A simulation study on the properties of KCV-SMOTE and KFS with application to credit card fraud prediction

STATS 771

Pao Zhu Vivian Hsu

McMaster University

Department of Mathematics & Statistics

April 24th, 2024

Agenda

- Introduction
- KCV-SMOTE & KFS Method
- Methods
- Results
- Discussion
- Conclusion

Introduction

Motivation

- Credit cards are an electronic form of payment that are popular for the convenience and protection they offer on everyday purchases
- Detecting credit card fraud is crucial for credit card companies to maintain customer satisfaction and minimize losses
- Machine learning techniques have become the predominant method to predict credit card fraud over the years
- Companies are constantly looking for ways to improve their predictions

Literature Review

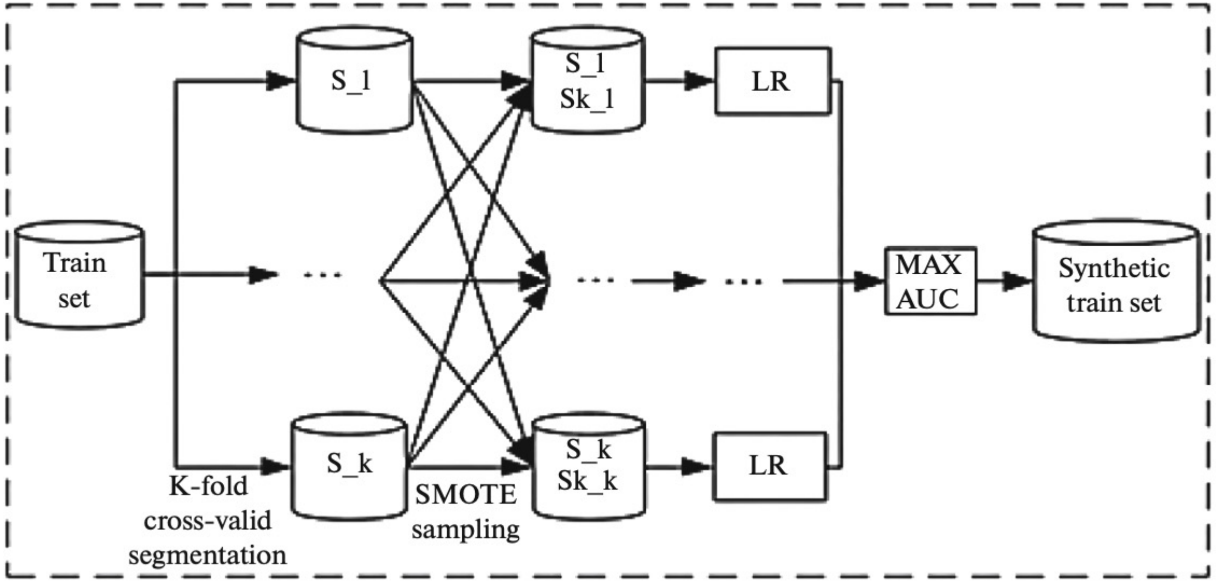
- Some techniques mentioned in recent literature include:
    - Random forest, neural networks, SVM, k-nearest neighbours, logistic regression, Naïve Bayes
- Common challenge that many researchers face when studying credit card fraud:
    - Imbalance of fraudulent and genuine transactions
- Kang and Zhang introduced the K-fold cross-validation and synthetic minority over-sampling technique (KCV-SMOTE) and key feature scanning (KFS) method to address these challenges
- The method was applied to a logistic regression approach and the final model had an AUC of 98.62%.

Research Gap

- While this approach is promising to improve model performance for imbalanced data, there hasn’t been enough exploration on how it would perform under varying signal-to-noise ratios (SNRs)
- Investigating this would provide insight into how robust and applicable this method is to a broader range of datasets
- Goal of our study:
  - Investigate the KCV-SMOTE and KFS method’s performance under different proportions of unbalanced data and ratios of signal-to-noise
  - We will focus on signal predictors vs. noise predictors when we talk about SNRs

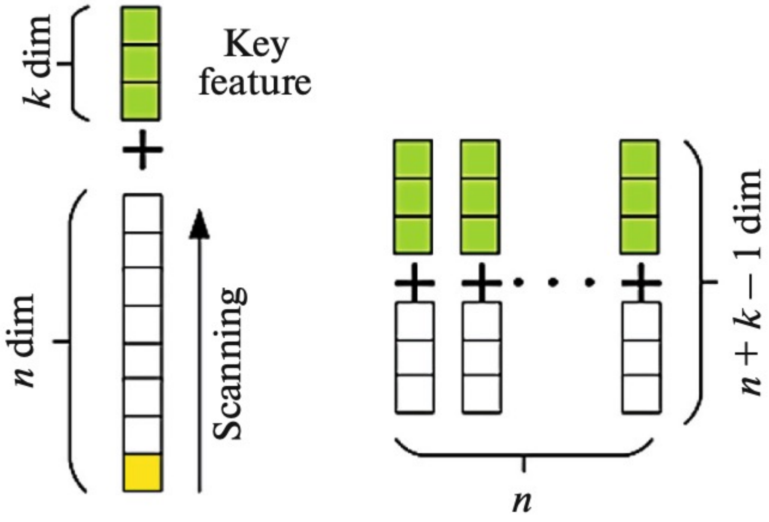
KCV-SMOTE & KFS METHOD

KCV-SMOTE



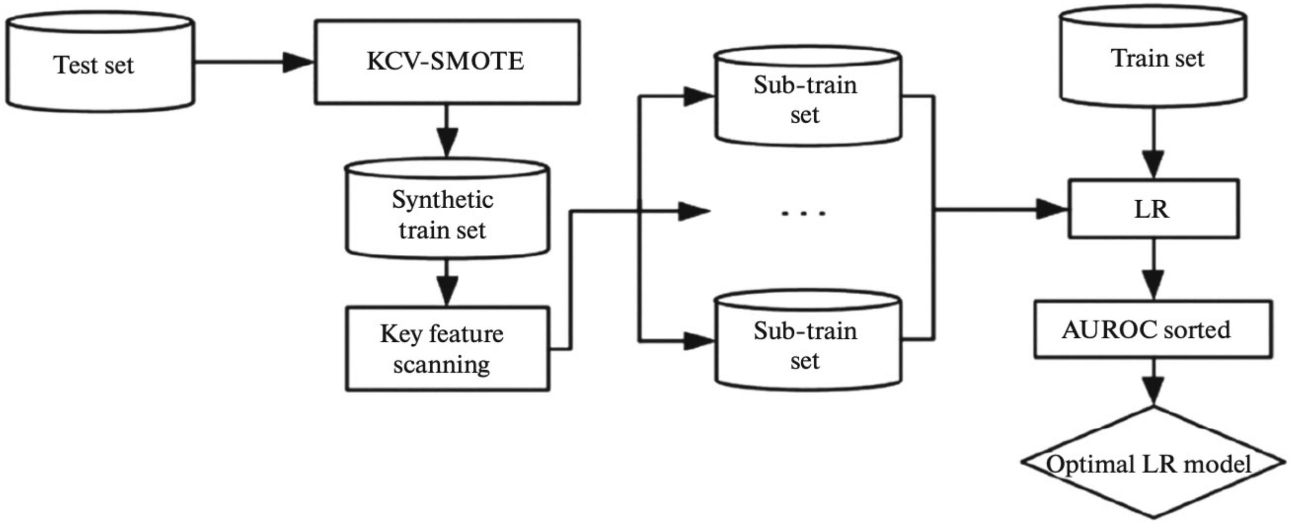
- Takes a training set and produces a synthetic training set
- Steps:
  1. Divide the training set into k subsets
  2. Perform SMOTE sampling (oversampling method) to obtain synthetic training sets for each subset
  3. Train a logistic regression model for each of the subsets and obtain an AUROC score
  4. Select the synthetic training set with the highest AUROC score

KFS



- Creates multiple datasets with different combinations of predictor variables
- Steps:
  1. Divide the training set into key features and general features
  2. Remove one feature from the general features and combine with key features to form a sub-training set
  3. Do this multiple times by scanning through the general features to obtain multiple sub-training sets

KCV-SMOTE & KFS Method

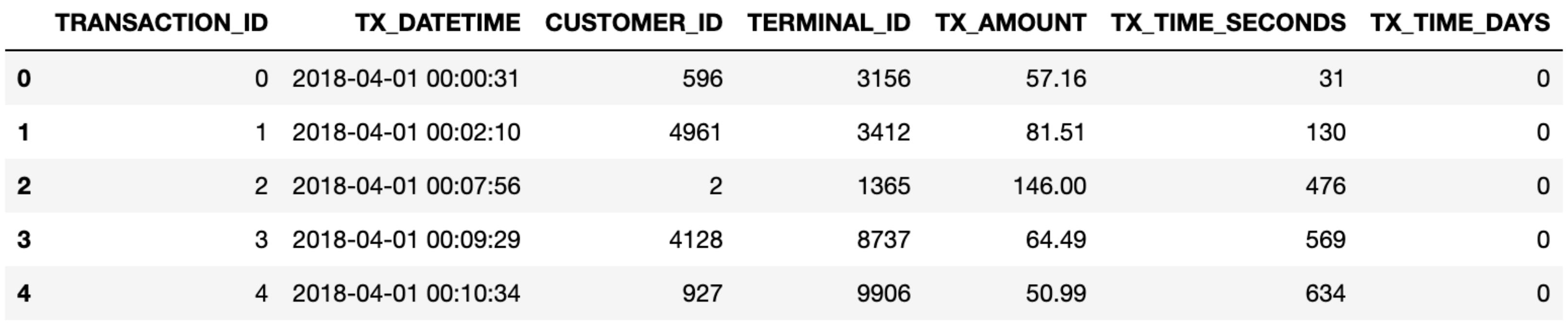


- Steps:
  1. Split data into a training set and a testing set.
  2. Use KCV-SMOTE on the training set to obtain a synthetic training set.
  3. Perform KFS on the synthetic training set to obtain sub-training sets.
  4. Use each sub-training set to build a separate logistic regression model. Get the AUROC values for each model, sort them, and select the top sub-training sets with the highest scores.
  5. Train the intersection of the sub-training sets from the previous step to obtain the best model, and input the test set into the best model to obtain the final prediction result.

Methods

Data Simulation

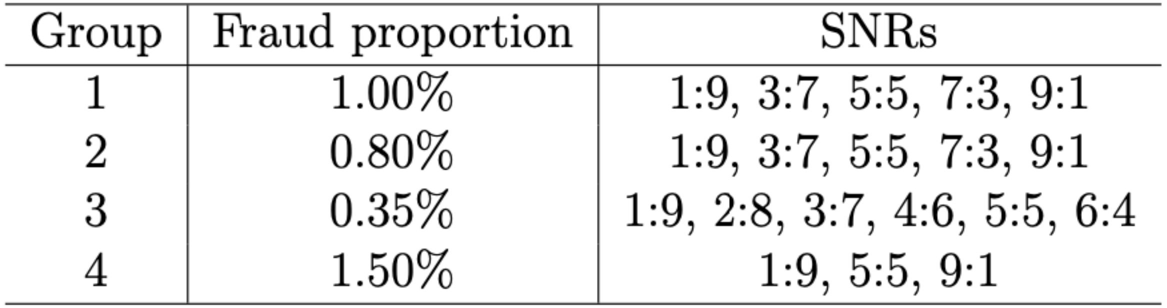
- Data is simulated using the transaction data simulator developed by the Machine Learning Group of ULB, the Université Libre de Bruxelles
- First, we created the baseline transaction dataset:
  1. Generated 5000 customer profiles and 10,000 terminal profiles
  2. Generated about 3.5M genuine transactions using the profiles and common statistical distributions (ex. transaction amount is normally distributed with mean and standard deviation taken from the spending habits of customers in customer profiles)



	TRANSACTION_ID	TX_DATETIME	CUSTOMER_ID	TERMINAL_ID	TX_AMOUNT	TX_TIME_SECONDS	TX_TIME_DAYS
0	0	2018-04-01 00:00:31	596	3156	57.16	31	0
1	1	2018-04-01 00:02:10	4961	3412	81.51	130	0
2	2	2018-04-01 00:07:56	2	1365	146.00	476	0
3	3	2018-04-01 00:09:29	4128	8737	64.49	569	0
4	4	2018-04-01 00:10:34	927	9906	50.99	634	0

Data Simulation

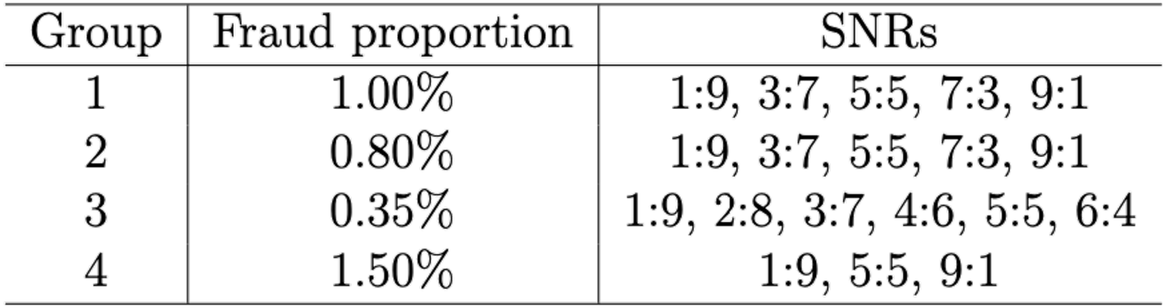
- Added fraud transactions to the baseline dataset to form 5 new datasets with a fraud proportions of 0.35%, 0.5%, 0.8%, 1%, and 1.5%
- Created 19 more datasets with varying SNRs of predictor variables
  - Assumed all simulated predictors thus far are **signal** predictors
  - Added **noise** predictors by simulating variables containing arbitrary integer values from 1 to 100.



Group	Fraud proportion	SNRs
1	1.00%	1:9, 3:7, 5:5, 7:3, 9:1
2	0.80%	1:9, 3:7, 5:5, 7:3, 9:1
3	0.35%	1:9, 2:8, 3:7, 4:6, 5:5, 6:4
4	1.50%	1:9, 5:5, 9:1

KCV-SMOTE & KFS Method

- Applied KCV-SMOTE & KFS method to all 24 simulated datasets
  - Unbalanced classes analysis
    - 5 model comparison
    - Confirm that the method performs well for unbalanced data as claimed by researchers
  - SNR analysis
    - 19 model comparison
    - Fix the fraud proportion and compare model performance for different SNRs

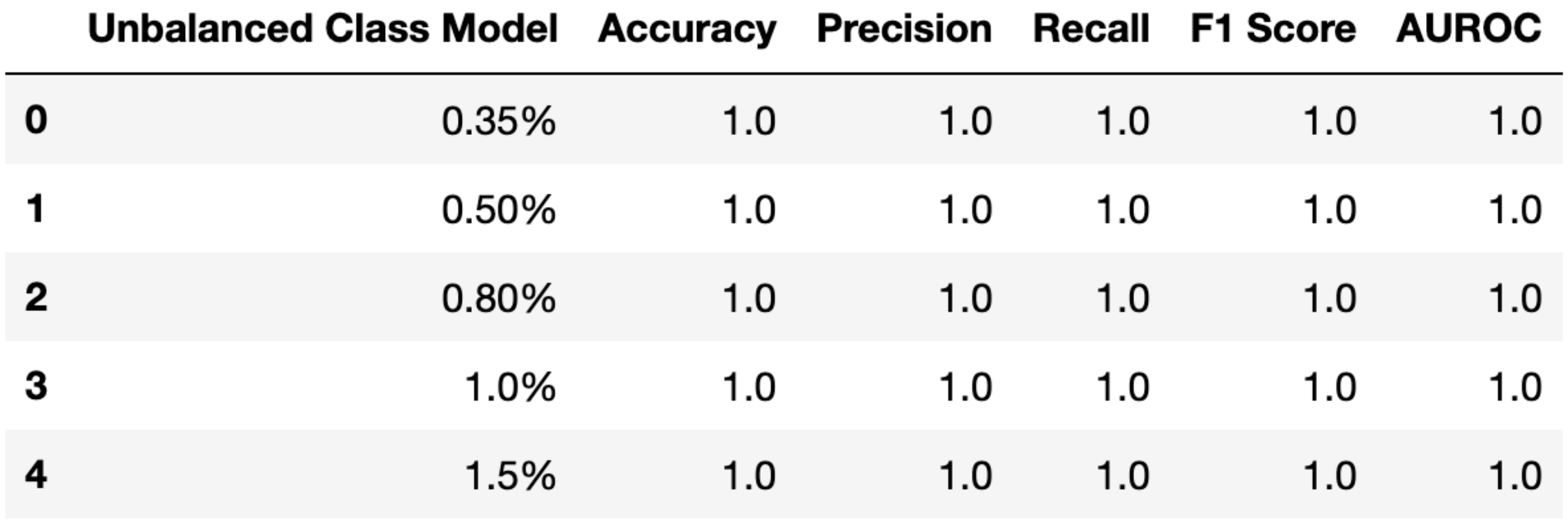


Group	Fraud proportion	SNRs
1	1.00%	1:9, 3:7, 5:5, 7:3, 9:1
2	0.80%	1:9, 3:7, 5:5, 7:3, 9:1
3	0.35%	1:9, 2:8, 3:7, 4:6, 5:5, 6:4
4	1.50%	1:9, 5:5, 9:1

Results

Unbalanced Classes Analysis

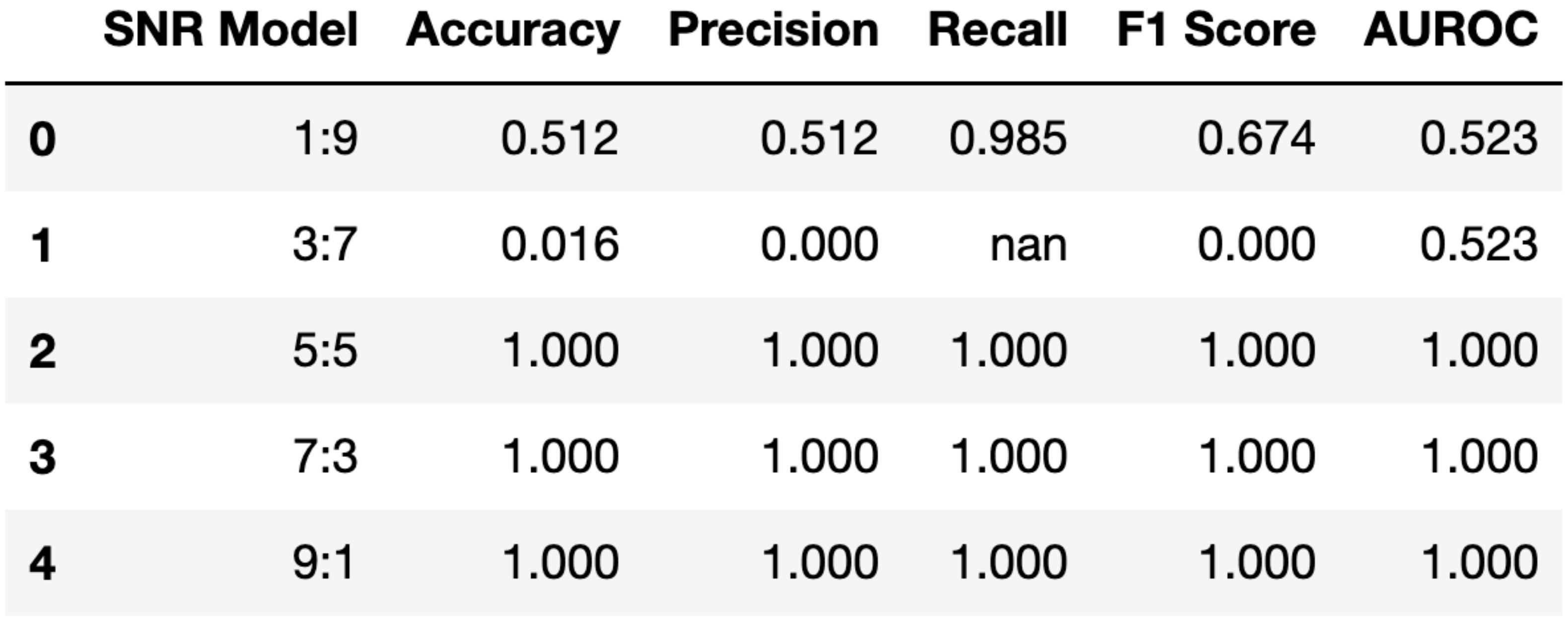
- Performs very well for each of the simulated datasets indicating that it can handle data with high degrees of unbalanced classes



	Unbalanced Class Model	Accuracy	Precision	Recall	F1 Score	AUROC
0	0.35%	1.0	1.0	1.0	1.0	1.0
1	0.50%	1.0	1.0	1.0	1.0	1.0
2	0.80%	1.0	1.0	1.0	1.0	1.0
3	1.0%	1.0	1.0	1.0	1.0	1.0
4	1.5%	1.0	1.0	1.0	1.0	1.0

SNR Analysis – 1% Fraud Proportion

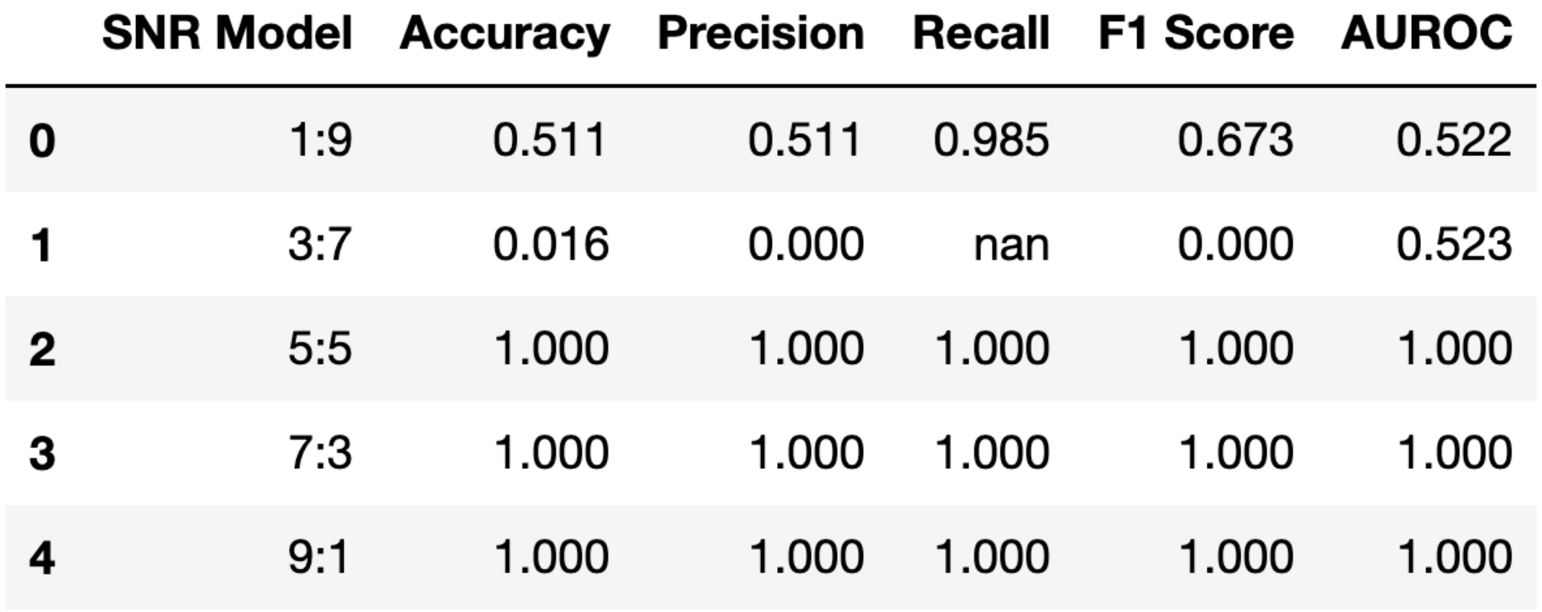
- KCV-SMOTE and KFS method has unstable and poor performance when there are more noise predictors compared to signal predictors
- Performs well when there is at least an even ratio of signal predictors to noise predictors.



	SNR Model	Accuracy	Precision	Recall	F1 Score	AUROC
0	1:9	0.512	0.512	0.985	0.674	0.523
1	3:7	0.016	0.000	nan	0.000	0.523
2	5:5	1.000	1.000	1.000	1.000	1.000
3	7:3	1.000	1.000	1.000	1.000	1.000
4	9:1	1.000	1.000	1.000	1.000	1.000

SNR Analysis – 0.8% Fraud Proportion

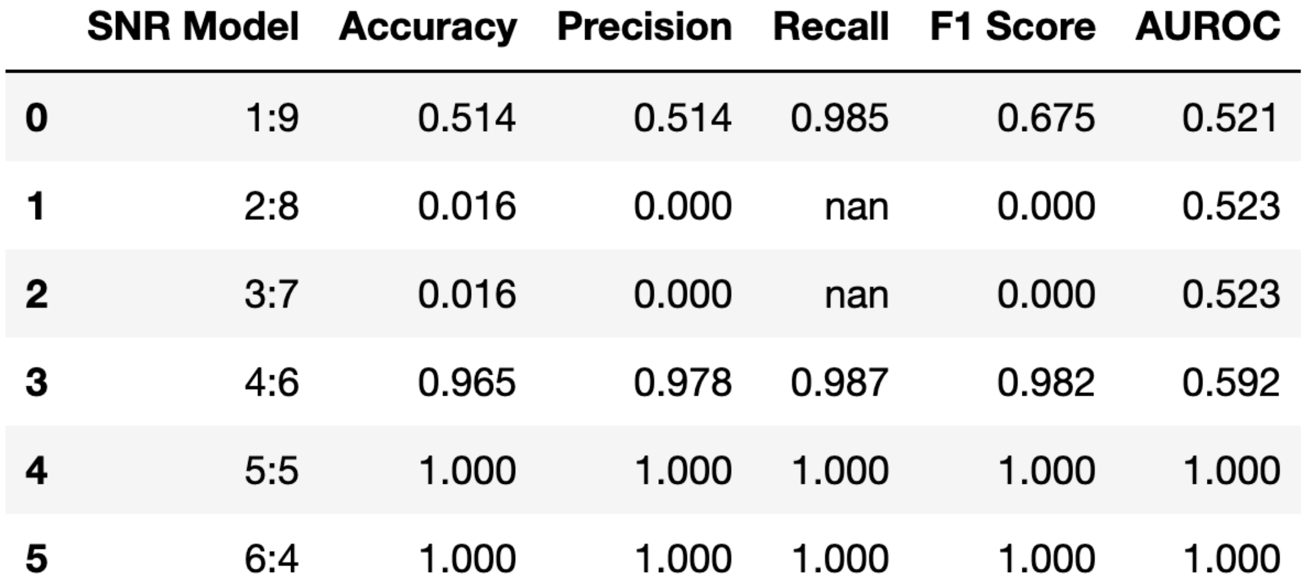
- Similar results when the percentage of fraud in the dataset is changed to 0.8%



	SNR Model	Accuracy	Precision	Recall	F1 Score	AUROC
0	1:9	0.511	0.511	0.985	0.673	0.522
1	3:7	0.016	0.000	nan	0.000	0.523
2	5:5	1.000	1.000	1.000	1.000	1.000
3	7:3	1.000	1.000	1.000	1.000	1.000
4	9:1	1.000	1.000	1.000	1.000	1.000

SNR Analysis – 0.35% Fraud Proportion

- Investigated more cases where there is a higher number of noise predictors than signal predictors
- Method also performs moderately well when there are 4 signal predictors and 6 noise predictors. Thus, there could be more noise predictors than signals.



	SNR Model	Accuracy	Precision	Recall	F1 Score	AUROC
0	1:9	0.514	0.514	0.985	0.675	0.521
1	2:8	0.016	0.000	nan	0.000	0.523
2	3:7	0.016	0.000	nan	0.000	0.523
3	4:6	0.965	0.978	0.987	0.982	0.592
4	5:5	1.000	1.000	1.000	1.000	1.000
5	6:4	1.000	1.000	1.000	1.000	1.000

SNR Analysis – Overall Results

- Similar results were obtained for 1.5% fraud proportion as well
- Having an SNR of at least 4:6 signal-to-noise predictors will yield promising results using KCV-SMOTE and KFS
- Changes in fraud proportion do not have a major impact on model performance outcomes even with varying degrees of SNRs

Discussion

Improvements to our study

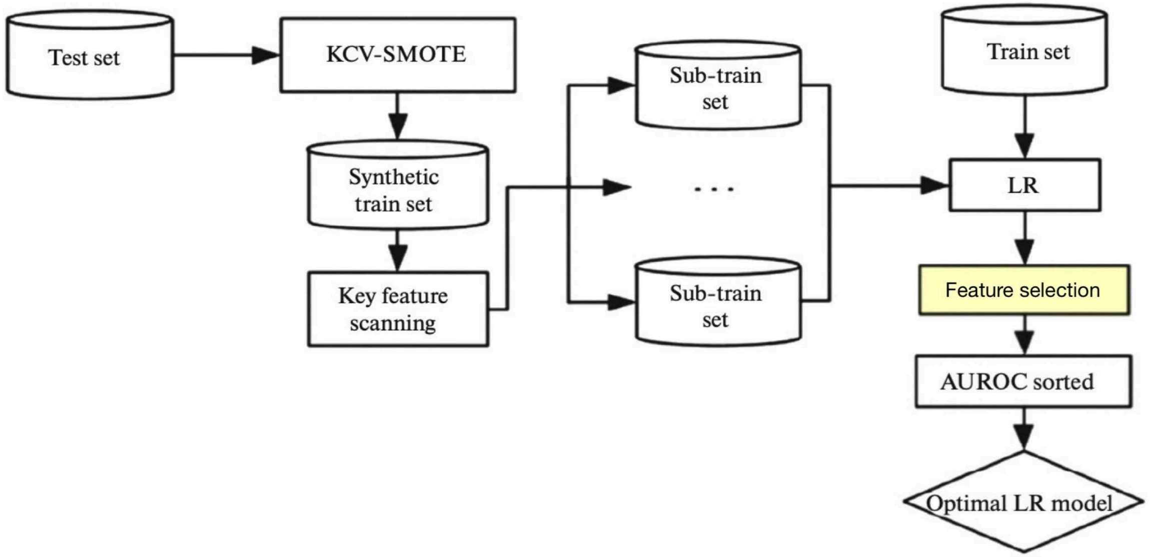
- Experiment on different values of k parts when implementing KCV
- Use a more rigorous approach to select key features when performing KFS
  - Having expertise from the credit card industry combined with the use of a classifier such as logistic regression can better inform our choice of key features
- Study more combinations of class imbalance and SNRs
  - Ex. If we have data with 30 variables, we could investigate how the KCV-SMOTE and KFS method performs under more specific SNRs
  - Provide us a greater understanding of what the SNR threshold would be for the KCV-SMOTE and KFS method to produce effective prediction results

Improvements to our study

- Perform a similar study using real data instead
  - We can treat the class imbalance proportion as fixed, assume all variables in the dataset are signal predictors and add simulated noise predictors into the study.
  - Since there is a greater risk involved when assuming real data as signal predictors, one can optionally perform a preliminary analysis to decide which predictors are most likely signals as opposed to noise

Proposed Modifications to KCV-SMOTE & KFS Method

- One way we could modify the KCV-SMOTE & KFS method to reduce the effects of low SNRs is to add feature selection to the process
- Backward stepwise elimination can be done after logistic regression is performed on each of the sub-training sets produced by KFS
- This can help filter out predictors that have a high chance of being noise in each model



Proposed Modifications to KCV-SMOTE & KFS Method

- There is a general debate on whether stepwise elimination should be applied on linear regression model fitting
    - To reduce the risk of potentially excluding important explanatory variables, we can use large significance levels during stepwise elimination
    - This is something that can be tuned by the researcher based on how comfortable they are with this risk in terms of their data
- Further work is required to test this proposed modification and its ability to improve model performance for datasets with low SNRs.

Conclusion

Conclusion

- Overall, our simulation study has provided greater insight into the effectiveness of Kang and Zhang’s method under different proportions of unbalanced data and ratios of signal-to-noise.
    - The method performed well on all tested percentages of highly imbalanced data, as claimed by Kang and Zhang
    - Having at least a 4:6 SNR of predictors will yield accurate results using KCV-SMOTE and KFS
    - SNRs lower than that will have poor and unstable performance
- Proposed a modification to the method that uses stepwise elimination to reduce the effects of low SNRs
- Further research is required to test this approach

Thank You!

References

[1] Fayaz Itoo, Meenakshi, and Satwinder Singh. Comparison and analysis of logistic regression, na¨ıve bayes and knn machine learning algorithms for credit card fraud detection. *International Journal of Information Technology*, 13:1503–1511, 2021.

[2] Haiyan Kang and Hao Zhang. A new improved method for online credit anti-fraud. *Automatic Control and Computer Sciences*, 56:347–355, 2022.

[3] Yann-A¨el Le Borgne, Wissam Siblini, Bertrand Lebichot, and Gianluca Bontempi. *Reproducible Machine Learning for Credit Card Fraud Detection - Practical Handbook*. Universit́ e Libre de Bruxelles, 2022.

[4] Rejwan Bin Sulaiman, Vitaly Schetinin, and Paul Sant. Review of machine learning approach on credit card fraud detection. *Human Centric Intelligent Systems*, 2:55–68, 2022.

[5] Worldline and Machine Learning Group - ULB. Credit card fraud detection, 2017.