LUND UNIVERSITY
School of Economics and Management

Comparative Analysis of Machine Learning Algorithms on Comprehensive and Cluster-Specific Data in the Auto Insurance Industry

Authors:

Veselina Balabanova
Shreeya Bhattarai

DABN01
Master's Thesis (15 credits ECTS)
May 2024
Supervisor: Simon Reese

Abstract

In recent years, businesses have been focusing on Customer Lifetime Value (CLV) to achieve better customer relationships and to identify high-value customers for more customized marketing strategies. This thesis contributes by comparing the performance of different machine learning models on cluster-specific data points and the complete dataset from the auto insurance industry. In addition, the study also discovers the most valuable customer cluster and devises customer retention strategies based on significant features that influence CLV.

For further empirical analysis, we have selected Principal Component Analysis (PCA) and *k*-means Clustering for customer segmentation. We have also used Random Forest, XGBoost, and Neural Networks, to predict CLV on comprehensive and cluster-specific data. Applied feature importance and hyperparameter tuning have been used for further insights. Overall, the findings suggest the best performance among the models is by Random Forest and its R^2 improved by 27% while RMSE dropped by 39% after applying the models to every cluster for predicting CLV. For future research, the findings from this study can also be adopted in other insurance industries to see how using clustering techniques helps improve the machine learning models' performances.

Keywords: Auto Insurance Industry, Machine Learning, Random Forest, XGBoost, Neural Network, k-Means Clustering, Principal Component Analysis, Customer Lifetime Value (CLV)

Acknowledgments

We would like to express our sincere gratitude to our thesis supervisor, Simon Reese, for providing us with his insights and time throughout the process. His teachings have brought significant growth to our knowledge. In addition, thanks to our friends who have made this process enjoyable by sharing their knowledge and motivating us through this journey. Lastly, special gratitude to our families for their unwavering support.

Shreeya & Veselina

Table of Contents

List of Tables

List of Figures

Abbreviations

CLV	Customer Lifetime Value
CNN	Convolutional Neural Network
CRISP-DM	Cross-Industry Standard Process for Data Mining
DNN	Deep Neural Networks
FCM	Fuzzy C-means Clustering
kNN	k-nearest Neighbors
ML	Machine Learning
PCA	Principal Component Analysis
ReLU	Rectified Linear Unit
RF	Random Forest
RFM	Recency, Frequency, and Monetary Value
RMSE	Root Mean Square Error
ROI	Return on Investment
SaaS	Software as a Service
SSE	Sum of Squared Error
SVM	Support Vector Machine
XGBoost	eXtreme Gradient Boosting

1. Introduction

With an increase in the significance of customer satisfaction, businesses have started to focus more on customer loyalty along with profitability. Businesses can be successful in managing customer relationships if they can identify the customer's true value which can lead to customized marketing strategies (Kim et al., 2006). Therefore, Customer Lifetime Value (CLV) has been gaining popularity in recent decades. CLV is the total revenue that a customer generates over the time period of its relationship with the company (Burelli, 2019). According to Gupta et al. (2006), the main reasons for the growing interest in CLV are the pressure to make marketing more accountable, the ability to identify profitable customers for more personalized marketing, and the improvement in technology for analyzing customer data.

Desirena et al. (2019) also highlight that there is an increasing interest in exploring customer retention strategies in the insurance industry due to the high churning rate. Customer retention in the insurance industry is quite challenging as it is becoming increasingly more difficult to engage customers even in a contract-based setting. Insurance companies can lose around 15% of their clients annually, which results in revenue loss (Desirena et al., 2019). Hence, the retention strategies need to be customer-specific and target high-value customer groups.

Several researches have been conducted to model and predict customer lifetime value (CLV) using advanced machine learning algorithms (Sun et al., 2023; Chen et al., 2018; Chen, 2018; Chamberlain et al., 2017; Jasek et al., 2018). In addition, the studies have been done across multiple industries such as online retail (Jasek et al., 2018; Sun et al., 2023), the software industry (Bakhshizadeh et al., 2022), the banking industry (Nekooei and Tarokh, 2015) and gaming industry (Burelli, 2019; Chen et al., 2018). Some commonly used machine learning algorithms for modeling CLV are Random Forest, AdaBoost, XGBoost, and deep learning methods (Chen et al., 2018; Win & Bo, 2020; Chen, 2018; Desirena et al., 2019; Sun et al., 2023). However, there aren't many studies that combine all of these models.

In addition, Tekin et al. (2022) found that using fuzzy clustering and then applying ensemble learning techniques in each cluster increases the overall success of the model. Through the literature review, it is also evident that there isn't much research done on using clustering techniques for segmenting the customer cluster in the auto insurance industry.

The main purpose of the thesis is to compare the performance of different machine learning models on the complete dataset and cluster-specific data on a dataset from the auto insurance industry. In addition, the study also discovers the most valuable customer cluster and devises tailored customer retention strategies based on significant CLV-influencing factors.

Similar to studies done by Nekooei and Tarokh (2015), Bakhshizadeh et al. (2022), and Rachid et al. (2018), the thesis uses *k*-means clustering to segment features. The models (Random Forest, XGBoost, and Neural Networks) are then trained both on cluster data and the whole dataset. The Random Forest model exhibited a 27% improvement in R^2 and a 39% decrease in $RMSE$ when evaluated on cluster-specific data as opposed to the complete dataset.

The paper is structured as follows: Section 2 includes the literature review of previous research regarding customer segmentation and customer lifetime value using machine learning in different industries. Section 3 explains the methodology used in this thesis. Section 4 discusses the dataset, exploratory analysis, data preparation, and data cleaning. Section 5 presents empirical analysis such as clustering, principal component analysis (PCA), and the performance of machine learning algorithms. Section 6 and Section 7 include data-driven business decisions, conclusions, limitations, and future research. Figure 1. also highlights the steps taken for analysis.

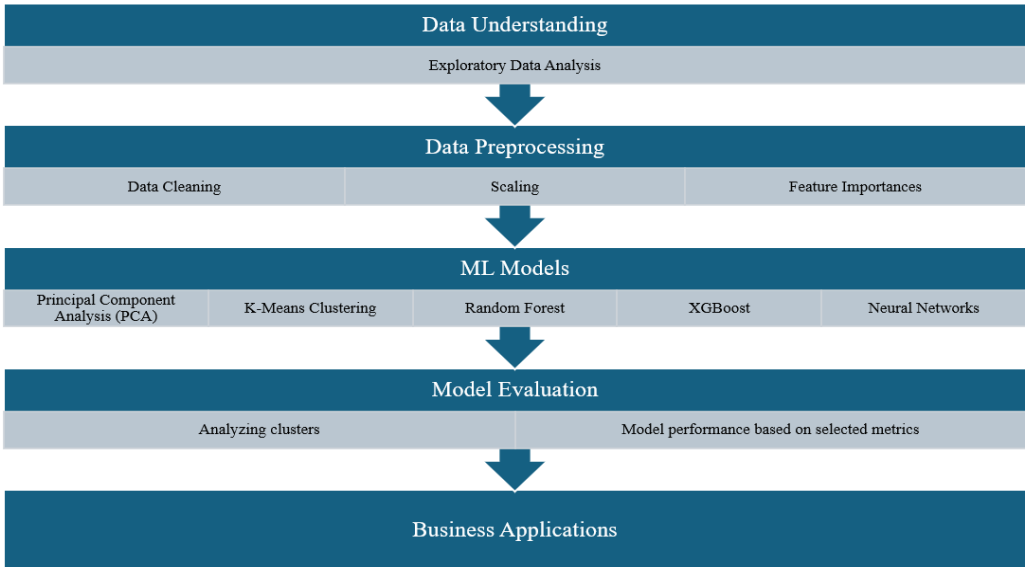


Figure 1. Process Flowchart

2. Literature Review

There has been a growing interest in Customer Lifetime Value (CLV) in recent years as a result of the high amount of customer transaction data available to draw insights into customers (Gupta et al., 2006). Customer Lifetime Value (CLV) is defined as the economic worth of a customer (Berger and Nasr, 1998). Additionally, CLV is also referred to as the total present value or estimated revenue that can be derived from a customer over its lifetime relationship with the company (Gupta et al., 2006).

As the cost of acquiring new customers is rising, online retailers must focus on retaining customers and specifically high-business value customers for long-term customer relationship management (Win and Bo, 2020). Furthermore, in online retail, Sun et al. (2023) highlighted that using purchasing information and adopting appropriate data analysis methods can support businesses to determine consumers who are bringing in more profit to the enterprise. In the software industry, Bakhshizadeh et al. (2022) discovered that CLV plays a crucial role in clustering customers, allocating resources, and planning effective strategies.

There have been several models used to predict CLV over the years. Initially, the literature favored parametric models which were easy to interpret. However, with the advancement in data mining and machine learning techniques, non-parametric techniques have started to focus on predictive ability (Gupta et al., 2006).

Gupta et al. (2006) also highlight several researches that combined different models to improve the predictions of CLV. The study by Win and Bo (2020) classified customers into different classes to decide on appropriate marketing promotions and campaigns using the Random Forest and AdaBoost models, in which Random Forest outperformed AdaBoost. The research done by Chen et al. (2018) in the video game industry focused on comparing the performance of parametric models with that of Deep Neural Networks (DNN) which included a Deep Multilayer Perceptron model and a Convolutional Neural Network (CNN). The study discovered that both neural networks outperformed parametric models while predicting CLV. Similarly, in the insurance industry, Bakhshizadeh et al. (2022) used a two-stage Neural Network to build a recommender system for maximizing CLV.

Furthermore, Sun et al. (2023) aimed to construct a customer segmentation model based on customer value measurement for non-contractual relationships in the online retail industry. The

different machine learning models used for the study were Support Vector Machine (SVM), *k*-nearest Neighbors (kNN), Naive Bayes, AdaBoost, etc. Moreover, a study in the aviation industry by Chen (2018) showcased the customer lifetime value using boosting techniques such as XGBoost to evaluate passenger network value for follow-up passenger segmentation and marketing promotions. Curiskis et al. (2023) introduced a new technique by using a hierarchical CLV modeling approach focusing on using the most recent data rather than historical data. The paper implemented a combination of machine learning models such as XGBoost, Gradient Boost, and kNN to conduct the analysis based on B2B SaaS companies.

Similarly, there have been several studies focused on customer segmentation to target high-value customers. Previously, the most traditional approach for grouping customers based on CLV was RFM models (Gupta et al., 2006). RFM stands for Recency, Frequency, and Monetary value of past customer purchases and the model provides scores for each group of customers. In the paper by Chan (2008), the study highlighted the importance of understanding the behavior of high-value customers for effective segmentation. Thus, the study adopted the RFM model for analyzing customer behavior based on the assumption that the future lifetime value of a consumer resembled the past and current patterns of their trading.

In recent research, several studies have used clustering to segment customer lifetime value into high and low-value customers (Najib et al., 2019; Cheng and Chen, 2009; Nekooei and Tarokh, 2015). Cluster analysis is an unsupervised learning algorithm for the classification of patterns into clusters based on similarity. There are two types of clustering - hard (or crisp) and soft. Hard clustering means assigning a specific data point to exactly one cluster. Soft clustering assigns a membership degree for every data point to a specific cluster (Omran et al., 2007).

For better customer relationship management, the study by Cheng and Chen (2009) discovered efficient results by segmenting customers based on RFM attributes and the *k*-means algorithm to target the right customer and maximize profitability. One method of soft clustering is Fuzzy C-means Clustering (FCM). The data points are given a membership grade between 0 and 1 which shows up to what degree the data point belongs to that specific cluster. This approach seems to be more applicable in scenarios where customers have various characteristics that belong to more than one cluster (Tekin et al., 2022). Tekin et al. (2022) used a mobile game's customers' marketing and gameplay data to predict customer lifetime value and define the clusters. The paper showed that performing the fuzzy clustering before ensembling methods yields better results than applying models on each customer individually.

The research by Bakhshizadeh et al. (2022) used the silhouette and the SSE indexes to find the optimal number of clusters (k) in *k*-means clustering. Then, customer lifetime value was calculated for every cluster and the clusters were assigned labels from highest to lowest value. This grouping technique helped the study define the customer clusters and create clearer business and marketing strategies. *k*-means clustering is also an appropriate technique to apply in the automobile insurance industry (Nekooei and Tarokh, 2015).

Some researchers have also performed feature importance to determine the most important features that influence CLV and develop strategies for better customer relationships. Win and Bo (2020) performed feature importance using the Random Forest method to rank the contribution of each feature on the target variable and selected six features from a total of 24 features. The model trained with fewer features had a slightly better performance than the model using all features. Similarly, the research on CLV systems at asos.com by Chamberlain et al. (2017) also ranked features based on Random Forest feature importance. In another study, feature selection was done using lasso coefficients (Srinivasan et al., 2023).

Hence, based on past studies, this thesis combines popular techniques which include Random Forest, XGBoost, and Neural Networks to discover the best model performance. In addition, k-means clustering, feature importance, and hyperparameter tuning are also considered to segment the dataset into clusters, determine the most significant features and overall optimize the model performance.

3. Methodology

This section provides an overview of the methods used in the study.

3.1. Methods for Customer Segmentation

In the thesis, we are using Principal Component Analysis (PCA) and *k*-means clustering for unsupervised pattern recognition. Thus, the aim is to discover groups in the data for customer segmentation (Everitt et al., 2011).

Principal Component Analysis applies a linear transformation to learn a low-dimensional representation of the data by preserving as much of the variance in the data as possible. The input x is projected to a representation z that best reconstructs the original data point. Then, a reconstruction error is calculated in terms of the mean squared error - the squared distance between the original data point and its reconstruction (Goodfellow et al., 2016). The goal is to minimize the reconstruction error concerning the training data.

Then, we apply *k*-means clustering on the PCA-reduced data. *k*-means clustering is an unsupervised learning method that tries to find similar x values and assigns hard cluster assignments (Lindholm et al., 2022). To group the data points, *k*-means uses the sum of squared Euclidean distances to the cluster centers, as the main measure of similarity:

$$\arg \min_{R_1, R_2, \ldots, R_M} \sum_{m=1}^{M} \sum_{\mathbf{x} \in R_m} \|\mathbf{x} - \widehat{\boldsymbol{\mu}}_m\|_2^2$$

In the equation above we have Mnumber of clusters and R_M number of data points in each cluster. $\widehat{\boldsymbol{\mu}}_m$is the cluster center, which is calculated as the mean of all data points in a specific cluster m(Lindholm et al., 2022).

Finding the optimal number of clusters before applying the *k*-means algorithm is the first step. One of the most popular methods to do this is the elbow method using the sum of squared error (SSE). The SSE index is the sum of squared error, i.e. distances of the data points in the cluster with the cluster center (Bakhshizadeh et al., 2022):

$$\mathrm{SSE} = \sum_{\mathrm{i}=1}^{\mathrm{k}} \sum_{\mathrm{x} \in \mathrm{C_i}} \mathrm{dist}^2 \left(m_i, x\right)$$

Therefore, the elbow method is a visual representation of a range of k values and the corresponding SSE index of every cluster. The plot will graphically resemble the shape of an elbow. The angle of the elbow corresponds to the optimal number of clusters (Umargono et al., 2020).

Along with the SSE index, the silhouette score is another key metric that we have used in determining the optimal number of clusters in our data. The values of the silhouette score can range from -1 to 1, and the closer the score is to 1, the better the result. It calculates the distance of a data point to its cluster as well as the distance with neighboring clusters. The silhouette score is defined as follows:

$$\mathrm{s(x)} = \frac{\mathrm{b(x)} - \mathrm{a(x)}}{\max[\mathrm{b(x)}, \mathrm{a(x)}]}$$

$a(x)$ is the distance of a particular data point from the other data points in its cluster, and $b(x)$ is the distance of the same data point from other data points in other clusters (Bakhshizadeh et al., 2022). The highest silhouette score corresponds to the optimal number of clusters. In this case, we have identified and separated the data into 3 clusters.

3.2. Predictive Models for CLV prediction

We have applied Random Forest, XGBoost, and Neural Networks to predict the Customer Lifetime Value (CLV) on the whole data and within each cluster separately.

Random Forest is a modification of the bootstrap aggregation (bagging) technique. The bootstrap aggregation method creates multiple datasets from the available training dataset by sampling with replacement. Therefore, some data points may be repeated and others may be completely ignored. This is how we get “B random but identically distributed bootstrapped datasets”. They are used to train an “ensemble of B base models”. In regression problems the predictions are averaged to predict a numerical value:

$$\widehat{y}_{\mathrm{bag}}\ (\mathbf{x}_\star) = \frac{1}{B}\sum_{b=1}^{B} \tilde{y}^{(b)}\ (\mathbf{x}_\star)$$

(Hastie et al., 2009)

Bagging fits decision trees of the same type and averages the final result. Random Forest builds on this approach by creating a collection of de-correlated trees and adds additional randomness when constructing each tree. It is achieved by picking a random subset of the training data whenever a node is split. Each decision tree will have its subset of training data, which makes the trees less correlated. The averaging afterward is more likely to result in a higher variance reduction. A potential drawback of this technique is that the variance of each tree will be higher because of the random sampling of training data (Lindholm et al., 2022).

The other technique used in the study is boosting, which is an ensemble learning technique mostly used for reducing bias in machine learning models. Unlike bagging, boosting models are trained sequentially, meaning that the outputs of one base model become the inputs of the following model. These data points are chosen by assigning weights $\left\{w_i^{(2)}\right\}_{i=1}^{n}$ to them. The method of calculating the weights and other model specifications depends on the type of boosting (Lindholm et al., 2022). We have used a popular boosting method called eXtreme Gradient Boosting (XGBoost), which builds upon the traditional Gradient Boosting technique. In Gradient Boosting, each decision tree considers only the mistakes from the previous base model and does not take the correct predictions into account. Therefore, the model calculates the sum of the residuals (difference between the predicted and actual values) from every decision tree to evaluate its performance. The goal is to minimize the value of the residuals. XGBoost works similarly but was specifically designed for speed improvement (Wade, 2020).

The objective of XGBoost can be summarized mathematically in the following equation:

$$\mathrm{obj}(\theta) = l(\theta) + \Omega(\theta)$$

The first term is the loss function, which is typically the Mean Squared Error (MSE) in regression problems:

$$l(\theta) = \sum_{i=1}^{n} (y_i - \widehat{y}_i)^2$$

This is the squared distance between the predicted $\widehat{y}_i$ and the actual values y_i. The second term is the built-in regularization term, which distinguishes XGBoost from other types of boosting. It improves accuracy by penalizing complexity to prevent overfitting and can be expressed as:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

The γ term scales the terminal nodes T and thus punishes the tree ensemble complexity. The regularization parameter represented by λ controls the penalty imposed on the squared weights w_j^2. The idea is to penalize larger weights to prevent overfitting. Therefore, we can conclude that XGBoost is a regularized version of Gradient Boosting (Wade, 2020).

We have also used feedforward neural networks - they are models that try to approximate a function $\boldsymbol{y} = f(\boldsymbol{x}; \boldsymbol{\theta})$ by learning the hidden parameters θ. The network is called feedforward because of the direction of the information - it flows from the inputs, through the hidden variables, and to the output (Goodfellow et al., 2016).

A neural network consists of combining multiple functions. Each one represents a hidden layer, whose output is unknown. This is why these layers are called hidden layers. Every layer consists of several hidden units called neurons. The neurons receive input data from the previous units and compute the value of their activation function. The units of a neural network are connected with weights $W_U^{(1)}$ and $W_U^{(2)}$, which are shown by every line in the figure below, and bias terms $b_1^{(1)}$ and $b_2^{(2)}$. The input layer consists of input units x_1 to x_p, one hidden layer q_1 , and the output layer y.

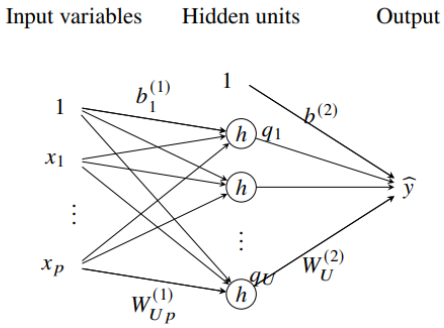


Figure 2. Neural Network Architecture

(Lindholm et al., 2022)

The activation function used in the hidden layers is the non-linear ReLU (rectified linear unit). It is the default function used in feedforward neural networks and is defined in the following way (Goodfellow et al., 2016):

$$g(z) = max(0, z)$$

The equation of the first hidden layer is as follows:

$$\boldsymbol{h} = g\left(\boldsymbol{W}^{\top}\boldsymbol{x} + \boldsymbol{b}\right)$$

(Goodfellow et al., 2016)

The activation function used in the final output layer is defined by the type of problem we are trying to solve. For our thesis, we are using the linear activation function, which is used to calculate the mean of a conditional Gaussian distribution:

$$p(\boldsymbol{y} \mid \boldsymbol{x}) = \mathcal{N}(\boldsymbol{y}; \hat{\boldsymbol{y}}, \boldsymbol{I})$$

(Goodfellow et al., 2016)

3.3. Cross-Validation and Hyperparameter Tuning

To optimize the performance of Random Forest and XGBoost, we have used 5-fold cross-validation. The data is split into k batches of data and the model is trained on $k-1$ batches (one is set aside for validation). The error is calculated for each validation set and the final result is the average of all validation errors. Hence, *k*-fold cross-validation reduces the chance of overfitting (Lindholm et al., 2022).

We can see the parameter descriptions, initialization and the optimal parameters for the Random Forest Regressor below:

- 'n_estimators': the number of trees in the forest.
- 'min_samples_split': minimum number of samples needed to split a node.
- 'min_samples_leaf': minimum number of samples of the data required to be at each node.
- 'max_features': ['auto', 'sqrt'] - maximum number of features before each split. ‘Sqrt’ refers to the squared maximum number of features.

(Sklearn.Ensemble.RandomForestRegressor, n.d.)

Hyperparameter	Initialization	Best parameter

n_estimators	[100, 200, 300, 400, 500, 600]	300
min_samples_split	[5, 6, 7, 8, 9, 10, 15]	5
min_samples_leaf	[5, 6, 7, 8, 9, 10, 15]	5
max_features	['None', 'sqrt', 'log2']	sqrt

Table 1. Best Parameters in Random Forest

However, the hyperparameter tuning for Random Forest did not show improvement in the performance metrics (R^2 and $RMSE$). This is why we have used the default parameters from the Random Forest Regressor.

The following booster parameters have been defined and tuned for the XGBoost Regressor:

- 'n_estimators': number of trees in the forest
- 'max_depth': maximum depth of the trees in the forest.
- 'learning_rate': it defines the step size at which weights are updated. It is tuned to prevent overfitting.
- 'gamma': the minimum decrease in loss required to split a leaf node.
- 'reg_lambda': L2 Regularization term. (XGBoost Parameters, 2022)

The table below shows the initialization and the best parameters chosen.

Hyperparameter	Initialization	Best parameter
n_estimators	[5000, 6000, 7000, 8000]	8000
max_depth	[3, 6, 9]	6
learning_rate	[0.1, 0.01, 0.001]	0.1
gamma	[0, 0.1, 0.2]	0.2
reg_lambda	[1e-6, 1e-4, 0.001, 0.01, 0.1, 1, 10, 100]	1e-6

Table 2. Best Parameters in XGBoost

The XGBoost Regressor is then fit to the training data with the optimal hyperparameters above and evaluated on the test dataset.

In the case of the feedforward neural network, we have experimented with several network architectures. The initial architecture of the neural network consisted of one hidden layer with 128 neurons and an output layer. Early stopping is used as a regularization method. It is a procedure that aims to save the model parameters when the error on the validation set is the lowest. In other terms, the model stops training when the validation error starts to increase (Goodfellow et al., 2016). After this, we experimented with network architecture by gradually adding two more hidden layers with 64 and 32 neurons respectively.

As an additional regularization method, we have added a dropout layer with a 0.5 dropout rate after the first hidden layer. The dropout technique removes different non-output units from the base network by multiplying their output values by 0. Thus, certain units are temporarily removed from the network at each layer. This makes the model more robust as it does not rely on the presence of specific neurons at every update. The dropout rate defines how many units will be dropped from one layer to the next (Goodfellow et al., 2016). This approach further improved the RMSE and the test mean absolute error.

The final model architecture is shown in Table 3.

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 128)	7936
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8256
dense_2 (Dense)	(None, 32)	2080
dense_3 (Dense)	(None, 1)	33

Total params: 18,305
Trainable params: 18,305
Non-trainable params: 0

Table 3. Final Model Architecture of Neural Network

The neural network was compiled with the Adam optimizer, mean squared error as the loss function, and mean absolute error and RMSE as evaluation metrics. The Adam optimizer is an adaptive learning rate algorithm for optimization. It is considered robust to the different choices of hyperparameters (Goodfellow et al., 2016). Overall, a neural network with 3 hidden layers and one dropout layer resulted in the optimal performance of the model.

3.4. Performance Metrics

Based on the related literature, suitable performance metrics for our study are the R^2 and Root Mean Squared Error (RMSE) values. RMSE shows the error square and punishes greater gaps between the actual and predicted values. Therefore, the closer the result is to 0, the better:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}}$$

(Tekin et al., 2021)

R^2 is also known as the coefficient of determination. It is the ratio of variance that can be predicted from the feature variables. Thus, the closer the result is to 1, the better:

$$\mathrm{R}^2 = 1 - \frac{\sum_{i=1} (\hat{y}_i - y_i)^2}{\sum_{i=1} (\bar{y}_i - y_i)^2}$$

(Chicco et al., 2021)

4. Data

The dataset is obtained directly from Kaggle and belongs to an auto insurance company in the USA. The data is recorded for a period of two months between 1st January 2011 and 28th February 2011. It contains information about every policyholder's vehicle, policy type, and socio-economic status. There are 24 columns and 9134 unique values in the dataset. The data has no missing values.

The target variable is Customer Lifetime Value - a numerical value that shows the future value of the customer to the company in USD (Kaggle, 2024). Although CLV is already given in the dataset, one of the common approaches to calculate CLV is cited in Gupta et al. (2006) as shown below:

$$CLV = \sum_{t=0}^{T} \frac{(p_t - c_t)\, r_t}{(1+i)^t} - AC,$$

where 'p_t' is the price paid by consumers at time t, 'c_t' is the cost for servicing the customer, and 'i' is the cost of capital for the company. In addition, 'r_t' is the probability of the customer's repeat purchase, 'AC' is the acquisition cost, and 'T' is the time period to estimate CLV. Overall, this calculation provides a CLV which is the net contribution from the customer throughout the relationship with the company after deducting the cost of servicing and acquiring the customer, and discounting the customer's contribution to the present time period.

The data has 23 features which can be divided into the following groups:

1) **Demographic** - Customer, State, Education, Employment Status, Income, Location Code, Marital Status, Gender

2) **Policy** - Coverage, Effective to Date, Monthly Premium Auto, Months Since Last Claim, Months Since Policy Inception, Number of Open Complaints, Number of Policies, Policy Type, Policy, Renew Offer Type, Total Claim Amount, Response

3) **Vehicle** - Vehicle Class, Vehicle Size

4) **Sales** - Sales Channel

The features have been further explained in Appendix A (Kaggle, 2024).

4.1. Exploratory Data Analysis

The dataset comprises 15 categorical variables and 9 numerical variables. The distribution of 8 numerical variables except ‘Effective To Date’, is shown in Appendix B.

The distribution clearly shows that the target variable - Customer Lifetime Value, is positively skewed. The majority of other numerical variables also have some positive skewness in the distribution. This suggests that scaling the data is essential for further processing. Given the skewness of the data, Customer Lifetime Value was also scaled using log-transformation. However, the log-transformed CLV showed oscillations with multiple peaks. It might mean that the data might have other underlying patterns that log-transformation doesn’t resolve. As a result, log-transformed CLV has been dropped.

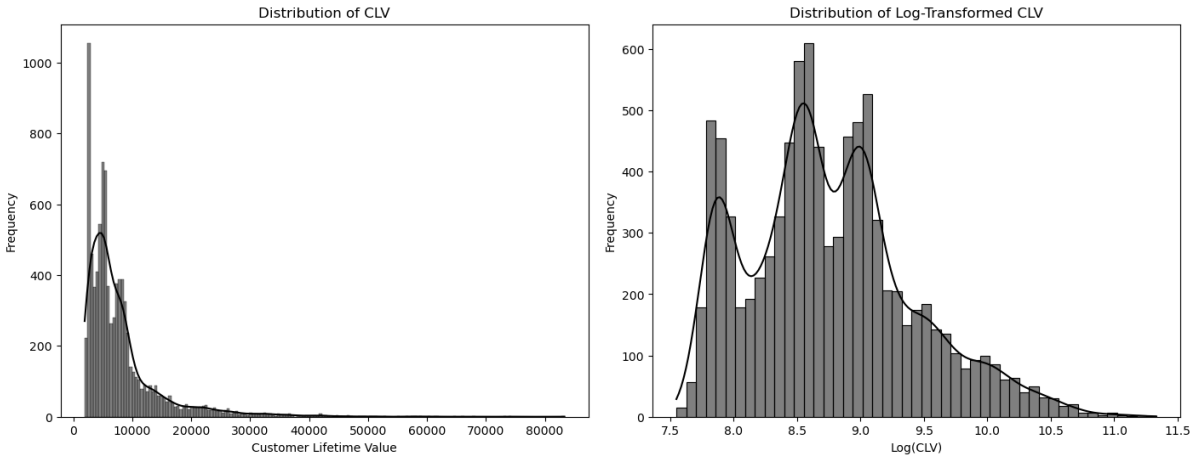


Figure 3. Distribution of CLV and Log-Transformed CLV

In terms of categorical variables, the distribution of 14 variables is shown in Appendix C, except ‘Customer’, which is merely a unique customer ID. The distribution of categorical variables also shows unequal distribution except for gender. The categorical variables have several subcategories; hence, some data cleaning to reduce the number of subcategories is done.

The summary statistics of numerical variables showcase the distribution of data, its average value, and its minimum and maximum value. Features such as ‘Number of Open Complaints’, ‘Months Since Policy Inception’, ‘Number of Policies’, and ‘Months Since Last Claim’ have similar mean and median. However, the average and median values differ mostly in ‘Customer Lifetime Value’, ‘Income’, ‘Monthly Premium Auto’, and ‘Total Claim Amount’. This difference is attributed to the effect of outliers.

	Customer Lifetime Value	Income	Monthly Premium Auto	Months Since Last Claim	Months Since Policy Inception	Number of Open Complaints	Number of Policies	Total Claim Amount
count	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000
mean	8004.940475	37657.380009	93.219291	15.097000	48.064594	0.384388	2.966170	434.088794
std	6870.967608	30379.904734	34.407967	10.073257	27.905991	0.910384	2.390182	290.500092
min	1898.007675	0.000000	61.000000	0.000000	0.000000	0.000000	1.000000	0.099007
25%	3994.251794	0.000000	68.000000	6.000000	24.000000	0.000000	1.000000	272.258244
50%	5780.182197	33889.500000	83.000000	14.000000	48.000000	0.000000	2.000000	383.945434
75%	8962.167041	62320.000000	109.000000	23.000000	71.000000	0.000000	4.000000	547.514839
max	83325.381190	99981.000000	298.000000	35.000000	99.000000	5.000000	9.000000	2893.239678

Table 4. Summary Statistics of Numerical Variables

Similarly, the comparison of CLV mean and median across categorical variables doesn't show a significant impact of outliers on average across sub-categories, except for 'Vehicle Class'. The CLV value across different vehicle classes differs as owners of luxury vehicles pay much larger premiums than others. The boxplot showcasing the relationship of categorical variables with CLV can be seen in Appendix D.

4.2. Feature Importance

While analyzing variables, research usually focuses on studying the association among variables and the strength of that relationship. Spearman Correlation Analysis is calculated with the ranks of the values of two variables rather than their actual values. The coefficient value ranges between -1 to +1. This method is also more robust to outliers (Schober et al., 2018).

The correlation analysis shows that 'Monthly Premium Auto', and the 'Number of Policies' seem to have a higher positive correlation with the target variable Customer Lifetime Value (CLV). Other variables such as 'Income', 'Total Claim Amount', 'Vehicle Class', and 'Coverage' also have some positive correlation with CLV as shown in Appendix E.

Furthermore, the relationship between CLV and other features is analyzed through statistical tests such as ANOVA and Pearson Correlation. CLV has a statistically significant correlation with 'Monthly Premium Auto', 'Total Claim Amount', 'Income', 'Number of Open Complaints', and 'Number of Policies'. The categorical features with statistically significant differences in CLV are 'Coverage', 'Employment Status', 'Marital Status', 'Renew Offer Type', and 'Vehicle Class'.

Thus, these features have a significant effect on CLV. Further analysis of feature importance is done using Random Forest and XGBoost feature importance in section 5.3.1. and 5.3.2.

4.3. Data Preprocessing and Transformation

First, we focused on finding duplicates and missing values. Based on the variable “Customer”, all customer IDs were found to be unique. In addition, there were no missing values in the dataset. Some variables are dropped from further processing such as ‘Customer’, and ‘Effective To Date’ as no additional information is provided by them.

Furthermore, the distribution shown in Appendix C highlights sparse subcategories in categorical variables. Hence, subcategories have been merged to reduce the sparsity and complexity of the model. In ‘Education’, ‘Master’ and ‘Doctor’ are merged into the ‘Master or above’ category. Under ‘Employment Status’, unemployed and employed are the large subcategories. Therefore, other employment statuses such as ‘Medical’, ‘Retired’, and ‘Disabled’ are categorized under ‘Others’.

In addition, we use one-hot encoding, which creates new binary columns for each subcategory of categorical variables. Now, the dataset has 62 columns including the target variable. For scaling the variables, standardization is used. This transforms data to have a mean of 0 and a standard deviation of 1 (Lindholm et al., 2022). Based on the Euclidean distance in the PCA space and a set threshold of mean plus 2 standard deviations, only 281 out of 9134 data points are considered outliers, which comprises around 3.07% of the dataset. As the ratio is very small, we have not removed the outliers, since they contain valuable insights about high-value customers.

In order to test the model performance of machine learning algorithms, it is an important step to split the dataset into a training and test set. In this thesis, the dataset has been split into 80% for the training dataset and 20% for the test dataset.

5. Empirical Analysis

This section outlines the results of the customer segmentation methods and machine learning models. It includes principal component analysis (PCA), *k*-means clustering, tree-based models, and neural networks. We will show the process of selecting the optimal number of clusters for the dataset and the results from the models based on R^2 and $RMSE$.

5.1. Principal Component Analysis

We have used the explained variance ratio to define the optimal number of principal components for dimensionality reduction. The figure below shows the explained variance (information) ratio by every component. The first principal component preserves the observations in which there is most variance. It is constructed in the direction in which the projections are as close as possible to the original data. The second principal component retains less variance in the data than the first one.

We have selected 20 components to visualize and see their corresponding ratio. The figure shows that two principal components manage to capture most of the variance in the data. Hence, we have chosen to apply dimension reduction to a two-dimensional space.

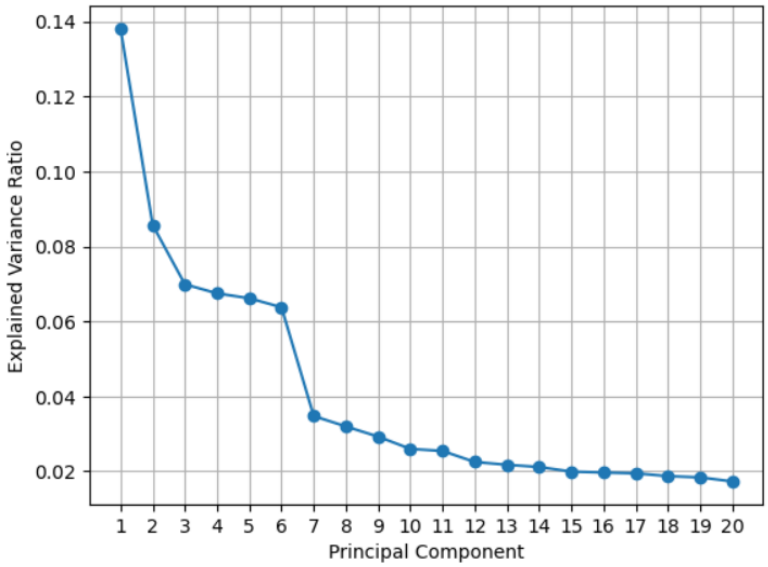


Figure 4. Explained variance ratio per component

As a result, we have plotted the distribution of the PCA-reduced data points in two dimensions below:

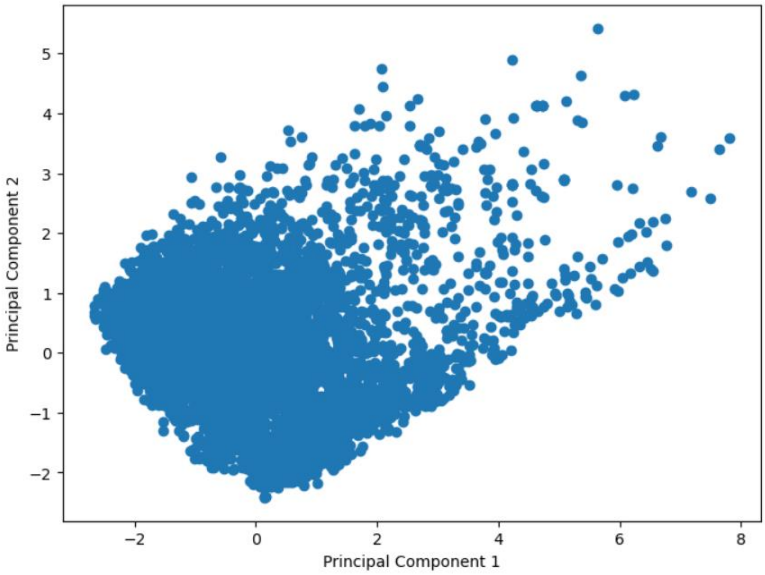


Figure 5. PCA-reduced Data Points

Feature	PC1 - Importance	PC2 - Importance
Total Claim Amount	0.636275	0.137117
Monthly Premium Auto	0.486770	0.571182
Income	0.416416	0.630183
Location Code_Suburban	0.191544	0.135841
EmploymentStatus _Employed	0.182135	0.276341
EmploymentStatus _Unemployed	0.153959	0.220087

Table 5. Principal Component Loadings

Table 5. shows the features with the highest loading scores in the principal components. These are the coefficients that show the degree of influence of the specific variable on that principal component. Their values range from -1 to 1: a positive loading signals that a variable

contributes to the component, and a negative value shows that the absence of that variable influences the component (Harvey & Hanson, 2024). There are no negative values in the scores which means that all features have a certain degree of influence on the principal component. Based on the table above, 'Total Claim Amount', 'Monthly Premium Auto', and 'Income' are the most important features in the first principal component. The second one is mostly influenced by 'Income', 'Monthly Premium Auto', and 'EmploymentStatus_Employed'. Therefore, these are the features causing most of the variance in the data. We can also conclude that they are the key drivers of the differences between our clusters.

5.2. *k*-Means clustering

The data is clustered based on all features to preserve as much of the information in the characteristics as possible. The elbow method based on the sum of squared error (SSE) and the silhouette score have been used to define the optimal number of clusters on the PCA-reduced data.

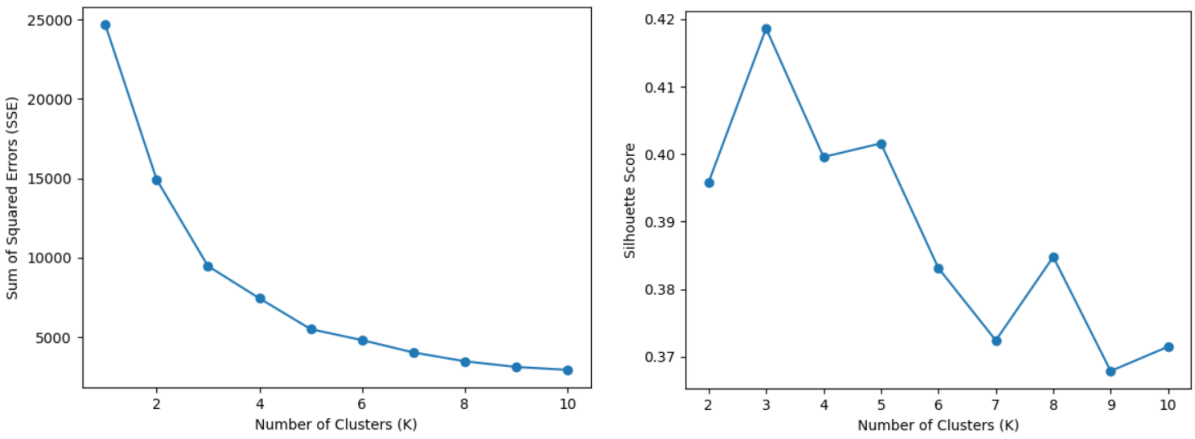


Figure 6. Elbow plot (left) and Silhouette score plot (right)

In Table 6. you can see the summarized SSE indexes and silhouette scores.

K	**SSE**	**Silhouette Score**
1	24698.211763	NaN
2	14926.884671	0.395827
3	**9467.435969**	**0.418598**
4	7437.113709	0.399559

5	5491.317032	0.401572
6	4797.512418	0.383110
7	4020.478922	0.372396
8	3465.014912	0.384765
9	3106.127988	0.367862
10	2924.303957	0.371475

Table 6. Summary of SSE indexes and Silhouette scores

Based on the results, the SSE index of 9467.44 and the corresponding silhouette score of 0.42 show the optimal number of clusters in the data, which is 3. The three clusters are visually represented on the graph below. We can see that all clusters are very distinctly separated. Therefore, we can conclude that a method of hard clustering such as *k*-means clustering is appropriate for this dataset.

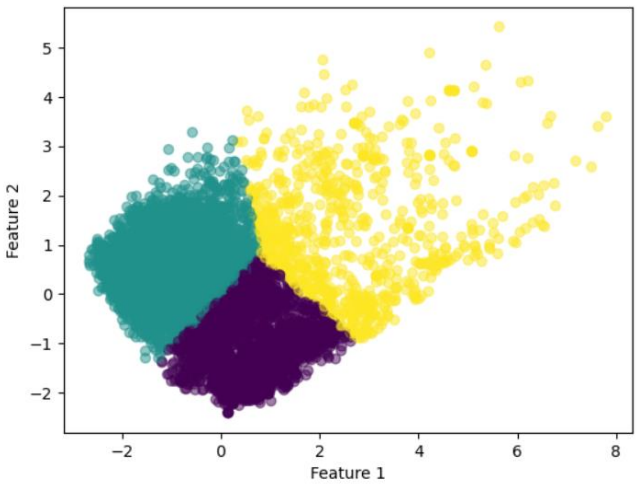


Figure 7. *k*-Means Clustering

The distribution of the data points and the corresponding percentage for each cluster is the following:

Cluster	Count	Percentage
1	3688	50.47
2	2772	37.94
3	847	11.59

Table 7. Distribution of Data Points on each Cluster

Cluster 1 is the biggest cluster with approximately half of the data points classified there, followed by Cluster 2 with almost 37.9% and Cluster 3 with 11.6%, which is the smallest in the data.

5.2.1. Cluster Characteristics

Grouping customers into three separate clusters will help us design specific strategies for customer retention. Before understanding the differences among clusters, there are some common customer characteristics that stand out in the data. Around 62% of the company's current customers are based in California and Oregon which also accounts for around 63% of total CLV. In addition, personal auto policy is the most popular policy type accounting for 74% of all policy types sold by the company. The majority of the customers (63%) are from the suburban population. In addition, the majority of them are approached by a sales agent in comparison with other sales techniques. In regards to vehicle class, four-door cars followed by SUVs contribute the highest to the customer lifetime value for the company. Customers in the three clusters also have a similar number of open policies with the company (on average, 2-3). You can see the customer-specific descriptions for every cluster below.

1) Cluster 1

The customers in cluster 1 are the ones with the highest income with a mean value of about 38 700 USD. They have the least number of open policies and complaints. In terms of demographics, the majority of the customers are married and employed. They need the policy for a four-door car with Basic coverage and the most popular offer among them is Offer 1. In this cluster, customers pay the least monthly premium auto and the total claim amount is slightly below the mean value for the feature in this cluster. The customer lifetime value in this cluster is around 8 000 USD and the maximum value is approximately 67 900 USD. The cluster mean is slightly below the mean value for the feature. This positions it as the second cluster out of the three clusters based on mean CLV.

2) Cluster 2

In terms of demographics and vehicle type, this cluster is similar to the first one. However, this is the cluster with the lowest average customer lifetime value - approximately 7 950 USD. An interesting observation is that the maximum CLV value in the cluster is higher than the one of the first cluster - around 73 200 USD. The total claim amount paid by the customer in USD is

also the lowest out of the three clusters. This is also the cluster with the most months passed since the last claim and the biggest number of months since policy inception. The income level and the monthly premium auto are similar to the ones in the first cluster.

3) Cluster 3

Customers in cluster 3 have the highest customer lifetime value (around 8 300 USD), which is above the mean for the feature in the cluster. The maximum CLV value in this cluster is around 83 300 USD. Intuitively, customers in this cluster also have the highest monthly premium auto payments and the most number of policies with the insurance company. They have also paid the most in total claim amount (USD) and have opened the biggest number of complaints. They seem to be more engaged in their relationship with the company as they have recently opened their policies and the lowest number of months has passed after their last claim. In terms of demographics, this cluster is identical to the previous two. However, customers in this cluster have also insured a bigger number of SUVs (apart from four-door cars) than the other two. Also, more customers have an Extended coverage type compared to Cluster 1 and 2. A controversial observation in this cluster is that the customers have the lowest income compared to the other two clusters (about 35 900 USD).

Overall, the customers in cluster 3 seem to be the highest-paying customers with the biggest estimated future profit that they will bring to the company. Table 8. shows the mean values for some features in each cluster.

	Customer Lifetime Value	Income	Monthly Premium Auto	Months Since Last Claim	Months Since Policy Inception	Number of Open Complaints	Number of Policies	Total Claim Amount	State_Arizona	State_California	...	Vehicle Class_Four-Door Car
Cluster mean	-0.001225	0.034567	-0.004822	-0.004115	0.000956	0.001774	0.002834	-0.001801	0.205267	0.287157	...	0.503968
Cluster mean	-0.007154	0.003359	-0.001417	0.009429	0.003846	0.001899	0.011885	-0.019227	0.200108	0.297451	...	0.504610
Cluster mean	0.047513	-0.056606	0.029005	-0.032721	-0.011708	0.035566	0.013166	0.051422	0.192444	0.289256	...	0.519481

	Vehicle Class_Luxury Car	Vehicle Class_Luxury SUV	Vehicle Class_SUV	Vehicle Class_Sports Car	Vehicle Class_Two-Door Car	Vehicle Size_Large	Vehicle Size_Medsize	Vehicle Size_Small
Cluster mean	0.016595	0.020202	0.191919	0.053752	0.213564	0.095960	0.703463	0.200577
Cluster mean	0.017082	0.022234	0.192245	0.056670	0.207158	0.106562	0.702278	0.191161
Cluster mean	0.022432	0.017710	0.213695	0.044864	0.181818	0.103896	0.714286	0.181818

Table 8. Mean Values of each Cluster for some Features

5.3. Model Performance

We have trained Random Forest, XGBoost, and Neural Networks both on the complete training data and separately on the three clusters we have previously defined.

5.3.1. Random Forest

Based on Grid Search Cross Validation, the following hyperparameters were chosen as optimal: 'max_features': 'sqrt', 'min_samples_leaf': 5, 'min_samples_split': 5, 'n_estimators': 300 (see Chapter 3: Methodology). The tuned model's performance achieved an R^2 = 0.56 and $RMSE$ = 0.694 on the test data. However, the Random Forest Regressor with its default parameters managed to achieve a better result - R^2 = 0.687 and $RMSE$ = 0.585 on the testing set. This is why we have used the default model within every cluster.

After using clustering to build on the model performance, we have seen a significant improvement in the performance metrics. The data points in every cluster were split into training and test data (80/20) and the random forest regressor was fit to the data. The advantage of this approach is that the model is trained on more homogenous data points within each cluster.

Random Forest				
	Cluster 1	**Cluster 2**	**Cluster 3**	**Overall Scores**
R^2	0.957	0.952	0.956	**0.955**
RMSE	0.167	0.19	0.335	**0.198**

Table 9. Model Performance of Random Forest after Clustering

The R^2 score increased by almost 27% and the $RMSE$ result dropped by 39%. Therefore, we can see a notable improvement in the model's performance when applied separately to each cluster. This is mostly because we are training the model on fewer and more homogenous data points within each cluster. There is less noise in the data and the results are more interpretable.

Random Forest's feature importances gives us useful insights into which features have the most significant impact on the Customer Lifetime Value (CLV). The key features for the RF model

are ‘Number of Policies’ and ‘Monthly Premium Auto’. The more policies a customer has and the more their monthly payment is, the bigger their CLV value becomes.

Feature	**Importance**
Number of Policies	0.472144
Monthly Premium Auto	0.249078
Months Since Last Claim	0.036929
Total Claim Amount	0.036448
Months Since Policy Inception	0.033995
Income	0.030177
Number of Open Complaints	0.005631
Education_High School or Below	0.005169
Policy_Personal L2	0.004043

Table 10. Features Importance with Random Forest

5.3.2. XGBoost

Based on hyperparameter tuning, the following parameters were chosen as the optimal ones: 'gamma': 0.2, 'learning_rate': 0.1, 'max_depth': 6, 'n_estimators': 8000, 'reg_lambda': 1e-06 (see Chapter 3: Methodology).

The model achieved an R^2 = 0.681 and $RMSE$ = 0.590, which shows a similar performance to the Random Forest model on the whole data. However, we can see a significant improvement when the model is applied separately to each cluster:

XGBoost				
	Cluster 1	**Cluster 2**	**Cluster 3**	**Overall Scores**
R^2	0.773	0.795	0.919	**0.801**
RMSE	0.385	0.392	0.457	**0.397**

Table 11. Model Performance of XGBoost after Clustering

The R^2 value increased by 12% and the $RMSE$ score decreased by 19%.

Similarly to Random Forest, ‘Number of Policies’ and ‘Monthly Premium Auto’ are the most important features of the model. Then, we see different features ranked as important in the creation of more decision trees. This model also considered the ‘Vehicle Size_Medsize’, ‘Coverage_Basic’, and ‘Sales Channel_Branch’ as key features.

Feature	**Importance**
Number of Policies	0.311062
Monthly Premium Auto	0.050787
Vehicle Size_Medsize	0.038238
Coverage_Basic	0.035372
Sales Channel_Branch	0.023465
Policy_Corporate L1	0.023017
Response_No	0.021914
Gender_F	0.020388
Vehicle Class_Luxury Car	0.020339

Table 12. Features Importance with XGBoost

Based on the two feature importances tables, we can conclude that the Random Forest model considered more numerical features to be important for making the decision trees and affecting the target. On the other hand, XGBoost focused more on the policy and vehicle characteristics.

Overall, Random Forest and XGBoost have a similar performance on the whole dataset. However, there is a greater difference when the models are trained separately on each cluster. Random Forest exhibits a much better performance than XGBoost. This is probably due to the averaging effect of the Random Forest model, which further helps to prevent overfitting.

5.3.3. Neural Networks

The neural network described in the Methodology section was also trained both on the whole data and cluster-specific data points. The neural network trained on the full dataset achieved an R^2 = 0.778 and $RMSE$ = 0.492 on the test data. The mean squared error and mean absolute error scores are 0.242 and 0.245, respectively. The loss scores are very low which shows a good model performance.

The table below shows a summary of the results per cluster.

Neural Network				
	Cluster 1	**Cluster 2**	**Cluster 3**	**Overall Scores**
R^2	0.821	0.810	0.874	**0.822**
RMSE	0.342	0.377	0.569	**0.386**

Table 13. Model Performance of Neural Network

As in the tree-based models, we see an improvement when the neural network is trained per cluster. There is a 4% increase in the R^2 score and a 11% decrease in the $RMSE$ result.

5.4. Model Comparison

Based on the tested models, we can see that the Random Forest shows the best performance on both metrics. Even though Neural Networks are more flexible, they usually require large datasets to be able to generalize well. In addition, Random Forest is better at capturing non-linear relationships than XGBoost. The averaging effect of the decision trees also contributes to the largely improved model performance and ability to generalize on new unseen data. The table below summarizes the results from the best models we have applied.

Best Models (per cluster results)			
	Random Forest	**XGBoost**	**Neural Network**
R^2	**0.955**	0.801	0.822
RMSE	**0.198**	0.397	0.386

Table 14. Overall Model Performance

6. Data-Driven Business Decisions

Before analyzing cluster-specific strategies, some general customer characteristics can also support in making better business-level strategies. In terms of the company's customer base, it is evident that over 60% of the current customers are based in California and Oregon, so more than 50% of the marketing budget should be allocated to target customers in these states as they contribute to over 50% of customer lifetime value for the company. The main focus should be to retain consumers in these regions by targeted marketing efforts based on specific cluster characteristics. In addition, upselling of personal auto policy can bring in more revenue for the company as it already accounts for over 70% of CLV. Upselling includes enhancing coverage limits, additional assistance, multi-vehicle discounts, etc. with an increase in premium.

Furthermore, as suggested by Curiskis et al. (2023), CLV-based metric to calculate return on investment (ROI) on marketing campaigns can also support the company to be more data-driven and optimize their marketing efforts to profitable customer clusters and channels. The company can focus more on the suburban customer base which accounts for 63% of the total existing customers by upgrading existing policy, conducting surveys, and hosting information sessions, to showcase the company's interest to meet the customer needs of this area.

Additionally, the cluster characteristics can help devise more specific retention and acquisition strategies in relation to high-value and mid and low-value customers.

Based on the cluster characteristics, customers from the low to mid-value segments are more passive in their relationship with the auto insurance company. They may not see an exceptional value in the offered services, but they keep renewing out of habit or necessity. As indicated by our research findings, engaging with a sales agent proves to be the most effective sales technique, since customers perceive a sense of care and receive personalized service compared to online interactions. Therefore, for mid and low-value clusters, assigning each customer a dedicated insurer who closely monitors their account and communicates important updates, such as expiration dates or new services, fosters a better connection and personalized approach that improves the customer experience. This approach not only reassures customers that their accounts are well-managed but also encourages their continued loyalty. This will outweigh the potential costs associated with researching alternative options. To enhance their engagement and maximize value, upselling personal auto policy in the mid-CLV segment should also be

considered as this cluster comprises individuals with the highest income. This could prove beneficial without necessarily introducing entirely new coverage options. For the low-value segment, regular communication and notification on policy updates can also enhance their relationship with the company. However, further personalized marketing efforts should be focused on the high-value cluster.

Cluster 3 is the premium cluster in our data. Its customers have the highest customer lifetime value and some of the highest CLV values in the data. These customers also have the largest monthly premium auto fees and more policies open with the company. A potential strategy to retain customers in this high-value segment is to create a loyalty club membership (Bakhshizadeh et al., 2022). Providing these customers with exclusive services and targeted promotions/discounts is a way to show them that the company values their relationship. Also, the additional services that they will have the opportunity to use will be much more discounted in comparison with requesting them from another insurance company. Quick communication with the company is another unique selling point for this cluster. The loyal customers should be able to establish faster and more timely communication with their insurers if they have any queries or requests (Bakhshizadeh et al., 2022). For instance, the company can automate the renewal process. If a customer agrees to an automatic renewal, minimal effort is required to renew the policy. Upon receiving a renewal notification, they only need to update vehicle or customer details on their online account if necessary, after which the company can proceed with the automatic renewal of their policy.

7. Conclusion

The objective of the study is to compare the performance of ML models on the complete dataset and within individual clusters. Additionally, we have identified the most valuable customer segment, important features that influence CLV, and the most suitable retention strategies for each cluster.

The dataset used for this study contains two months of customer data from an auto insurance company in the USA. Based on the correlation analysis, 'Monthly Premium Auto' and 'Number of Policies' have the most positive relation with CLV. In addition, the statistical significance test showed that 'Number of Open Complaints', 'Total Claim Amount', and 'Income' are also important in the data. Among the categorical features, 'Coverage', 'Employment Status', 'Marital Status', 'Renew Offer Type', and 'Vehicle Class' seem to be the most significant.

We have applied Principal Component Analysis (PCA) to reduce the dimensionality of the data, and *k*-means clustering to perform customer segmentation. This allows us to identify three as the optimal number of clusters in the data based on the elbow method with the sum of squared error (SSE) indexes and the silhouette score. The 3 clusters range from low, to mid and high business value. The third cluster comprises customers with the biggest CLV value (on average, 8 300 USD). The mean values of clusters 1 and 2 are similar (on average, 8 000 USD and 7 950 USD). This is why we have designed similar customer retention strategies for these two clusters under Business Applications.

We have evaluated tree-based models (Random Forest and XGBoost) and Neural Networks. Each of the three models showed significantly improved performance when applied separately to each cluster and averaged, in contrast to when applied across all data points in the dataset. This improvement probably results from the more homogenous data points within every cluster. However, one disadvantage to this approach is that it uses less data for model training. On the other hand, when models are trained on the entire dataset, more data is utilized, potentially enhancing the model's generalizability.

In terms of model performance, Random Forest showed the highest R^2 = 0.955 and the lowest $RMSE$ = 0.198 scores. In terms of model feature importance, Random Forest and XGBoost also took 'Number of Policies' and 'Monthly Premium Auto' as the most important features in

the models. The rest of the Random Forest feature importances were ‘Months Since Last Claim’ and ‘Total Claim Amount’, while XGBoost focused on the vehicle and coverage type. In summary, RF mostly considered the numerical features as important, and XGBoost took the vehicle and policy characteristics into account.

Overall, employing unsupervised learning for pattern recognition, such as *k*-means clustering, before model training yields improved CLV predictions compared to training and running ML models on the entire dataset.

7.1. Limitations and Future Research

One of the limitations of this study is the time period of the data. The data only spans for a two-month period which might not fully represent the customer's relationship with the company. It further hinders discovering any seasonal patterns. In addition, the analysis is limited to the available features and precalculated CLV. Hence, it might be interesting to see if the models yield similar findings for other auto insurance datasets.

Despite the limitations, this study is a good starting point for evaluating and predicting CLV in the insurance industry. For further exploration, other researchers can also try more machine learning models and clustering techniques. The findings from this study can also be adopted in other insurance industries to see if using clustering techniques helps improve the machine learning models’ performances. Another potential direction for future research is to perform feature engineering and then try to cluster customers based on high-value or low-value features.

References

Bakhshizadeh, E., Aliasghari, H., Noorossana, R. & Ghousi, R. (2022). Customer Clustering Based on Factors of Customer Lifetime Value with Data Mining Technique (Case Study: Software Industry), *International Journal of Industrial Engineering & Production Research*, vol. 33, no. 1, pp.18–33

Berger, P. D. & Nasr, N. I. (1998). Customer Lifetime Value: Marketing Models and Applications, *Journal of Interactive Marketing*, vol. 12, no. 1, pp.17–30

Burelli, P. (2019). Predicting Customer Lifetime Value in Free-to-Play Games, in G. Wallner (ed.), *Data Analytics Applications in Gaming and Entertainment*, 1st edn, Series: Data analytics applications: Auerbach Publications, pp.79–107

Chamberlain, B. P., Cardoso, Â., Liu, C. H. B., Pagliari, R. & Deisenroth, M. P. (2017). Customer Lifetime Value Prediction Using Embeddings, in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.1753–1762, Available Online: https://dl.acm.org/doi/10.1145/3097983.3098123

Chan, C. (2008). Intelligent Value-Based Customer Segmentation Method for Campaign Management: A Case Study of Automobile Retailer, *Expert Systems with Applications*, vol. 34, no. 4, pp.2754–2762

Chen, P. P., Guitart, A., Del Rio, A. F. & Perianez, A. (2018). Customer Lifetime Value in Video Games Using Deep Learning and Parametric Models, in *2018 IEEE International Conference on Big Data (Big Data)*, pp.2134–2140, Available Online: https://ieeexplore.ieee.org/document/8622151/

Chen, S. (2018). Estimating Customer Lifetime Value Using Machine Learning Techniques, in C. Thomas (ed.), *Data Mining*, InTech, Available Online: http://www.intechopen.com/books/data-mining/estimating-customer-lifetime-value-using-machine-learning-techniques

Cheng, C.-H. & Chen, Y.-S. (2009). Classifying the Segmentation of Customer Value via RFM Model and RS Theory, *Expert Systems with Applications*, vol. 36, no. 3, pp.4176–4184

Chiang, L.-L. & Yang, C.-S. (2018). Does Country-of-Origin Brand Personality Generate Retail Customer Lifetime Value? A Big Data Analytics Approach, *Technological Forecasting and Social Change*, vol. 130, pp.177–187

Chicco, D., Warrens, M. J. & Jurman, G. (2021). The Coefficient of Determination R-Squared Is More Informative than SMAPE, MAE, MAPE, MSE and RMSE in Regression Analysis Evaluation., *PeerJ. Computer science*, vol. 7, p.e623

Curiskis, S., Dong, X., Jiang, F. & Scarr, M. (2023). A Novel Approach to Predicting Customer Lifetime Value in B2B SaaS Companies, *Journal of Marketing Analytics*, vol. 11, no. 4, pp.587–601

Desirena, G., Diaz, A., Desirena, J., Moreno, I. & Garcia, D. (2019). Maximizing Customer Lifetime Value Using Stacked Neural Networks: An Insurance Industry Application, in *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pp.541–544, Available Online: https://ieeexplore.ieee.org/document/8999077/

Everitt, B. S., Landau, S., Leese, M. & Stahl, D. (2011). Cluster Analysis, Chichester: John Wiley & Sons, Ltd, Available Online: https://onlinelibrary.wiley.com/doi/book/10.1002/9780470977811

Goodfellow, I., Bengio, Y. & Courville, A. (2016). Deep Learning, MIT Press, Available Online: https://www.deeplearningbook.org/

Gupta, S., Hanssens, D., Hardie, B., Kahn, W., Kumar, V., Lin, N., Ravishanker, N. & Sriram, S. (2006). Modeling Customer Lifetime Value, *Journal of Service Research*, vol. 9, no. 2, pp.139–155

Harvey, D. T. & Hanson, B. A. (2024). Understanding Scores and Loadings

Hastie, T., Tibshirani, R. & Friedman, J. (2009). The Elements of Statistical Learning, [e-book] New York, NY: Springer New York, Available Online: http://link.springer.com/10.1007/978-0-387-84858-7

Jasek, P., Vrana, L., Sperkova, L., Smutny, Z. & Kobulsky, M. (2018). Modeling and Application of Customer Lifetime Value in Online Retail, *Informatics*, vol. 5, no. 1, pp.2

Kaggle. (2024). IBM Watson Marketing Customer Value Data, Available Online: https://www.kaggle.com/datasets/pankajjsh06/ibm-watson-marketing-customer-value-data/data

Kim, S.-Y., Jung, T.-S., Suh, E.-H. & Hwang, H.-S. (2006). Customer Segmentation and Strategy Development Based on Customer Lifetime Value: A Case Study, *Expert Systems with Applications*, vol. 31, no. 1, pp.101–107

Lindholm, A., Wahlström, N., Lindsten, F. & Schön, T. B. (2022). Machine Learning: A First Course for Engineers and Scientists, 1st edn, Cambridge University Press, Available Online: https://www.cambridge.org/highereducation/product/9781108919371/book

Markoulidakis, I., Rallis, I., Georgoulas, I., Kopsiaftis, G., Doulamis, A. & Doulamis, N. (2021). Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem, *Technologies*, vol. 9, no. 4, p.81

Najib, M., Pratomo, E. A. & Mulyati, H. (2019). Customer Segmentation Analysis Based on the Customer Lifetime Value Method, *Jurnal Aplikasi Manajemen*, vol. 17, no. 3, pp.408–415

Nekooei, A. & Tarokh, M. J. (2015). Customer Clustering Based on Customer Lifetime Value: A Case Study of an Iranian Bank, *ITRC*, vol. 7, no. 2, pp.71–90

Omran, M. G. H., Engelbrecht, A. P. & Salman, A. (2007). An Overview of Clustering Methods, *Intelligent Data Analysis*, vol. 11, no. 6, pp.583–605

Rachid, A. D., Abdellah, A., Belaid, B. & Rachid, L. (2018). Clustering Prediction Techniques in Defining and Predicting Customers Defection: The Case of E-Commerce Context, *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 4, p.2367

Schober, P., Boer, C. & Schwarte, L. A. (2018). Correlation Coefficients: Appropriate Use and Interpretation, *Anesthesia & Analgesia*, vol. 126, no. 5, pp.1763–1768

Sklearn.Model_selection.GridSearchCV. (n.d.)., Available Online: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

Srinivasan, R., Rajeswari, D. & Elangovan, G. (2023). Customer Churn Prediction Using Machine Learning Approaches, in *2023 International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF)*, pp.1–6, Available Online: https://ieeexplore.ieee.org/document/10083813/

Sun, Y., Liu, H. & Gao, Y. (2023). Research on Customer Lifetime Value Based on Machine Learning Algorithms and Customer Relationship Management Analysis Model, *Heliyon*, vol. 9, no. 2, p.e13384

Tekin, A. T., Kaya, T. & Cebi, F. (2021). Customer Lifetime Value Prediction for Gaming Industry: Fuzzy Clustering Based Approach, *Journal of Intelligent & Fuzzy Systems*, vol. 42, no. 1, pp.87–96

Umargono, E., Suseno, J. E. & Vincensius Gunawan, S. K. (2020). K-Means Clustering Optimization Using the Elbow Method and Early Centroid Determination Based on Mean and Median Formula:, in *Proceedings of the 2nd International Seminar on Science and Technology (ISSTEC 2019)*, Available Online: https://www.atlantis-press.com/article/125944915

Understanding the Classification Report through Sklearn. (2023). Available Online: https://muthu.co/understanding-the-classification-report-in-sklearn/

Ventosa, J. C. (2021). Models to Improve Customer Retention, Available Online: https://www.kaggle.com/code/juancarlosventosa/models-to-improve-customer-retention [Accessed 15 May 2024]

XGBoost Parameters. (2022). Available Online: https://xgboost.readthedocs.io/en/latest/parameter.html#general-parameters

Wade, C. (2020). Hands-On Gradient Boosting with XGBoost and Scikit-Learn: Perform Accessible Machine Learning and Extreme Gradient Boosting with Python, First published., Birmingham: Packt

Win, T. T. & Bo, K. S. (2020). Predicting Customer Class Using Customer Lifetime Value with Random Forest Algorithm, in *2020 International Conference on Advanced Information Technologies (ICAIT)*, pp.236–241, Available Online: https://ieeexplore.ieee.org/document/9261792/

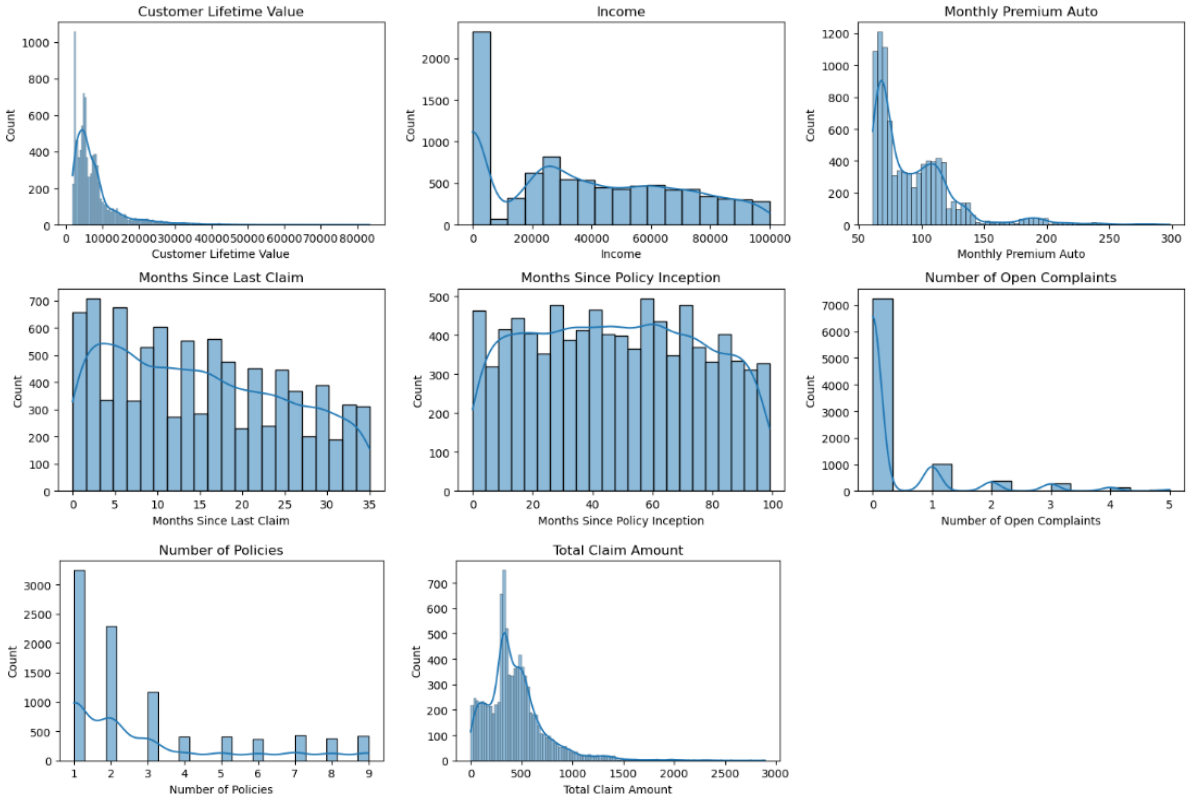
Appendices

A. Variable Description

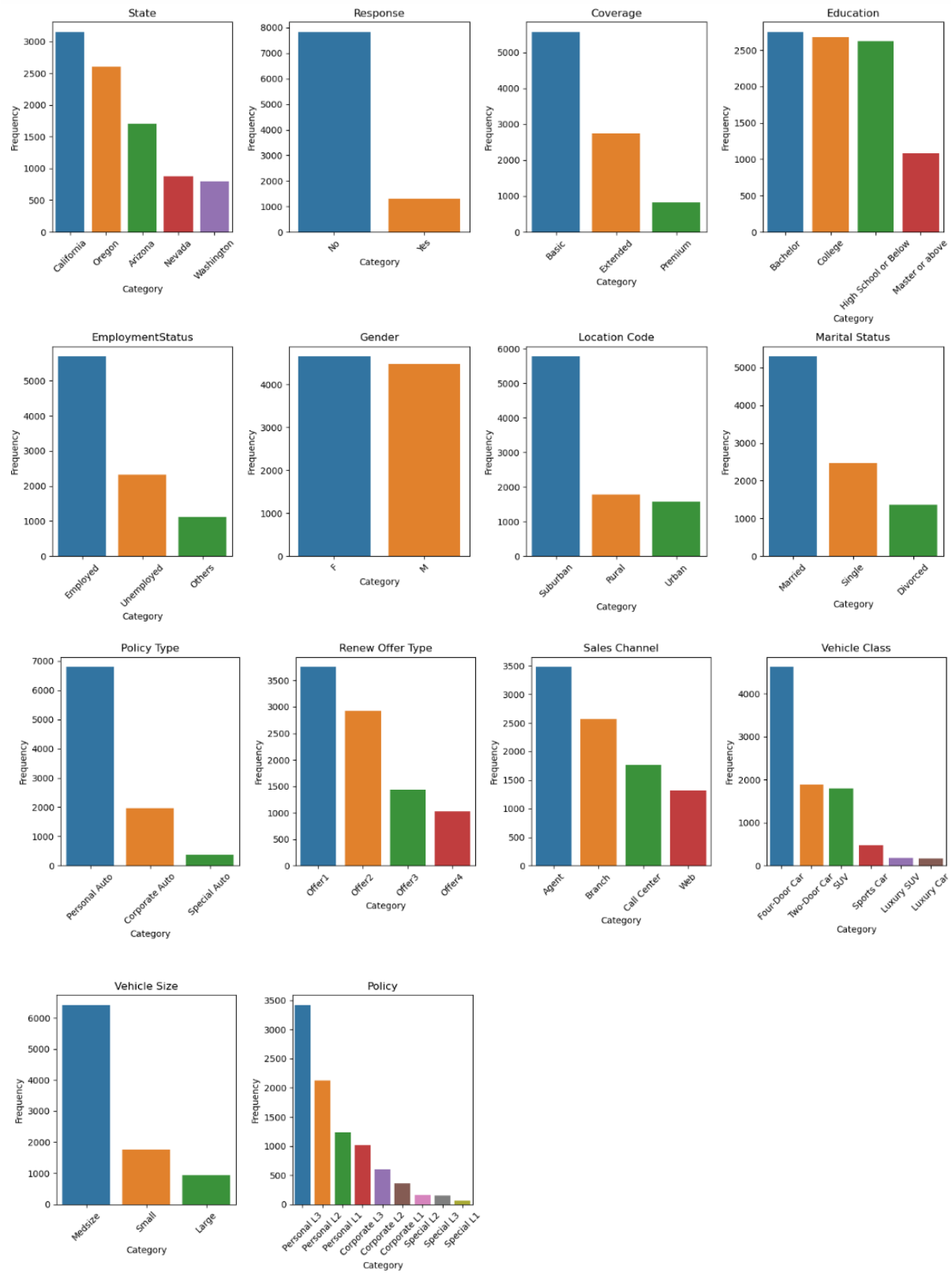
Variable Name	Description	Data Type
Customer	Unique customer code	Categorical
State	Customer's state	Categorical
Customer Lifetime Value	Future value of the customer to the company in USD	Numerical
Response	Customer's response to renewal offer: Yes/No	Categorical
Coverage	Type of insurance package	Categorical
Education	Customer's level of education	Categorical
Effective to Date	Policy's date of expiry	Date
Employment Status	Customer's employment status	Categorical
Gender	Customer's gender	Categorical
Income	Customer's Annual Income	Numerical
Location Code	Customer's location area	Categorical
Marital Status	Customer's marital status	Categorical
Monthly Premium Auto	Average monthly payment	Numerical
Months Since Last Claim	Number of months since the last claim	Numerical
Months Since Policy Inception	Number of months after the start of the policy	Numerical
Number of Open Complaints	Number of complaints opened by the customer	Numerical
Number of Policies	Number of active policies under the same customer	Numerical
Policy Type	Type of policy: Personal Auto, Corporate Auto, Special Auto	Categorical
Policy	Policy number	Categorical
Renew Offer Type	Type of offer for renewal	Categorical
Sales Channel	How the customer was contacted	Categorical
Total Claim Amount	Cumulative amount of claims in USD	Numerical
Vehicle Class	Type of vehicle	Categorical
Vehicle Size	Size of vehicle	Categorical

(Ventosa, 2021)

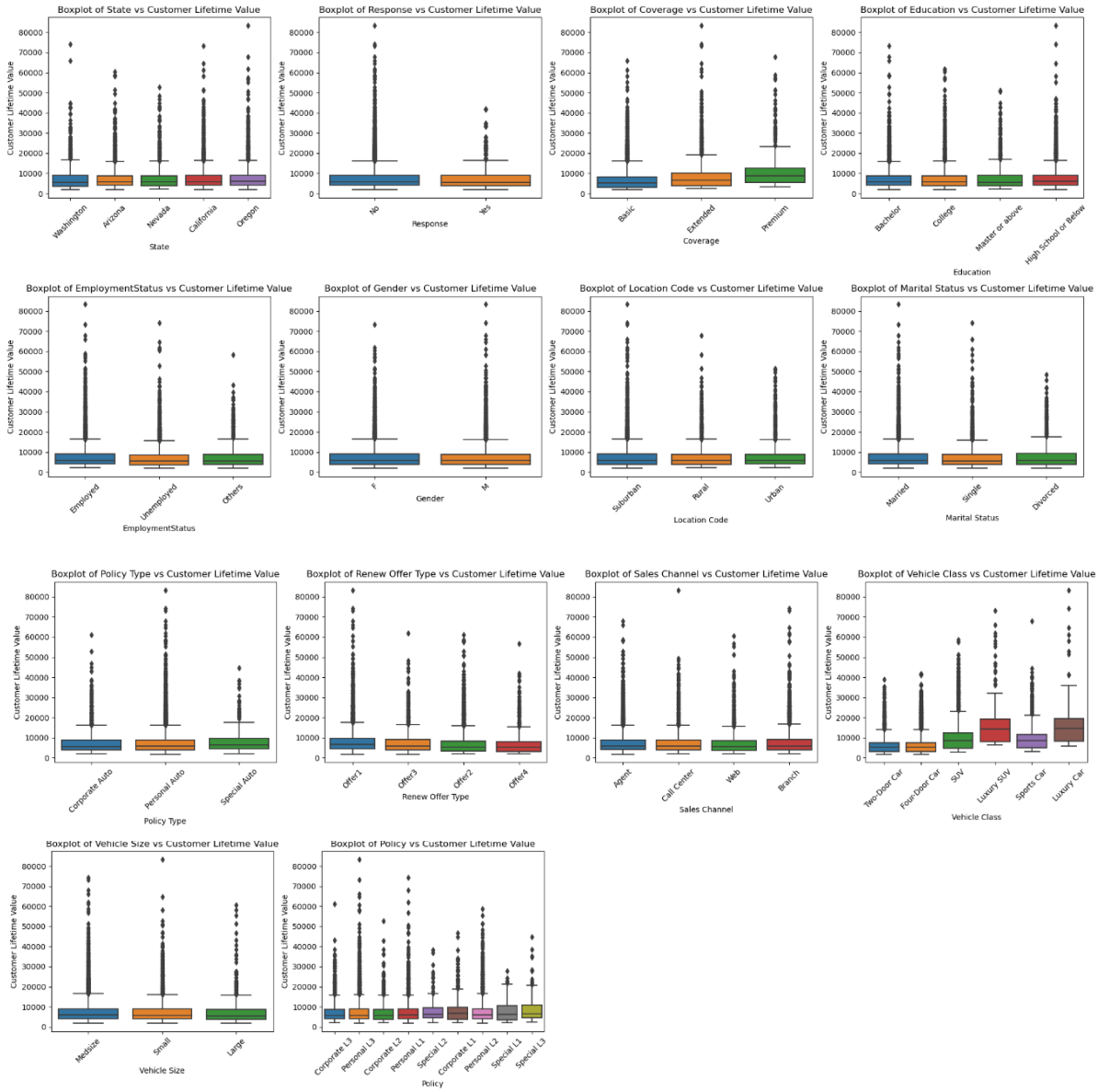
B. Distribution of Numerical Variables



C. Distribution of Categorical Variables



D. Boxplot showcasing the relationship between categorical variables and CLV



E. Correlation Analysis