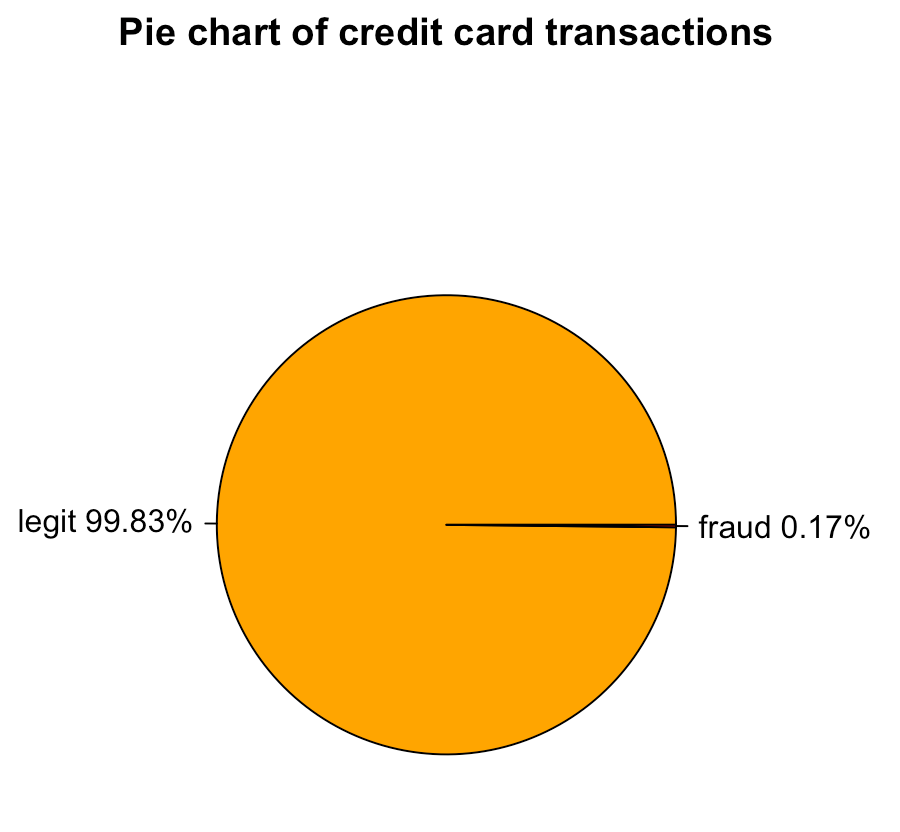
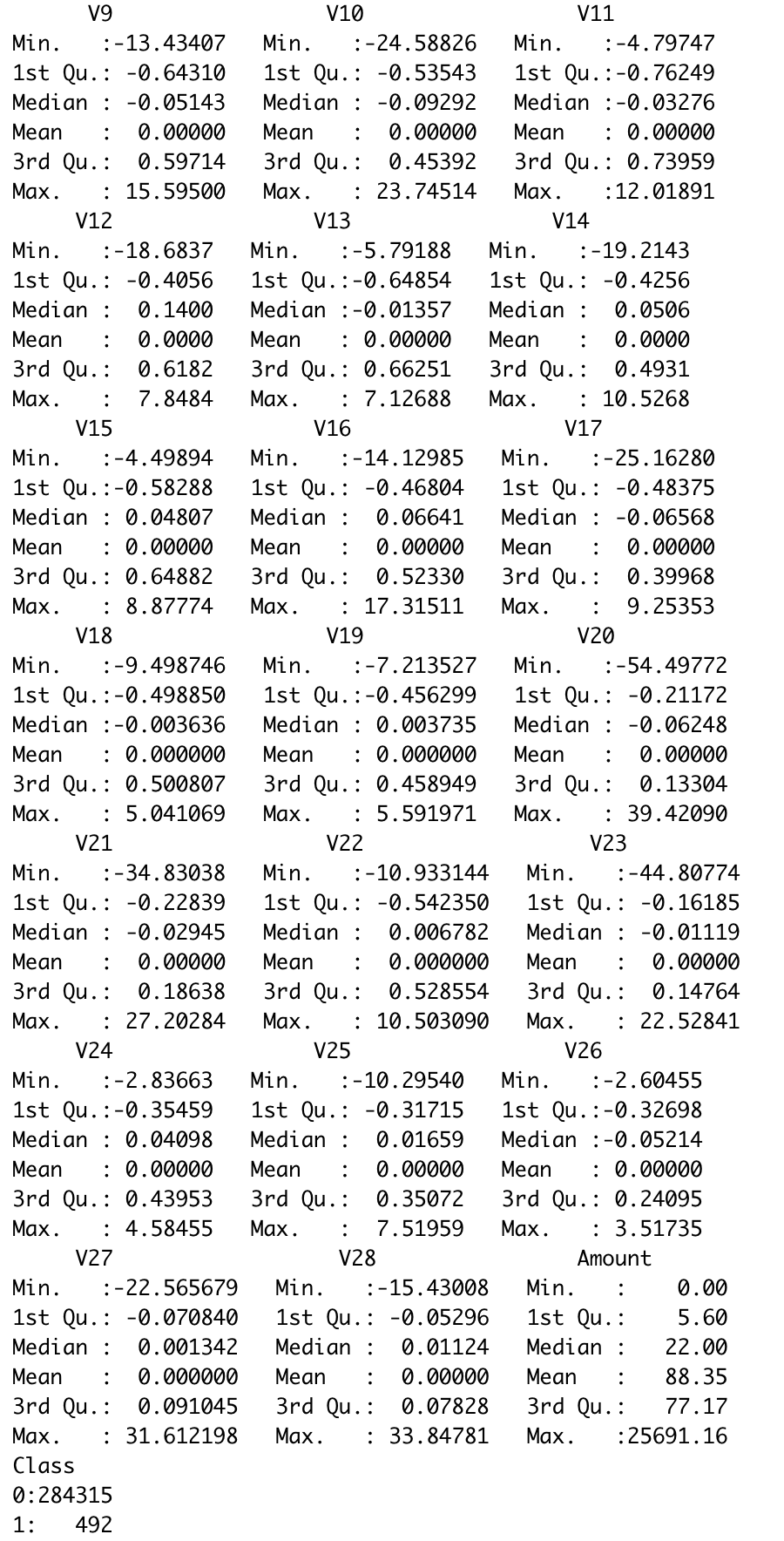
Credit Card Fraud Detection in Machine Learning

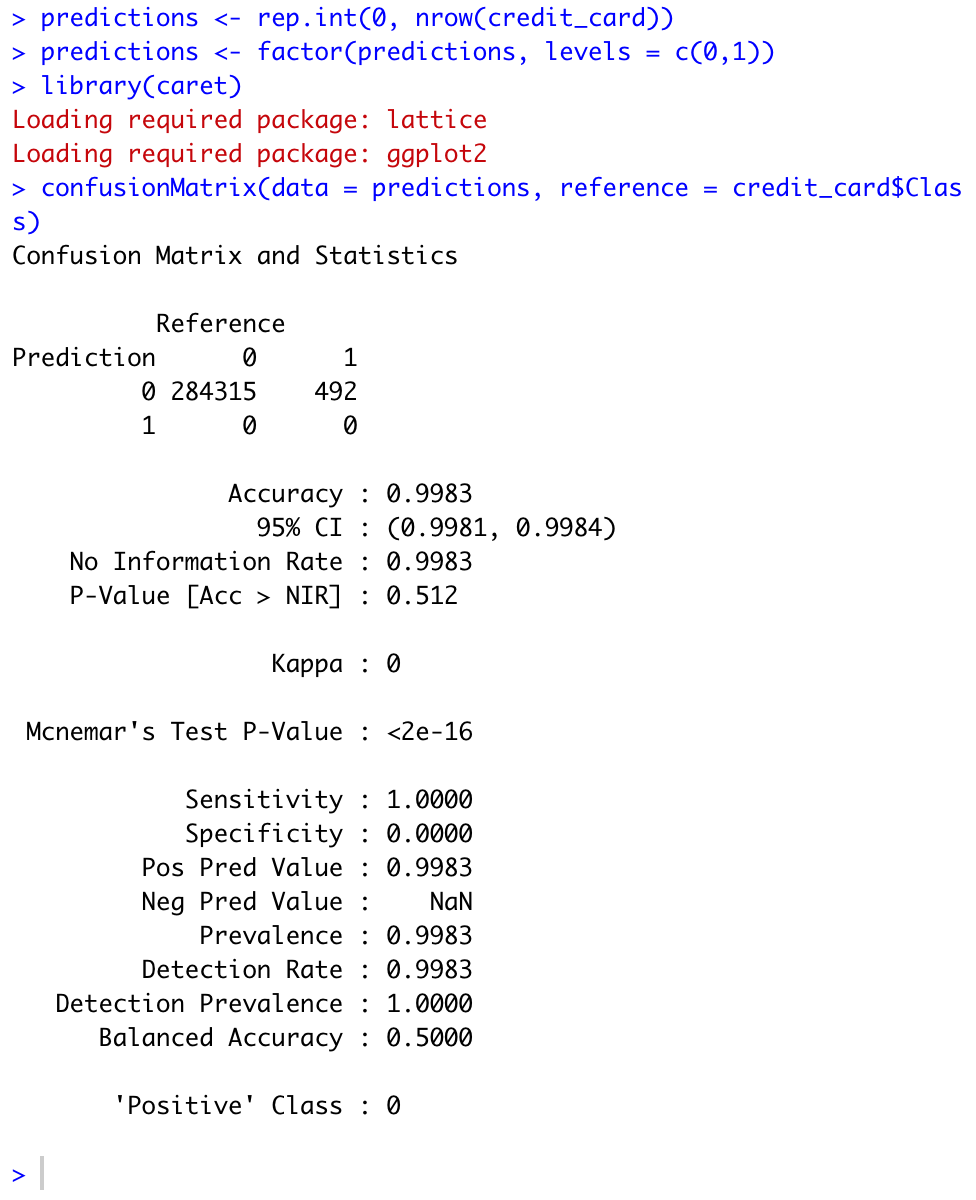
The dataset used in this project is from September 2013 credit card transactions by European cardholders. The list of transactions comes from a two-day duration. There are 492 fraudulent transactions from the total of 284,807. The dataset contains numerical input variables which are the result of a Principal Component Analysis (PCA) transformation. V1 – V28 are the principal components. Time is the seconds between each transaction. Amount is the transaction amount. When working to identify fraudulent transactions we are dealing with a classification task in which each sample transaction has predefined labels tells if it is fraud (1) or legitimate (0). The challenge when dealing with this dataset is that it is highly unbalanced, less than 1% of transactions are fraudulent. Classifiers usually favor the majority class which will lead to incorrect flagging and this may lead to the loss of real customers. Classifiers learn better from balanced data. Using R, and various packages included in the R code file, three sampling methods were performed. Sampling was used on the training set to build a model then used on the unbalanced test set. Random Over-Sampling (ROS) is done by increased the number of fraudulent transactions by duplication. The issue with this method is that creating duplicates already present does not explain variance in the dataset. Random Under-Sampling (RUS) is done by decreasing the number of legitimate cases to get a balanced distribution. The issue with this method is that we throw away a large amount of useful data. In order to solve issues with these methods Synthetic Minority Over-Sampling Technique (SMOTE) was performed on the dataset. SMOTE is done by oversampling the fraudulent transactions by creating synthetic fraudulent transactions. The method finds n nearest neighbor, randomly chooses one of the neighbors and adds a synthetic sample that lies on the line that joins two points.

Beginning with the importing the dataset, the class must be converted to a factor variable.

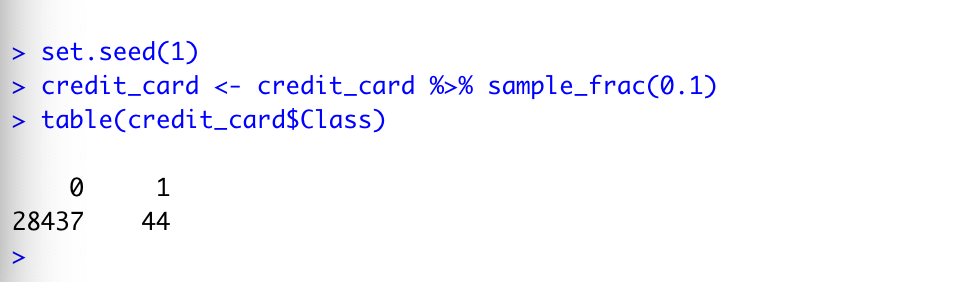


This shows how unbalanced the dataset is. This must be addressed.

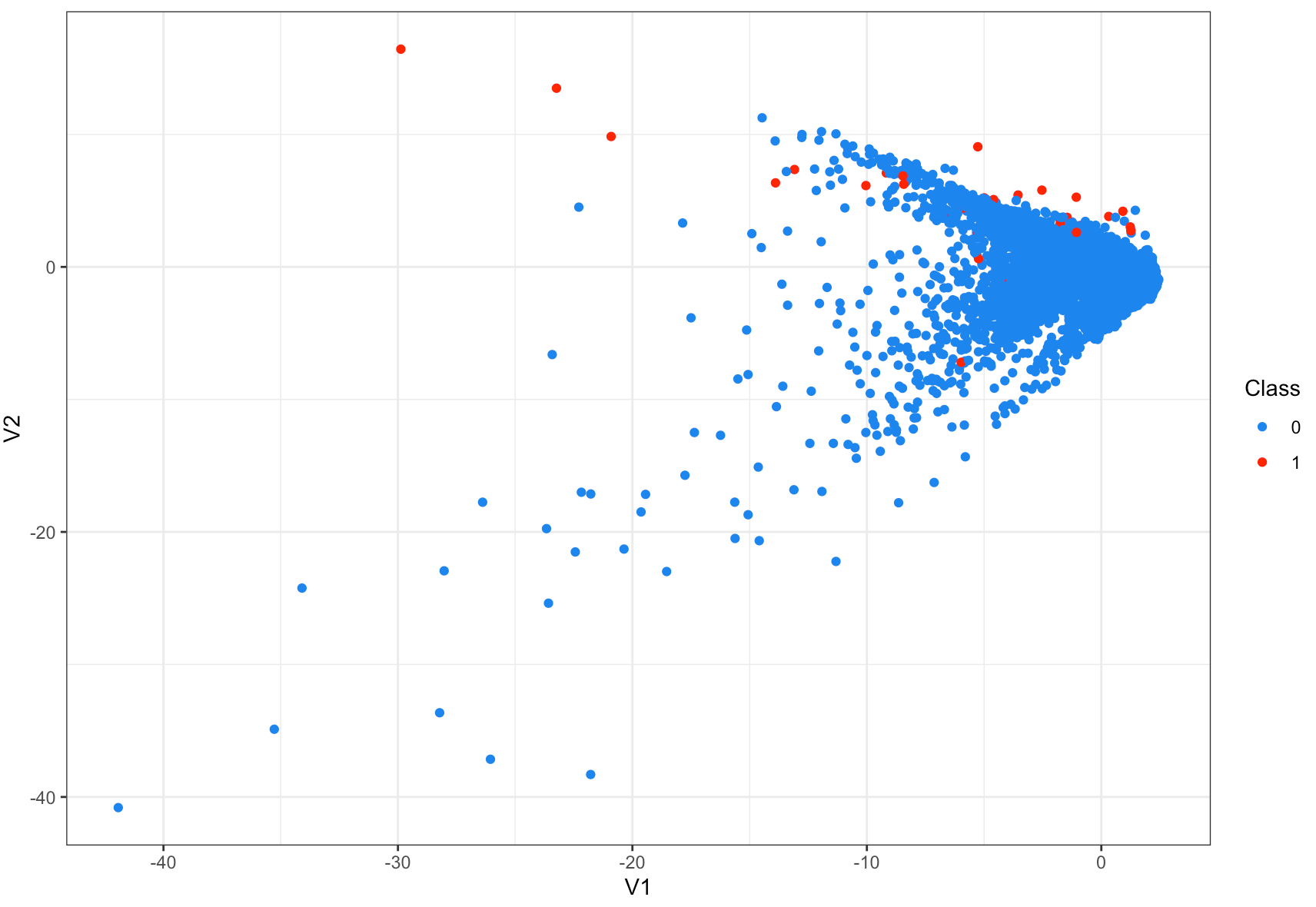
First a no model prediction was preformed, this flagged all the transactions are legitimate. The goal is to maximize true negative cases in order to classify fraud transactions. The package caret was used to build a confusion matrix.



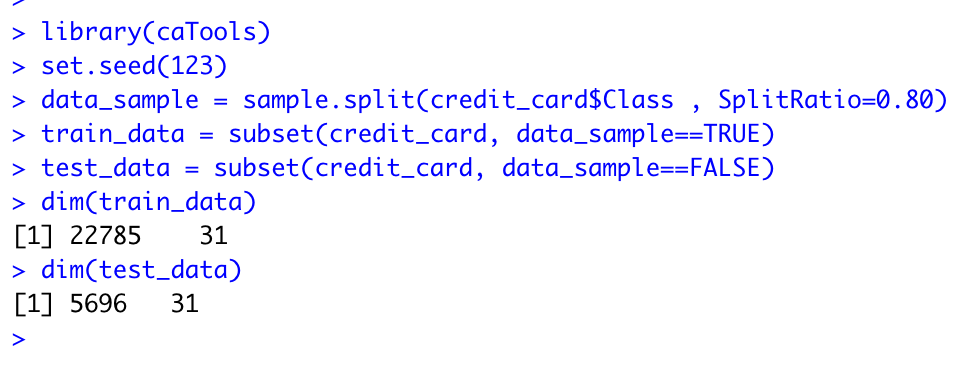
A smaller version of the dataset was taken in order to compute faster. Ten percent of the dataset was used. The package dplyr was used in order to achieve this.



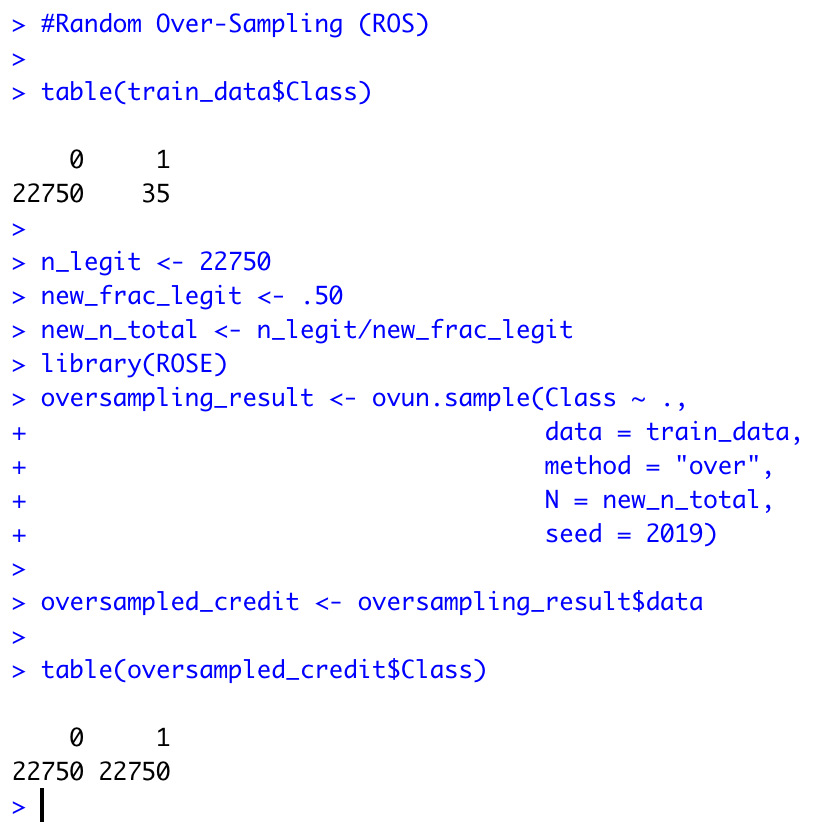
A scatterplot was created using the ggplot2 package. The x-axis is V1 and the y-axis is V2. Observe the imbalance of the points.



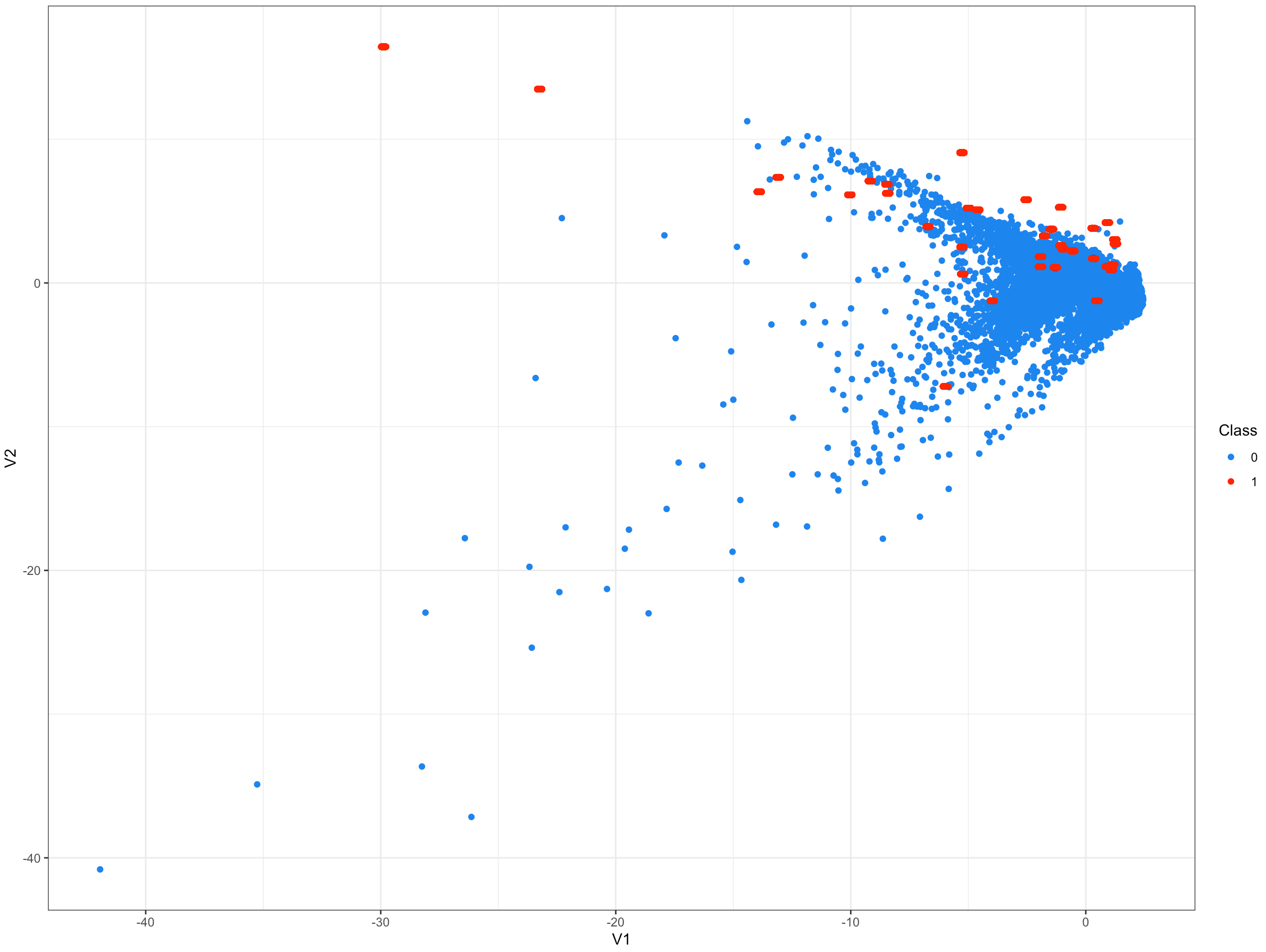
Balancing of the dataset is only done on the training set. The training and test sets are built using the package caTools. Eighty percent of the rows will be the train set and twenty percent will be in the test set.

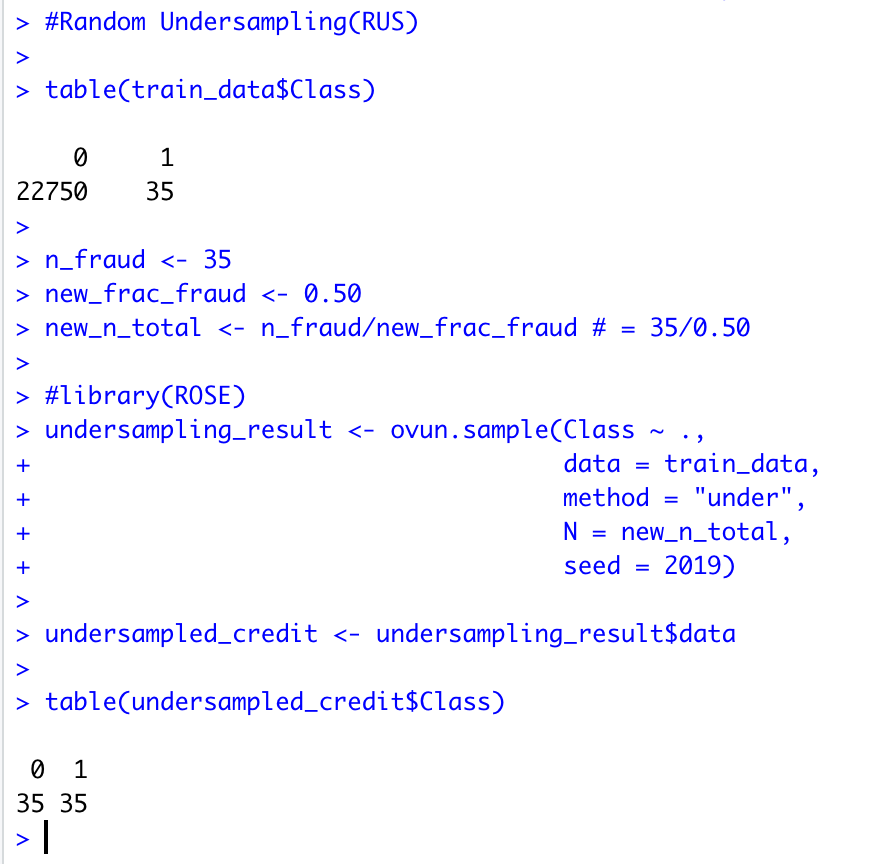


Preforming Random Over-Sampling on the training set was done using the ROSE package. Fifty percent of the rows will be legitimate transactions and fifty percent will be fraudulent transactions. The dataset is now balanced.

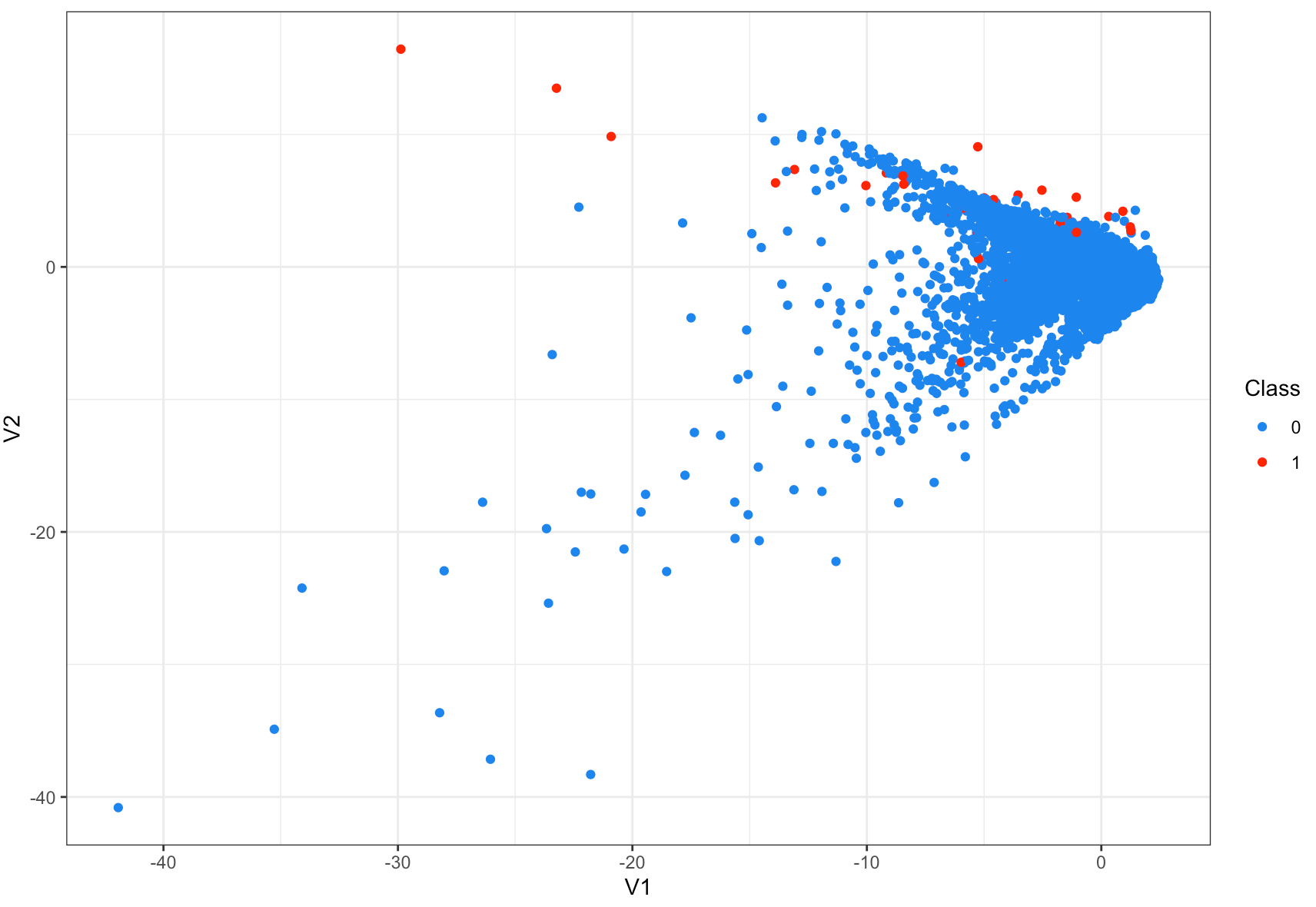
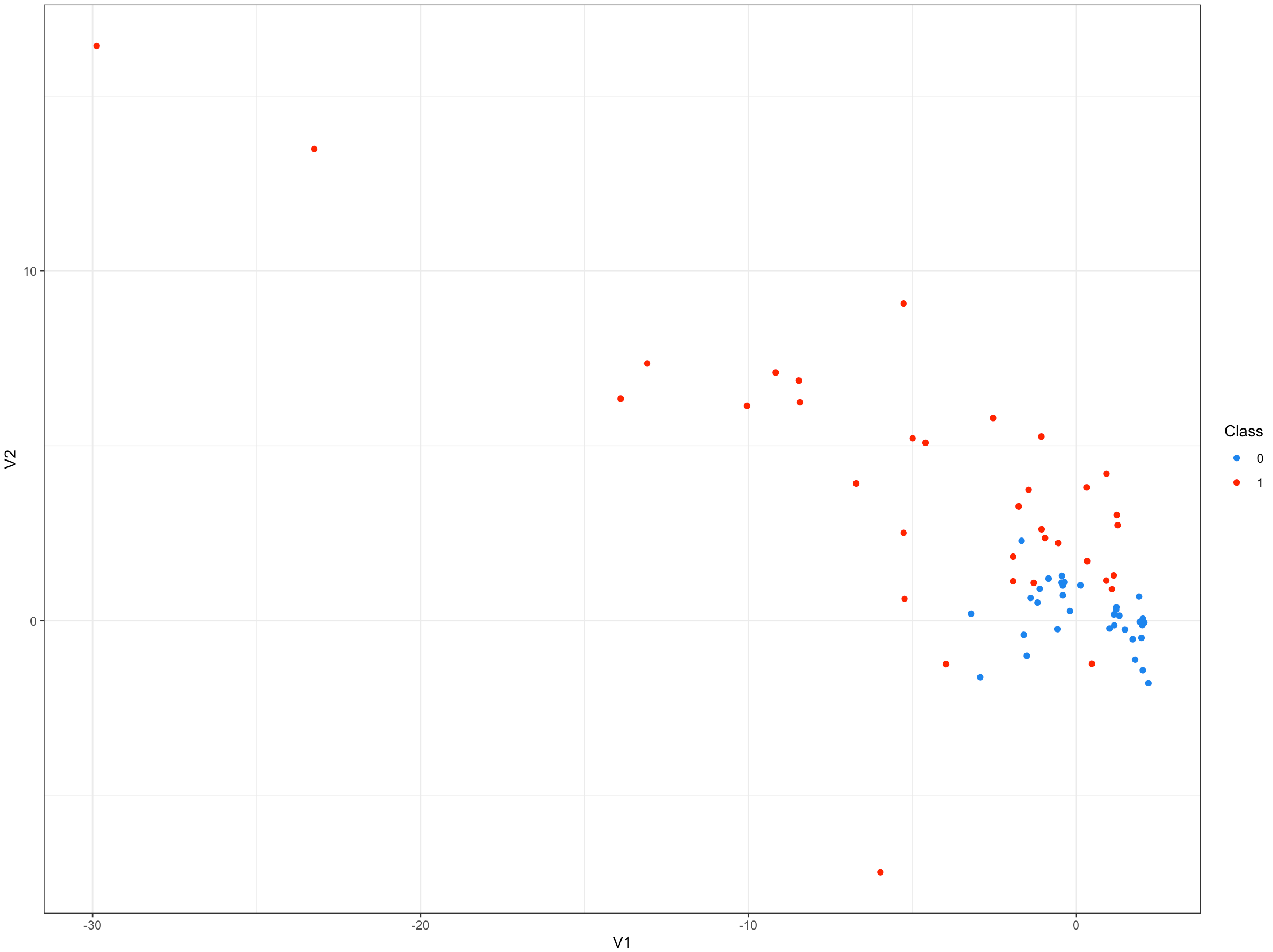


Using jitter in order to see the points stacked on top of each other because they are duplicated.

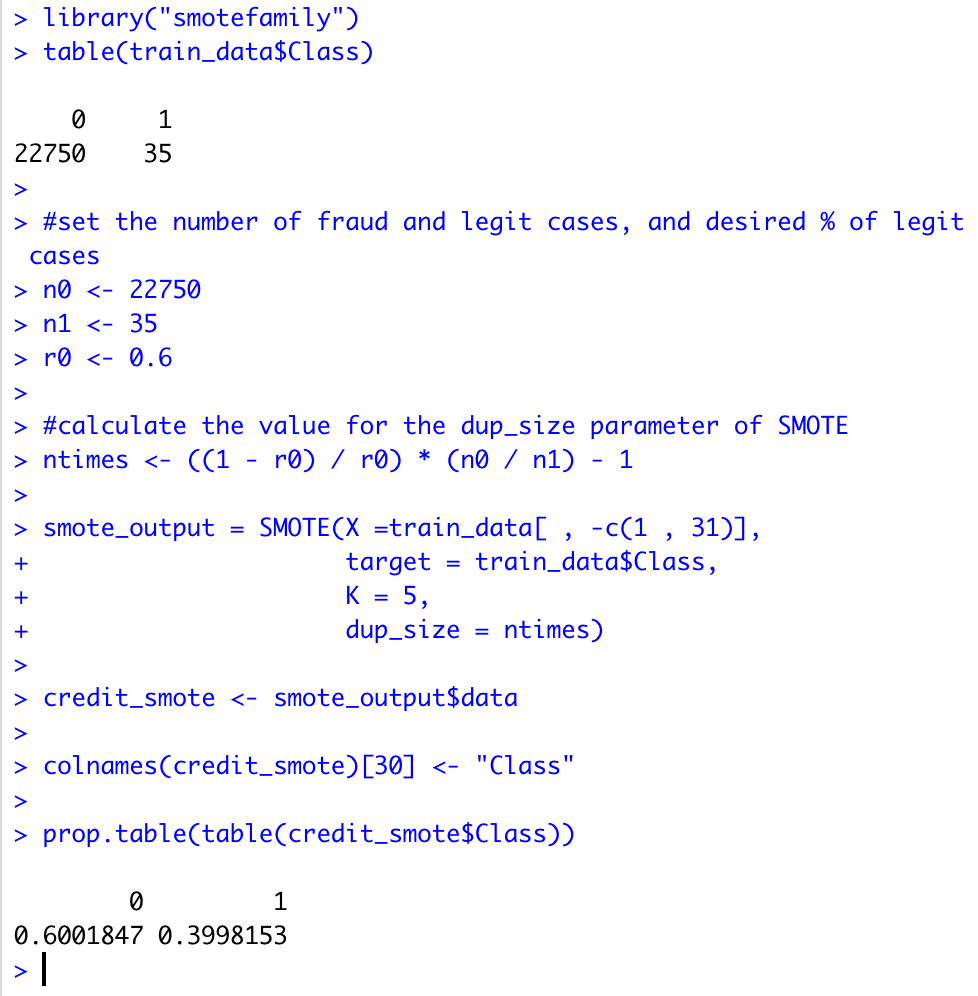


Preforming Random Under-Sampling on the training set was done using the same ROSE package and changing the method from “over” to “under”. 

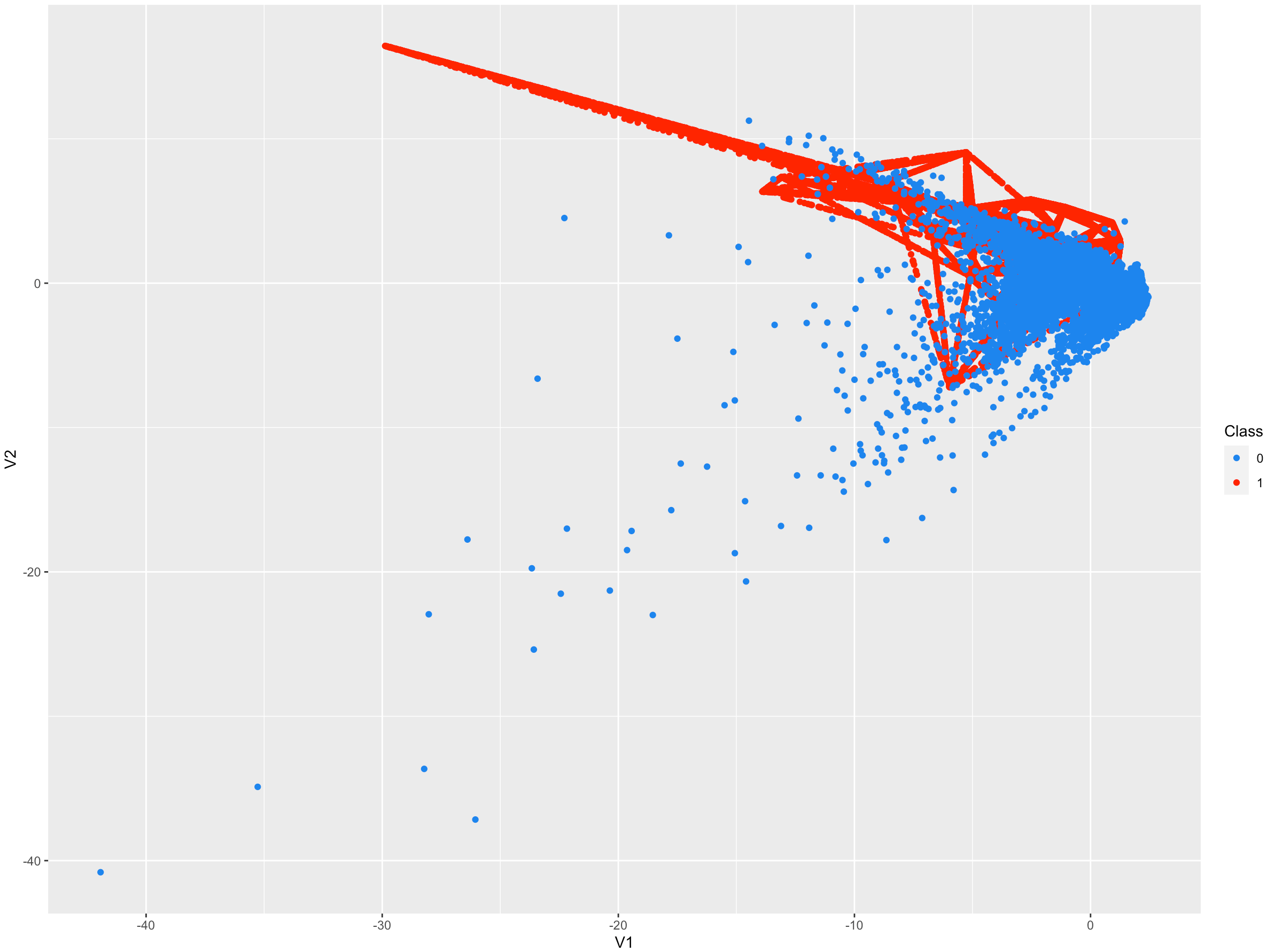
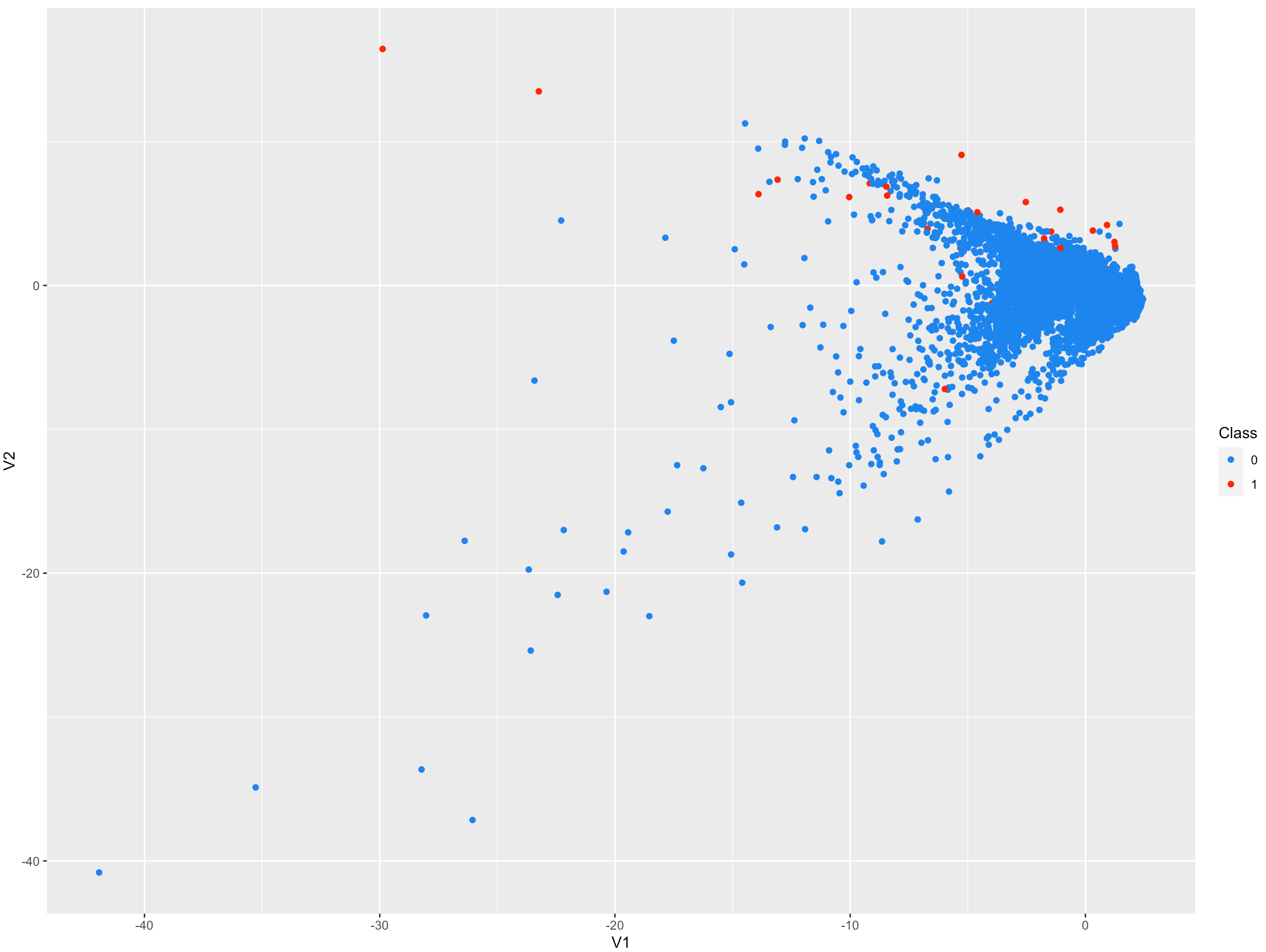
The data is equally distributed. Removing a very large amount of data is not ideal. As seen in the first scatter plot, the dataset looks very different and is not a good representation of overall data.



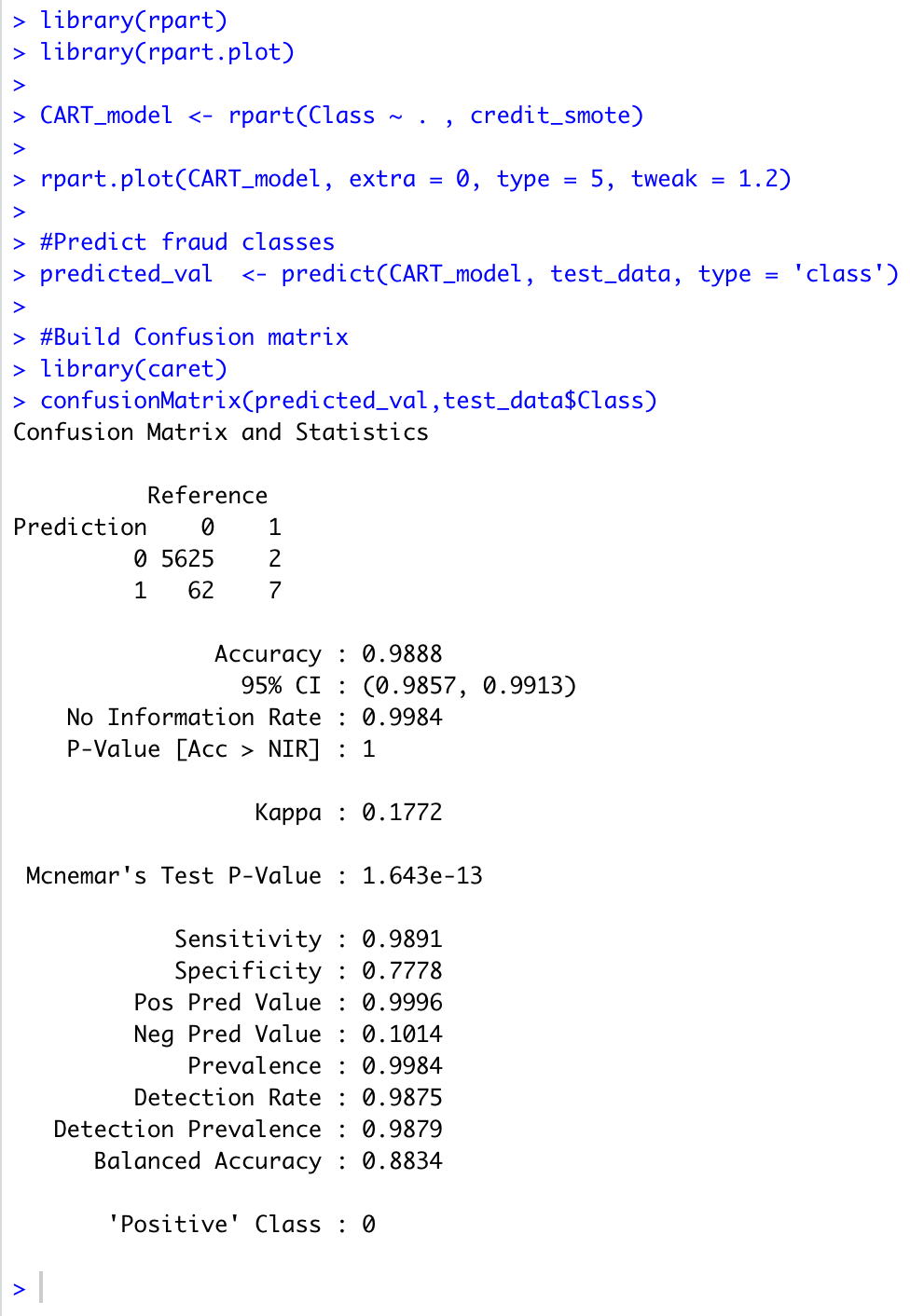
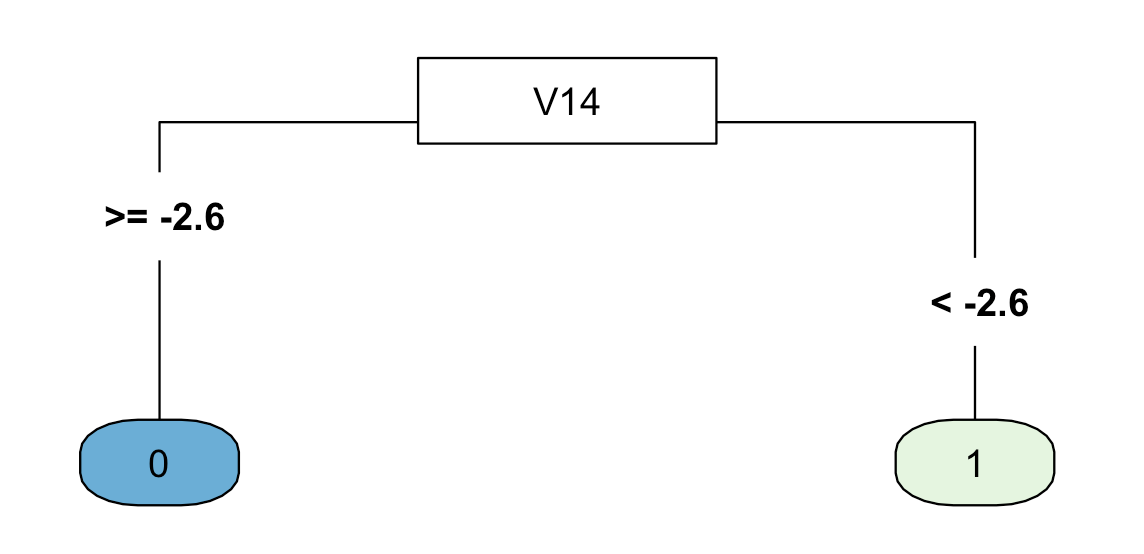
Now preforming Synthetic Minority Over-Sampling Technique is done by using the smotefamily package. Set the number of fraudulent and legitimate cases and the desired percentage of legitimate cases. Calculate the value for the dup\_size parameter of SMOTE. The argument X is a data frame or matrix of numeric-attributed dataset. The argument target is a vector of a target class attribute corresponding to a dataset X. the argument K is the number of nearest neighbors during the sampling process. The argument dup\_size is the number or vector representing the desired times of synthetic minority instances over the original number of majority instances.



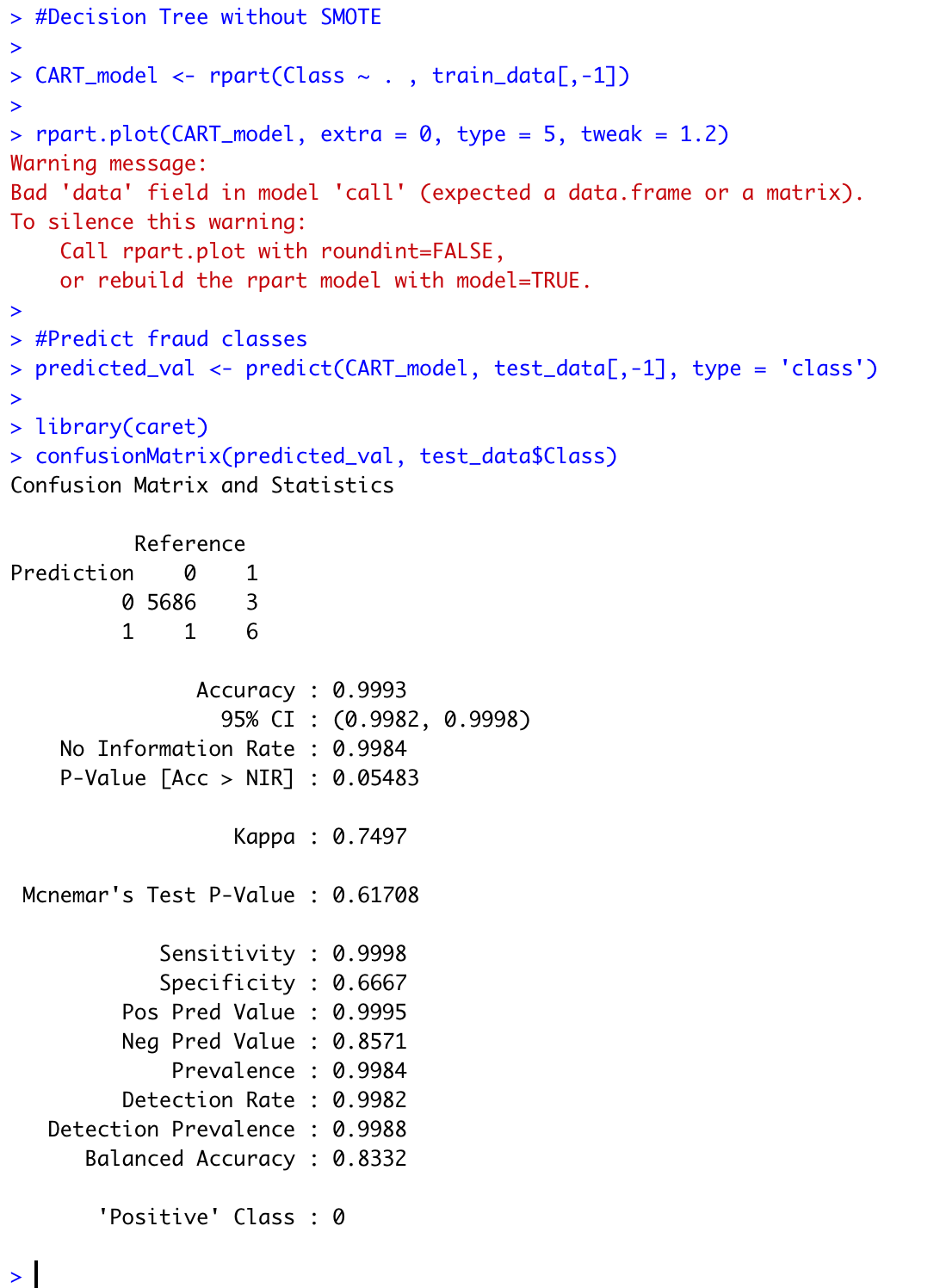
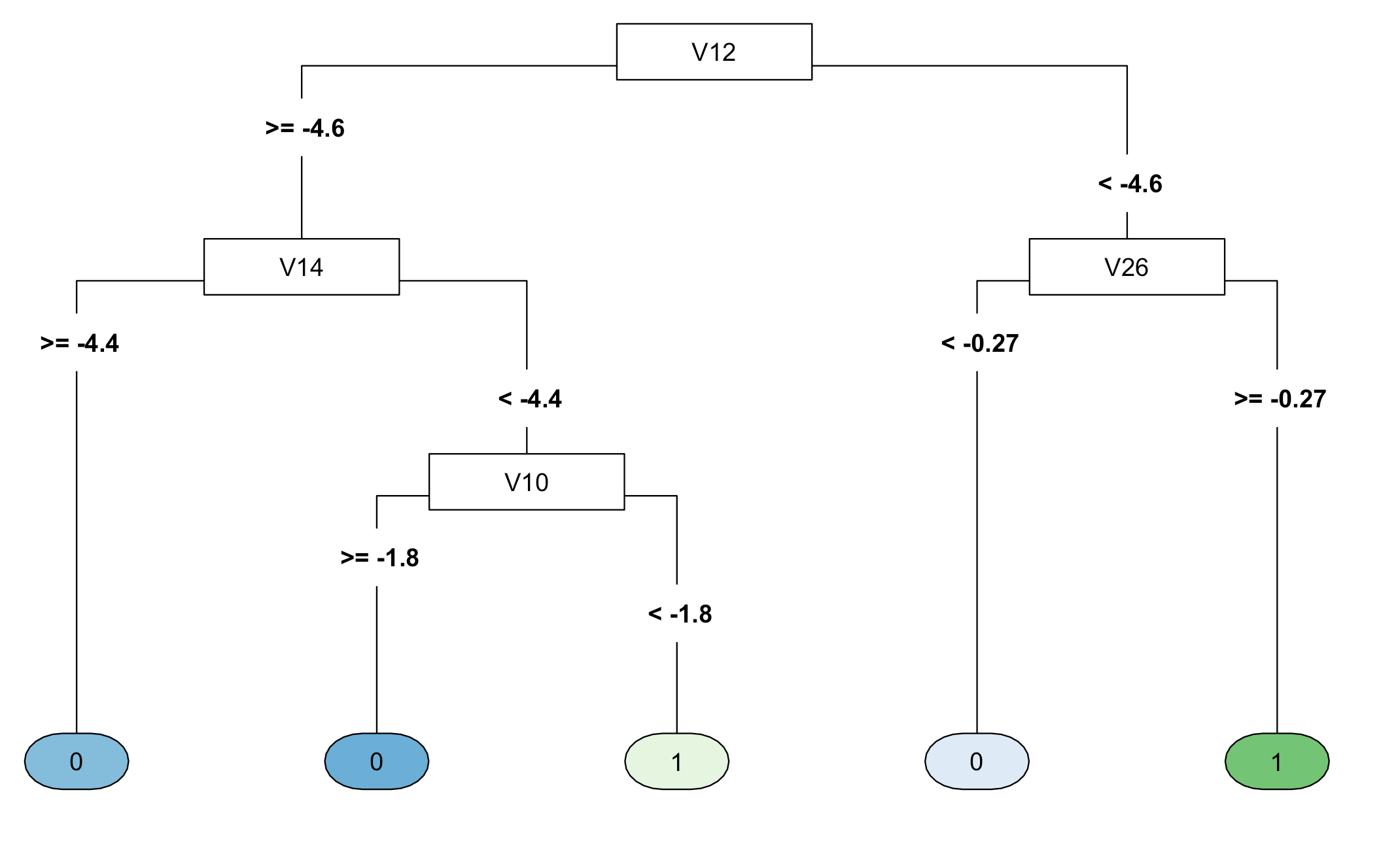
Comparing scatterplots, observe a much more accurate representation of the data while using Synthetic Minority Over-Sampling Technique.

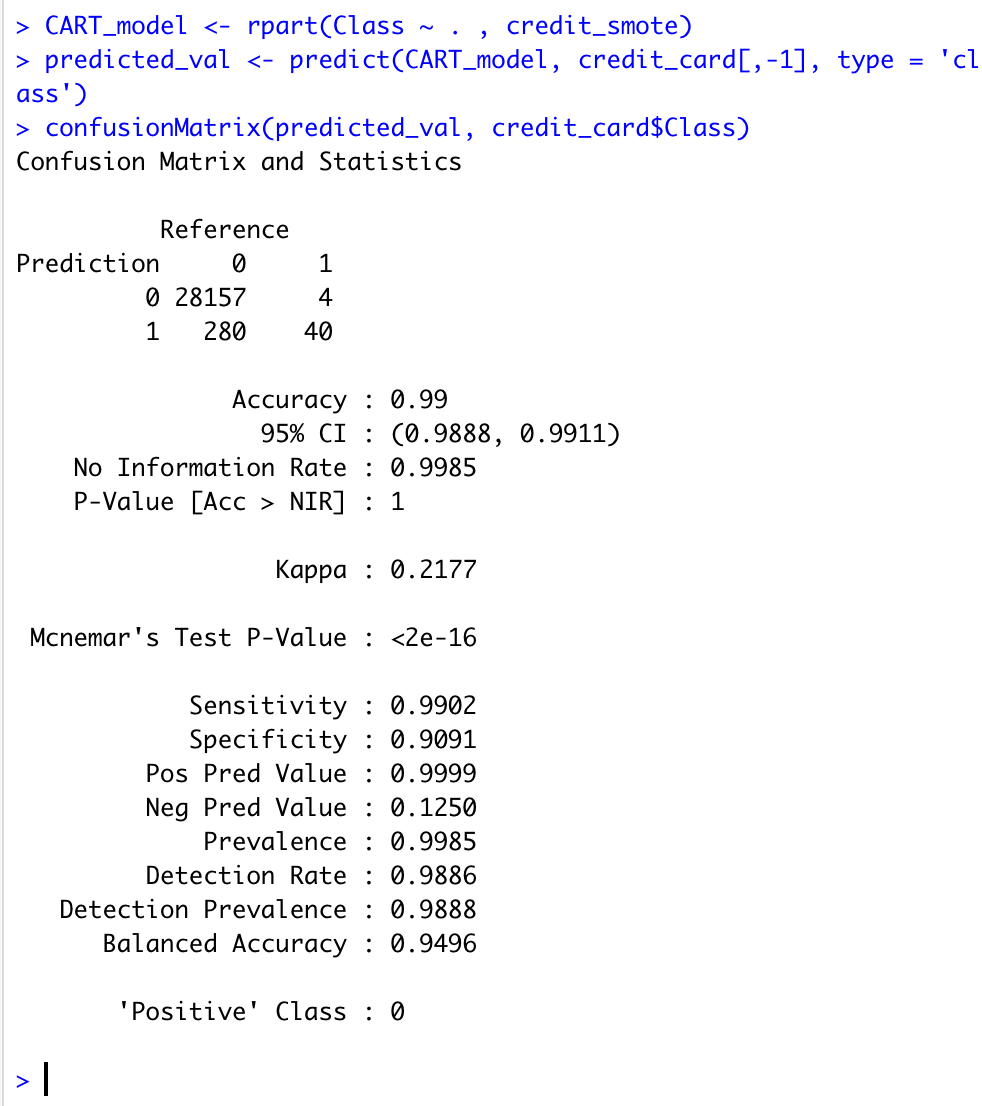
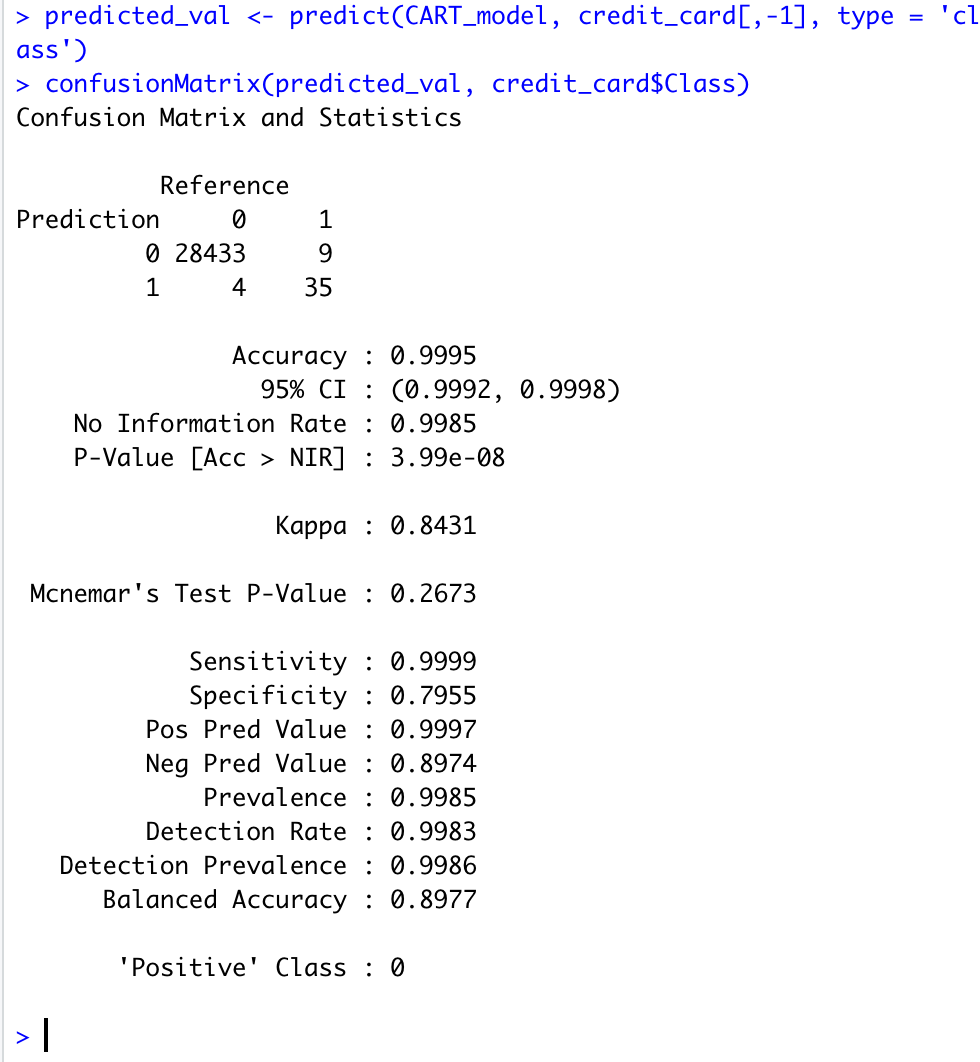
To predict if transactions are fraudulent or legitimate using Synthetic Minority Over-Sampling Technique the package rpart was used to build a decision tree. The package rpart.plot was used to plot the decision tree. The caret package was used to build a confusion matrix in order to compare results. Seven out of nine fraudulent transactions were predicted correctly using Synthetic Minority Over-Sampling Technique on the test data.

A decision tree was also used to predict without using the Synthetic Minority Over-Sampling Technique. Six out of nine fraudulent transactions were classified correctly.

The model built on Synthetic Minority Over-Sampling Technique was used on the train data. Forty out of forty-four fraudulent transactions were correctly classified. Using SMOTE detected five more fraudulent transactions than without using SMOTE. Overall SMOTE is preferable because it balances the dataset synthetically then predicts fraudulent transactions.

In the future I would like to build a model on the entire dataset using Amazon’s AWS. My computer would not be able to process the dataset unless I took a small portion of it. I would like to try using different transaction datasets. Also, I would like to preform Adaptive Synthetic (ADASYN) sampling on the dataset.

“The essential idea of ADASYN is to use a weighted distribution for different minority class examples according to their level of difficulty in learning, where more synthetic data is generated for minority class examples that are harder to learn compared to those minority examples that are easier to learn. As a result, the ADASYN approach improves learning with respect to the data distributions in two ways: (1) reducing the bias introduced by the class imbalance, and (2) adaptively shifting the classification decision boundary toward the difficult examples.” <https://sci2s.ugr.es/keel/pdf/algorithm/congreso/2008-He-ieee.pdf>

References:

<https://www.kaggle.com/mlg-ulb/creditcardfraud>

<https://data-flair.training/blogs/data-science-machine-learning-project-credit-card-fraud-detection/>

<https://rpubs.com/slazien/fraud_detection>

<https://learn.datacamp.com/courses/fraud-detection-in-r>

<https://rpubs.com/kieroneil/321413>

<https://blog.revolutionanalytics.com/2016/09/fraud-detection.html>

<https://www.kaggle.com/chrisbow/credit-card-fraud-prediction-in-r>

<https://medium.com/geo-university-learning-content/credit-card-fraud-detection-using-keras-and-r-d1fa59a7326>

<https://rikunert.com/SMOTE_explained>

<https://www.researchgate.net/publication/287601878/figure/fig1/AS:316826589384744@1452548753581/The-schematic-of-NRSBoundary-SMOTE-algorithm.png>

<https://towardsdatascience.com/having-an-imbalanced-dataset-here-is-how-you-can-solve-it-1640568947eb>

<https://sci2s.ugr.es/keel/pdf/algorithm/congreso/2008-He-ieee.pdf>