (4.03)
loan_amnt	10.2 (7.0)	12.2 (7.2)	13.6 (9.2)	15.2 (9.6)	14.1 (6.6)	22.7 (6.4)	22.4 (5.1)	17.3 (6.5)	15.7 (8.1)
revol_util	63.5 (21.99)	62.8 (21.62)	67.44 (18.78)	69.26 (18.05)	62.66 (19.75)	68.25 (18.85)	47.64 (23.01)	48.05 (23.01)	62.08 (21.54)
tot_cur_bal	44.9 (62.6)	49.6 (68.1)	121.6 (125.0)	153.4 (140.6)	149.3 (138.3)	157.6 (142.7)	273.8 (158.9)	265.9 (188.2)	142.6 (150.0)
emp_length	Experienced	Senior	Junior	Senior	Experienced	Senior	Senior	Experienced	Experienced
grade	D	C	D	C	B	C	B	A	B
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	36	60	60	36	36
Agglomerative, Ward's Criterion	22.8%	9.4%	19.4%	7.5%	17.7%	3.1%	8.9%	11.3%	21191
annual_inc	49.2 (21.6)	49.3 (19.0)	73.3 (33.2)	80.9 (33.4)	81.1 (33.7)	87.8 (49.0)	100.2 (56.9)	107.5 (42.2)	74.2 (40.0)
bc_util	83.12 (15.19)	40.62 (17.86)	79.4 (15.05)	68.94 (20.23)	83.78 (14.01)	20.0 (13.69)	53.48 (24.39)	59.96 (22.01)	70.26 (24.28)
dti	19.23 (7.68)	16.92 (7.99)	19.38 (7.13)	11.56 (4.66)	19.98 (6.8)	13.33 (7.93)	17.41 (7.55)	15.17 (6.75)	17.8 (7.58)
fico_score	678.22 (13.37)	702.27 (21.69)	691.75 (16.02)	687.16 (14.88)	685.05 (16.32)	765.16 (20.81)	711.02 (25.85)	717.89 (20.05)	695.08 (25.41)
int_rate	16.94 (2.0)	13.76 (2.89)	12.13 (1.23)	12.64 (1.56)	18.16 (2.79)	7.72 (2.12)	15.12 (3.58)	7.94 (0.94)	14.14 (4.03)
loan_amnt	9.6 (5.1)	9.7 (4.8)	14.5 (6.1)	13.8 (5.9)	23.4 (6.6)	14.2 (7.3)	24.1 (6.5)	18.0 (6.1)	15.7 (8.1)
revol_util	70.55 (16.43)	36.81 (13.56)	70.15 (15.25)	61.69 (18.9)	74.61 (14.27)	19.9 (12.16)	48.91 (20.06)	54.71 (19.57)	62.08 (21.54)
tot_cur_bal	65.3 (81.1)	76.1 (85.4)	179.0 (137.2)	34.8 (35.7)	152.5 (132.3)	188.4 (194.4)	232.8 (185.9)	263.5 (185.0)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Experienced	Senior	Experienced	Senior	Senior	Experienced
grade	C	B	B	B	C	A	B	A	B
home_ownership	RENT	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	60	36	60	36	36

B.4 GMM Model Selection

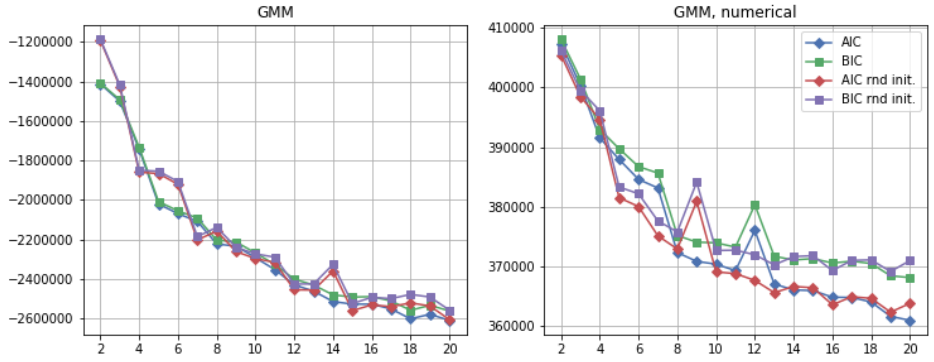

Figure 27: AIC and BIC scores shown for GMM models using K-means or random initialization while using all core features or only the metric features. Results are shown for the data of 2014-2018 Q1.

Table 16: Cluster statistics for GMM with 6 clusters on data of 2013Q1. Mean and standard deviation are given for numerical features and the mode for categorical features. *annual_inc*, *loan_amnt* and *tot_cur_bal* are given in US$1000.

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**	**Cluster 6**	**Overall**
GMM K-means init.	4.2%	27.8%	4.9%	5.7%	44.1%	13.3%	21191
annual_inc	50.5 (34.2)	59.5 (34.1)	75.4 (39.9)	76.1 (38.1)	82.3 (41.5)	84.2 (37.3)	74.2 (40.0)
bc_util	76.47 (22.69)	76.84 (22.01)	71.5 (22.91)	81.2 (21.43)	65.08 (24.54)	66.63 (24.77)	70.26 (24.28)
dti	18.95 (7.95)	18.38 (7.65)	19.08 (7.62)	19.84 (7.3)	16.66 (7.44)	18.69 (7.3)	17.8 (7.58)
fico_score	679.62 (16.27)	681.15 (16.94)	693.11 (19.33)	679.48 (15.33)	703.97 (26.06)	706.86 (26.31)	695.08 (25.41)
int_rate	18.45 (2.62)	16.56 (1.63)	15.96 (3.38)	22.31 (1.15)	10.82 (2.3)	14.6 (2.75)	14.14 (4.03)
loan_amnt	10.9 (8.0)	12.5 (8.2)	20.2 (8.8)	22.6 (8.0)	15.0 (6.7)	21.6 (5.5)	15.7 (8.1)
revol_util	65.93 (19.81)	66.25 (20.21)	63.49 (20.98)	71.38 (19.14)	58.23 (21.87)	60.41 (21.79)	62.08 (21.54)
tot_cur_bal	51.4 (90.5)	85.1 (108.1)	138.5 (146.7)	120.6 (131.1)	173.2 (162.2)	200.5 (152.6)	142.6 (150.0)
emp_length	Experienced	Experienced	Senior	Experienced	Experienced	Senior	Experienced
grade	C	C	D	E	B	C	B
home_ownership	OWN	RENT	OWN	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	60	60	36	60	36
GMM K-means init., num. variables	23.8%	13.8%	20.5%	9.2%	21.8%	10.9%	21191
annual_inc	57.2 (27.6)	70.5 (42.8)	78.9 (31.7)	80.0 (51.8)	80.1 (36.9)	90.5 (53.0)	74.2 (40.0)
bc_util	72.69 (18.1)	93.08 (5.68)	80.44 (14.74)	47.59 (25.78)	70.41 (19.41)	35.82 (20.98)	70.26 (24.28)
dti	17.87 (7.58)	17.99 (7.54)	20.99 (6.75)	16.68 (7.54)	17.38 (7.2)	13.22 (7.12)	17.8 (7.58)
fico_score	687.17 (17.12)	680.51 (13.78)	690.13 (18.35)	707.02 (30.62)	696.71 (20.32)	726.67 (34.83)	695.08 (25.41)
int_rate	13.9 (2.78)	16.42 (3.05)	17.15 (3.33)	15.3 (4.41)	10.89 (2.62)	11.65 (4.11)	14.14 (4.03)
loan_amnt	12.7 (5.4)	11.2 (6.9)	24.4 (6.0)	12.6 (9.1)	14.8 (5.7)	16.0 (8.4)	15.7 (8.1)
revol_util	65.5 (18.06)	71.0 (16.62)	73.24 (14.87)	44.26 (18.05)	63.39 (18.55)	34.78 (20.88)	62.08 (21.54)
tot_cur_bal	29.0 (18.7)	110.2 (131.6)	176.8 (142.1)	142.4 (163.6)	249.7 (129.3)	153.0 (186.6)	142.6 (150.0)
emp_length	Experienced	Experienced	Senior	Experienced	Experienced	Experienced	Experienced
grade	B	C	C	C	B	A	B
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	60	36	36	36	36
GMM, rnd init.	12.8%	7.1%	2.1%	61.5%	0.2%	16.3%	21191
annual_inc	61.8 (36.6)	66.3 (40.3)	69.4 (34.5)	71.3 (36.7)	77.6 (41.8)	98.7 (45.1)	74.2 (40.0)
bc_util	77.42 (22.58)	77.88 (21.94)	81.81 (22.77)	71.77 (22.45)	82.02 (24.19)	54.05 (25.99)	70.26 (24.28)
dti	18.82 (7.71)	19.47 (7.6)	19.49 (7.45)	17.96 (7.5)	18.15 (6.45)	15.47 (7.25)	17.8 (7.58)
fico_score	681.1 (17.3)	681.25 (16.8)	678.32 (15.35)	692.65 (20.79)	673.74 (10.07)	723.59 (28.7)	695.08 (25.41)
int_rate	18.59 (0.68)	19.75 (2.69)	23.37 (0.33)	13.75 (2.14)	24.79 (0.09)	8.4 (1.87)	14.14 (4.03)
loan_amnt	14.0 (9.3)	17.9 (10.0)	20.2 (8.2)	15.3 (7.7)	21.0 (7.8)	16.9 (6.6)	15.7 (8.1)
revol_util	66.81 (20.39)	67.99 (19.64)	71.73 (19.91)	63.44 (20.3)	69.41 (20.27)	49.34 (22.92)	62.08 (21.54)
tot_cur_bal	93.5 (121.5)	96.6 (129.8)	98.1 (105.8)	136.6 (137.1)	88.0 (134.3)	229.8 (188.5)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Experienced	Experienced	Senior	Experienced
grade	D	E	F	B	G	A	B
home_ownership	RENT	RENT	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE
term	36	60	60	36	60	36	36
GMM rnd inti., num. variables	23.5%	13.7%	11.1%	16.1%	24.3%	11.3%	21191
annual_inc	58.2 (29.5)	73.6 (44.4)	77.7 (43.8)	77.8 (30.9)	81.9 (38.1)	83.1 (53.6)	74.2 (40.0)
bc_util	70.39 (19.86)	93.69 (5.28)	65.79 (26.06)	78.32 (16.37)	67.98 (21.06)	39.49 (25.0)	70.26 (24.28)
dti	17.83 (7.59)	18.47 (7.43)	15.23 (7.6)	21.56 (6.72)	17.28 (7.07)	15.24 (7.65)	17.8 (7.58)
fico_score	688.12 (18.13)	682.41 (14.79)	700.55 (28.82)	689.7 (18.23)	699.02 (23.08)	718.65 (37.07)	695.08 (25.41)
int_rate	13.88 (2.93)	16.4 (3.22)	13.23 (3.89)	17.72 (3.17)	11.22 (2.88)	14.03 (4.74)	14.14 (4.03)
loan_amnt	12.4 (5.3)	12.3 (7.6)	15.6 (8.0)	25.1 (5.9)	15.9 (6.1)	12.9 (9.1)	15.7 (8.1)
revol_util	62.12 (18.67)	70.5 (16.77)	65.79 (26.06)	71.23 (15.86)	60.64 (19.36)	38.21 (20.35)	62.08 (21.54)
tot_cur_bal	29.9 (20.3)	127.2 (143.4)	140.8 (161.2)	155.2 (138.8)	252.7 (131.1)	142.1 (173.9)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Senior	Senior	Experienced	Experienced
grade	B	C	B	C	B	B	B
home_ownership	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	36	36	36

B.5 Cluster Characteristics 2013Q1

Table 17: Cluster statistics for K-means, K-prototypes, Agglomerative clustering and GMM with 6 clusters on data from 2013Q1. Mean and standard deviation are given for numerical features and the mode for categorical features. *annual_inc*, *loan_amnt* and *tot_cur_bal* are given in US$1000.

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**	**Cluster 6**	**Overall**
K-means	17.2%	18.7%	14.7%	17.6%	10.3%	21.5%	21191
annual_inc	47.6 (19.7)	52.3 (15.6)	55.7 (21.7)	85.1 (33.6)	93.9 (44.5)	108.7 (45.3)	74.2 (40.0)
bc_util	84.75 (12.46)	79.67 (14.89)	43.3 (17.06)	85.52 (13.29)	30.39 (16.58)	75.59 (15.02)	70.26 (24.28)
dti	14.98 (6.25)	26.35 (4.59)	14.61 (6.81)	19.55 (6.27)	14.61 (7.35)	14.89 (5.91)	17.8 (7.58)
fico_score	678.38 (13.58)	691.4 (17.2)	696.58 (19.81)	681.64 (14.93)	742.98 (25.57)	698.54 (18.27)	695.08 (25.41)
int_rate	16.63 (2.7)	14.29 (2.54)	14.28 (3.01)	18.2 (2.94)	9.18 (2.97)	11.0 (2.53)	14.14 (4.03)
loan_amnt	8.6 (4.2)	13.2 (5.4)	11.4 (6.0)	23.9 (6.9)	17.8 (7.6)	18.7 (7.0)	15.7 (8.1)
revol_util	72.04 (14.74)	69.31 (14.71)	38.92 (13.04)	76.61 (13.97)	29.14 (13.97)	67.58 (14.51)	62.08 (21.54)
tot_cur_bal	36.7 (48.3)	109.0 (89.0)	59.8 (70.3)	163.5 (137.4)	214.3 (170.9)	261.2 (177.3)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Senior	Experienced	Senior	Experienced
grade	C	B	B	C	A	B	B
home_ownership	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	36	36	36
K-prototypes	21.4%	13.8%	15.6%	18.4%	11.3%	19.5%	21191
annual_inc	47.0 (18.4)	55.4 (22.5)	57.2 (17.6)	83.8 (34.1)	97.0 (44.8)	108.6 (46.0)	74.2 (40.0)
bc_util	85.71 (12.09)	42.88 (18.03)	75.87 (16.25)	82.14 (15.41)	33.23 (17.71)	78.29 (14.5)	70.26 (24.28)
dti	17.69 (7.24)	14.07 (6.84)	25.51 (5.09)	20.19 (6.63)	14.36 (7.02)	14.11 (5.45)	17.8 (7.58)
fico_score	678.23 (13.38)	697.21 (21.53)	695.71 (17.59)	685.45 (16.59)	739.9 (26.09)	694.75 (18.0)	695.08 (25.41)
int_rate	16.8 (2.58)	14.46 (3.06)	12.94 (2.18)	18.15 (2.71)	8.98 (2.82)	11.16 (2.35)	14.14 (4.03)
loan_amnt	9.5 (4.6)	11.1 (6.0)	13.4 (5.5)	24.3 (6.6)	18.0 (7.4)	18.1 (7.0)	15.7 (8.1)
revol_util	73.26 (14.5)	38.25 (13.61)	66.37 (15.15)	73.34 (15.18)	31.55 (14.91)	70.16 (14.53)	62.08 (21.54)
tot_cur_bal	37.7 (44.0)	45.8 (57.4)	139.9 (100.8)	173.3 (136.9)	226.9 (171.9)	250.1 (180.6)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Senior	Experienced	Senior	Experienced
grade	C	B	B	C	A	B	B
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	36	36	36
Agglomerative	22.8%	9.4%	26.9%	17.7%	8.9%	14.4%	21191
annual_inc	49.2 (21.6)	49.3 (19.0)	75.4 (33.4)	81.1 (33.7)	100.2 (56.9)	103.2 (44.5)	74.2 (40.0)
bc_util	83.12 (15.19)	40.62 (17.86)	76.49 (17.29)	83.78 (14.01)	53.48 (24.39)	51.27 (26.3)	70.26 (24.28)
dti	19.23 (7.68)	16.92 (7.99)	17.21 (7.41)	19.98 (6.8)	17.41 (7.55)	14.77 (7.06)	17.8 (7.58)
fico_score	678.22 (13.37)	702.27 (21.69)	690.48 (15.84)	685.05 (16.32)	711.02 (25.85)	728.17 (28.09)	695.08 (25.41)
int_rate	16.94 (2.0)	13.76 (2.89)	12.27 (1.35)	18.16 (2.79)	15.12 (3.58)	7.9 (1.3)	14.14 (4.03)
loan_amnt	9.6 (5.1)	9.7 (4.8)	14.3 (6.1)	23.4 (6.6)	24.1 (6.5)	17.2 (6.5)	15.7 (8.1)
revol_util	70.55 (16.43)	36.81 (13.56)	67.8 (16.78)	74.61 (14.27)	48.91 (20.06)	47.14 (23.19)	62.08 (21.54)
tot_cur_bal	65.3 (81.1)	76.1 (85.4)	138.9 (134.6)	152.5 (132.3)	232.8 (185.9)	247.2 (189.6)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Senior	Senior	Experienced	Experienced
grade	C	B	B	C	B	A	B
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	60	36	36
GMM	23.5%	13.7%	11.1%	16.1%	24.3%	11.3%	21191
annual_inc	58.2 (29.5)	73.6 (44.4)	77.7 (43.8)	77.8 (30.9)	81.9 (38.1)	83.1 (53.6)	74.2 (40.0)
bc_util	70.39 (19.86)	93.69 (5.28)	65.79 (26.06)	78.32 (16.37)	67.98 (21.06)	39.49 (25.0)	70.26 (24.28)
dti	17.83 (7.59)	18.47 (7.43)	15.23 (7.6)	21.56 (6.72)	17.28 (7.07)	15.24 (7.65)	17.8 (7.58)
fico_score	688.12 (18.13)	682.41 (14.79)	700.55 (28.82)	689.7 (18.23)	699.02 (23.08)	718.65 (37.07)	695.08 (25.41)
int_rate	13.88 (2.93)	16.4 (3.22)	13.23 (3.89)	17.72 (3.17)	11.22 (2.88)	14.03 (4.74)	14.14 (4.03)
loan_amnt	12.4 (5.3)	12.3 (7.6)	15.6 (8.0)	25.1 (5.9)	15.9 (6.1)	12.9 (9.1)	15.7 (8.1)
revol_util	62.12 (18.67)	70.5 (16.77)	65.79 (26.06)	71.23 (15.86)	60.64 (19.36)	38.21 (20.35)	62.08 (21.54)
tot_cur_bal	29.9 (20.3)	127.2 (143.4)	140.8 (161.2)	155.2 (138.8)	252.7 (131.1)	142.1 (173.9)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Senior	Senior	Experienced	Experienced
grade	B	C	B	C	B	B	B
home_ownership	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	60	36	36	36

Table 18: Cluster statistics for all models with 8 clusters on data from 2013Q1. Mean and standard deviation are given for numerical features and the mode for categorical features. *annual_inc*, *loan_amnt* and *tot_cur_bal* are given in US$1000.

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**	**Cluster 6**	**Cluster 7**	**Cluster 8**	**Overall**
K-means	11.4%	11.4%	16.2%	14.1%	13.7%	5.9%	13.2%	14.1%	21191
annual_inc	37.9 (14.2)	51.4 (19.4)	54.1 (15.4)	68.5 (25.1)	80.1 (30.2)	87.6 (42.4)	100.5 (41.9)	114.5 (48.6)	74.2 (40.0)
bc_util	84.31 (13.76)	39.68 (16.49)	79.24 (14.53)	80.5 (13.34)	85.22 (13.67)	21.81 (13.99)	52.84 (14.49)	85.07 (10.69)	70.26 (24.28)
dti	19.59 (6.77)	15.17 (7.06)	26.11 (4.61)	11.33 (4.27)	21.54 (5.83)	12.91 (7.12)	16.21 (6.55)	15.33 (5.62)	17.8 (7.58)
fico_score	677.99 (13.46)	697.58 (20.21)	693.47 (16.93)	681.86 (14.58)	681.3 (14.86)	757.09 (22.43)	713.18 (18.42)	692.44 (17.19)	695.08 (25.41)
int_rate	17.49 (2.35)	14.52 (3.02)	13.81 (2.32)	14.72 (2.67)	19.06 (2.72)	8.79 (2.96)	10.46 (3.04)	11.86 (2.73)	14.14 (4.03)
loan_amnt	6.7 (2.9)	10.4 (5.6)	13.6 (5.2)	13.7 (5.6)	24.2 (6.8)	16.0 (7.6)	19.8 (7.1)	19.3 (7.3)	15.7 (8.1)
revol_util	71.19 (15.59)	36.27 (12.61)	69.06 (14.29)	69.3 (14.46)	76.19 (14.13)	21.43 (11.78)	47.89 (11.53)	76.86 (11.87)	62.08 (21.54)
tot_cur_bal	38.7 (49.3)	54.6 (64.6)	116.8 (90.3)	47.4 (53.1)	153.9 (123.5)	183.1 (168.1)	246.1 (163.0)	297.1 (175.6)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Experienced	Senior	Experienced	Senior	Senior	Experienced
grade	C	B	B	B	C	A	A	B	B
home_ownership	RENT	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	60	36	36	36	36
K-prototypes	15.2%	8.8%	12.3%	13.7%	15.8%	16.4%	6.8%	11.1%	21191
annual_inc	40.5 (12.9)	50.8 (22.1)	60.3 (18.8)	65.7 (26.6)	82.8 (30.5)	85.8 (37.6)	90.4 (41.5)	125.5 (52.1)	74.2 (40.0)
bc_util	84.8 (13.0)	39.83 (17.89)	52.71 (16.35)	82.22 (12.96)	86.37 (10.5)	82.99 (14.15)	24.0 (14.52)	65.52 (16.4)	70.26 (24.28)
dti	24.17 (5.51)	12.73 (6.25)	21.64 (6.61)	11.49 (4.34)	18.38 (6.63)	20.62 (6.64)	13.27 (7.0)	14.37 (6.18)	17.8 (7.58)
fico_score	680.87 (14.7)	696.58 (21.91)	702.65 (18.05)	680.38 (14.08)	688.13 (16.21)	686.13 (16.74)	752.84 (24.13)	711.02 (17.63)	695.08 (25.41)
int_rate	16.77 (2.62)	14.99 (3.2)	12.81 (2.46)	15.46 (2.92)	12.51 (1.98)	18.21 (2.74)	8.75 (2.87)	9.31 (2.5)	14.14 (4.03)
loan_amnt	9.2 (4.5)	9.3 (5.4)	15.0 (6.2)	13.2 (6.0)	14.6 (6.3)	24.9 (6.3)	16.7 (7.7)	20.9 (6.6)	15.7 (8.1)
revol_util	72.32 (14.7)	35.6 (13.51)	47.15 (12.21)	70.81 (14.57)	77.0 (12.27)	74.17 (14.32)	23.6 (12.44)	59.06 (14.74)	62.08 (21.54)
tot_cur_bal	47.8 (45.8)	33.5 (47.2)	133.9 (96.5)	29.2 (28.0)	221.2 (129.2)	182.5 (140.8)	198.7 (169.6)	303.1 (199.1)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Experienced	Senior	Senior	Experienced	Senior	Experienced
grade	C	B	B	C	B	C	A	A	B
home_ownership	RENT	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	36	60	36	36	36
Agglomerative	22.8%	9.4%	19.4%	7.5%	17.7%	3.1%	8.9%	11.3%	21191
annual_inc	49.2 (21.6)	49.3 (19.0)	73.3 (33.2)	80.9 (33.4)	81.1 (33.7)	87.8 (49.0)	100.2 (56.9)	107.5 (42.2)	74.2 (40.0)
bc_util	83.12 (15.19)	40.62 (17.86)	79.4 (15.05)	68.94 (20.23)	83.78 (14.01)	20.0 (13.69)	53.48 (24.39)	59.96 (22.01)	70.26 (24.28)
dti	19.23 (7.68)	16.92 (7.99)	19.38 (7.13)	11.56 (4.66)	19.98 (6.8)	13.33 (7.93)	17.41 (7.55)	15.17 (6.75)	17.8 (7.58)
fico_score	678.22 (13.37)	702.27 (21.69)	691.75 (16.02)	687.16 (14.88)	685.05 (16.32)	765.16 (20.81)	711.02 (25.85)	717.89 (20.05)	695.08 (25.41)
int_rate	16.94 (2.0)	13.76 (2.89)	12.13 (1.23)	12.64 (1.56)	18.16 (2.79)	7.72 (2.12)	15.12 (3.58)	7.94 (0.94)	14.14 (4.03)
loan_amnt	9.6 (5.1)	9.7 (4.8)	14.5 (6.1)	13.8 (5.9)	23.4 (6.6)	14.2 (7.3)	24.1 (6.5)	18.0 (6.1)	15.7 (8.1)
revol_util	70.55 (16.43)	36.81 (13.56)	70.15 (15.25)	61.69 (18.9)	74.61 (14.27)	19.9 (12.16)	48.91 (20.06)	54.71 (19.57)	62.08 (21.54)
tot_cur_bal	65.3 (81.1)	76.1 (85.4)	179.0 (137.2)	34.8 (35.7)	152.5 (132.3)	188.4 (194.4)	232.8 (185.9)	263.5 (185.0)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Experienced	Senior	Experienced	Senior	Senior	Experienced
grade	C	B	B	B	C	A	B	A	B
home_ownership	RENT	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	60	36	60	36	36
GMM	15.9%	14.4%	9.8%	15.7%	11.0%	9.0%	17.4%	6.8%	21191
annual_inc	57.9 (24.1)	63.2 (31.2)	65.3 (35.8)	74.9 (34.5)	77.7 (43.9)	78.8 (48.2)	90.0 (37.9)	94.6 (59.9)	74.2 (40.0)
bc_util	82.22 (13.53)	64.65 (20.44)	65.6 (26.92)	67.41 (21.26)	66.03 (25.66)	94.62 (4.02)	75.97 (17.13)	27.63 (16.1)	70.26 (24.28)
dti	21.69 (7.25)	15.45 (6.68)	17.61 (7.53)	17.05 (7.35)	15.25 (7.6)	19.9 (7.11)	19.27 (6.81)	13.31 (7.16)	17.8 (7.58)
fico_score	684.45 (16.33)	691.67 (18.93)	688.84 (19.64)	696.53 (21.48)	700.14 (28.17)	681.16 (13.97)	696.28 (21.27)	739.69 (33.72)	695.08 (25.41)
int_rate	16.91 (3.22)	12.97 (2.87)	16.53 (3.27)	11.11 (2.71)	13.24 (3.88)	15.7 (3.14)	14.71 (4.15)	11.64 (4.58)	14.14 (4.03)
loan_amnt	17.9 (7.4)	12.2 (5.4)	9.4 (6.7)	12.6 (4.7)	15.7 (8.0)	12.9 (7.6)	24.2 (5.6)	16.2 (8.9)	15.7 (8.1)
revol_util	68.36 (14.36)	58.94 (19.26)	52.53 (15.98)	57.08 (16.9)	66.03 (25.67)	87.56 (6.3)	67.45 (15.63)	25.58 (13.62)	62.08 (21.54)
tot_cur_bal	35.9 (20.6)	30.7 (23.0)	96.2 (112.6)	226.5 (124.4)	140.8 (161.5)	147.9 (155.3)	272.7 (132.3)	165.0 (198.3)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Experienced	Experienced	Senior	Senior	Experienced	Experienced
grade	C	B	C	B	B	C	C	A	B
home_ownership	RENT	RENT	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	36	36	60	36	36

C Additional Model Results: Tracking Segments

C.1 K-Prototypes

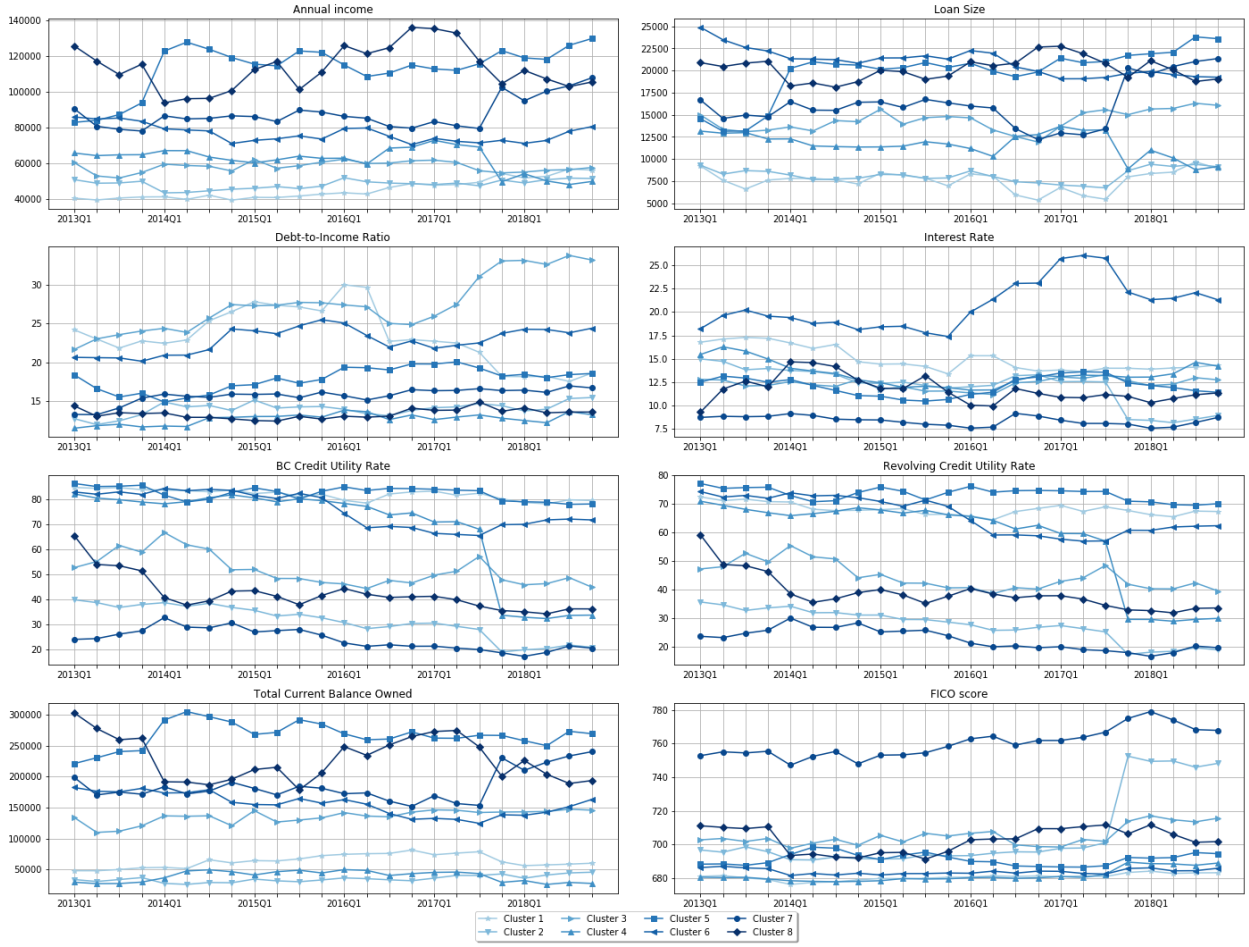

Figure 28: Customer segment trends of metric features between 2013Q1 and 2018Q4 by K-prototypes clustering. The customer segments are ranked in terms of annual income as measured in 2013Q1.

C.2 Agglomerative Clustering

Table 19: Cluster statistics for Agglomerative clustering with 8 clusters on data of 2013Q1 and 2013Q2. Mean and standard deviation are given for numerical features and the mode for categorical features. *annual_inc*, *loan_amnt* and *tot_cur_bal* are given in US$1000.

Agglomerative	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Cluster 8	Overall
2013Q1	22.8%	9.4%	19.4%	7.5%	17.7%	3.1%	8.9%	11.3%	21191
annual_inc	49.2 (21.6)	49.3 (19.0)	73.3 (33.2)	80.9 (33.4)	81.1 (33.7)	87.8 (49.0)	100.2 (56.9)	107.5 (42.2)	74.2 (40.0)
bc_util	83.12 (15.19)	40.62 (17.86)	79.4 (15.05)	68.94 (20.23)	83.78 (14.01)	20.0 (13.69)	53.48 (24.39)	59.96 (22.01)	70.26 (24.28)
dti	19.23 (7.68)	16.92 (7.99)	19.38 (7.13)	11.56 (4.66)	19.98 (6.8)	13.33 (7.93)	17.41 (7.55)	15.17 (6.75)	17.8 (7.58)
fico_score	678.22 (13.37)	702.27 (21.69)	691.75 (16.02)	687.16 (14.88)	685.05 (16.32)	765.16 (20.81)	711.02 (25.85)	717.89 (20.05)	695.08 (25.41)
int_rate	16.94 (2.0)	13.76 (2.89)	12.13 (1.23)	12.64 (1.56)	18.16 (2.79)	7.72 (2.12)	15.12 (3.58)	7.94 (0.94)	14.14 (4.03)
loan_amnt	9.6 (5.1)	9.7 (4.8)	14.5 (6.1)	13.8 (5.9)	23.4 (6.6)	14.2 (7.3)	24.1 (6.5)	18.0 (6.1)	15.7 (8.1)
revol_util	70.55 (16.43)	36.81 (13.56)	70.15 (15.25)	61.69 (18.9)	74.61 (14.27)	19.9 (12.16)	48.91 (20.06)	54.71 (19.57)	62.08 (21.54)
tot_cur_bal	65.3 (81.1)	76.1 (85.4)	179.0 (137.2)	34.8 (35.7)	152.5 (132.3)	188.4 (194.4)	232.8 (185.9)	263.5 (185.0)	142.6 (150.0)
emp_length	Experienced	Experienced	Experienced	Experienced	Senior	Experienced	Senior	Senior	Experienced
grade	C	B	B	B	C	A	B	A	B
home_ownership	RENT	RENT	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	60	36	60	36	36
2013Q2	4.5%	12.9%	9.5%	19.2%	10.7%	18.9%	6.0%	18.3%	28750
annual_inc	38.2 (17.5)	46.7 (22.1)	66.2 (31.8)	69.2 (33.1)	69.6 (34.1)	83.6 (46.6)	84.0 (50.2)	86.8 (46.3)	71.5 (40.9)
bc_util	40.84 (19.34)	85.34 (12.57)	83.81 (15.04)	77.15 (14.67)	55.95 (25.17)	80.78 (15.7)	22.65 (14.72)	51.07 (19.18)	67.58 (25.08)
dti	13.27 (8.08)	17.62 (7.89)	20.06 (7.33)	17.78 (7.25)	16.45 (7.5)	19.58 (6.9)	13.17 (7.57)	15.05 (6.79)	17.2 (7.59)
fico_score	697.16 (21.82)	679.25 (14.75)	681.18 (15.11)	691.84 (17.44)	688.86 (20.41)	688.38 (18.03)	757.62 (26.16)	710.7 (20.97)	695.88 (26.5)
int_rate	13.76 (3.47)	17.3 (2.21)	16.87 (1.88)	12.2 (1.25)	17.7 (2.37)	19.29 (3.18)	9.17 (3.11)	10.91 (3.09)	14.88 (4.31)
loan_amnt	7.1 (3.6)	8.6 (4.5)	11.0 (6.1)	13.0 (5.7)	14.1 (7.9)	22.6 (6.9)	14.3 (8.1)	16.6 (7.1)	14.7 (8.0)
revol_util	35.77 (14.23)	72.67 (15.73)	71.36 (16.37)	67.48 (15.0)	48.95 (19.5)	70.96 (16.03)	21.41 (12.28)	46.32 (16.29)	59.13 (22.03)
tot_cur_bal	18.2 (25.3)	29.9 (31.5)	158.3 (112.2)	144.1 (136.0)	60.6 (70.6)	173.9 (154.8)	177.8 (164.1)	203.2 (171.4)	134.6 (144.6)
emp_length	Experienced	Experienced	Senior	Experienced	Experienced	Senior	Experienced	Senior	Experienced
grade	B	C	C	B	C	C	A	B	B
home_ownership	RENT	RENT	MORTGAGE	MORTGAGE	RENT	MORTGAGE	MORTGAGE	MORTGAGE	MORTGAGE
term	36	36	36	36	36	60	36	36	36