Identifying Outliers using Spectral Clustering

Vinayak Nambiar
AI22MTECH13005
Department of Artificial Intelligence
Indian Institute of Technology, Hyderabad
ai22mtech13005@iith.ac.in

Roshin Roy
AI22MTECH13006
Department of Artificial Intelligence
Indian Institute of Technology, Hyderabad
ai22mtech13006@iith.ac.in

Sarvesh Kumar Purohit
AI22MTECH14006
Department of Artificial Intelligence
Indian Institute of Technology, Hyderabad
ai22mtech14006@iith.ac.in

Parth Nitesh Thakkar
CS22MTECH14005
Department of CSE
Indian Institute of Technology, Hyderabad
cs22mtech14005@iith.ac.in

Rahul Vigneswaran K
CS23MTECH02002
Department of CSE
Indian Institute of Technology, Hyderabad
cs23mtech02002@iith.ac.in

Rithik Agarwal
CS22MTECH11004
Department of CSE
Indian Institute of Technology, Hyderabad
cs22mtech11004@iith.ac.in

Code is available at Google Collab Link

Abstract

Spectral clustering is a popular unsupervised learning algorithm used for clustering and partitioning data points. This paper presents a spectral clustering algorithm applied to a tax dataset containing seven features and three ratio variables. We use the eigenvectors of a similarity matrix derived from the dataset to perform the clustering. We show that the algorithm can effectively partition the data points into meaningful clusters. Our results demonstrate the potential of spectral clustering as a tool for analyzing datasets with complex structures.

1. Problem Statement

Tax fraud is a serious issue that causes financial losses for governments and taxpayers worldwide. Traditional methods for detecting tax fraud are inadequate due to the growing complexity and volume of tax data. Machine learning algorithms have shown promise in detecting fraudulent tax activities by identifying patterns and anomalies in tax data [2]. In the context of tax fraud detection, spectral clustering can be used to identify groups of taxpayers with similar tax behaviours or patterns, potentially uncovering instances of fraudulent activity [3]. Other machine learning algorithms, such as logistic regression, decision trees, and neural networks, have limitations [4] in handling complex, non-linear, high-dimensional data and require large amounts of training data to perform well. Logistic regression assumes a linear relationship between the dependent variable and independent variables, making it less suitable for complex, non-linear data structures. Decision trees can be limited in their ability to handle complex, high-dimensional data. They may also suffer from overfitting or underfitting. Neural networks require large amounts of training data to perform well which can be challenging to obtain in the context of tax fraud detection. Tax fraud is a relatively rare occurrence, and labelled fraud data may be limited or non-existent, making it challenging to train a neural network effectively [2].

Spectral clustering, on the other hand, can handle non-linear and complex data structures [1] while also detecting and handling outliers [6], making it more suitable for tax data analysis. It is also an unsupervised learning algorithm, making it flexible and efficient for detecting unknown or emerging fraud patterns. This study aims to investigate the effectiveness of spectral clustering in detecting tax fraud and identify relevant tax variables for analysis.

2. Description of the Dataset

The dataset consists of 1199 data-points with features extracted from a larger dataset. The dataset contains seven features (cov1 to cov7) and three ratio variables (sal_pur_rat, igst_itc_tot_itc_rat, and lib_igst_itc_rat).-

- cov1 to cov7 represent seven different features or variables that have been extracted from the original data.
- sal_pur_rat represents the ratio of sales to purchases. The ratio of sales to purchases is a financial metric that compares the amount of revenue generated by a company through the sale of its goods or services to the cost of the goods or services purchased to generate that revenue. The ratio can be used to evaluate a company's profitability, efficiency, and overall financial health. A high ratio indicates that a company is generating a lot of revenue relative to its costs, while a low ratio suggests that a company may be struggling to generate revenue or is operating with inefficient processes.
- igst_itc_tot_itc_rat represents the ratio of the Integrated Goods and Services Tax (IGST) Input Tax Credit (ITC) to the Total Input Tax Credit (ITC). The ratio is a measure of the proportion of input tax credit claimed on IGST compared to the total input tax credit claimed. This ratio indicates the extent to which IGST, which is applicable to inter-state supplies, has been used to set off the taxpayer's tax liability. If the ratio is high, it suggests that the taxpayer has a large volume of inter-state supplies and is effectively using IGST to set off their tax liability. If the ratio is low, it suggests that the taxpayer is primarily engaged in intra-state supplies, and may have lower inter-state transactions.
- lib_igst_itc_rat represents the IGST Input Tax Credit (ITC) ratio to the Total Input Tax Credit (ITC). The ratio indicates the proportion of input tax credit claimed by a taxpayer for IGST paid on inter-state transactions as compared to the total input tax credit claimed on all purchases. A higher ratio indicates that a larger proportion of the input tax credit claimed by the taxpayer is for IGST paid on inter-state transactions, which could be an indicator of the nature of their business or operations.

3. Algorithm Used

We use the eigenvectors of a similarity matrix derived from the dataset to perform the clustering. We show that the algorithm can effectively partition the data points into meaningful clusters. We also perform a sensitivity analysis to investigate the effects of varying algorithm parameters.

The similarity matrix is generated using a radial kernel (Trials were also conducted using a Laplacian kernel and the same results were obtained). The similarity matrix contains the pairwise similarities between all data points in the dataset, where the similarity between two data points is computed based on their distance in the feature space. The matrix thus generated is made sparse using a technique mentioned in [7]. This paper suggests a methodology to reduce the computation time complexity by making the adjacency matrix sparse by limiting the number of neighbourhoods to a set of nodes with high similarity values by using a threshold to neglect low-similarity neighbourhoods. This threshold according to the paper lies in the interval $[\mu - \sigma, \mu)$.

This will isolate the outliers and construct a sparse matrix, thus making our computations easier. We then create a graph from the adjacency matrix thus obtained. Then generate a degree matrix and form a normalized graph Laplacian matrix from it which is defined as:

$$L_{\text{norm}} = I - D^{-\frac{1}{2}}.A.D^{-\frac{1}{2}}$$

where A is the adjacency matrix and D is the diagonal degree matrix [5].

We then compute the smallest 'k' eigenvalues and corresponding eigenvectors of the normalized Laplacian matrix. We have considered the first 15 eigenvalues for our paper. These eigenvalues represent the "spectrum" of the normalized Laplacian matrix and can be used to analyze the properties of the graph, such as the number of connected components or the clustering behaviour [6]. In this case, the first two eigenvalues are zero, which indicates that the graph is disconnected (has multiple connected components). The remaining eigenvalues can be used to determine the number of clusters in your data. Refer to **Figure 1** for the plot of k-values vs eigenvectors.

Based on the results observed from the plot, there are several 'elbows' in the eigenvalue spectrum, which could potentially indicate the presence of different numbers of clusters. One common approach is to look for a 'gap' in the eigenvalues, which can be used to estimate the number of clusters. Based on the relatively large difference between the first and second eigenvalues, it suggests that there are two clusters in the data.

It's worth noting that the choice of the number of clusters is subjective and depends on the context and goals of the analysis [5]. We will further experiment with different numbers of clusters and evaluate their performance based on the Mean Squared Error for different numbers of clusters. The no. of clusters resulting in the lowest MSE is chosen to be the optimum number of clusters. In our experiments, we observed that the minimum value of MSE was obtained for two clusters. The results are placed at.

The algorithm that summarizes all the steps described above are mentioned below in a structured way.

Algorithm 1 Spectral Clustering Algorithm

```
Input: CSV file containing dataset
Output: Optimum number of clusters
Create similarity matrix (S) using rbf kernel
Find $\mu$ and $\sigma$ for S
Select a threshold in the interval $[\mu - \sigma, \mu)$
Initialize Adjacency Matrix (A)
for all $e \in S$ do
    if $S$[e] $> threshold$ then
        Set $A$[e] = 1
    else
        Set $A$[e] = 0
    end if
end for
Compute Degree Matrix (D) from Adjacency Matrix (A)
Compute normalised Laplacian (L) : $I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$
Compute eigen values and eigen vectors of L and plot the first 15 smallest eigen values
Identify first large spectral gap from plot of elbow points
Compute Mean square error using kMeans algorithm for different number of clusters.
Identify the number of clusters that resulted in the least MSE
Return optimum number of clusters
```

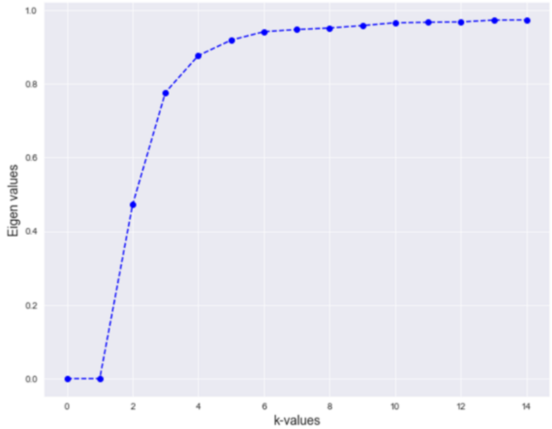

Figure 1. **k-values vs eigenvectors.** Observe that the largest gap difference is between first and second eigenvalues.

Table 1. Performance Analysis of Clusters Based on Mean Squared Error

No of Clusters	Mean Squared Error
2 Clusters	**0.003411**
3 Clusters	0.196929
4 Clusters	0.250881
5 Clusters	0.821733
6 Clusters	1.265832
7 Clusters	1.677417

4. Results

Typically, the clusters represent the normal behaviour or patterns, and any data points outside the clusters or far from the centroid of the cluster could be considered potential outliers or anomalous behaviour. In the context of tax data, this could mean that most of the tax returns fall within the clusters and exhibit normal tax behaviour, while any returns that lie outside the clusters may indicate potentially fraudulent activity or atypical tax behaviour. Our results demonstrate the potential of spectral clustering as a tool for analyzing datasets with complex structures.

References

[1] Michael Jordan Andrew Ng and Yair Weiss. On spectral clustering: Analysis and an algorithm. *NIPS'01: Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, page 849–856, 2001. 1

[2] María Jesús Delgado Rodríguez César Pérez López and Sonia de Lucas Santos. Article on tax fraud detection through neural networks: An application using a sample of personal income taxpayers. *Special Issue of Future Intelligent Systems and Networks 2019; Future Internet 2019, 11(4), 86*, page 2272–2279, 2019. 1

[3] Andrés Moreno Mariadel Pilar Villamil Daniel de Roux, Boris Pérez and César Figueroa. Tax fraud detection for under-reporting declarations using an unsupervised machine learning approach. *KDD '18: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, page 215–222, Jul, 2018. 1

[4] Asri Ngadi Salima Omar and Hamid H. Jebur. Machine learning techniques for anomaly detection: An overview. *International Journal of Computer Applications, Volume 79 – No.2*, page 0975 – 8887, Oct, 2013. 1

[5] Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing 17(4), arXiv:0711.0189*, 2007. 2

[6] Yuan Wang, Xiaochun Wang, and Xia Wang. *A Spectral Clustering Based Outlier Detection Technique*, volume 9729, pages 15–27. 01 2016. 1, 2

[7] Peng Yang and Biao Huang. A spectral clustering algorithm for outlier detection. In *2008 International Seminar on Future*

Information Technology and Management Engineering, pages 33–36, 2008. 2