Predicting Home Credit Default Risk

Nitu Sharaff
sharaff@usc.edu
University of Southern California
Los Angeles, California

Vaishnavi Patil
patilvai@usc.edu
University of Southern California
Los Angeles, California

Srushti Yadhunath
yadhunat@usc.edu
University of Southern California
Los Angeles, California

Abstract

Data mining's strength lies in determining or predicting trends in areas where there is a large volume of data and a pattern may not necessarily be easily visible to the human eye. The area of loan applications and loan defaulting is one such field where there is scope for applying prediction algorithms to better help classify loan applicants for financial institutions. Clustering algorithms have been applied to a Kaggle data set obtain from a financial institution for providing these institutions with a more wholesome risk analysis of an applicants ability to repay a loan.

Keywords Clustering, K-Means, Gaussian mixture model, Loan Prediction, CURE, Hierarchical Clustering

1 Introduction

Loan default is a significant source of loss for banks and financial institutions that lend money. It is an area that is paid attention to carefully when lending money to customers. The financial industry sees this area as a massive opportunity for applying prediction and machine learning algorithms to gather more insight when making a decision for loan applications. [9]

Data mining is powerful in this area as it helps in two primary ways: 1) converting raw data and clean it into a form that can be processed and 2) discovering trends or predicting trends from large volumes of data that is not visible easily to human observation. This paper explores the application of mining procedures on credit risk. Specifically, clustering algorithms are being applied in for this application to predict an applicants loan repayment ability.

2 Problem

Credit risk is a decision that needs to be carefully considered when bankers or lenders choose to give money. Financial institutions and money lenders look at and evaluate a consistent set of factors when evaluating a customer or applicants ability to repay a loan including some of the following:

1. Credit rating and Credit history
2. Collaterals
3. Age
4. Occupation
5. Surplus funds

Financial Inclusion is lacking when it comes to lending loans to customers with a limited credit history or when any of the above factors are lacking. Many of these customers are left out of opportunities to borrow money since they don't qualify under the traditional banking methods mentioned above. Additional factors could be included to evaluate an applicant's ability to repay loans to improve financial inclusion in this area. [4] The data provided includes factors such as telecommunication and transactional information among others in an effort to evaluate the customers ability in a more wholesome manner. Bankers and financial institutions are increasingly giving more information in forming a consolidated risk management system. Advances in technology, artificial intelligence and machine learning methods has given rise to this primarily.

3 Challenges

The challenges of applying clustering algorithms and challenges of predicting risk in the financial industry will be addressed in this section. Clustering algorithms or cluster analysis [15] aims to divide data into groups based on some similarity measure. It helps to improve our perception of the data and show groups in a seemingly dissimilar dataset. The greater the similarity in a group and the greater the differences between groups, the more effective the clustering we achieve. Clustering often does not use previously provided class labels, hence in this report, we removed the class labels that were provided and allowed our algorithms to predict the labels itself. Cluster attributes can be binary, discrete and continuous. In our case, since applicants can fall into one of two categories as either being approved or not approved, our cluster values are binary. The classic challenge of clustering that we faced in our example was the "curse of dimensionality". Data sparsity increases with dimensions and becomes tougher to cluster. One way of combating this issues is to visually plot a histogram version of the data and gain an understanding of their pairwise distances. One of the ways we used in this report to combat this issue was to experiment with multiple clustering algorithms and gain an understanding of the dataset through an ensemble of algorithms instead of just one. It allowed us to understand which algorithm would be better suited to this dataset and also deal with a large dataset (900 MB). Obstacles to predicting risk or risk management [3] in the financial industry include the massive rate at which data is being generated, the diversity and complexity of the data since it often comes from different business lines and is often a bottom-up process which makes it difficult to aggregate when we reach the higher levels and attempt to merge risks from different avenues. Although in the context of our report, we are specifically dealing with home loans, these are issues worth considering when the algorithm is applied to a larger application or larger institution with more factors to deal with when prediction and modeling is done.

4 Dataset Description

Data was obtained from a Kaggle dataset provided by a financial institution based in the United States known as HomeCredit. One of HomeCredits sustainability vision is to improve financial inclusion since this is often lacking when it comes The aim of the dataset released was for better algorithms to be applied to the dataset and evaluate if financial inclusion could be improved by factoring in additional criteria in evaluating an applicants ability to repay a loan. The data provided is vast in terms of the number of factors being included in an applicants consideration. There are seven tables overall. The list of the tables and their description is provided below:

1. **Application | Train and Test** - The train table consists of applicant information that can be used to train our model. The test table consists of applicant information that we can test our model against. Both tables consist of class labels which are 0 and 1 to indicate if an applicant qualifies for a loan or not. While training our algorithm, class labels were removed. Algorithm labels that were predicted were then compared with the provided labels to computer precision and recall. Each row of the table represents a loan application
2. **Bureau** - The customers prior credits provided by other lenders or banks and were reported to the credit bureau are in this table. For each loan, there are as many rceords as the credits the client has in the bureaue prior to their application date.
3. **Bureau Balance** - The previous credits in the credit bureau and their monthly balances are found in this table.
4. **POS Cash Balance** - The records of previous Point-Of-Sales and cash loans that a customer had with the Home Credit institution is provided here. The rows correspond to a month-wise credit history of the applicant with respect to HomeCredit.
5. **Credit Card Balance** - The credit cards owned by the applicant with HomeCredit is reported in this table. The table has a month-wise credit history of the applicant.
6. **Previous Application** - If a customer has filed for previous loans with HomeCredit, these are provided in this table.
7. **Installment Payments** - The history of the applicants repayment of all the previously given credits in Home Credit related to the loans in our sample. There are records for every payment and every payment missed. One record is equivalent to one payment of one installment OR one installment corresponding to one payment of one previous Home Credit credit related to loans in our sample.

5 Solutions

The overview of the entire procedure followed (Figure 1) is given as follows:

1. Create data frame by inner join of all files using foreign keys.
2. Preprocessing and cleaning data:
   a. Remove high categorical columns.
   b. Grouping similar ID rows by their means.
   c. Imputing NULL values by means if the columns provide at least 30 percent of information, else drop columns.
   d. Removed target attribute to use for evaluation only.
3. Implement algorithms for clustering - K-means, Gaussian Mixture Models, CURE.
4. Generate combinations for clustering to compare with target values.
5. Evaluation of precision, recall and F1 score for all three clustering techniques.
6. Comparison of algorithms and visualizations

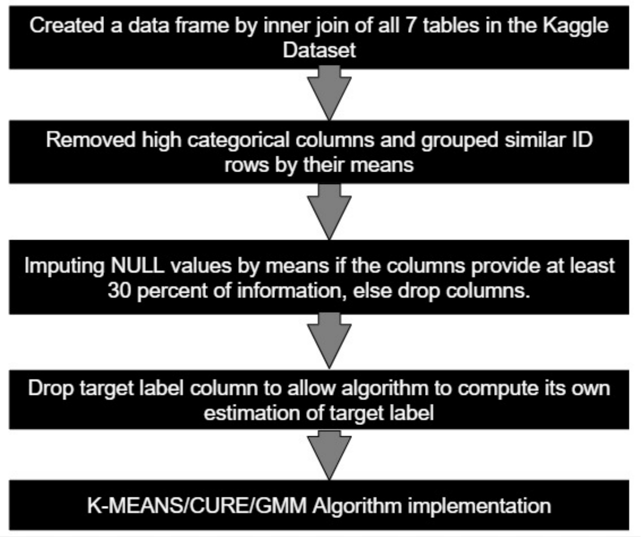
Created a data frame by inner join of all 7 tables in the Kaggle Dataset
Removed high categorical columns and grouped similar ID rows by their means
Imputing NULL values by means if the columns provide at least 30 percent of information, else drop columns.
Drop target label column to allow algorithm to compute its own estimation of target label
K-MEANS/CURE/GMM Algorithm implementation

Figure 1. Steps taken for preprocessing

5.1 Pre-processing and Data Cleaning

The raw data obtained from Kaggle dataset of Predicting Home Credit Default Risk contains about 7 files providing details about a persons current applications, previous credit history, loan installments, details of bureau they borrowed the loans, etc. The first step was to inner join all these files together by the foreign keys as specified by the company itself. Next, since the data contains categorical attributes which is not in Euclidean dimensions, we convert the attributes with just two categories to unique numbers, otherwise we drop those attributes. Also, as a person could have has many loan applications, to make the primary key unique, we group those rows together by taking averages of them. We find that most attributes have a lot of rows with Null values. We remove those features that provide us less than 30 percent of the data. That is, if 70 percent of the values for a feature is Null then we drop those attributes else we impute the null values with the mean of the columns. Finally, after the preprocessing is completed we have a dataset that contains 100 columns and 0.25 million rows.

In this paper, we implement 3 algorithms to find clustering. The first technique used is the K-means clustering that allows for hard margin grouping of data points. Secondly, we use Gaussian Mixture Models that allows for soft margin clustering giving a probability for a data point to occur in more than one cluster. Finally, we implement the CURE algorithm to handle large data by using hierarchical clustering on a sample of data and then cluster assignment to each data point. For each of these algorithms, we try to find exactly 2 clusters to group people into whether providing loans to them is risky or safe.

5.2 K-means Clustering

The K-means clustering algorithm [8] is an unsupervised learning technique that aims to group data points into exactly k clusters such that each observation belongs to the group which has the closest mean to it.

The k-means algorithm calculates distance of a point to every other point and merges the ones with smallest distance to a group. The cluster is then represented its centroids which is the mean of the data points in the cluster. Each data point x is assigned to a cluster based on:

$$\operatorname{argmin}_{c_i \in C} dist(c_i, x)^2 \tag{1}$$

where dist(Âů) is usually the Euclidean distance. The centroids for the cluster are further calculated as:

$$c_i = \frac{1}{|S_i|} \sum_{x_i \in S_i} x_i \tag{2}$$

Since the data set is huge and k-means requires to compute Euclidean distance between every pair of data points, we use a scalable system of Apache Spark to implement the K-means algorithm and implement clustering. We evaluated K-Means by varying the hyper parameters. We supply the number of clusters, K and the converging distance. We have set K=2, as our intention was to segregate users into two groups of whether they can repay the loan or not. We experimented with different values of converging distance to get the best clustering analysis. The running time for K-Means in Spark is about 3 minutes.

However, clustering faces few challenges of mis-grouping of data points due to hard assignment and it only works for spherical clusters and not for complex shapes or structures.

5.3 Gaussian Mixture Models

In order to overcome the drawback of K-means clustering that provides a hard margin to the data points to be exactly in one clusters, we used Gaussian Mixture Models.This model provides a soft margin to each data point. This means that each data points receives a probability of belonging to each of the clusters. This helped us to comprehend how much safe or risky is it to provide loan to a particular person. The Gaussian Mixture Model also helps to create complex cluster structures instead of only spherical ones.

The Gaussian Mixture Model [2] follows the Expectation-Maximization technique where the probability of some point x belonging to the a cluster is given by :

$$P(C_k|x_i) = \frac{\phi_k N(x_i|\mu_k, \sigma_k)}{\sum_{j=1}^{K} \phi_k N(x_i|\mu_j, \sigma_j)} \tag{3}$$

where

μ, σ, ϕ are the means, standard deviation and weight terms.

These terms are then updated in every iteration until convergence. We implemented Gaussian Mixture Models using the Python Scikit-learn library to generate a probability distribution over the data points and then set a threshold for which it would be safe for a person to get the loan. The threshold was set as per the evaluation of clusters through hit-and-trial method.

5.4 Clustering Using Representatives

The final data set that we obtained was huge about 0.25 million points and computing Euclidean distances of every pair of points takes a long time. Though the scalable approach of K-means through Spark was faster, we implemented another approach of CURE algorithm to understand the time complexity of the algorithm.

We implemented the CURE[6] algorithm by taking a sample of 0.2 of the entire dataset. This gave about 50,000 data points. Hierarchical clustering[7] with single linkage given by equation below was applied to this sample to get exactly 2 clusters.

$$\min \{\, d(a,b) : a \in A,\ b \in B \,\}. \tag{4}$$

Representative points for each clusters were then calculated and the entire data was assigned a cluster through these representative points.

The CURE algorithm helped to deal with the data as the entire data was not completely loadable to the disk.

6 Applications

Unsupervised machine learning [5] allow us to uncover previously unknown patterns in data, but most of the time these patterns are unsatisfactory approximations of what supervised machine learning can achieve. Additionally, we are dealing with unsupervised algorithms and thus there is no way to determine how accurate they are, making supervised machine learning more applicable to real-world problems.

So, we realize that the best time to use unsupervised machine learning is when one does not have data on desired outcomes, like determining a target market for an entirely new product that your business has never sold before. Supervised learning is optimal when we know our target labels.

Some applications of unsupervised machine learning techniques include:

Clustering [13] allows to automatically split the dataset into groups according to similarity. Overestimation of similarity is a problem with unsupervised algorithms. For this reason, cluster analysis is a poor choice for applications like customer segmentation and targeting. Some widely used applications include anomaly detection, where one can automatically discover unusual data points in the dataset. Especially useful in pinpointing fraudulent transactions, discovering faulty pieces of hardware, or identifying an outlier caused by a human error during data entry.

Association mining like recommendation engines sets of items that frequently occur together in the dataset. Widely used by retailers for basket analysis, because it allows analysts to discover shopping patterns over time and develop more effective marketing and merchandising strategies.

Apart from that Clustering is popular in fields of Biology, computational biology and bioinformatics, medicine, World Wide Web, Finance, Crime detection and several others.

Unsupervised Machine Learning algorithms were the basis for our data set -Home Credit Default Risk. K-Means Clustering, Gaussian Mixture Models and CURE are the unsupervised machine learning algorithms we implemented.

6.1 Applications of K-Means

Segmentation: E-retailers or e-commerce companies [12] are the focal points in the retail industry. They offer luring offers and discounts and the ease of the access is their highest selling point. They aim to move from discount led to convenience or differentiation led offering over a period in time. One of the large retailer wanted to segment the customers based on customer spend patterns and understand price sensitivity of the customers. K-Means is the widely used clustering algorithm for segmentation. Some of the segmentation variables considered are âĂŞ total spend, value of discounts, percent discounts across transactions, number of items bought on discounts etc and used k-means clustering to find the discount orientation of the customers. Thus, proving to be a comprehensive solution.

Document Classification: Cluster documents [14] in multiple categories based on tags, topics, and the content of the document. This is a very standard classification problem and k-means is a highly suitable algorithm for this purpose. The initial processing of the documents is needed to represent each document as a vector and uses term frequency to identify commonly used terms that help classify the document. The document vectors are then clustered to help identify similarity in document groups. The inverse document frequency and term frequency together give a good measure of clustering for documents.

Cyber-Profiling Criminals: Cyber-profiling [17] is the process of collecting data from individuals and groups to identify their involvement in crimes. The idea of cyber profiling is derived from criminal profiles, which provide information on the investigation division, thus proving to be a classic K-Means problem to classify the types of criminals who were at the crime scene.

6.2 Applications of Clustering Using Representatives

Finding Patterns in Large Data sets: Big Data is a challenge for K-Means. CURE is a efficient algorithm in dealing with large data size. The increasing enrolment of students at any University is one Big Data challenge which equates to increase of students database which can be mined to discover patterns in large data sets. Patterns extracted can be converted to understandable information that can be useful to the organization.

Gene Expression Data[11]: Gene expression data hides vital information required to understand the biological process that takes place in a particular organism in relation to its environment. Deciphering the hidden patterns in gene expression data offers with the preference to strengthen the understanding of functional genomics. The millions of data comprising the complexity of biological networks and the volume of genes present increase the challenges of comprehending and interpretation of the resulting mass of data. Such data inhibits vagueness, imprecision, and noise. Therefore, the use of CURE is a first step toward addressing these challenges, which is essential in the data mining process to reveal natural preference and identify interesting patterns in the underlying data. CURE proves to be an optimal choice for clustering of gene expression data in making known the natural structure evident in gene expression data, fine tuning the gene functions, cellular processes, and subtypes of cells, mining useful information from noisy data, and understanding gene regulation.

6.3 Applications of Gaussian Mixture Model

Anomaly Detection [1]: To prevent hard clustering we turn to Gaussian Mixture Model to get a softer analysis from results. Image processing applications amount to the detection of small objects (anomalies) that are different from the dominating background texture and clutter. For example, in landscaped/Portrait mode of aerial photo interpretation, one may want to detect small man-made objects from fields and vegetation. Additionally, in manufacturing quality inspection, one may want to detect small defects, outliers in fabric and paper. One example of anomaly detection, in which the object (anomaly) of interest is a parachute in the middle of the image in the presence of background texture and clutter. The detection results are presented as a prediction error image (anomaly) and a spatial average of the squared prediction error (smoothed anomaly). Bright spots in the latter image correspond to likely anomalies. The results produces better detection and fewer false alarms, illustrate the efficacy of the Gaussian mixture model.

Tracking Multiple Moving Objects Using Gaussian Mixture Model: For object recognition, navigation systems and surveillance systems, object tracking is an indisputable first-step. The conventional approach to object tracking is based on the difference between the current image and the background image, proving to be very crude. The algorithms based on the difference image are useful in extracting the moving objects from the image and track them in consecutive frames. GMM when implemented would be done in three stages of color extraction, foreground detection using Gaussian Mixture Model and object tracking. Initially color extraction is done to extract the required color from a particular picture frame, after that color extraction the moving objects present in the foreground are detected using Gaussian Mixture Model analysis is applied on consecutive frames of video sequence and giving us feedback on moving objects. Thus allowing us to track multiple moving objects.

7 Evaluation of Results

We ran three unsupervised clustering algorithms: K-Means in Spark, Clustering Using Representatives(CURE) using Jupyter Notebook and Gaussian Mixture Models in Python.

The intend of clustering via three different algorithms was to get the best evaluation of results when compared with the gold standard of target label. For the purpose of clustering we removed the target label from our data set.

Original size of our data was around 688 MB. This huge size of data promoted us to use Spark. After data cleaning and pre-processing we reduced the data set to contain two hundred thousand rows and 99 columns. For evaluation purposes for each of the three algorithms we are printing a classification report along with a 3D scatter plot by importing matplotlib library.

7.1 K-Means

K-Means[16] algorithm is very popular and used in a variety of applications such as market segmentation, document clustering, image segmentation and image compression, etc. The goal usually when we undergo a cluster analysis is either:Get a meaningful intuition of the structure of the data weâĂŹre dealing with. An example of this goal is prediction of whether the user is a credit defaulter or not via clustering the data into two clusters. Another goal is Cluster-then-predict where different models will be built for different subgroups if we believe there is a wide variation in the behaviors of different subgroups.

7.2 Clustering Using Representatives

We implemented CURE in python using Jupyter Notebook. The time complexity for assigning representative points for the entire data set was very high. Thus, we choose to take a fraction of our original data set for CURE. We trained the model using a sample of 0.1 fraction of our data. We set the hyper parameters for hierarchical clustering, K=2, alpha was set to 0.1 and number of representatives were set to 4.

Table 1. Classification Report for K-means

Label	Precision	Recall	f1-score	support
0	0.92	0.48	0.63	226890
1	0.08	0.52	0.13	19212
micro avg	0.48	0.48	0.48	246103
macro avg	0.50	0.50	0.38	246103
weighted avg	0.86	0.48	0.59	246103

Table 2. Classification Report for Clustering Using Representatives

Label	Precision	Recall	f1-score	support
0	0.92	0.97	0.94	22642
1	0.08	0.03	0.05	1968
micro avg	0.89	0.89	0.89	24610
macro avg	0.50	0.50	0.49	24610
weighted avg	0.85	0.89	0.87	24610

Table 3. Classification Report for Gaussian Mixture Model

Label	Precision	Recall	f1-score	support
0	0.93	0.60	0.73	226891
1	0.09	0.46	0.15	19212
micro avg	0.59	0.59	246103	
macro avg	0.53	0.44	246102	
weighted avg	0.86	0.59	0.68	246103

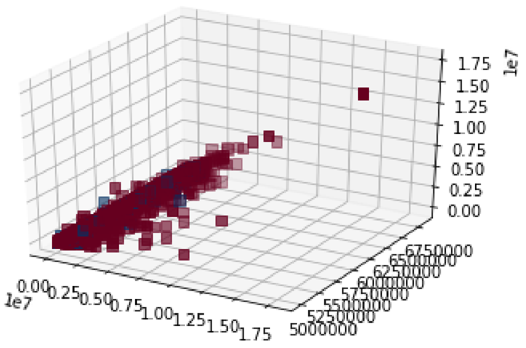

Figure 2. 3D visualization of the original data without Clustering.

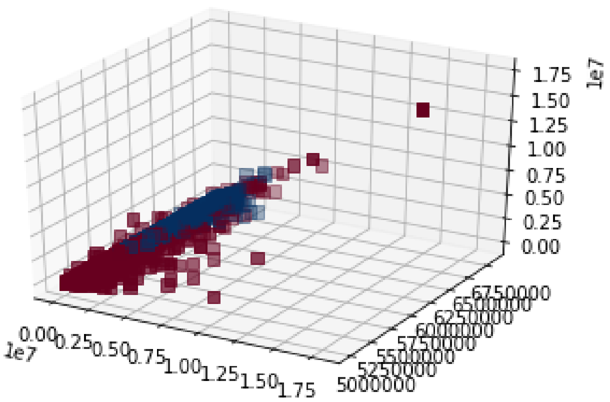

Figure 3. 3D Clustering visualization using K-Means Clustering

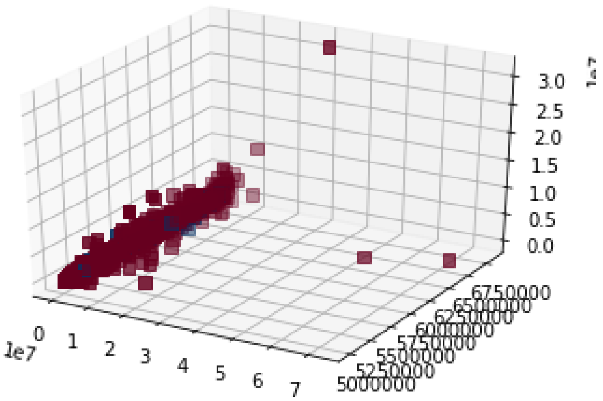

Figure 4. 3D Clustering visualization using CURE Clustering.

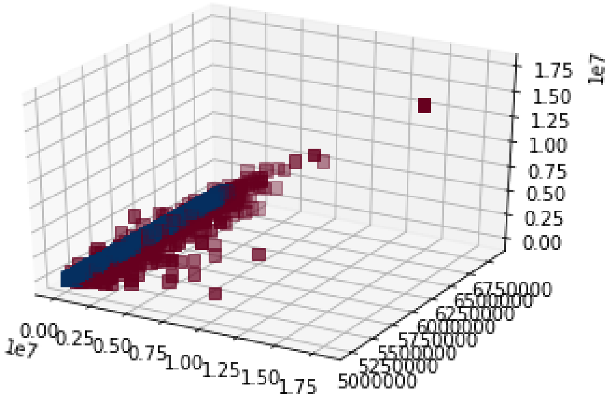

Figure 5. 3D Clustering visualization using Gaussian Mixture Model Clustering.

7.3 Gaussian Mixture Model

A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. One can think of mixture models as generalizing k-means clustering to incorporate information about the covariance structure of the data as well as the centers of the latent Gaussians.

We set the hyper parameters for GMM with the number of mixture components to 2, covariance type to tied so that each component has its own general covariance matrix, regularization to defaults of 1e-6 such that non-negative regularization is added to the diagonal of covariance. Allows to assure that the covariance matrices are all positive.

8 Visualizations

This section details the three dimensional plots we used to visualize the clusters obtained from clustering. For distinction of the two clusters we used red and blue colour. One colour represents the user group who are loan defaulters and the other are user groups who can repay the loan.
We demonstrate plots for all the three algorithms and also for the original dataset to compare the differences between them. The axis of the plots are the top three columns with highest variance. The high variance of a column relates to the fact that the column provides maximum information in the dataset and therefore we select those columns only for visualization purposes.

9 Comparison of Results for all Algorithms

All the three models are substantially close to each other in terms of their results. However, due to class imbalance [10] in the original data set the detection rates of 1 as compared to 0 is less. Thus, presenting us a overall reduction in the precision and recall scores. The analysis of the three algorithms gave us satisfactory results. The results for Clustering Using Representatives were surpassed by K-Means and Gaussian Mixture Models. Reason for CUREs inefficiency could be attributed to the complexity of the size of data set for performing hierarchical clustering. The time complexity of assigning clusters via euclidean distance resulted was very high thus we chose to choose a fraction of the original data set. Eventually resulting in better performances by K-Means and Gaussian Mixture Model as compared to CURE.

This is a two dimensional figure of precision, recall and f1-scores of Gaussian Mixture Model, CURE and K-Means. We observe that for clustering algorithms K-Means and Gaussian Mixture Model, the distribution is very close. CURE has a distinct variation in its values when compared to CURE and Gaussian Mixture Model.

10 Concluding Remarks

We attempted to use K-Means, Gaussian Mixture Model and Clustering Using Representatives clustering to predict a customer‘s loan default risk. Whether the customer is capable of repaying the loan sanctioned to them. Our algorithms attempted to calculate target labels to see how efficient the unsupervised clustering results are when compared to the gold standard of the labeled data. Due to class imbalanced nature of our data, we saw stark variation in the precision, recall and f1-score values for the two target values 0 and 1.

After comparing with given labels we concluded that: Gaussian Mixture Models gave us the highest precision and recall. K-Means fairing a close second followed by CURE.

Table 4. Comparison of Classification Report for all the algorithms

Algorithm type	Avg. Precision	Avg. Recall	Avg. f1-score
CURE	0.47	0.48	0.485
Gaussian M.M.	0.515	0.52	0.52
K-Means	0.5	0.49	0.495

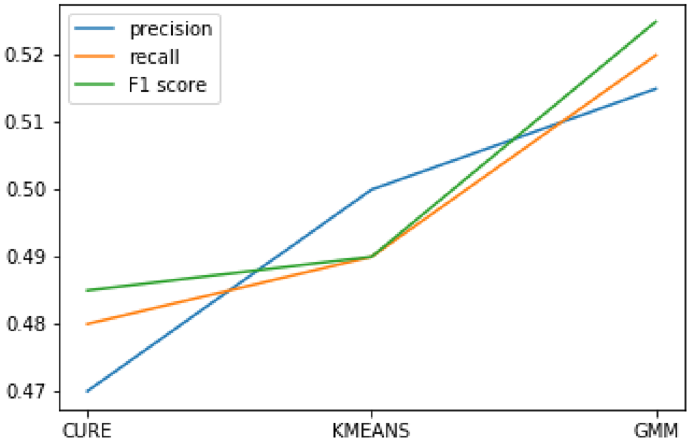

Figure 6. 2D plot comparing f1-scores of CURE,K-Means and GMM

We observe that for clustering algorithms K-Means and Gaussian Mixture Model, the distribution is close.

Having an ensemble of clustering algorithms helped us to expose the data to varying margins of clustering. We implemented hard clustering and soft clustering and saw the analysis for each. For our data set, soft clustering lead to more optimal results.

Clustering via K-Means, Gaussian Mixture Model and Clustering Using Representatives is one of the possible framework that can be used in the finance and banking industry for risk evaluation.

Acknowledgments

We would like to thank Prof. Wensheng Wu for giving us the opportunity to work on this project and guiding us through it. We would also like to thank the teaching instructors for their support.

References

[1] [n. d.]. Gaussian Mixture Model. https://www.sciencedirect.com/topics/computer-science/gaussian-mixture-model

[2] Jeffrey D Banfield and Adrian E Raftery. 1993. Model-based Gaussian and non-Gaussian clustering. *Biometrics* (1993), 803–821.

[3] Christine Cumming and Beverly Hirtle. 2001. The challenges of risk management in diversified financial companies. *Economic Policy Review* 7, 1 (2001).

[4] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. 1996. From data mining to knowledge discovery in databases. *AI magazine* 17, 3 (1996), 37–37.

[5] Nizar Grira, Michel Crucianu, and Nozha Boujemaa. 2004. Unsupervised and semi-supervised clustering: a brief survey. *A review of machine learning techniques for processing multimedia content* 1 (2004), 9–16.

[6] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. 1998. CURE: an efficient clustering algorithm for large databases. In *ACM Sigmod Record*, Vol. 27. ACM, 73–84.

[7] Stephen C Johnson. 1967. Hierarchical clustering schemes. *Psychometrika* 32, 3 (1967), 241–254.

[8] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. 2002. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 7 (2002), 881–892.

[9] K Kavitha. 2016. Clustering loan applicants based on risk percentage using K-means clustering techniques. *International Journal of Advanced Research in Computer Science and Software Engineering* 6, 2 (2016), 162–166.

[10] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. 2009. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39, 2 (2009), 539–550.

[11] Jelili Oyelade, Itunuoluwa Isewon, Funke Oladipupo, Olufemi Aromolaran, Efosa Uwoghiren, Faridah Ameh, Moses Achas, and Ezekiel Adebiyi. 2016. Clustering algorithms: Their application to gene expression data. *Bioinformatics and Biology insights* 10 (2016), BBI–S38316.

[12] Siddheswar Ray and Rose H Turi. 1999. Determination of number of clusters in k-means clustering and application in colour image segmentation. In *Proceedings of the 4th international conference on advances in pattern recognition and digital techniques*. Calcutta, India, 137–143.

[13] D Hari Hara Santosh, Poornesh Venkatesh, P Poornesh, L Narayana Rao, and N Arun Kumar. 2013. Tracking multiple moving objects using gaussian mixture model. *International Journal of Soft Computing and Engineering (IJSCE)* 3, 2 (2013), 114–119.

[14] Vivek Kumar Singh, Nisha Tiwari, and Shekhar Garg. 2011. Document clustering using k-means, heuristic k-means and fuzzy c-means. In *2011 International Conference on Computational Intelligence and Communication Networks*. IEEE, 297–301.

[15] Michael Steinbach, Levent Ertöz, and Vipin Kumar. 2004. The challenges of clustering high dimensional data. In *New directions in statistical physics*. Springer, 273–309.

[16] Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrödl, et al. 2001. Constrained k-means clustering with background knowledge. In *Icml*, Vol. 1. 577–584.

[17] Prayudi Zulfadhilah, Yudi Prayudi, and Imam Riadi. 2016. Cyber Profiling using Log Analysis and K-Means Clustering. *International Journal of Advanced Computer Science and Applications* 7, 7 (2016).