Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Unsupervised Machine Learning Models for Focused Target Customer Segmentation

Blessington Naveen Palaparthi [1], Dr. S. Akthar [2]
[1] *Department of Computer Science and Engineering, Acharya Nagarjuna University, Guntur, India. Email: blessingtonp@outlook.com*
[2] *Department of Computer Science, Government College for Women (A), Guntur, India. Email: shahedaakthar76@gmail.com*

ARTICLE INFO

Received: 14 Dec 2024
Revised: 17 Feb 2025
Accepted: 28 Feb 2025

ABSTRACT

This research investigates novel methods for customer segmentation using advanced machine learning models. As digital marketing rapidly expands, accurately identifying target segments is crucial for optimizing campaign reach and enhancing return on investment (ROI). The study utilizes various clustering algorithms to identify the most effective approach. Essential processes include processing and preparing customer data, followed by applying widely-used machine learning algorithms.

Keywords: Digital marketing, Marketing automation, Customer segmentation, clustering, machine learning, artificial intelligence, K-means, DBSCAN, Affinity Propagation, BIRCH, Bisecting K-Means, digital campaign strategy optimization, smart customer segmentation.

INTRODUCTION

Digital marketing is a rapidly evolving landscape for modern businesses, presenting a constant challenge to stay ahead of the competition. Both outbound campaigns and inbound user responsesgenerate large volumes of data quickly. While faster hardware can facilitate rapid data processing, intelligent systems are necessary to make the data more meaningful and actionable. Businesses are progressively leveraging Artificial Intelligence (AI) and Machine Learning (ML) techniques to enhance marketing strategies, such as customer segmentation, personalization, campaign planning, predictive analytics, and performance evaluation. Among these, customer segmentation serves as the crucial starting point.

Customer segmentation enables businesses to pinpoint the right focus groups for their campaigns. Meeting customers' specific needs increases their likelihood of engaging and developing into loyal, long-term partners. Effectively attracting and retaining ideal clients, who regularly engage with a company's products or services, relies heavily on a deep understanding of their purchasing habits.

Behera et al. [1] emphasized that for companies to succeed in the digital landscape, they must create strategies that offer targeted and measurable methods for engaging customers through digital marketing. They introduced a model that provides real-time, personalized marketing insights to recommend products for both online and offline customers using various sales strategies. This model can also define e-markets by clustering items, customers, and unique selling propositions, while collecting, storing, and processing transaction data to display personalized marketing information that aids customer decision-making.

In their published work, Bhagat et al. [2] concluded that AI-driven customer segmentation is transforming modern marketing. By transcending traditional demographic categories, AI uncovers intricate customer insights. This technology allows businesses to detect complex patterns, behaviors, and preferences, facilitating highly personalized marketing strategies. Such a profound understanding fosters stronger connections and more effective engagement with individual customers.

Dullaghan et al. [3] highlighted the potential of machine learning applications in guiding marketing strategies. They acknowledged the role of data mining techniques in customer segmentation and employed naïve Bayesian modeling to segment telecommunication customers based on their billing and socio-demographic characteristics. Their

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

findings emphasized the critical importance of billing and usage features in customer segmentation. However, the decision tree algorithm yielded different results and feature preferences compared to the Bayesian model.

Alsac et al. [4] utilized the Random Forest (RF), Gaussian Naïve Bayes (GaussianNB), k-Nearest Neighbor (kNN), and Logistic Regression (LogReg) classifiers on a dataset of 64,000 marketing emails. The analysis showed that the precision and recall metrics are similar for all the three algorithms RF, GaussianNB, and kNN along with the F1 score indicating the success of email marketing campaigns. Conversely, LogReg demonstrated low performance in terms of recall and f1-score.

Vijilesh et al. [5] introduced a customer segmentation approach that utilizes the K-means algorithm to divide an establishment's clientele into segments based on their characteristics and attributes. This concept aims to help B2C companies secure a competitive advantage by developing distinctive products and services while efficiently targeting potential customers.

Monil et al. [6] explored the capability of hybrid clustering algorithms to outperform individual models in terms of performance. They assessed K-means, DBSCAN, and Affinity Propagation algorithms, concluding that each has its own strengths and weaknesses. They suggested that a hybrid approach, combining these algorithms, could be advantageous depending on the specific situation and requirements, allowing for a more tailored strategy.

Jing et al. [7] observed that the DBSCAN partitioning method was overly simplistic. They also observed that the GETNEIGHBORS query providedby Spark accesses the datarepetitively. In an attempt to address these issues, they proposed DBSCAN-PSM, which incorporates new data partitioning and merging techniques. Experimental results showed that this novel method improves both parallel efficiency and the performance of the clustering algorithm.

Maureen et al. [8] utilized marketing analysis tools, such as the RFM model, to study customer segmentation. Their findings indicated that this model facilitates customer segmentation across multiple parameters, enabling the identification of high-value customers based on factors such as purchase behavior. By combining a decision tree classifier with a gradient-boosting classification approach, they addressed the classification challenge and discovered that the gradient-boosting classifier outperformed the others. Besides contributing to the application of these models in digital marketing strategies their research also advanced the predictive modeling in customer behavior.

Vamsi Katragadda [9] conducted a study on applying machine learning techniques for dynamic customer segmentation. In the research processboth K-means and DBSCAN algorithms are used and the results suggested strong adaptability to real-time data inputs, facilitating timely adjustments in marketing strategies and personalized customer interactions. Recommendations for future research involve examining additional machine learning methods, assessing the long-term impact of dynamic segmentation on customer loyalty, and addressing ethical concerns in data-driven marketing strategies.

The goal of this research is to implement both novel and popular clustering algorithms. The study aims to systematically compare the results to identify the best-performing algorithms using datasets from Kaggle.

MACHINELEARNING MODELS FOR CUSTOMER SEGMENTATION

There arethree different approaches in Machine learning based on the data to be analyzed and processed.

Supervised Learning:

A supervised learning model is employed when the source data is labeled, with eachinputdata point having a clear and specific output. Algorithms utilize labeled data to recognize patterns and assign appropriate labels to previously unlabeled data points.Supervised learning is broadly categorized into two types:

- ***Classification***-predicts the category to which a specific data element belongs such as sentiment analysis, churn predictions, spam detection and similar tasks.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

- ***Regression***–predicts a numerical value by analyzing past data. For instance, it can predict the clickthrough rate of a campaign, stock prices, house prices, and more.

Semi-Supervised Learning:

A semi-supervised learning model is applied when the source data is partially labeled, especially in cases where the dataset is extensive and labeling every data point is costly. To train models for tasks like classification and regression, a blend of supervised and unsupervised learning techniques can be used to processsuch partially labeled data. While semi-supervised learning is typically used in the same scenarios as supervised learning methods, it stands out by incorporating various techniques that integrate unlabeled data into the model training process, alongside the labeled data that conventional supervised learning relies on.

Unsupervised Learning:

When the data is entirely unlabeled, unsupervised learning models are employed allowing the algorithm to uncover hidden patterns within the dataset. With the ability to discover similarities and differences in the data, unsupervised learning models are aperfectchoice for exploratory data analysis, cross-selling strategies, customer segmentation and similar cases. These modes are used for three main tasksas explained below:

- ***Clustering***- Clustering is a data mining technique that organizes unlabeled data based on their similarities or differences. Clustering algorithms group raw, unclassified data objects, revealing underlying structures or patterns within the dataset.
- ***Association***- An association task is a rule-based method used to identify relationships between variables within each dataset. These techniques are often used for market basket analysis, allowing companies to gain deeper insights into the connections between various products.
- ***Dimensionality reduction*** - Dimensionality reduction is a method applied when a dataset contains an excessively large number of features or dimensions. Although additional data can enhance accuracy, it may also hinder the performance of machine learning algorithms (e.g., by causing overfitting) and complicate the visualization of the dataset. This technique is often applied during the data preprocessing stage, as it reduces the number of data inputs to a manageable size while maintaining as much of the dataset's integrity as possible.

In practice, dimensionality reduction can simplify data before applying clustering algorithms. It can also use clustering to identify patterns that can then be further analyzed with association rules. These techniques complement each other, enhancing the overall data analysis.

TYPES OF CLUSTERING MODELS

A cluster consists of data points that share similarities, defined by their relationships with neighboring data points. The data elements grouped in this manner are known as clusters. Clustering can aid analysts in grouping customers with similar purchasing behaviors, which can then be used to target appropriate markets, provide product recommendationsfor upsell/cross, customer segmentation and analytics. Different types of clustering algorithms are designed to manage various kinds of unique data effectively as explored here.

Centroid-Based Clustering:

Centroid-based clustering is a popular partitioning method that organizes data points around central vectors called centroids, which represent the clusters. Data points are assigned to clusters based on their squared distances from the centroids. To determine the similarity measure between clusters, Euclidean distance, Manhattan distance, or Minkowski distance can be employed.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Density-Based Clustering:

Density-based clustering, a type of model-based method, defines clusters as regions with high data density, separated by areas of low data density within the dataset. Unlike centroid-based methods, it automatically determines the number of clusters. One of the key advantages of this technique is that clusters can take on any shape and are not restricted by specific constraints.

Hierarchical-Based Clustering:

Hierarchical-based clustering, also known as connectivity-based clustering, organizes data into a tree structure of nested clusters in a top-down manner. While it is more restrictive compared to other clustering methods, it is particularly well-suited for certain types of datasets. There are two types of hierarchical clustering:

- Agglomerative (bottom-up) approach: Clusters are repeatedly merged into larger ones until a single cluster remains.
- Divisive (top-down) approach: Begins with all data in a single cluster and successively splits it into smaller clusters until each cluster contains only one data point.

Distribution-Based Clustering:

Distribution-based clustering involves generating data points from a combination of statistical distributions, such as Gaussian or Poisson. This approach uses a central point as the core of the cluster, with the likelihood of a data point being part of the cluster diminishing as its distance from the center grows. The Gaussian Mixture Model is the most commonly employed algorithm in distribution-based clustering.

Fuzzy Clustering:

Fuzzy Clustering is an unsupervised machine learning model where a data point can be a part of more than one cluster. In other words, a data point belongs to a cluster with a degree of fuzzy membership or probability. This behavior is very different from traditional algorithms where a data point belongs to a single cluster. It is the model of choice to process complex clusters.

CLUSTERING MACHINE LEARNING ALGORITHMS

K Means Clustering:

K-Means is an unsupervised, centroid-based machine learning algorithm designed to group unlabeled data elements into similar clusters. For example, in digital marketing, K-Means can be used to segment customers based on purchases made and spending patterns, creating groups like Budget Shoppers, Frequent Buyers, and Big Spenders for targeted marketing.

K-Means is a partitional clustering algorithmvalued for its simplicity, ease of implementation, and numerous demonstrated successes across fields such as business, science, and medicine. It was independently developed in multiple locations during the 1950s and 1960s, eventually gaining widespread recognition. This algorithm is known as Lloyd's algorithm. K-Means algorithm needs three main parameters to be defined:

1. *Initialization criteria*: Arthur et al. [10] introduced the k-means++ initialization scheme to generate initial centroids that are distant from one another, increasing the likelihood of achieving better results. Additionally, Random point's generator can be used for initialization. Ongoing research focuses on developing the most efficient seeding methods for the K-Means algorithm, with one such study based on Independent Component Analysis.
2. *Number of clusters*: Determining the number of clusters is the most challenging part of configuring this algorithm, as there are no definitive mathematical criteria for it. However, various heuristic and simplified

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

approaches have been devised. The elbow method is one of the simplest and most widely used techniques. Additionally, advanced methods for selecting the optimal number of clusters include:

- Minimum Message Length (MML)
- Minimum Description Length (MDL)
- Bayes Information Criterion (BIC)
- Akaike Information Criterion (AIC)
- Dirichlet Process
- Gap statistics

3. *Distance metric*: The Euclidean metricis the most widely used method to calculate distance between data points. It is also referred to as the spherical k-means model. As the name indicates its primary limitation is that it identifies only spherical-like groups and may become distorted in highly multi-dimensional analyses, a challenge known as the "curse of dimensionality." Alternative options include:

- Mahalonobis distance (high computational cost)
- Itakura-Saito distance
- L1 distance
- Cosine distance
- Bregman distance

Steps in the K-Means Algorithm:

The K-Means algorithmconsists of three major steps:

1. initialize '**k**' centroids where '**k**' is equal to the number of clusters chosen for a specific dataset.
2. Assigneach data point to its closest centroid based on distance (usually Euclidean). This stepis called the expectation step.
3. Computethe *mean* of all the points for each cluster and reassignthe cluster center, or centroid. This step is called the maximization step.
4. This process is repeated until the centroid positions converge or the maximum number of iterations is reached.

NOTE: The centroid, or cluster center, is either the *mean* or *median* of all the points within the cluster based on the characteristics of the data.

The objective is to minimize the sum of distances between data points and their assigned clusters. This algorithm aims at minimizing an objective function known as squared error function given by:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} \left(\left\| x_i - v_j \right\| \right)^2$$

where,

'$||x_i - v_j||$'–gives the euclidean distance between the variables $x_i$and v_j.

'c_i'- number of data points in cluster 'i'

'c'- number of cluster centers.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Data points that are nearest to a centroid are grouped together within the same category. A higher 'k' value, representing the number of clusters, indicates smaller, more detailed clusters, whereas a lower 'k' value results in larger, less detailed clusters.

It is a common practice to perform the analysis using multiple initialization points as the outcome of clustering is influenced by that criterion. Finally, the input that yields the lowest resultant inertia is selected. The algorithm often encounters the issue of local minima, which can be mitigated through advanced algorithms. Enhanced Firefly Algorithms serve as a notable example of such improvements.

Ongoing research and variations proposed to K-Means include the following:

- K-Medoid - centroid is the most centrally located object.
- K-Median - centroid is calculated using median instead of a mean.
- Fuzzy C-means model.

Benefits:

1. *Simple*: K-means clustering is straightforward to understand and implement. It stands as the most widely used unsupervised machine learning technique.
2. *Fast*: K-means clustering utilizes a straightforward iterative approach, making it computationally more efficient compared to hierarchical clustering, which requires building tree-like cluster structures and calculating pairwise distances between all data points.
3. *Scalable*: K-means is a highly scalable model for large datasets. It handles clusters of diverse shapes and sizes, making it well-suited for cluster analysis.

Drawbacks:

1. *Input Dependent*: K-means clustering relies on correct initialization to achieve meaningful cluster results. Inadequate centroid initialization can result in longer run times and suboptimal cluster assignments. Significant research efforts have been dedicated to enhancing the centroid initialization process.
2. *Low performance*: K-means algorithm faces challenges with datasets that exhibit high variability. When certain assumptions are not met, K-means may generate low-quality clusters. For example, uneven cluster sizes can skew centroids toward larger clusters, leading to bias and misclassification in smaller ones. This limitation can be mitigated by generalizing K-means with probabilistic models such as the Gaussian Mixture Model.
3. *Outlier impact*: Since the algorithm calculates centroids by averaging the values within a cluster, it is highly sensitive to outliers. This sensitivity can increase the likelihood of overfitting by incorporating these outliers.

DBSCAN

The Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm defines clusters as dense regions within the data space, separated by lower-density areas. It groups closely packed data points and classifies outliers as noise based on their density in the feature space.Martin Ester et al. [11] introduced this density-based algorithm, which was published in 1996. The fundamental idea of DBSCAN is based on dense regions, operating under the assumption that natural clusters are formed by closely grouped points. This requires a precise definition of what constitutes a "dense region."Key parameters required for DBSCAN algorithm are:

- **eps**, ε– distance
  - defines the radius of the neighborhood around a data point. Choosing the**eps** is critical.
  - If **eps**is too small, most points will be classified as noise.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

- If **eps**is too large, clusters may merge, and the algorithm may fail to distinguish between them.
- Two data points aretreated as neighbors, if the distance between them is less than or equal to the value of epsilon (**eps**). k-distance graph is a common technique to specify**eps**.

- **MinPts** – The minimum number of points required within the epsilon (eps) radius to define a dense region.
  - A typical guideline is to set **MinPts**>= **D**+1, where **D** is the number of dimensions in the dataset.
  - Mostly, a minimum value of **MinPts = 3** is recommended.

DBSCAN classifies data points into three categories:

1. *Core points:* These have enough neighboring points within a specified radius (epsilon).
2. *Border points:*Data points that are positioned close to core points but lackenough neighbors to be classified as core points themselves.
3. *Noise points:*Data points that do not belong to any cluster.

By iteratively expanding clusters from core points and linking density-reachable points, DBSCAN creates clusters without depending on strict assumptions about their shape or size.

Steps in the DBSCAN Algorithm:

1. **Identify Core Points**: For every point in the dataset, calculate the number of neighboring points within its epsilon (eps) neighborhood. If the count is equal to or greater than MinPts, classify the point as a core point.
2. **Create Clusters**: For every core point not already assigned to a cluster, create a new cluster. Recursively identify all density-connected points within the epsilon (eps) radius of the core point and add them to the cluster.
3. **Density Connectivity**: Two points, aa and bb, are regarded as density-connected if there exists a sequence of points where each consecutive pair lies within the epsilon (eps) radius, and at least one point in the sequence is a core point. This ensures that all points within a cluster are interconnected through a network of dense regions.
4. **Label Noise Points**: After processing all points, any point that does not belong to a cluster is classified as noise.

Benefits:

- The number of clusters need not be predetermined for the algorithm to execute, instead it relies on the eps and MinPts parameters.
- Overcomes the limitations of convex-shaped assumptions by detecting clusters in various shapes and sizes.
- It effectively manages noise and can identify points that do not belong to any cluster, categorizing them as outliers.

Drawbacks:

- The computational cost is high as neighborhood queries need to be executed for each data point.
- It struggles to accurately identify clusters with varying densities.
- The Euclidean distance measure, commonly used in DBSCAN, becomes less effective in high-dimensional data, a challenge referred to as the curse of dimensionality.

Affinity Propagation:

Affinity Propagation is a relatively recent unsupervised clustering algorithm [32]. It groups the data points based on their similarityinto multiple clusters.It does not require to specify the numbers of clusters before implementation. It was introduced by Freyet al. [12] in 2007.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The algorithm leverages "message-passing" between data points to determine cluster centers, known as exemplars. Each exemplar represents a typical data point within a cluster, characterizing other points in the same cluster. The algorithm seeks to identify the most representative exemplars to cluster data into compatible groups. This approach is especially useful when the number of clusters is unknown or when the data displays complex, non-spherical, or non-convex cluster shapes.

- **Exemplars**: The exemplars serve as the "leaders" of the groups. Each group has one representative data point (the exemplar), and other data points are assigned to the group based on their similarity to the exemplar.
- **Message-passing**: Each individual (data point) communicates with others to determine who should become the leader (exemplar) of their group.

Messages are exchanged back and forth, allowing the data points to gradually agree on the most suitable leaders (exemplars). Once the leaders are selected, the remaining data points join the group led by the most appropriate leader (exemplar).

Key Steps of Affinity Propagation:

Affinity Propagation clustering algorithm has the followingsteps:

1. *ComputeSimilarity*: The first step of the algorithm is to constructa similarity (or dissimilarity) matrix that measures the similarity between pairs of data points. The choice of similarity metric varies based on the nature of the data and the problem being addressed. Commonly used metrics include Euclidean distance, negative squared Euclidean distance, and others.
2. *Responsibility Update*: Next, the algorithm initializes the responsibility matrix (R), where R(i,k) represents the responsibility of data point ii to act as the exemplar for data point kk. Using their similarity and the responsibilities received from other data points, the data points engage in a competition to become exemplars.

$$r(i,k) \leftarrow s(i,k) - \max_{k' \neq k} \{a(i,k') + s(i,k')\}$$

3. *Availability Update*: The algorithm then initializes the availability matrix (A). This matrix A(i, k) represents the availability of data point kto designate data point i as its exemplar. The data points with higher availability are more likely to be selected.

$$a(i,k) \leftarrow \min\left(0, r(k,k) + \sum_{i' \notin \{i,k\}} \max(0, r(i',k))\right) \text{ for } i \neq k \text{ and}$$
$$a(k,k) \leftarrow \sum_{i' \neq k} \max(0, r(i',k))$$

4. *Update Responsibility and Availability*: Next, the algorithm iteratively updates the responsibility and availability matrices until convergence is achieved, which occurs when the matrices exhibit minimal changes between iterations.
5. *Net Responsibility Calculation*: The net responsibility for each data point is then calculated by adding both responsibility and availability.
6. *Exemplar Selection*: Next task is to identify exemplars, which aredata points with high net responsibility. These exemplars serve as the cluster centers.
7. *Cluster Assignment*: Finally, the algorithm assigns each data point to its nearest exemplar based on similarity, thereby forming clusters.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Benefits:

1. The number of clusters need not be predefined, providing greater flexibility in clustering tasks.
2. High-quality clusters are produced, even when the data points differ in size or density.
3. Well-suited for clustering data with complex relationships and non-linear patterns.
4. Versatile across a wide range of applications, including image segmentation, customer profiling, and gene expression analysis.

Drawbacks:

1. May require significant computational resources, particularly for large datasets, rendering it impractical for large-scale clustering tasks.
2. Might not consistently yield optimal results when compared to other clustering methods, like K-Means or Gaussian Mixture Models.
3. Sensitivity to the selection of the similarity metric can influence how similarities between data points are measured.
4. May generate multiple exemplars within a single cluster, complicating the interpretation of the clustering results.

BIRCH

Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) is a density-based hierarchical clustering technique tailored for handling large-scale datasets. Introduced by Zhang et al. [13] in 1996, it overcomes the shortcomings of traditional clustering algorithms, which face challenges with scalability and data quality as dataset sizes increase.BIRCH algorithm involves three tunable parameters. Unlike K-means, it does not require to specify the optimal number of clusters (k), as this is automatically determined during the process:

1. *threshold*: The maximum number of data points a sub-cluster in the leaf node of the CF tree can hold.
2. *branching_factor*: Specifies the maximum allowable number of CF sub-clusters within each internal node of the tree structure.
3. *n_clusters*: Specifies the number of clusters to be returned after the BIRCH algorithm is complete. If set to None, the final clustering step is skipped, and intermediate clusters are provided instead.

Steps in BIRCHAlgorithm:

The BIRCH clustering algorithm consists of two stages, hence called Two Step Clustering.

1. Build the CF Tree: The algorithm summarizes large datasets into smaller and dense regions called Clustering Feature (CF) entries. ACF entry is defined as an ordered triple, (N, LS, SS) where:
   a. 'N' is the number of data points in the cluster.
   b. 'LS' is the linear sum of N data points, a measure of the location of the cluster.
   c. 'SS' is the squared sum of N data points, a measure of the spread of the cluster.

A CF entry can be composed of other CF entries, making it a CF Tree where each leaf node contains a sub-cluster.

2. Global Clustering: An existing clustering algorithm is applied to the leaf nodes of the CF tree. Optionally, these clusters can be further refined until each leaf node reaches a specified maximum number of entries, known as the threshold.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The clustering solution can be affected by the order in which data records are input due to the nature of the CF tree's hierarchical structure. BIRCH clustering can be applied to several random data samples and evaluate the results to fix this.

Benefits:

1. *Scalability*: BIRCH efficiently processes large datasets by leveraging memory-optimized data structures and a hierarchical clustering method.
2. *Rapid clustering*: The CF tree structure enables quick traversal and clustering, making it ideal for real-time scenarios.
3. *Automatic clusters:*Based on the given threshold,BIRCH adaptively splits clusters and thereby eliminating the need to predefine the number of clusters.

Drawbacks:

1. *Sensitivity to parameter tuning*: Achieving optimal clustering results relies on carefully configuring parameters such as the branching factor and threshold.
2. *Limited to spherical clusters*: BIRCH excels with spherical clusters but may face difficulties when handling clusters with complex or irregular shapes.

Bisecting K-Means:

Bisecting K-means is a hybrid ofdivisive Hierarchical (top-down) and K-means clustering.Rather than dividing the dataset into K clusters in each iteration, the algorithm bisects one cluster into two sub-clusters at each step using K-means, repeating this process until a total of K clusters is achieved.

Karypis et al. [14] noted that bisecting K-means outperforms regular K-means, as it generally forms clusters of more uniform sizes. In contrast, regular K-means often produces clusters with significantly varying sizes. While smaller clusters generally exhibit higher quality, their influence on the overall quality measure is limited due to the weighting by cluster size. In contrast, larger clusters often show lower quality and significantly reduce the overall cluster quality.

Steps in Bisecting K-Means Algorithm:

1. Select a cluster to divide.
2. Use the basic K-means algorithm to create two sub-clusters (bisecting step).
3. Repeat the bisecting step (step 2) 'n' times and choose the split that results in the clustering with the greatest overall similarity. Here, 'n' represents the number of trial bisections conducted in each phase of Bisecting K-means.
4. Continue repeating steps 1, 2, and 3 until the target number of clusters is achieved.

Benefits:

1. Bisecting K-means is more efficient when K is large. It focuses on a single cluster and two centroids per bisecting step, significantly reducing computation time.
2. Bisecting K-means typically generates clusters of similar sizes, whereas K-means often results in clusters with widely varying sizes.

Drawbacks:

1. The quality of the resulting clusters is highly influenced by the choice of initial centroids.
2. The clusters formed vary significantly in size, resulting in imbalance and potentially leading to the creation of empty clusters.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

QUALITY ASSESSMENT OF CLUSTERS

In unsupervised machine learning, evaluating cluster quality and validity is vital to implement clustering solutions. There are two types of evaluation metrics:

Internal Evaluation: These metrics are applicable in cases wherethe ground truth labels are unavailable, offering a means to evaluate the quality of clustering by analyzing the inherent characteristics of the data.

- Inertia (Within-Cluster Sum of Squares)
- Silhouette Coefficient
- Davies-Bouldin Index
- Calinski-Harabasz Index (Variance Ratio Criterion)

External Evaluation: These metrics are applicable where the ground truth labels are available.

- Rand Index (RI)
- Adjusted Rand Index (ARI)
- Normalized Mutual Information (NMI)

Silhouette Score:

For this research, silhouette score is considered as the quality assessment metric as it iswidely used to assess how effectively data points are grouped into clusters. It ensures that clusters are both well-formed and distinct, making it an essential tool for various applications, including marketing and image analysis. This score is determined by assessing each data point's similarity to its assigned cluster and its dissimilarity to other clusters. By integrating the Silhouette score with visualizations and domain expertise, clustering solutions can become more meaningful and dependable.

Steps to Calculate Silhouette Score:

1. Calculate the average distance (a_i)between each data pointto other data points within the same cluster so that the similarity of the data pointto others is measured.
2. Calculate the average distance (b_i)between each data point, to other data points in the other clusters it does not belong to. This value reflects its dissimilarity to data points in different clusters.
3. For each data point, calculate the Silhouette Score with the given formula.

Silhouette Score = $(b_i - a_i) / \max(a_i, b_i)$

4. The overall Silhouette Score is calculatedas an average of all the scores, thus providing the quality assessment.

The values of score range from -1 to +1:

- Positive values indicate that the data points belong to the correct clusters.
- Ascore of zero indicates overlapping clusters suggesting that the data points areclose to multiple clusters.
- Poor clustering results are indicated by the negative values that the data points are assigned to incorrect clusters.

RESULTS AND OBSERVATIONS

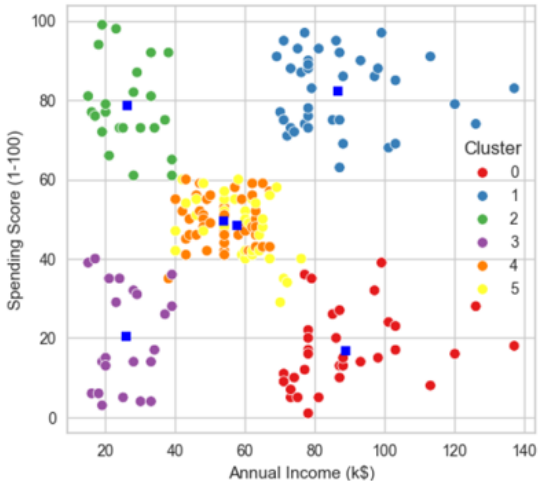
K-Means:

The K-Means algorithm generated six clusters, as illustrated in Figure-1. Each segment's income-to-spending score is provided. To evaluate the consistency of the results, multiple runs were conducted. With each run, the number of

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

data points in the clusters fluctuated significantly, leading to random clusters and highlighting the stochastic nature of the algorithm. Nevertheless, the silhouette score is favorable compared to other algorithms.

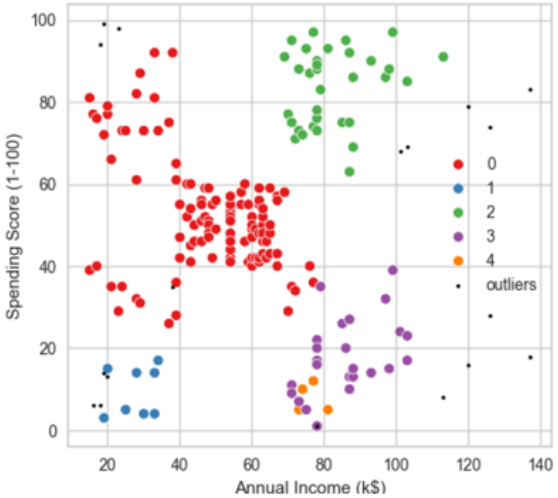


	Clusters
0	Medium income - Low spending score
1	High income - high spending score
2	Low income - high spending score
3	Low income - low spending score
4	Low income - medium spending score
5	Low income - medium spending score
	Silhouette Score = 0.45

Figure 1

DBSCAN:

The DBSCAN algorithm produced six clusters, as shown in Figure2. One prominent cluster, characterized by medium income and medium to high spending scores, is ambiguous and comprises 56% of the input data points. Additionally, outliers were identified, accounting for 9% of the total data. These two factors render the results impractical for use. Furthermore, the silhouette score is the lowest when compared to other algorithms.



	Clusters
0	Low income - medium spending score
1	Low income – low spending score
2	High income - high spending score
3	Medium income – low spending score
4	Low income - low spending score
*	Outliers
	Silhouette Score = 0.25

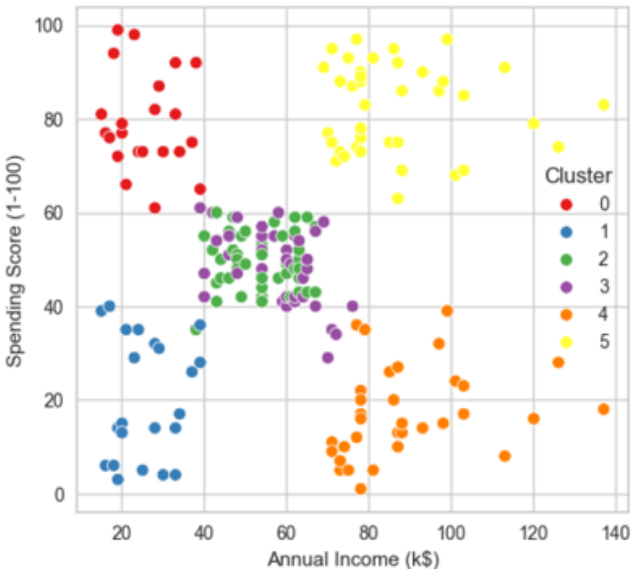
Figure 2

Affinity Propagation:

The advanced Affinity Propagation algorithm produced six well-defined clusters, each exhibiting a good silhouette score, as illustrated in Figure 3. This reflects the high quality of the clusters. The various clusters, characterized by different income-to-spending score combinations, are also presented. Notably, the cluster sizes remained consistent across multiple runs.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

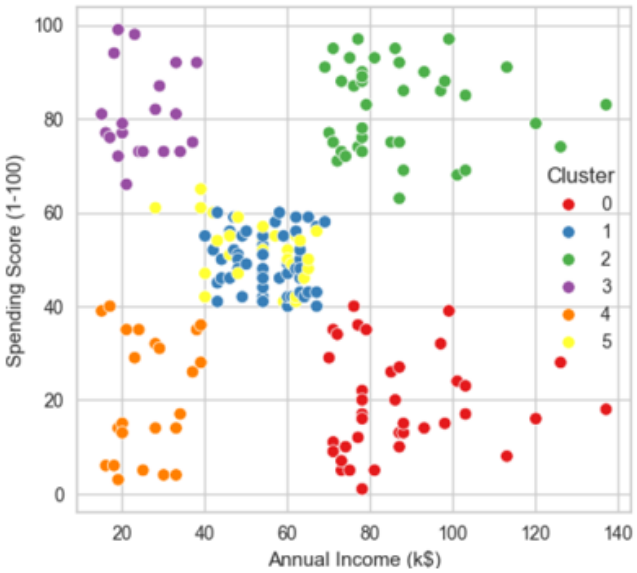


	Clusters
0	Low income – high spending score
1	Low income – low spending score
2	Low income - medium spending score
3	Low income – Medium spending score
4	Medium income - low spending score
5	High income – high spending score
	Silhouette Score = 0.45

Figure 3

BIRCH:

The BIRCH algorithm generated six clusters, as depicted in Figure (4), with corresponding income-to-spending scores. The algorithm achieved a silhouette score of 0.43, which is higher only when compared to K-Means and Affinity Propagation algorithms. Additionally, the cluster sizes were observed to remain consistent across multiple runs.



	Clusters
0	Medium income - Low spending score
1	Low income - medium spending score
2	High income - high spending score
3	Low income - high spending score
4	Low income - low spending score
5	Low income - medium spending score
	Silhouette Score = 0.43

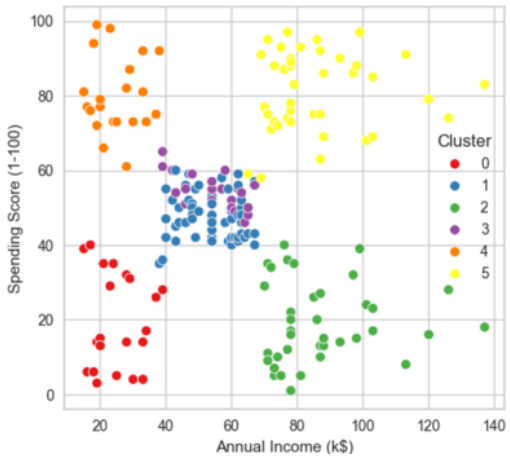
Figure 4

Bisecting K-Means:

The Bisecting K-Means algorithm consistently produced uniform cluster sizes across multiple runs, in contrast to the K-Means algorithm. Figure (5) illustrates the segmentsformed, along with their annual income-to-spending score details. Notably, the silhouette score is slightly lower than that achieved by the K-Means algorithm.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article



	Clusters
0	Low income - Low spending score
1	Low income - medium spending score
2	Medium income - Low spending score
3	Low income – Medium spending score
4	Low income - high spending score
5	High income - High spending score
	Silhouette Score = 0.39

Figure 5

Across all the evaluated algorithms, clusters based on customers' age are widely distributed, as depicted in Figure (6). Distinct clusters for specific age groups could not be formed, indicating the characteristics of the source data and the lack of significant variation within it.

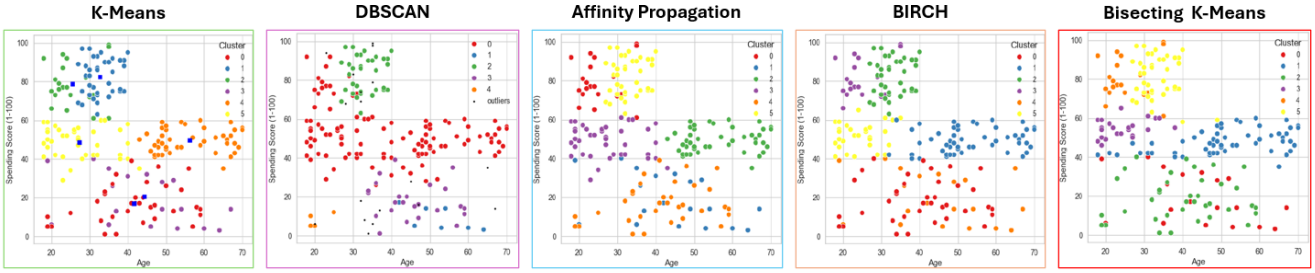


Figure 6

Unsupervised machine learning algorithms from various clustering categories were applied and analyzed using data samples sourced from Kaggle. The dataset comprises customer ID, Gender, Age, Annual Income, and Spending Score. The below Figure (7)offers a consolidated overview of the study's results, enabling comparison of cluster size percentages. Additionally, the silhouette score reflects the quality of each cluster.

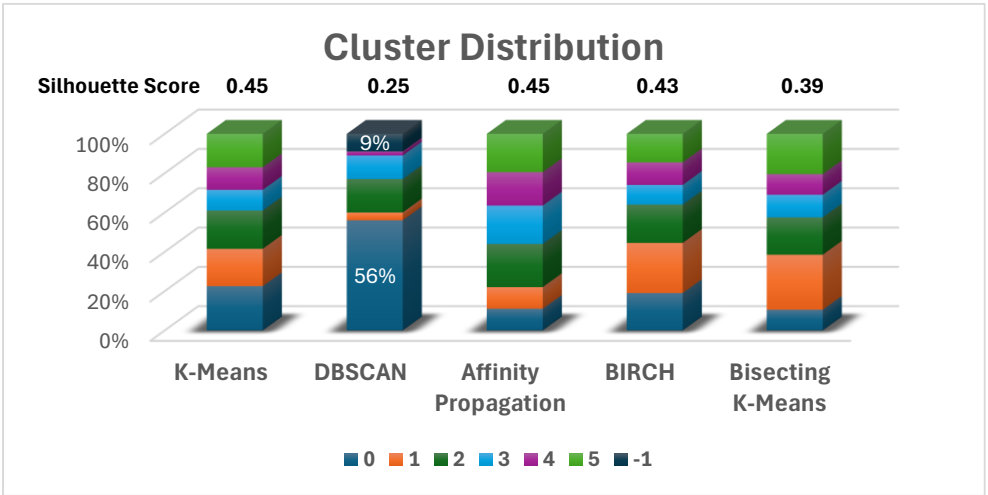


Figure 7

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

CONCLUSION

Based on the data analyzed and results generated thereof, below conclusions can be drawn.

Despite its popularity for its simplicity and efficiency, the centroid-based K-means algorithm often produced inconsistent results due to its stochastic nature, even when the random state is set to a fixed integer value, such as one.

The density-based DBSCAN algorithm struggled to produce meaningful clusters, as it identifies clusters based on the density of points. When a cluster is less dense compared to others, DBSCAN might not identify it as a valid cluster, resulting in suboptimal outcomes. Moreover, the notably lower silhouette scores compared to other algorithms emphasize the poor quality of the clusters produced.

Based on the number of clusters generated and the quality assessment from the silhouette scores, Affinity Propagation algorithm clearly performed best compared to other algorithms. The results are found to be optimal and consistent across multiple runs. For the input data source used in this research, Affinity Propagation proved to be a good fit for customer segmentation.

BIRCH, a hybrid of density-based and hierarchical clustering, produced a stable number of data points across multiple runs and achieved a commendable silhouette score, slightly lower than that of the Affinity Propagation algorithm. Therefore, it can be concluded that BIRCH delivered the second-best performance.

Bisecting K-Means, a hybrid of centroid-based and hierarchical clustering approaches, consistently produced a stable number of data elements within each cluster, unlike the standard K-Means algorithm. However, its silhouette score was significantly lower.

Future research opportunities could focus on the exploration of fuzzy clustering algorithms or hybrid approaches combining multiple algorithms. With the rapid advancements in Artificial Intelligence and Machine Learning, there is significant potential for the development of new algorithms, innovative approaches, and improved processing hardware.

REFERENCES

[1] Behera, Rajat Kumar & Gunasekaran, Angappa& Gupta, Shivam & Kamboj, Shampy& Bala, Pradip Kumar, 2020. "Personalized digital marketing recommender engine," Journal of Retailing and Consumer Services, Elsevier, vol. 53(C). https://ideas.repec.org/a/eee/joreco/v53y2020ics0969698918307987.html

[2] Bhagat, Abhiraj, Pranav Bhandari, Anup Lal Yadav, and Navjot Singh Talwandi. "AI-Powered Customer Segmentation For Marketing." *Available at SSRN 4811907* (2024).

[3] Dullaghan, Cormac, and Eleni Rozaki. "Integration of machine learning techniques to evaluate dynamic customer segmentation analysis for mobile customers." arXiv preprint arXiv:1702.02215 (2017).

[4] Alsac, Ali. (2022). Developing Customer Segmentation Models for Digital Marketing Campaigns using Machine Learning.

[5] KATRAGADDA, VAMSI. "Dynamic customer segmentation: using machine learning to identify and address diverse customer needs in real-time." *IRE Journals* 5, no. 10 (2022): 278-279.

[6] Vijilesh, V., A. Harini, M. Hari Dharshini, and R. Priyadharshini. "Customer Segmentation Using Machine Learning." *Elementary Education Online. http://doi. org/10.17051/ilkonline* 335 (2021).

[7] Monil, Patel, Patel Darshan, Rana Jecky, Chauhan Vimarsh, and B. R. Bhatt. "Customer segmentation using machine learning." International Journal for Research in Applied Science and Engineering Technology (IJRASET) 8, no. 6 (2020): 2104-2108.

[8] Jing, Weipeng, Chuanyu Zhao, and Chao Jiang. "An improvement method of DBSCAN algorithm on cloud computing." *Procedia computer science* 147 (2019): 596-604.

[9] Maureen, Akazue, Esiri Kesiena Henry, and Clive Asuai. "APPLICATION OF RFM MODEL ON CUSTOMER SEGMENTATION IN DIGITAL MARKETING." *Nigerian Journal of Science and Environment* 22, no. 1 (2024): 57-67.
[10] Arthur, David, and Sergei Vassilvitskii. *k-means++: The advantages of careful seeding*. Stanford, 2006.
[11] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96). AAAI Press, 226–231.
[12] Frey, Brendan J., and Delbert Dueck. "Clustering by passing messages between data points." *science* 315, no. 5814 (2007): 972-976.
[13] Zhang, Tian, Raghu Ramakrishnan, and Miron Livny. "BIRCH: an efficient data clustering method for very large databases." *ACM sigmod record* 25, no. 2 (1996): 103-114.
[14] Karypis, Michael Steinbach George, Vipin Kumar, and Michael Steinbach. "A comparison of document clustering techniques." In *TextMining Workshop at KDD2000 (May 2000)*, pp. 428-439. 2000.
[15] Das, S., Nayak, J. (2022). Customer Segmentation via Data Mining Techniques: State-of-the-Art Review. In: Nayak, J., Behera, H., Naik, B., Vimal, S., Pelusi, D. (eds) Computational Intelligence in Data Mining. Smart Innovation, Systems and Technologies, vol 281. Springer, Singapore. https://doi.org/10.1007/978-981-16-9447-9_38.
[16] Wang, Chenguang. "Efficient customer segmentation in digital marketing using deep learning with swarm intelligence approach." *Information Processing & Management* 59, no. 6 (2022): 103085.
[17] Balkaya, A., Tüzünkan, E., Ayaz, K., Akçay, Y., Abut, F., Akay, M.F., Erdem, S. and Alsaç, A., Developing Customer Segmentation Models for Digital Marketing Campaigns using Machine Learning.
[18] Talosig, E., 2022. Improving Digital Marketing for Attracting the Target Customer Segments.
[19] Nica, E., Gajanova, L. and Kicova, E., 2019. Customer segmentation based on psychographic and demographic aspects as a determinant of customer targeting in the online environment. *Littera Scripta*, *12*(2), pp.108-126.
[20] S. R. Regmi, J. Meena, U. Kanojia and V. Kant, "Customer Market Segmentation using Machine Learning Algorithm," 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2022, pp. 1348-1354, doi: 10.1109/ICOEI53556.2022.9777146. keywords: {Machine learning algorithms; Consumerbehavior;Clusteringalgorithms;Companies;Machinelearning;Marketresearch;Productdevelopment;MachineLearning;Classification;Clustering;CustomerSegmentation;Spendingbehavior;K-MeanAlgorithm},
[21] Patankar, Nikhil & Dixit, Soham &Bhamare, Akshay & Darpel, Ashutosh & Raina, Ritik. (2021). Customer Segmentation Using Machine Learning. 10.3233/APC210200.
[22] Dullaghan, C. and Rozaki, E., 2017. Integration of machine learning techniques to evaluate dynamic customer segmentation analysis for mobile customers. *arXiv preprint arXiv:1702.02215*.
[23] R. Amutha, Aazmaan Ahmed Khan, Customer Segmentation using Machine Learning Techniques
[24] Pyla. Srinivas Dileep, Dr. M. Seshashayee, CUSTOMER SEGMENTATION USING MACHINE LEARNING, e-ISSN: 2582-5208, International Research Journal ofModernization in Engineering Technology and Science, Volume:04/Issue:05/May-2022
[25] Yash Parab,Jugal Dave Customer Segmentation Using Machine Learning: A Comprehensive Research Study, © 2023 IJNRD | Volume 8, Issue 6 June 2023 | ISSN: 2456-4184 | IJNRD.ORG
[26] Yu, Hui & Chen, LuYuan& Yao, Jingtao& Wang, XingNan. (2019). A three-way clustering method based on an improved DBSCAN algorithm. Physica A: Statistical Mechanics and its Applications. 535. 122289. 10.1016/j.physa.2019.122289.
[27] Jing, Weipeng& Zhao, Chuanyu& Jiang, Chao. (2019). An improvement method of DBSCAN algorithm on cloud computing. Procedia Computer Science. 147. 596-604. 10.1016/j.procs.2019.01.208.
[28] Arthur, David &Vassilvitskii, Sergei. (2007). K-Means++: The Advantages of Careful Seeding. Proc. of the Annu. ACM-SIAM Symp. on Discrete Algorithms. 8. 1027-1035. 10.1145/1283383.1283494.
[29] Refianti, Rina &Mutiara, Achmad &Juarna, Asep&Suhendra, Adang. (2018). A Preference Model on Adaptive Affinity Propagation. International Journal of Electrical and Computer Engineering (IJECE). 8. 1805. 10.11591/ijece.v8i3.pp1805-1813.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

[30] P.Ling and his team proposed and “Adjustable Preference Affinity Propagation (APAP)” algorithm which can produce better clustering results due to improvement to the element preference calculations

[31] H. Wenlong and his team proposed “Transfer affinity propagation-based clustering” which out-performs current algorithm in a case of a very small dataset.

[32] Jihan Alameddine, Kacem Chehdi, Claude Cariou, "Optimization of unsupervised affinity propagation clustering method," Proc. SPIE 11155, Image and Signal Processing for Remote Sensing XXV, 111550C (7 October 2019); https://doi.org/10.1117/12.2533164