EFFECT OF DATASET REDUCTION TECHNIQUES ON COMPUTATIONAL COMPLEXITY AND PREDICTIVE PERFORMANCE OF CLASSIFICATION PROBLEM

by
SUAT AKKAŞ

Submitted to the Graduate School of Social Sciences
in partial fulfilment of
the requirements for the degree of Master of Science

Sabancı University
December 2025

ABSTRACT

EFFECT OF DATASET REDUCTION TECHNIQUES ON COMPUTATIONAL COMPLEXITY AND PREDICTIVE PERFORMANCE OF CLASSIFICATION PROBLEM

SUAT AKKAŞ

Data Science M.Sc. THESIS, January 2025

Thesis Supervisor: Asst. Prof. Ezgi KARABULUT TÜRKSEVEN

Keywords: Sampling, Dimensionality Reduction, Similarity, Classification, Computational Performance

The usage of big data in the industry increases day by day. This situation exists also in the financial industry. The usage of big data in the financial sector leads to enormous improvement in the areas of financial problems such as credit scoring problems. However, the usage of big data also increases the computational time and usage of available resources enormously. Therefore, this issue makes the usage of big data in some applications and some situations inefficient.

To handle inefficiency in the usage of big data, we have focused on the sampling methods in this study. By using row-wise sampling algorithms and dimensionality reduction in data, we aimed to reduce computational time for solving credit scoring problems. However, our aim in this study is not just a reduction in computational time but also the performance of the model usage in credit scoring in the case of usage of big data. We have used also feature selection and transformation algorithms in order to observe the effect of selection and transformation algorithms on different sample sizes of sampled data in terms of predictive power. Moreover, to validate whether the sample dataset represents the main dataset or not, we have used a bunch of similarity metrics for different data types that exist in the dataset.

By using this methodology, we have observed the relation between the computational time, power and data representativeness for different sample sizes of sampled data. According to our findings from our study, it is possible to preserve the predictive power of models until some sample size, with decreasing the computational amount in significant amounts. By demonstrating the relation between the computational time versus predictive power relations with different sample sizes and different feature reduction methods, we aim to propose the sample size and feature reduction selection for one's main concerns.

ÖZET

VERI KÜMESI AZALTMA TEKNIKLERININ SINIFLANDIRMA PROBLEMININ HESAPLAMA KARMAŞIKLIĞI VE TAHMIN PERFORMANSI ÜZERINDEKI ETKISI

TEZ YAZARI

Tez Danışmanı: Yrd. Doç. Dr. Ezgi KARABULUT TÜRKSEVEN

Anahtar Kelimeler: Örnekleme, Boyut İndirgeme, Benzerlik, Sınıflandırma, Hesaba Dayalı Peformans

Büyük verinin endüstride kullanımı her geçen gün artmaktadır. Bu durum finans endüstrisinde de mevcuttur. Büyük verinin finans sektöründe kullanımı, kredi puanlama sorunları gibi finansal sorunlar alanında muazzam iyileştirmelere yol açmaktadır. Ancak, büyük verinin kullanımı aynı zamanda hesaplama süresini ve mevcut kaynakların kullanımını da muazzam şekilde artırmaktadır. Bu nedenle, bu sorun bazı uygulamalarda ve bazı durumlarda büyük verinin kullanımını verimsiz hale getirmektedir.

Büyük verinin kullanımındaki verimsizliği ele almak için bu çalışmada örnekleme yöntemlerine odaklandık. Satır bazlı örnekleme algoritmaları ve sütun bazlı boyut indirgeme kullanarak, kredi puanlama sorunlarını çözmek için hesaplama süresini azaltmayı amaçladık. Ancak, bu çalışmadaki amacımız sadece hesaplama süresini azaltmak değil, aynı zamanda büyük verinin kullanımı durumunda kredi puanlamasında model kullanımının performansını da azaltmaktır. Ayrıca, tahmin gücü açısından örneklenen verilerin farklı örnek boyutlarında seçim ve dönüştürme algoritmalarının etkisini gözlemlemek için özellik seçimi ve dönüştürme algoritmalarını da kullandık. Ayrıca, örnek veri setinin ana veri setini temsil edip etmediğini doğrulamak için, veri setinde bulunan farklı veri tipleri için bir dizi benzerlik metriği kullandık.

Bu metodolojiyi kullanarak, örneklenen verilerin farklı örnek boyutları için hesaplama süresi, güç ve veri temsiliyeti arasındaki ilişkiyi gözlemledik. Çalışmamızdan elde ettiğimiz bulgulara göre, hesaplama miktarını önemli miktarda azaltarak, modellerin tahmin gücünü belirli bir örnek boyutuna kadar korumak mümkündür. Farklı örnek boyutları ve farklı özellik azaltma yöntemleriyle hesaplama süresi ile tahmin gücü ilişkileri arasındaki ilişkiyi göstererek, ana endişeler için örnek boyutu ve özellik azaltma seçimini önermeyi amaçlıyoruz.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Assist. Prof. Dr. Ezgi Karabulut Türkseven for her invaluable guidance, insightful feedback, and unwavering support throughout the course of my research. Their expertise and encouragement have been instrumental in shaping this thesis and my academic growth.

I would also like to extend my heartfelt thanks to the members of my thesis jury for their invaluable feedback and constructive suggestions. Their insights have significantly enhanced the quality of my work, and I am deeply grateful for their time and effort.

I am profoundly thankful to my family for their unconditional love, patience, and constant encouragement. Your support has been my foundation, inspiring me to persevere and achieve this milestone. To my parents, your belief in me has been my greatest motivation, teaching me the value of hard work and resilience. To my sister, thank you for always being my cheerleader and for your endless encouragement that has kept me focused and determined. To my beloved wife, I owe my deepest gratitude for your incredible support, endurance, and unwavering belief in me. Your encouragement and understanding have been my greatest strength throughout this journey, and I am truly blessed to have you by my side.

Thank you all for being part of this incredible journey.

Dedication page
To my beloved family

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

The dramatic increase in data generation has affected the organization's decision-making processes and the usage of data frequencies. Big data is beneficial for decision making processes by improving processes' accuracy in the organizations, it gives opportunities for retrieving more informative insights, However, it also brings some challenges for storage, computational efficiency such as considerably long computational times and lack of processing power to mine this data or extracting information from it. Big data management is challenging for organizations, especially while aiming for operational efficiency and minimum time-consuming operations. As mentioned in He & Garcia (2009) study, organizations mostly face considerable problems in learning from large and complex datasets, especially dealing with imbalanced data sets, which are mostly common in real-world applications.

Those issues about big data emerged in most of the sectors. One of those sectors is financial sector. Usage of big data in financial sector is becoming crucially important for decision making processes such as credit scoring. Applications of machine learning algorithms on big data in credit scoring decisions have become powerful and practical tools for credit risk assessment and how effective in processing complex, multidimensional data for accurate predictions is demonstrated by Lessmann, Baesens, Seow & Thomas (2015). Moreover, Bao, Xie, Song & Song (2019) have mentioned that although more sophisticated models they have used as integrated approach of unsupervised and supervised learning methods, reached superior results, they require large amounts of computational resources, and more complex optimization strategies to overcome this problem.

In order to handle these computational challenges led by usage of big data, several approaches used such as GPU acceleration, distributed computing systems and algorithmic optimizations. The optimization of model architecture and data preprocessing suggested by Xia, Hu, Hu, Shi, Bai, Zhong, Lu & Zhang (2017), for an answer to this problem. In later years,Zhang, Luo & Du (2021) emphasized the importance of balancing model complexity with computational efficiency, while demonstrating the effectiveness of ensemble methods in credit scoring. Thus, the main issue arises the question that in order to reach the optimal predictive performance, should the entire data set be used or if samples selected from the data set carefully is enough while significantly reducing computational complexity?

Considering the advanced methodologies and recent algorithms used in credit scoring, this thesis focused on sampling methodologies on applications of machine learning algorithms in the area of credit scoring. The main problem and the challenging one in this sampling approach is to ensure that the sampled data maintains the representativeness of the original dataset and captures the main patterns of the original data set by also maintaining the essential relationship between features and target as in the main data. This relationship is crucial in terms of predictive power of the machine learning models, especially in classification problems of credit scoring. This challenge of representativeness of the sample data requires careful consideration of sampling techniques that can capture the data distribution across both numerical and categorical variables as well documented in Lessmann et al. (2015).

The solution to this problem could be the computers with high computational power. Although this approach is effective in reducing computational time, this is not a proper answer to our problem, in the case of limited resources such as GPU, and limited access to highly advanced technologies. Moreover, as the sample size of data gets much bigger, there would be still a necessity for datasets with relatively smaller sizes in terms of efficient problem solving. Because of those issues, we have focused on effect of sampling methods in our study to overcome this problem discussed above. Moreover, still, it is worth to discuss the alternative solution to sampling methodologies. Methods that work with the whole dataset but uses subsamples during its steps such as mini batch method. However, this approach is based on using whole dataset but using sub-samples of it iteratively. Because, this method uses whole dataset eventually, this does not reduce the computational time. This method could be effective only if the usage of parallelization while applying this method. Even if mini batching applied with parallel processing, reduction in computational power, will not be realized in the usage of GPU. There would be still high memory usage.

To reach correct answers to those questions, we have used a credit application dataset, this data set includes 518546 rows, in other words credit applications, and 1312 distinct features. Those features that exist in data consist of properties of this specific application. Some of those features have categorical characteristics. However, some of those features have continuous characteristics. Suppose the features are grouped according to the information each of those features carries. In that case, we can say that we have feature groups such as demographic, financial situation, application properties-based, and account-based features. The summary of the features is provided in Table 1.1. As this dataset includes both numerical variables and categorical variables, it provides an ideal environment for our experiments. Also, the scale and complexity of our dataset used for credit application scoring align with the high dimensional challenges addressed by Bao et al. (2019) in their integrated machine-learning approach.

In demographic features, we have age, hometown, location, and demographic information-based features. Those features are categorized in our dataset most as categorical. Moreover, the second group of features in our dataset is features related to the application's financial situation. Those features include the income information of the applicant; total accounted debt, and the financial risk of the applicant. As we named before, the third group of features are grouped as application properties features. Those features mostly consist of mixed-type features. In other words, those features include both categorical and numerical data-type features. The continuous part of that feature group are a number of credit applications of an applicant, number of rejected applications, and also combination of those features. The categorical part of this feature group consists of how the application occurred and which way the applicant applied for one credit. The final feature group consists of features related to the accounts of the applicants. Those features are transaction information of applicant's accounts in a time-based manner. The remaining features of this group are number of delinquencies, which means the credit debt of this account is not paid on time. Thus, the total number of features with those groups becomes 1202 for numerical features and 110 features for categorical features.

Table 1.1 Feature Data Types According to Feature Groups

Feature Groups/Data Type	**Numerical**	**Categorical**
Demographic	25	52
Financial Situation	738	11
Application Properties	238	22
Account Properties	201	25
Total	**1202**	**110**

This dataset includes a number of applications in one year to the bank, and we also have 46424 applications for a one-month period, which is derived from one month later from our development data. Thus, this dataset will be used in order to validate our foundations coming from the results of our experiments. A target column also exists in our dataset. The definition of this target column is the default information of this application. This means that if an application defaults according to the definition of default stated, then the target column becomes 1. Otherwise, the target column is labeled as 0. The definition of default is the three consecutive unpaid installments in one year. Thus, from a one-year perspective, starting from the application date, if the application does not pay its monthly installments three consecutive times, this application's target value is labeled as one. By using this definition for target value, when we investigate the target ratio of the data, we have observed that this is a highly imbalanced dataset. The target ratio of our development dataset is observed as 0.0261, and for the test period, it is observed as 0.033. This means that we need to consider this situation when applying model algorithms, sampling, and feature reduction techniques. The structure of the data is summarized in Table 1.2.

Table 1.2 Training and Test Data Target Ratios

	Training	**Test**
Number of Rows	518546	46424
Target Ratio	0.026	0.033

The structure of this thesis is designed in order to investigate data sampling representativeness and its contribution to efficiency systematically. Thus, in the first step, we conduct comprehensive similarity control between the original dataset and its different-sized row-wise random samples to find optimal sampling thresholds. We will also investigate how the feature transformation and feature selection algorithms affect sampled data compared to the original dataset. Finally, we will observe the impact of machine learning models such as logistic regression and boosting models on sampled data again compared to its original dataset in terms of model performance metrics. To conclude, our study aims to conduct a comparative analysis on the representativeness of different-sized sample data according to its original data set in terms of column-wise similarity, feature selection, and transformation algorithms' effects on sampled data, finally the comparison of predictive performance between original dataset and sampled data of different sizes by aiming to gain computational efficiency.

The thesis is organized as follows: Section 2 focuses on sampling, and similarity methods used to calculate representativeness. Section 3 describes PCA algorithm's application to sample data and main data. Section 4 reveals the results of classification algorithm performance on different sample sizes. We conclude this study and provide future research directions in Section 5.

This comprehensive approach allows us to not only address the computational challenges of big data in credit scoring but also provide practical guidelines for maintaining model performance while significantly reducing computational complexity.

2. Sampling and Similarity Metric Results

2.1 Structure of Data and Data Sampling

Our problem consists of a highly imbalanced dataset. In addition, our study focuses on reducing the dataset row-wise and column-wise (by using feature transformation algorithms), and we need those transformations and sampling to represent the dataset. Therefore, our study must consider imbalance target ratios while applying those algorithms.

In this section, we will be giving the results of our experiments on similarity metrics used for continuous features and categorical features. This part includes only row sampling methodology, and the investigation of similarity test results of this sampling method whether similar to those of the main data. We have used a stratified random sampling method for our row-wise samples. According to the definition by Xia et al. (2017), stratified random sampling is a probability sampling technique that divides a population into distinct subgroups and selects samples from each stratum. The primary aim is to ensure that each subgroup is adequately represented in the sample, thus improving the precision and generalizability of the findings. We have used this method because of its simplicity and the representative power of the main dataset. We do not use other methods used mostly for imbalanced datasets because our focus is on the whole representation of data without changing its properties, distribution, and, more importantly, target ratio. Moreover, the reason not to use direct random sampling instead stratified based on the same concerns explained in the previous sentence. We have applied stratification only on target feature in other words response variable.

Thus, by preserving the target ratio in each data sample derived from the main development data, we have selected 6 levels of sample size. The sample sizes of the main dataset consist of 20%,10%, 5%, 1%, 0.5%, and 0.1% of the size of the main dataset. Moreover, each of those datasets includes the same ratio for the target. After sampling the main dataset according to this methodology, we have started to investigate the representativeness of those samples for the main dataset. In order to compare those samples' representativeness, we also have sampled 20 times for each sample size with different random states. Thus, we have tried those results for each sample size 20 times.

In order to measure the representativeness of samples, we have investigated some of the tests which suit our case. In order to do that, firstly we have investigated the normality of our features. However, most of the features we have investigated do not fit the normality assumption. As can be seen from Figure 2.1 below, when we plotted histogram plots of those features with the correct binning, we have observed that most of the features skewed by violating the normal distribution. We have provided one example of our features in Figure 2.1.

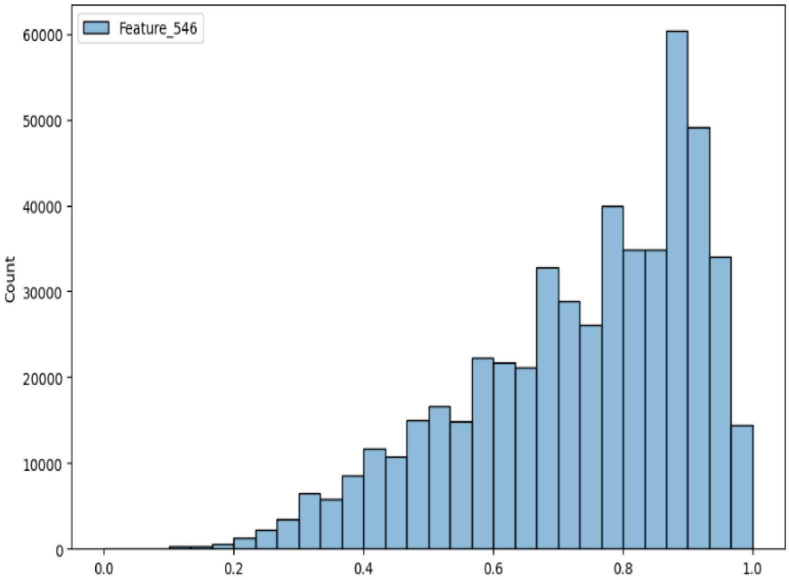

Figure 2.1 Histogram Plot of Feature 546

Therefore, we eliminated parametric options for similarity testing that assume normality, such as the t-test and z-test as stated in Delacre, Lakens & Leys (2017). After that, we investigated other similarity tests and metrics that are robust to deviation from normality distributions and non-parametric.

2.2 Similarity Metrics for Continuous Features

From those non-parametric and distribution-free similarity tests, we have started our experimentation with KS-Test. The test was developed by the studies conducted by Kolmogorov and Smirnov as explained in Dodge (2008). The test approach is based on the maximum deviation from one distribution to another. The test formula provided below represents the supremum of the distance between the empirical cumulative distribution functions of two samples. In the formula F_1 and F_2 represent empirical cumulative distribution function of two sample data, $sample_1$ and $sample_2$ respectively. The KS test statistic is shown as:

$$D = \sup |F_1(x) - F_2(x)| \tag{2.1}$$

The simplicity of this approach makes this test applicable to real-world problems. As Jr. (1951) demonstrated, while the test is powerful at finding location and shape differences, it is not sensitive to the differences in tails of distribution. Moreover, because the main focus is on maximum deviation for the test, it is not powerful at detecting small differences throughout the distribution. Although it is a distributional-free method, the decision of this test relies on the p-value. P-value is unreliable when a large dataset is used. Recent advances in the availability of big data also have revealed the limitation of p-value-based statistical tests. The analysis conducted by Lipsmeyer (2013) has shown us that even in the small or trivial differences between data sets, the p values could become extremely sensitive as the size of the samples increases. Thus, this situation showed that with large samples, the statistical significance becomes a trivial statement, as shown in their study. However, as mentioned in the study by Wasserstein & Lazar (2016), the statistical difference does not mean a real difference between two data sets, especially for the big data. What is important in their work is that with a large enough sample size, any trivial difference can yield a statistically significant result. As imagined, this situation makes relying on p-value lead to misleading results for real-world applications in which most of the samples are classified as large sample sizes.

In Figure 2.2, the x axis represents the sample ratios, the bar graph on the plot represents the KS statistics value. The line graph in Figure 2.2, represents the p-value of KS statistic. The situation mentioned in the previous paragraph for p-value calculation, can be seen in Figure 2.2; as the sampling ratio decrease, the p-value gets larger and larger. However, this means that if the sample size is relatively much smaller than the main data, it is more probable to have a larger p-value. This situation is not logical, so it is also observed by our experimentation that this p-value cannot be used for a decision of representativeness. However, as it can be derived from Figure 2.2, the statistic value is not working as a p-value and does not depend on the sample size. Thus, this makes the statistic value of the KS Test reliable for usage in similarity testing.

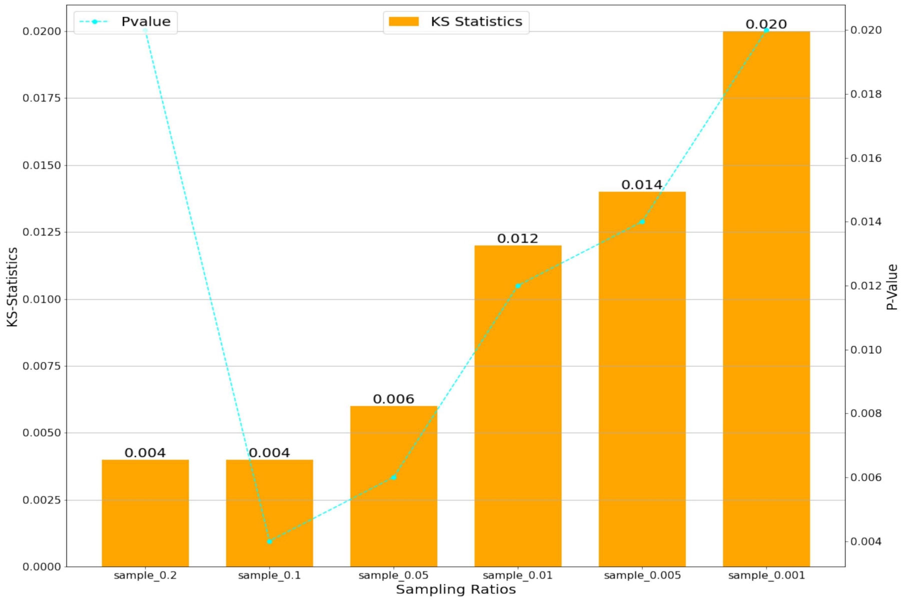

Figure 2.2 KS Test Statistics and P Value Change with Sampling Ratio

In our experiments, we have observed that scaling the data does not impact the KS statistics p-value. Whether the scaling is applied or not, the KS test statistic gives the same results. However, this is not the case for all methods we have used for similarity. Thus, while using those test statistics, we have scaled our continuous feature set with the min-max scaling method.

Before moving on to the similarity metric results, decision thresholds for accepting a test result indicate whether the two datasets are similar to each other or not. Although we did the research to find the right threshold for the methods used, a proper threshold doesn't exist for some of the metrics we have used. To overcome this problem about the threshold point, we have applied a method by using uniform and normal distributions. Firstly, we randomly selected 500000 samples from a dataset which is uniformly distributed between 0 and 1. To that data, we added a random noise variable which is normally distributed, with a mean equal to zero, and different standard deviation levels at 0.1, 0.05, 0.01, and 0.001. Then, for each standard deviation, we have created similar results for each test metric or statistic used, and calculated their corresponding thresholds.

After determining all of those thresholds for each of the metrics we would use for our experiments, we have started to apply our tests to each sample with different sample sizes. In addition, we repeated this process for each sample size every time with different random states. For each sample size, and for each feature we take the average of all those twenty trials, and thus, detected the result of metric or statistical test.

The KS test is the first test to be used for similarity testing in our experiments. This test is preferable for us because it is able to give insights about the overall distribution similarity between two data sets and is more robust to outliers. Figure 2.3 below illustrates the KS Test similarity score between the main data and sample data with different ratios for all of the features.

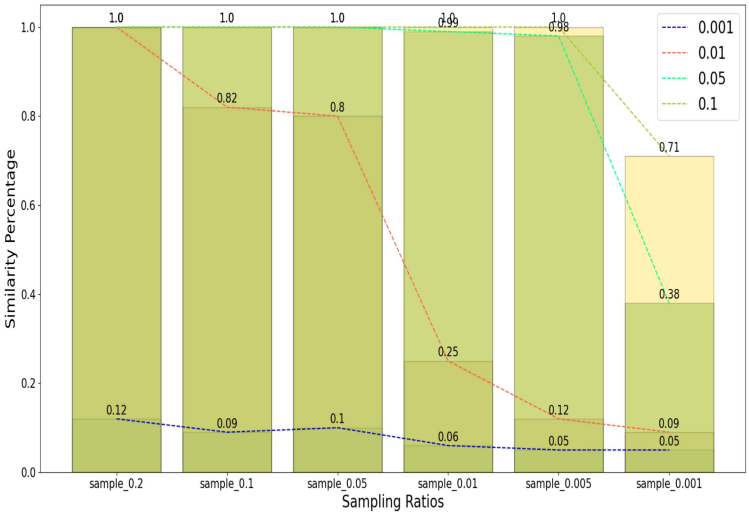

Figure 2.3 KS Test Statistics Similarity by Sample Size and Threshold

In Figure 2.3, the x axis represents the sampling ratios derived from the main. The y axis represents the similarity percentage of the main data between sample datasets. This similarity percentage is calculated as follows:

- Calculate the KS Test static value of relative sample size, which means 1202 continuous features
- Calculate how many features pass the test for each threshold value calculated based on using different standard deviations
- Divide those features that passed the test by the total number of continuous features; reach the calculation of similarity percentage.
- Calculate these results twenty times for different random states, and report their average as the last result of the similarity percentage value

Thus in Figure 2.3 each bar represents the similarity percentages, and line graphs demonstrates the trends as the sample size decreases. Also, those line graphs are colored differently to indicate that each different colored line represents the different standard deviation-based thresholds. As we can see from the figure, there is a very high similarity when the sampling ratio equals 0.2. Except for the smallest ratio used, which indicates 0.001 standard deviations, the data performs nearly perfectly for other threshold values. All of those test results are very close to 1. Even though sampling ratios get smaller, in the thresholds based on standard deviations 0.005, and 0.1, the samples pass the test in greater percentages. With the deviation 0.01, until sample 0.01, the sample datasets mostly pass the test. Therefore, this situation actually indicates that overall distribution of the main and sample test preserved especially until the sampling ratio 0.05 of main data. However, just by looking the KS test, it is hard to say the data preserves all structure even if in its extreme values.

The next metric we examine is the Wasserstein Distance. This test relies on the magnitude of distribution differences rather than statistical significance. One of those novel methods is the Wasserstein distance, in other words Earth Mover's Distance(EMD). This method suggested by Peyré & Cuturi (2020) has a formula for two probability distributions P and Q which is defined as below:

$$W(P,Q) = \inf\{\mathbb{E}[d(X,Y)] : X \sim P, Y \sim Q\} \tag{2.2}$$

In the formula provided above, $X \sim P$ represents the random variable derived from the probability distribution P, and $Y \sim Q$ represents the random variable derived from the propability distribution of Q. $d(X,Y)$ stands for the distance between points X and Y and the formula is basically infimum of expected value of this distance. The Wasserstein distance additionally quantifies how much the two distributions differentiate from each other and whether they differ. Thus, this insight, given by the Wasserstein distance, makes this metric also valuable for real-world applications. Also, the calculation of this metric for distribution with finite samples is as follows:

$$\hat{P}_n = \frac{1}{n}\sum_{i=1}^{n} \delta_{x_i} \tag{2.3}$$

$$\hat{Q}_m = \frac{1}{m}\sum_{i=1}^{m} \delta_{y_i} \tag{2.4}$$

In this formula provided above δ_x stands for Dirac mass value at point x, for these empirical distributions, Sommerfeld & Munk (2017) showed that the Wasserstein distance can be computed as:

$$W^1(\hat{P}_n, \hat{Q}_m) = \min_{\pi \in \Pi(\hat{P}_n, \hat{Q}_m)} \sum_{i,j} \pi_{i,j} d(x_i, y_j) \tag{2.5}$$

In the formula $\Pi(\hat{P}_n, \hat{Q}_m)$ represents the set of $n \times m$ matrices with row sum $1/n$ and column sums $1/m$. The Wasserstein Distance is a more comprehensive metric than the KS Test. This is because the optimal distance for converting one distribution to another can be observed.

From Figure 2.4, it can be seen that the Wasserstein Distance behaves more powerfully than the KS Test based on its comprehensive approach. As can be seen from Figure 2.4, a 0.01 standard deviation-based threshold dramatically decrease the Wasserstein Distance by increasing the sample size. However, when we look at the standard deviation point, is labeled as 0.05, it can still be concluded that the sample is similar to the main data until the sampling ratio is 0.01.

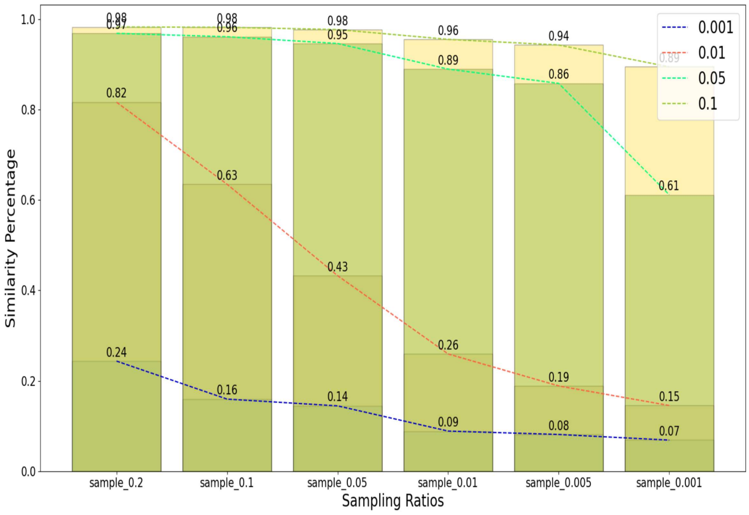

Figure 2.4 Wasserstein Test Statistics Similarity by Sample Size and Threshold

Moreover, we also investigated density function behaviour of our main data and related samples. In order to do that, we first need the estimates of the pdf of our main and sample datasets because the distribution is unknown for most of the features and it is hard to fit a known distribution. Therefore, we have used kernel density estimation to achieve this. After predicting density functions of our datasets, we have created 100 samples in the interval of each feature's minimum and maximum values. After we got the results of those random samples in the predicted density function, we obtained sufficient conditions to compare the two dataset's density function behavior. We have used two different metrics for this aim. The first metric we have used is KL divergence. As proposed by Kullback & Leibler (1951) provided the formula as follows:

$$KL(P||Q) = \int p(x)\log(p(x)/q(x))dx \tag{2.6}$$

As indicated in the formula itself, it measures the relative entropy between two probability distributions, which are stated as P and Q and their probability density functions as $p(x)$ and $q(x)$ in the formula above. However, there are some limitations of this test. One of the limitations of this method is its asymmetry, and dependency of $Q(x)$ value's being denominator, which means $Q(x)$'s value to be zero, leading the result of the metric being infinity. Thus, in order to overcome those limitations, Wang, Nagy, Gilg & Kuang (2012) offer a methodology in their study by demonstrating the effectiveness of Kernel density estimation (KDE). Using KDE leads to more robust estimates especially for the case of limited samples. The results of this metric can be seen in Figure 2.5. Because the KL divergence is not a symmetric metric, we examine the main data distribution with sample data by giving priority to the main data distribution. When the results are observed from the figure, it is the most powerful test for deviation from the main distribution. Although it dramatically labels datasets as not representative according to other tests we have used, actually it is correlated with their performance on the classification algorithms.

For the maximum threshold value based on 0.1 standard deviation, the first three sample sizes gives the highest similarity results. However, there is a sharp decrease in KL divergence, after the sampling ratio decreased to 0.01. This is the most correlated part of the model performance results according to the sampling ratios. When we change threshold, the KL divergence becomes a much more powerful test, and those results become much more powerful than can be accepted. Thus, it can be determined for this test that a standard deviation of 0.1 is more suitable for deciding on the similarity scores. In addition to KL divergence, Lin (1991) proposed a method called The Jensen-Shannon (JS) divergence, which handles some of the limitations that KL divergence has. As formulated in the original work conducted by Lin (1991), the method is defined as follows:

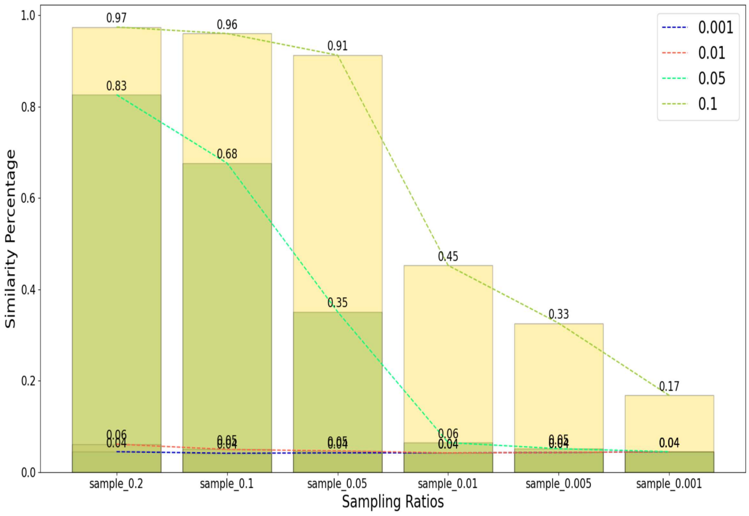

Figure 2.5 KL Divergence Similarity by Sample Size and Threshold

$$JSD(P||Q) = \frac{1}{2}KL(P||M) + \frac{1}{2}KL(Q||M) \tag{2.7}$$

In this equation, M represents $\frac{1}{2}(P+Q)$ which is the average distribution of two samples. In his study, Nielsen & Dane-Nielsen (2014) demonstrated that the square root of the JS divergence satisfies the triangle inequality, which makes this metric symmetric, which is not the case for KL divergence. This property makes this test more preferable in terms of applicability. Thus, the difference between the two metrics is one is symmetric, and the other one is not symmetric. Therefore, JS divergence can be defined as symmetric version of KL divergence. The results of JS divergence more conservative on eliminating feature similarities. This is demonstrated in Figure 2.6 provided below for JS Divergence results.

For the threshold values based on standard deviations 0.1 and 0.05, the metric behaves more steadily. Except for the last sampling ratio, all sampling ratios have performed well in terms of JS Divergence. However, there is a sharp difference between the threshold values for 0.01, 0.001, and 0.1, 0.05. Thus, those thresholds for standard deviations 0.01 and 0.001 are unreliable results.

As a result of those analyses, we can conclude that based on our experiments, until the sampling ratios 0.005 and 0.001, the row-wise stratified random sampling represented the main data until the sampling ratio of 0.01. Moreover, test statistics such as KS Test, results statistics data, which demonstrate representativeness, reach sampling ratios with 0.005 of the size of the main data. In addition to those tests, we

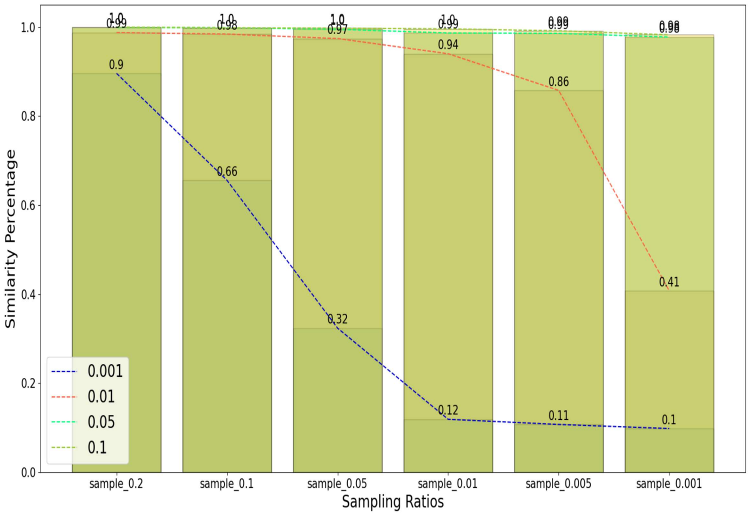

Figure 2.6 JS Divergence Similarity by Sample Size and Threshold

have also applied experiments for Mann Whitney U test and Anderson Darling Test. Mann & Whitney (1947) developed a method for two independent samples. The test statistic shown below does not make any normality assumptions and provides a powerful alternative to the distribution-based tests mentioned above.

$$U = n_1 n_2 + [n_1(n_1 + 1)/2] - R_1 \tag{2.8}$$

In the formula, n_1, n_2 are the sample sizes of the sample 1 and sample 2, and R_1 represents the sum of ranks of the first sample. The study conducted by Mann & Whitney (1947) is based on the Wilcoxon Test as stated in Noether (1992) , which also proposes a rank-based method similar to this test. According to this paper, the test's approach is combining both sample data, ranking all observations coming from all of the two sample data, and then analyzing this ranking accordingly. This method is not sensitive to outliers in data with this approach. According to Blair & Higgins (1980), the Mann-Whitney test achieves 95% of the power of the t-test even if the data is distributed normally. Considering the robustness of outliers, deviation from normality, and efficiency compared to other methods, the Mann-Whitney U Test can be classified as a powerful statistical analysis tool.

The study conducted by Anderson & Darling (1952) improved the approach of Cramer von Mises by modifying this approach to make it more tail-weighted. Their test statistic formulated for finite samples is as follows:

$$A^2 = -n - (1/n)\sum_i (2i-1)[\ln F_1(Y_i) + \ln(1 - F_2(Y_{n+1-i}))] \tag{2.9}$$

In the formula n represents the sample size, F_1 and F_2 represent the cumulative distribution function of This enhanced approach, which gives more weight to the tails of the distribution differences, makes this approach essential, especially for applications in which extreme values are of special importance, such as financial applications.

Because the high dependency on a sample size of the Whitney U-test leads to inconsistent results for the similarity of the main data and sample data. The results of an experiment on sample data and main data similarity using the Mann Whitney-U test have a very high percentage of features that pass the Mann Whitney-U test such that 99% of features pass the test even if the sample size equals 0.005 of the main data size. These results are obtained by using the threshold values derived from 0.001 standard deviations. Even if we use this threshold, the result of representativeness did not change or reduce. Thus, the test is not reliable for us to use it.

However, because of the ability of the Anderson Darling test to detect dissimilarity between two datasets, especially on the tail, it is worth sharing the experiment results. Thus, we have provided the results of the test in Figure 2.7.

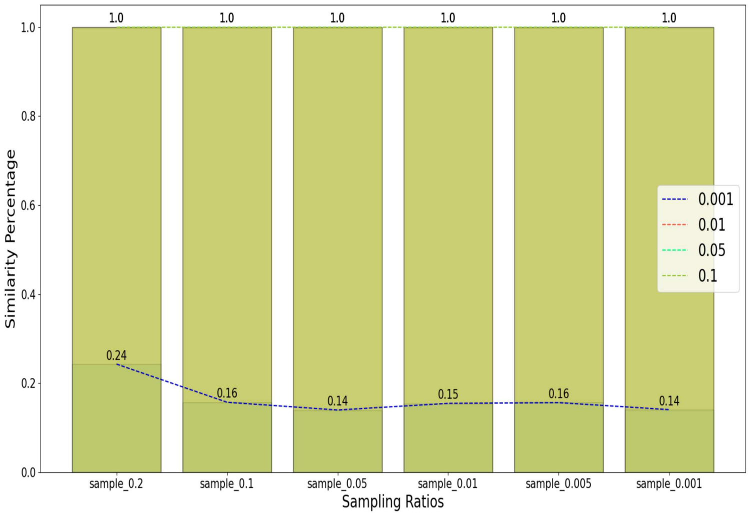

Figure 2.7 Anderson Darling Test Statistics Similarity by Sample Size and Threshold

However, Anderson Darling's results show the effect of sample size; thus, they are not reliable for our case. Thus, we relied on the tests we applied above to calculate similarity between the main dataset and related sample datasets.

2.3 Similarity Tests for Categorical Features

The second part of our similarity calculations is based on suitable tests and metrics for categorical features. We have also applied those similarity metrics to continuous features by binning them into 10 equal quantiles. Those metrics we will be using for this part are the Population Stability Index, Chi Square test, Hellinger Test, Overlapping Coefficient, Total Variance Deviation, and, finally, JS divergence.

Before applying those tests to categorical, we investigated the frequencies of each unique value of those features. If the frequency of a unique value is less than 0.001, then we have grouped it as a low-frequency category. This is because when the frequencies get very small, the PSI or Chi-Square Test does not work because the probability of observing this unique group in my test data becomes very small. Thus, PSI and Chi-Square test could not produce a result. However, we do not use this property for continuous features. Instead, we have used quantile binning for continuous features based on main dataset values. Also, we apply PSI separately for null ratios of features. This means that null ratio difference has become another experiment for us to develop.

PSI is the first test we have started our similarity score calculation for categorical and binned continuous features. Population Stability Index (PSI) is an essential tool in credit scoring applications especially for monitoring purposes such as investigating population drift. As Potgieter, van Zyl, Schutte & Lombard (2023) demonstrated, PSI has a formula as:

$$\sum_i (P_{1i} - P_{2i}) \ln(P_{1i}/P_{2i}) \tag{2.10}$$

In the formula P_{1i} is the probability of the i^{th} category or event in the first distribution, and P_{2i} is the probability of the i^{th} category or event in the second distribution. As Taplin & Hunt (2019) mentioned some application issues of this index, those include the correct binning for continuous data, and preventing zero frequencies may occur in bins by introducing small constant which avoids undefined logarithms. Extensive studies have been conducted in order to determine validated threshold values for PSI. One of those studies which is conducted by Potgieter et al. (2023), demonstrated PSI values below 0.1 does not indicate any significant population change. However, PSI values between 0.1 and 0.25 could mean a moderate shift between two populations compared.

Although PSI has a great field of usage for detecting population shifts in credit scoring problems, the test did not perform well in our case. The test results of PSI get very high scores in terms of our similarity scores. Even if the sampling ratio equals 0.001 of the main dataset row size, the similarity score has reached 0.99. Thus, from our point of view, the results are not sufficient and enough to say that the two datasets are similar to each other. Another test used for population or, in other words, categorical group change comparison is the Chi-square statistic. The test's main aim is to calculate deviations from two distributions and calculating the differences between observed and expected frequencies of two distributions or two independent samples. In their study, Serna, Vargas Cardona, González, Cárdenas-Peña & Orozco Gutierrez (2020) demonstrates its effectiveness for classifying categorical data by proposing a method using the Chi-square statistics and t-SNE. Also, in their study, Boriah, Chandola & Kumar (2008) indicate the effectiveness of Chi-Square in detecting categorical relationships by evaluating various similarity measures For Chi-square tests, also the results are nearly perfect in similarity between the main dataset and related sample datasets. Thus, we could not reach the desired solutions for those two tests. The reason behind this could be that we have used predetermined thresholds which derived from the literature. Therefore, we have used other metrics discussed above.

Tyler, Du, Feng, Bai, Xu, Horowitz, Stone & Celi (2018) investigated the Overlapping Coefficient in their study. Tyler et al. (2018) state that the Overlapping Coefficient measures the shared area between two probability distributions. They investigated this metric's application in detecting the overlapping of clinical lab values. The study demonstrated the utility of this metric in medical research. Also, in another study,Franklin, Rassen, Ackermann, Bartels & Schneeweiss (2014) used OVL for aiming to evaluate covariate balance in cohort studies. They demonstrated this metric's effectiveness in detecting overlapping regions between two distributions. From Figure 2.8, the result of the overlapping coefficient can be seen. As can be derived from Figure 2.8, the first four sampling ratios perform well for every threshold tried. After the sampling ratio reduction to 0.005 of the main dataset, the similarity still can be considered acceptable levels; however, for small size standard deviation thresholds, the similarity value decreased sharply. It is also a logical result

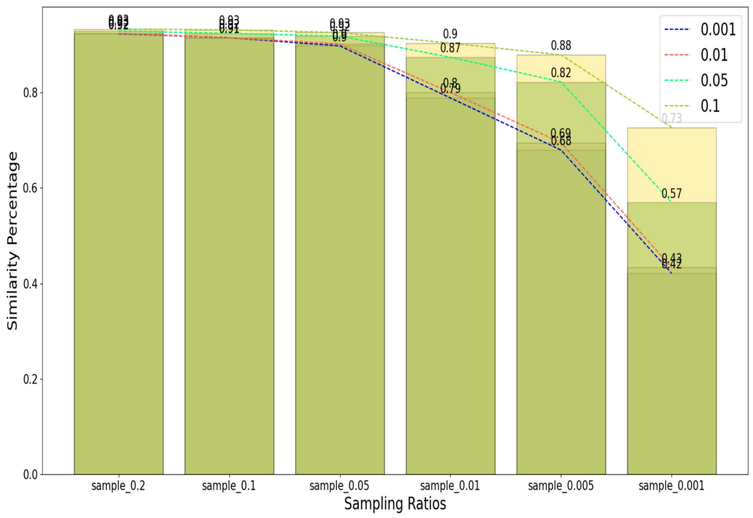

Figure 2.8 OVL by Sample Size and Threshold

for this similarity metric as the sample size decreases dramatically compared to the main dataset; similar frequencies for each bin or category would be much harder. As a result it becomes observable to have such reduction in very small thresholds.

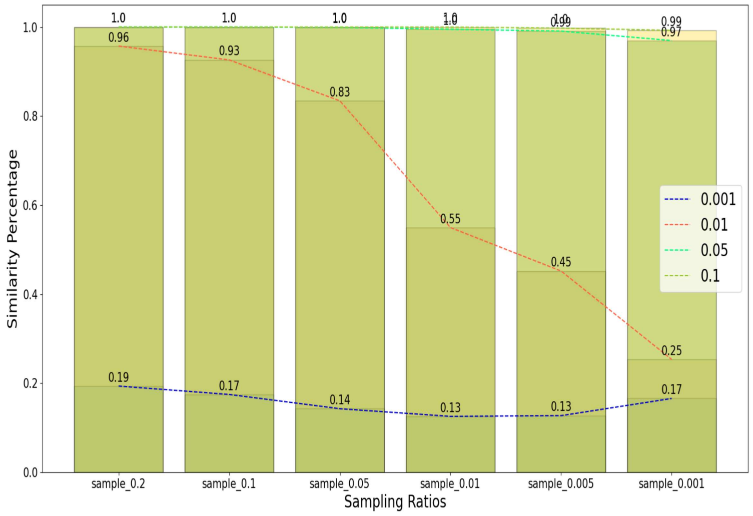

Figure 2.9 Hellinger Distance by Sample Size and Threshold

Another metric used to quantify the similarity between distributions is the Bhattacharyya Distance. A study by Becker & Becker (2023) provides the information that the metrics effectively detects the shift in data distributions while capturing the geometric similarity of probability distributions. Furthermore, another research about distribution similarity conducted by Cuadras, Cuadras & Greenacre (2006), uses the Hellinger Distance which is defined as a variation of Bhattacharyya Distance. The study explored this metric's effectiveness in handling complex distribution similarities. Additionally, Ayeldeen, Mahmood & Hassanien (2015) highlight its robustness to categorical datasets with high sparsity in the use of classification problems. Until the small deviations, Hellinger distance behaves similarly to chi-square and PSI results as in Figure 2.9. This situation is unsurprising because the Hellinger distance equals the square root of population density differences. Thus it has very similar logic to PSI and Chi square. However, the difference between those two metrics is that the Hellinger distance gives more weight to smaller densities by modifying the effects of all densities more equally. Therefore, as the standard deviation-based thresholds get smaller, a sharp decrease can be observed. The reason is probably based on the increase in the deviation of categories or bins of features which have less frequencies. However, despite all of those observations, first three sampling ratios still have a good performance on the similarity of features. It is not worth relying on the last threshold, which is based on the smallest standard deviation of 0.001. The second metric is Jensen Shannon divergence. However, this time, we derived calculation based on direct densities of categories without estimating a density function. This approach is also applied to manually binned continuous features to measure densities. The results are shown in Figure 2.10.

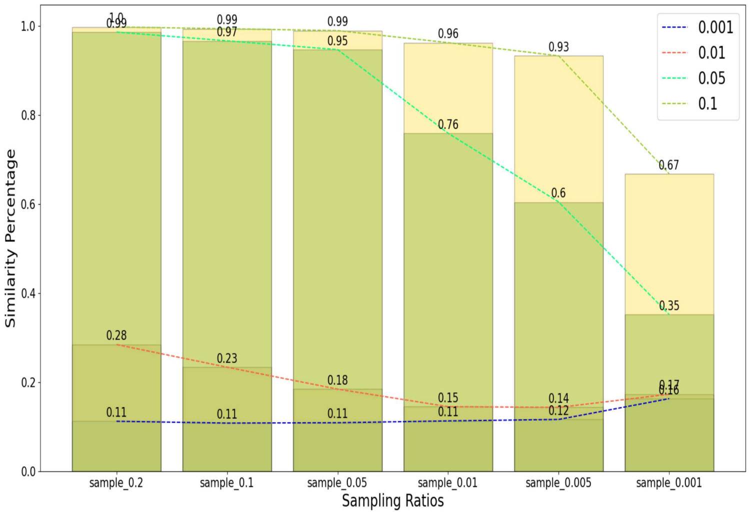

Figure 2.10 JS Divergence by Sample Size and Threshold

As Figure 2.10 indicates, even in the 0.05 deviation-based thresholds, there is a decrease starting from the sampling ratio 0.05. This situation is acceptable since, as we will discuss in the following sections, the model performance acts similarly to those similarities. The trend could be considered similar to the trend in model performances. From this perspective, Jensen-Shannon divergence can be classified as a reliable metric for categorical and binned continuous feature similarities. The last metric for the similarity control of all features is the total variance deviation metric. Total Variation Distance is also a robust metric for categorical dissimilarity. It is an especially useful metric in shifts in categorical data.

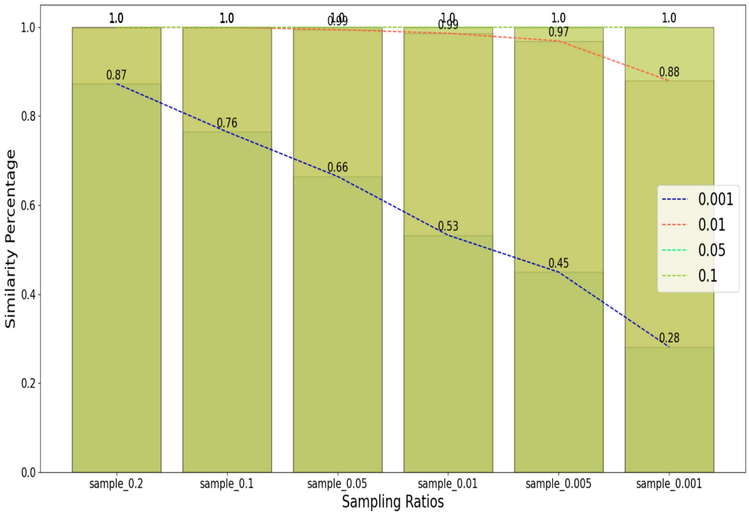

Figure 2.11 TVD by Sample Size and Threshold

From Figure 2.11, we also got highly similar features of the sample dataset with corresponding features of the main dataset. However, the results demonstrated in the figure indicated there is nearly no difference between absolute values of frequencies of categories or bins between the original main dataset and the sample dataset. However, at least in the smallest sample size, a sharp decrease in similarity score is expected; this is not the case for the first three standard deviation-based thresholds. Thus, this indicates that deviation from absolute values of categories and bins do not seem to be a comprehensive method for similarities between main data and sample data.

To conclude, we have used several similarity tests in order to find the representativeness of our main data set and its random sample. Moreover, the repetition of each test twenty times for each sampling ratio of sample datasets. As a result, it can be concluded that both our main dataset and sample dataset are similar to each other until the sampling ratio is 0.005. However, the most robust sampling ratios of datasets in terms of representativeness are 0.2, 0.1 and 0.05. We have done experiments for similarity with different aspects by using a wide range of metrics which focus on different parts of the two datasets' similarity.

3. Similarity Based on Feature Transformation

In this section of the study, we have conducted experiments using PCA. Principal Component Analysis (PCA) is a dimensionality reduction technique that transforms correlated features into a smaller set of uncorrelated variables or principal components, also by capturing the maximum variance with those principal components, this method becomes an effective tool for feature reduction especially existence of multicollinearity according to definition derived from the study by Gewers, Rodrigues Ferreira, Arruda, Silva, Comin, Amancio & da F. Costa (2018). From the study conducted by Reid & Spencer (2009), it is demonstrated that PCA could be regarded as an efficient algorithm in terms of computational time. Thus, this property makes this algorithm suitable for the large-scale datasets used in credit scoring problems. Also, PCA has been used in the study by Maldonado, Perez & Bravo (2017) to preprocess datasets and handle multicollinearity and reduction in feature sets while preserving the information derived from maximum variance. In addition to those studies, Abid, Zhang, Bagaria & Zou (2018) propose different aspects of the usage of PCA; in the study, they analyzed the variance explained by principal components in order to determine whether a sample dataset includes a similar structure compared to the other one.

We have fitted PCA algorithm to our main dataset by setting different components of PCA. First of all, in order to satisfy the requirements of PCA, we have applied some transformations to our main dataset. We have applied imputing, scaling and encoding algorithms to our main dataset. We have applied target encoding on our categorical dataset in order to handle categorical variables. The reason behind using this algorithm is that it does not create additional columns and give higher values if target ratio of one category is higher than others. Thus, in this way it sorts the unique values according to target ratios which help learning process for classification algorithms. For imputing, we impute null values with the median value of each continuous feature in order to prevent effects outliers. Finally, we have applied standard scaler to the data in order to effect of high valued feature effect on PCA algorithm.

After all transformations, we have fitted PCA algorithm on our transformed main dataset, for number of component with 10, 20, 30, 40 and 50. Overall, the main data is fitted with different PCA object having different components five times. After obtaining 5 different PCA models, we have applied the same procedure to the sample datasets as well. Thus, for each sampling ratio, we have 10 random samples each, and we have fitted 5 different PCA models with different number of components. We have used those PCA models initially for similarity concerns. For this purpose, we have used explained variance and controlled reconstruction error after PCA transformation which will be described in detail. We have derived test results with the following combinations:

- PCA model is trained on main data, and tested on main data. Model fitting is repeated for 5 different component sizes. **(Main Model Main Error)**
- PCA model is trained on main data, and tested on sample data. Model fitting is repeated for 5 different component sizes and 6 sampling ratios. **(Main Model Sample Error)**
- PCA model is trained on sample data, and tested on main data. Model fitting is repeated for 5 different component sizes and 6 sampling ratios. **(Sample Model Main Error)**
- PCA model is trained on sample data, and tested on the same sample data. Model fitting is repeated for 5 different component sizes and 6 sampling ratios. **(Sample Model Sample Error)**

One of the metrics we have used is the reconstruction error, which comes from the mean squared error between inverse transformed data and raw data. Since PCA gives a transformation of the data from a higher dimension to a lower dimension, its inverse also exists with some loss of information. We transform the test data with our PCA model, and inverse transform it, and compare the difference for the loss of information. The formula of reconstruction error we have used provided below:

$$\text{Reconstruction Error} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}(X_{ij} - \hat{X}_{ij})^2 \tag{3.1}$$

In the formula, X_{ij} represents j^{th} feature of i^{th} row, of raw data X. $\hat{X}_{ij}$ represents j^{th} feature of i^{th} row, of inverse transformed data $\hat{X}$. Then, we used a reconstruction method for PCA models for each of those transformed datasets with different PCA algorithms. This method is equal to the transformed dataset's inverse transform with principal components obtained from different PCA algorithms. We have also repeated the process for 10 different random state samples, taking the average of those results. In this part, we discuss the results of reconstruction error for PCA models with 20 components and with 50 components.

As it can be observed from Figure 3.1, we have compared PCA model fitted by main data and PCA model fitted by sample datasets having different sample sizes. From Figure 3.1, we have observed that:

- The reconstruction error results for **Main Model Main Error** did not change as the sampling ratio decreases as expected, because the result of this part does not depend on the sampled datasets.
- The reconstruction error results for **Sample Model Sample Error** decreases as the sampling ratio decreases. This indicates overfitting for the PCA algorithm, although it seems very successful on the data which is trained, it has a very poor performance on the main dataset as it can be seen it from **Sample Model Main Error** from Figure 3.1.
- Reconstruction error of **Main Model Sample Error** experiment, increases most of the time as the sampling ratio decreases. However, the change in this experiment is very small. The reason behind this behaviour is although the sampling ratio to be tested changes and gets smaller, the PCA model fitted on main data does not change as the sampling ratio changes. Thus, those results just indicate the main model which is trained in main data performance on sampled datasets with different sampling ratios. Thus there is not so much information derived from those results in terms of representativeness.
- The reconstruction error results for **Sample Model Main Error** and **Main Model Sample Error** increases steadily as the sampling ratio decreases. The behaviour of this case fits to our expectations. Because as the sampling ratio decreases to 0.001 which is a very small sampling ratio, there would be expected less generalizable PCA model which leads to bigger reconstruction error on test data. Also, because the main aim in this experiment to test the representativeness of sample datasets with main data, this case gives the best chance for comparison those results with **Main Model Main Error** results.

Thus, observations we have derived from the results of reconstruction error experiments, led us to focusing on **Main Model Sample Error** and **Main Model Main Error** for understanding the meaning of those results. By just investigating those tow cases, it can be derived the representativeness of sampled datasets for main dataset. The similar values for reconstruction error of **Main Model Main Error** and **Sample Model Main Error** means that PCA model trained with sampled datasets have similar representativeness for main data compared to PCA model trained with main data. Thus, we can say that if two reconstruction error results for each sample ratio (**Main Model Main Error** and **Sample Model Main Error**) is very close to each other, the sampled dataset with given sampling ratio is representative for main dataset. We can say that as

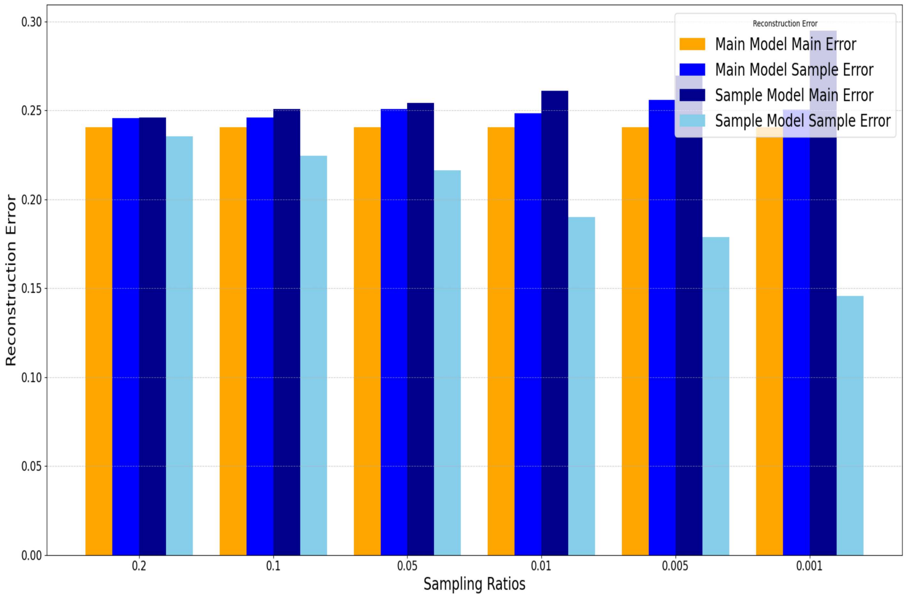

Figure 3.1 Reconstruction Error on Different Sample Sizes with 20 Components

However, it still can be derived that in terms of reconstruction error, the sample datasets of sample sizes 0.2,0.1 and 0.05 of the main dataset have representativeness properties. This means that for those datasets similarity of variance in the whole data captured by the PCA model with 20 components. However, this is of course not perfect similarity. Therefore, in order to observe that there is an improvement in PCA algorithm's representativeness power on main data, we have provided another figure below. In Figure 3.2, this we have applied same experiment for PCA model which has 50 principal components. The expectation of those PCA models have less reconstruction error on main data and sample datasets by not looking at whether it is PCA model fitted by main data or PCA model fitted by one of the sample datasets.

As it can be observed, our expectations have been met by the main model and sample models. The reconstruction error for **Main Model Main Data** was around 0.24 by PCA model with 20 principal components. However, this error decreased sharply in the reconstruction error results of PCA models with 50 components. Even though, as a first insight, it can be derived from the figure with 20 components PCA models, that the reason of similarity for reconstruction error based on having less number

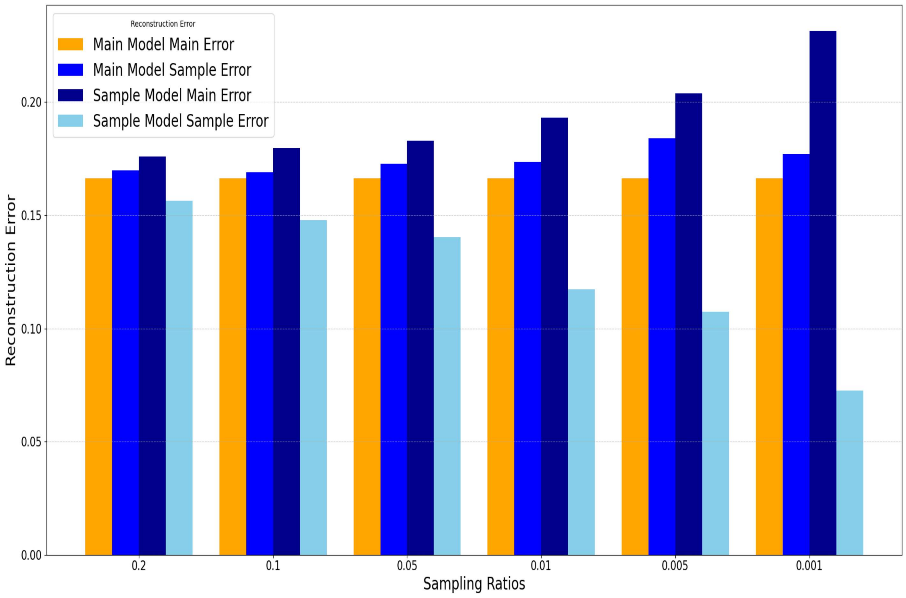

Figure 3.2 Reconstruction Error on Different Sample Sizes with 50 Components

of components for both of the PCA models, which means having less representative transformation occurred while applying those. The results with the PCA models with 50 components actually falsify this insight, because although reconstruction error sharply decreased in the main dataset by the PCA model fitted using main data with 50 components, the reconstruction error of PCA model fitted by sample datasets having sample size of 0.2 of main dataset, also very close reconstruction error result on main data to one with the main PCA model has. However, there are still dramatic deviations starting from sample datasets having a sample size of 0.01 of the main dataset. Thus, those two different figures are consistent with each other.

For the second step, we have reported our different PCA model's explained variance of principal components. The explained variance error formula is demonstrated as follow:

$$\text{Explained Variance Difference} = \frac{1}{k}\sum_{i=1}^{k}\left(PC_i^x - PC_i^y\right)^2 \tag{3.2}$$

In the formula provided above, PC_i^x, i^{th} principal component of data X and PC_i^y i^{th} principal component of data Y. The symbol k means number principal components, the formula means mean squared error between principal components difference of data X and data Y. In other words, we compare the explained variances of the i^{th} principal components of models X and Y, and analyze how similar their explanation power is. For a given number of components, model X always refers to the model trained on the main data, and model Y refers to the model trained on the sample, for the specified sampling ratio. The results have been shown in Figure 3.3. When the figure is investigated, for the sample dataset with sampling ratio of 0.2 has very low MSE results. However what is worth to discuss here, when the model complexity increases, MSE of sample and main models increases as well. Thus, the more complex models actually have more meaningful results in terms of similarity of those features. This is an important aspect for us on comparing model scores as well. Again, the MSE of models does not exceed the 0.1 until the sampling ratio is decreased to 0.01 as indicated in the figure. This insight is repeated insight for other metric discussed in the previous sections as well. Therefore, it can be concluded generally, in terms of similarity of two dataset behaviours, the sampling ratio of 0.05 is the edge point for acceptable similarity.

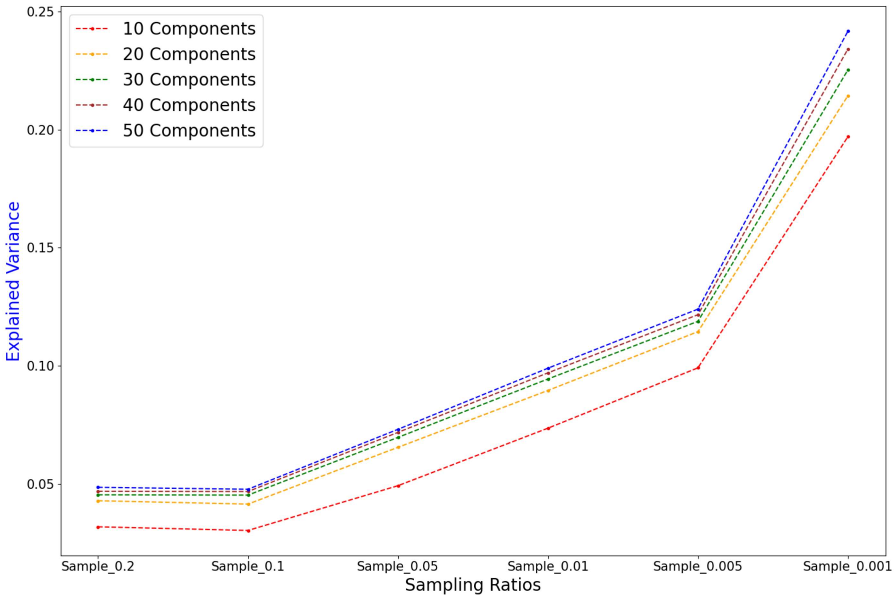

Figure 3.3 Explained Variance Difference MSE of Main Data and Sample Dat

4. Classification Algorithms Performance Results

4.1 PCA Transformed Data Classification Performance

Due to the similar results of the two datasets, we move on to applying modelling trails for our main purpose. For this purpose, in addition to the PCA models, we also use XGBoost to model a credit-scoring problem with a binary classification structure. The algorithm consists of decision trees that construct an ensemble model in order to optimize predictive performance by using gradient boosting techniques. According to Sahin (2020), XGBoost minimizes residual errors by consecutively constructing a decision tree ensemble for each iteration. This structure of algorithm makes it very effective in predictive power of modeling. However, despite its success in the prediction of the model, its performance is highly dependent on the hyperparameter tuning of the algorithm; thus, proper regularization and optimization techniques become highly vital while using it, as mentioned by Shaik, Jongkittinarukorn & Bingi (2024). Moreover, the study conducted by Li, Cao, Li, Zhao & Sun (2020) demonstrated the effectiveness of XGBoost in credit scoring while detecting non-linear relationships between features and imbalanced target handling. However, because our dataset size changes a lot as the sample dataset used for modeling changes, the parameters of the model fitted should also be changed. Therefore, we have applied hyperparameter optimization to datasets transformed by PCA algorithms. After selecting hyper parameters, we have fitted the model accordingly. Thus, we have fitted a total of 35 models with different optimal parameters. As the main purpose of this thesis is to reduce computational time for modelling, we have also calculated computational time for PCA, hyperparameter optimization and final training time of the model for the main dataset and six different sample datasets. In our hyperparameter optimization, we have used Bayesian optimization technique which uses a Gaussian objective function in order to find the best parameters. As a result of those experiments, we first report the average precision score versus the computational time. Average precision score, is a metric that summarizes the precision-recall curve into a single value according to Su, Yuan & Zhu (2015). It is especially useful for imbalanced datasets where positive classes are rare. The formulation stated below from the study:

$$\text{AP} = \sum_{n=1}^{N} (R_n - R_{n-1}) P_n \tag{4.1}$$

In the formula, R_n, The change in recall between the n^{th} and $(n-1)^{th}$ thresholds. P_n, is the precision at the n^{th} threshold and N represents number of all thresholds. The results are shown in Figure 4.1.

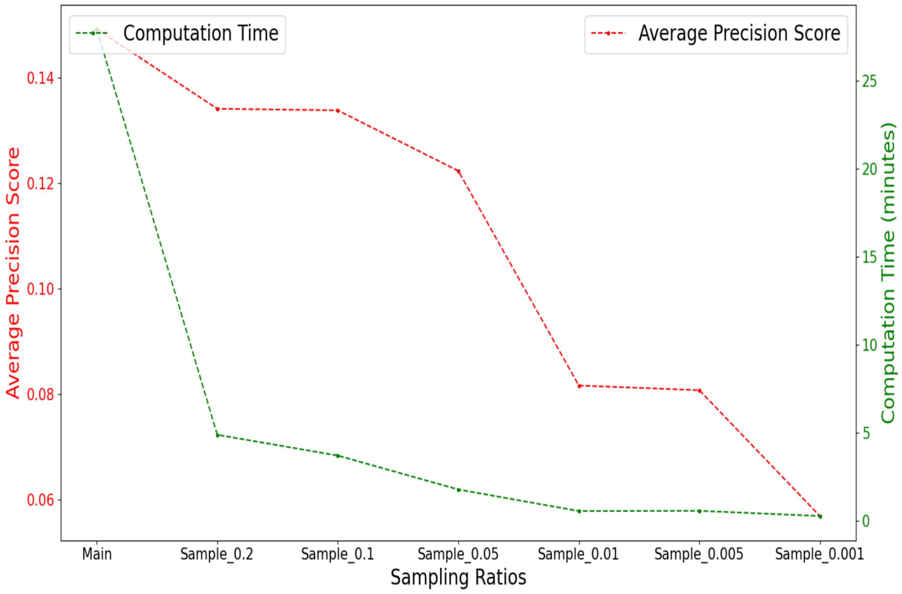

Figure 4.1 Average Precision Score with PCA 20 Components vs Computational Time

From Figure 4.1, as sample ratios decrease, the computational time also decreases. However, what is important information derived from this figure is that there is a sharp decrease in computational time even if using the sampling ratio of 0.2.

Moreover, we have compared the PCA models trained by sample datasets' performance or similarity score on the main dataset. However, while comparing the main model score and sample model scores, we compare their performance on a test dataset, which is independent of the main data set and, as a result, sample datasets. As it can be seen from Figure 4.1, our experiments show that there is only 1 point decrease in the sample ratio which is 0.2 of main dataset. There is also another important point which is even though the sampling ratio decreases from 0.2 to 0.1, the model score does not change. However, it cannot derive the same conclusion for computational time. Computational time sharply decreased from 25 minutes to 4 minutes from main data to 0.2 sampling ratio, this is nearly 1/6 of the main model computational time. Although model performance does not change moving from 0.2 sample ratio to 0.1 sample ratio, the computational time continues to decrease. Thus, this situation makes it preferable to choose sample data that has a sample ratio of 0.1. However, as moving to the sample ratio 0.05, the model decreases around 1 point. However, computational time does not decrease by considerable amounts. For this experiment, it is best to choose the sample dataset with 0.1 of the main dataset according to the results of it.

We have also obtained the score results of the ROC-AUC score, which is also a useful metric for the imbalanced targets. In Su et al. (2015), the ROC-AUC score is described as a performance metric for classification models, particularly binary classification. It measures the trade-off between true positive rate (TPR) and false positive rate (FPR) at various threshold settings.

$$\text{ROC-AUC} = \int_0^1 \text{TPR} \times \text{FPR}^{-1}(x)\, dx \tag{4.2}$$

As the ROC-AUC score increases, the discrimination power of the model also increases. This means that the model is succesful in discriminating positive and negative labeled target values better in high ROC-AUC score. Also, the ROC-AUC scores more on the ranking of instances rather than not only focusing on positive classes. On the other hand, the average precision score tends to be affected by more number of positives than ROC-AUC score.

Therefore, we have also observed the ROC-AUC score results for better understanding the results insight. However, there is a similarity between the results of those metrics derived from a comparison of 6 different sampled datasets with main dataset-based models. The one difference between those datasets, the decrease in ROC-AUC score constantly continues to decrease, thus one can decide which sample to use according to their concern on performance and computational time.

We have done experiments for all five different component sizes, however in this section we provide the PCA model results with 50 components in order to have better comparison with models with 20 components. Thus in Figure 4.3, we have also examined the results for test dataset with PCA models with 50 components.

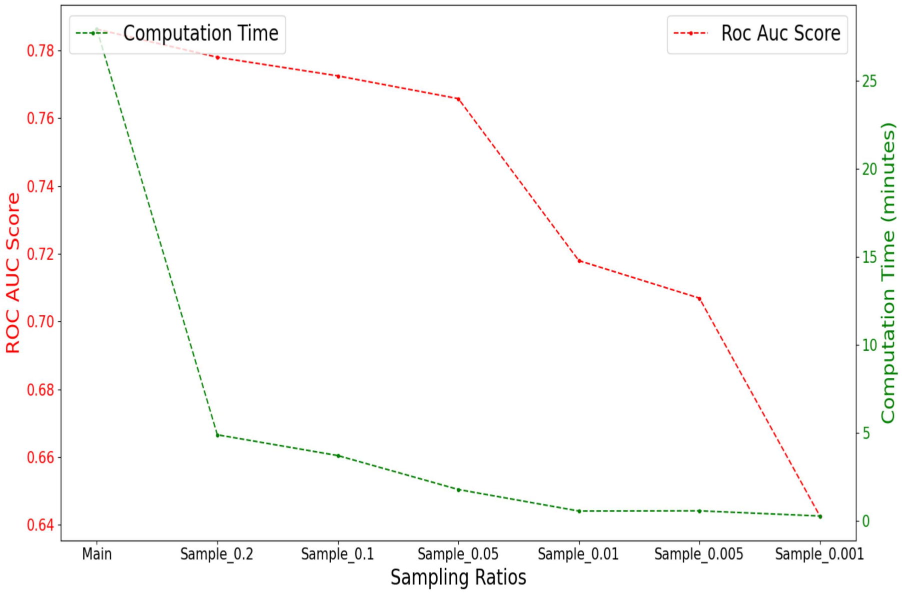

Figure 4.2 ROC-AUC Score with PCA 20 Components vs Computational Time

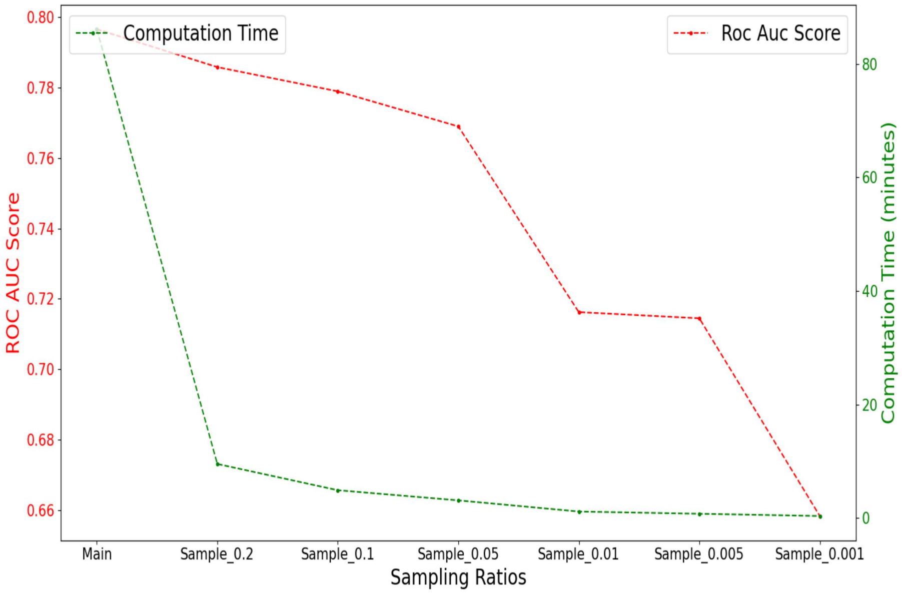

Figure 4.3 ROC-AUC Score with PCA 50 Components vs Computational Time

We have also observed an increase in the model performance, and again, there is around a one-point decrease in the model performance score in the sample dataset with a 0.2 sample ratio of the main dataset. Although there is a steady decrease in ROC-AUC score in each deviation from sample ratio, this decrease in the model performance is less than 1 point until the sample ratio of 0.05. For computational time concerns, there is a dramatic change from the main model to the sample model with 0.2 of the main data size. The computational time decreased from 86 minutes to 9 minutes which almost 1/10 decrease which is much higher than experiments with 20 components PCA models. The computation from sample data to sample data also decreases from 9 minutes to 5 minutes, however again there exists a tradeoff between computational time and model performance. If the model performance results are much more important, even a nearly 1 point decrease, then one should continue with the sample dataset having a sample ratio of 0.2 of the main dataset. However in the case of computation time much more important, then it should be chosen sample ratio with 0.1. However the aim of those experiments is to show that with the huge gain in computational time manner, there does not exist a considerable amount of decrease in the sample datasets. Also we have provided the results of average precision score in Figure 4.4.

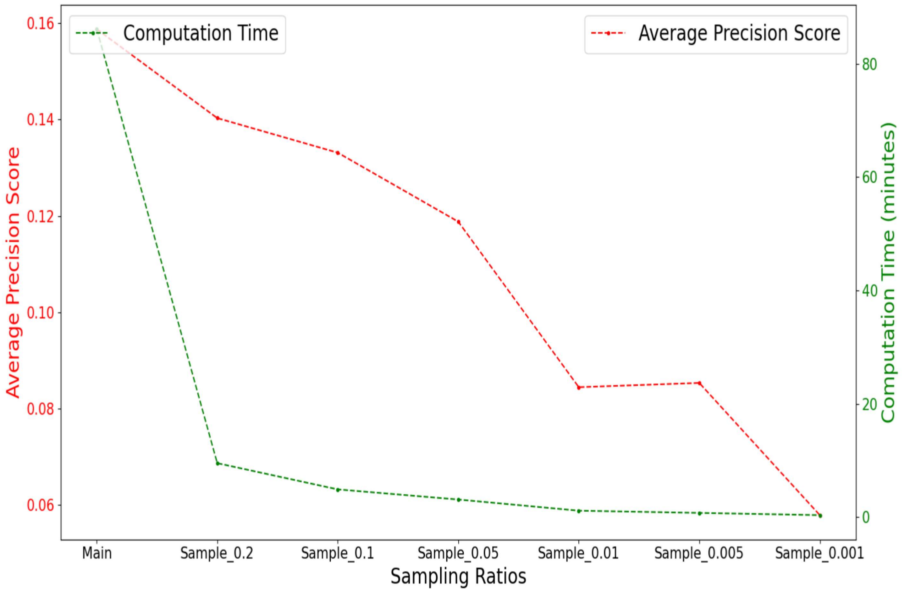

Figure 4.4 Average Precision Score with PCA 50 Components vs Computational Time

The same trend exists in these results. Thus, the average precision score is not discussed further.

4.2 Feature Selection Algorithms Applied Data Classification Results

In addition to those model experiments, we have applied experiments on model performance score without PCA transformation. Instead of using PCA transformation, we have used other techniques for feature reduction. Because we have a huge feature set, we first used the Shap algorithm to select the first 100 features and then used the SFS forward selection algorithm to select 20 features. According to study Uncu & Türksen (2007) called Sequential Feature Selection (SFS). This algorithm is defined as a greedy algorithm where features are added one by one according to the improvement in the model score defined in the study. Also, there is a version of this as a backward algorithm in which features are removed from the model feature set iteratively. The study demonstrates that algorithms are preferable because of their ability to identify the most predictive features, reducing dimensionality while maintaining accuracy. These properties make this algorithm advantageous for our research while using tabular data for credit scoring problems. However, as Nalić, Martinović & Žagar (2020) demonstrated in their study, this algorithm can be computationally demanding in cases such as high-dimensional datasets. Also, Stanczyk (2015) mentioned in his study that the algorithm tends to overfit for small datasets; the reason behind this situation is the selected features' inability to generalize well for unseen data.

In credit scoring, this algorithm is also used in many studies. One of those studies is the one of Koutanaei, Sajedi & Khanbabaei (2015), in which demonstrated how well performed this algorithm in selecting repayment-predictive features while using in hybrid ensemble models.

Thus, for again six different sample ratios, we have done our experiments on Shap algorithm initially. The results of the Shap algorithm derived from an initial XGBoost model select the first most important 100 features for sample datasets and the main dataset. After getting those most important feature list results, we have investigated which of those features overlap the main model's top 100 feature importances. To measure the similarity of feature sets of sample dataset-based models with main model dataset-based features, we have used Jaccard Similarity Metric. Jaccard Similarity as explained in Ye (2014), is a statistical measure used to compare the similarity between two sets. It quantifies the overlap between the sets relative to their combined size. The formula provided in the study as:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{4.3}$$

According to the results based on this similarity metric, we have provided Table 4.1.

Table 4.1 Sample Ratio vs Jaccard Similarity Score on Shap Importance Results

Sample Ratio	Jaccard Similarity Score
0.2	0.63
0.1	0.53
0.05	0.40
0.01	0.34
0.005	0.29
0.001	0.20

According to those results, the similarity of those feature sets does not overlap too much. The maximum value it took for Jaccard similarity is around 0.6; thus, we will look deeper into the reason behind this. The first thing we investigated is that the features that existed in the sample feature list were not in the top 100 main feature list. The number of those features is 22. Also, 22 features exist not in the top 100 feature list but in the main features. Thus, the first control we conducted was for comparing those features to check the correlation between those 22 features that exist only in the sample dataset and 22 features that exist only in the main dataset. The results obtained for those two feature sets listed, which have the maximum correlated features in the feature set, do not exist in the main dataset in Table 4.2.

Table 4.2 Correlation with Features after Shap

Feature Name	Correlation
Feature1	0.9784
Feature 2	0.9860
Feature 3	0.9655
Feature 4	0.8657
Feature 8	0.8392
Feature 9	0.7945
Feature 10	0.6496

When we look at the correlation values, the correlated feature list includes highly correlated features. This situation can potentially change this list with correlated features for each trial. Our experiment also validated this assumption for the same sampled dataset with different random states. Even though the same-sized sample datasets were derived from the main dataset, has low Jaccard Similarity; therefore, it is reasonable that those correlated features led to the selecting one of them while using Shap.

Table 4.3 Feature Names and Their Relative VIF Values

Feature Name	Vif Value
Feature1	576.69
Feature 2	55.00
Feature 3	16.78
Feature 4	12.27
Feature 5	11.29
Feature 6	6.78
Feature 7	6.07
Feature 8	3.63
Feature 9	2.34
Feature 10	1.98
Feature 11	1.95
Feature 12	1.86
Feature 13	1.60
Feature 14	1.40
Feature 15	1.34
Feature 16	1.32
Feature 17	1.23
Feature 18	1.16
Feature 19	1.15
Feature 20	1.01
Feture 21	1.00
Feature 22	1.00
Feature 23	1.00

Moreover, as indicated in Table 4.3, we have calculated VIF for the feature set that does not exist in the top 100 main feature list with top 100 feature list of main model. The Variance Inflation Factor(VIF) is a measure used to capture multicollinearity as stated in Shrestha (2020). From Shrestha (2020), the formula of VIF stated below:

$$VIF = \frac{1}{1-R^2} \tag{4.4}$$

From the Table 4.3, it can be derived from at least 7 features with high VIF values, which means there is a high collinearity of those features with the features of the main dataset. Thus, it is not surprising that those features are used interchangeably with other features in the main dataset. However, until this point, we only investigated the linear relation. The non-linear relations of features are also worth investigating. Also the model fitted also has a nonlinear behavior. Therefore, by also conducted VIF calculation for non-linear actually XGBoost model. In order to conduct such a test, we have calculated R-squared of XGBoost Regressor model which has set of predictors includes top 100 main features and target the each feature of sets of feature which does not exist in main model but sample model. The results cover the remaining open parts which linear model VIF is not enough to explain.

Table 4.4 Feature Names with Related VIF Values Based on Boosting Algorithm

Feature Name	Vif Value
Feature 2	43.64
Feature 12	29.89
Feature 5	24.23
Feature 4	14.53
Feature 1	12.44
Feature 3	12.31
Feature20	12.27
Feature 22	11.71
Feature 6	10.83
Feature 7	9.07
Feature 21	7.04
Feature 10	6.00
Feature 8	5.62
Feature 11	5.11
Feature 9	3.66
Feature 14	3.21
Feature 17	3.20
Feature 13	2.78
Feature 15	2.17
Feature 16	2.11
Feature 21	1.74
Feature 18	1.47
Feature 19	1.39

After observing the results of XGBoost VIF, it can be said that almost half of the features could be predicted by the features of the main dataset model. Moreover, in VIF results based on linear regression, features not having high VIF values, in this calculation gets much higher values. As a result, we can conclude that those features exists in top 100 features of main model dataset can be interchangeably used for features exist in 100 features of sample model dataset.

By using that information, the results of Shap-based important features of the main dataset are similar to those used in the sample dataset. Also, the actual representativeness in our experiment will be observed in terms of model performance calculation. In our experiment, we also have used SFS in order to reduce feature

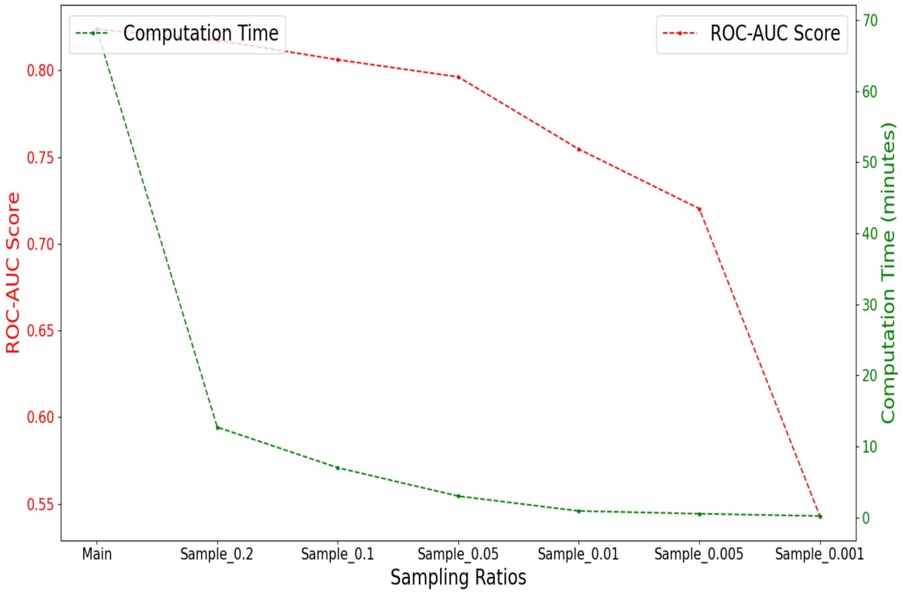

Figure 4.5 ROC AUC Score Classification Results of Features of SFS Selected Feature

datasets of main data and sampled data to 20 features. Those top 100 most important features selected by the Shap algorithm for the main dataset and sampled datasets were reduced to 20 feature lists by selecting those features also using the SFS algorithm.

The final 20 features were selected by the SFS algorithm for the main dataset, and six different sample datasets were used to fit seven different models. Also, we have compared those feature main model performances with six different models of the sampled dataset. After fitting those models with their respective datasets and features using the XGBoost model, we calculated the model performance of those models on test datasets. In Figure 4.5, we have demonstrated the ROC-AUC score on the test dataset, and the computational time for SFS. The similar trend is followed as in the PCA transformed models. However, the reduction in model scores in terms of ROC-AUC score is lower than the models trained with PCA transformed data. Moreover, computational time reduction is a similar amount also. Therefore, those insights from the model scores demonstrate that sampled data is more robust to feature selection algorithms compared to feature transformation such as PCA. However, the reason also might be based on that PCA is effective in capturing the linear relations, however nonlinear relation information might be omitted by this transformation. Also, we have provided the results in terms of average precision score in Figure 4.6.

Additionally, for the SFS process, as we have mentioned before, we applied the Shap algorithm, and then according to the Shap importance results, we eliminated

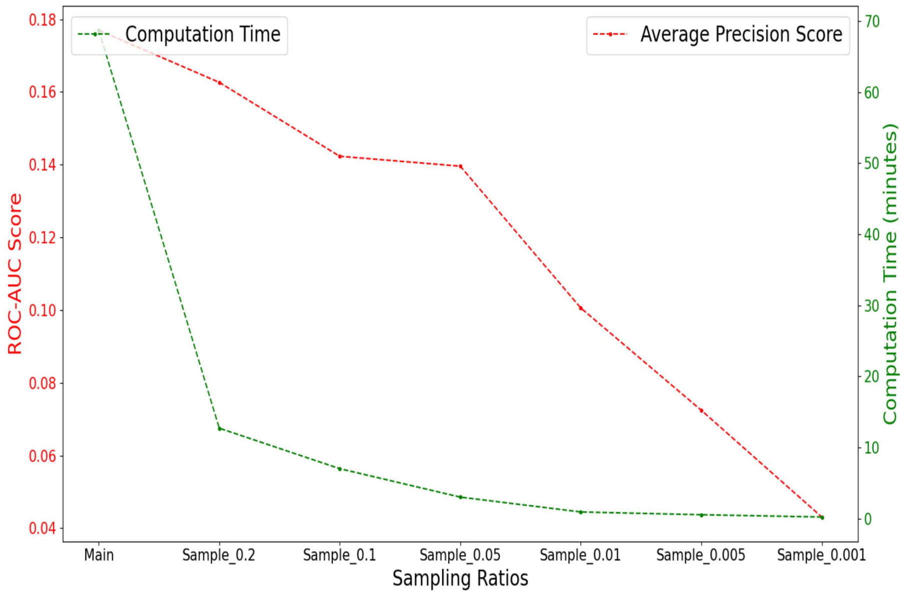

Figure 4.6 Average Precision Score Classification Results of Features of SFS Selected Feature

correlated features before moving towards to the SFS algorithms for all seven models separately. This means that among the correlated features the one with higher Shap importance value remains our feature list, the other one is eliminated from our list. The threshold value for correlation is determined as 0.85. Moreover, there is also computational time for Shap algorithms also which also provided below in Table 4.5.

Table 4.5 Computational Times with Features after Shap

	Computational Time (minutes)
Main	22
Sample 0.2	4
Sample 0.1	2
Sample 0.05	1
Sample 0.01	0.5
Sample 0.005	0.3
Sample 0.001	0.15

5. Conclusion and Future Work

To conclude, we have investigated, firstly, the data similarity between our main dataset with randomly sampled datasets. Our test results demonstrated that until some selected sample size, the random sampled datasets are actually represent the main dataset. Some test results show more power for this context, however other less such as Chi Square tests and PSI. Moreover, we have observed that, also the explainable variance for sample dataset and main data behave similarly, and PCA models', based on those sampled data and main data, reconstruction error of main data again give very similar results with of sampled data until the sample size of 0.05 of main data. However, although most of the tests give reliable results, some tests give more proper results than performance scores of sample datasets on unknown data called test data in our study. The JS Divergence on both categorical and continuous data, the Wassertein Distance and KS Statistics behave give very similar results compared with classification performance results. Moreover, the main conclusion of our study would be that it is not necessary to use all data to obtain a high predictive performance. Although models based on the main dataset give better performance on unknown data, both features with selected feature selection algorithms and transformed with PCA algorithm, the reduction in performance results are negligible especially until the sample size of 0.05 of main datasets, however the reduction in computational time is very high. Therefore, especially for those studies with time concerns, it is better to move with random stratified sampled data when the target ratio is imbalanced.

Future work of this study would be to investigate this random sampling method on unsupervised algorithms performance other than PCA algorithms. The main focus of this study on classification problem, however as a future work it should be investigated the results on regression problems, and also algorithms which use Deep Learning algorithms. Finally, the last future work of this study is to obtain an optimal sample size selection by using an optimization algorithm by defining proper cost functions using similarity scores and computational times. Moreover, it can be extended to also automated sample size selection by using Reinforcement Learning algorithms.

Another way for future work would be to investigate effect of various sampling methods in time series data. The focus of the work could be concentrated on exploring the effect of sequential sampling, window-based sampling, and stratified sampling on the prediction performance of time-dependent datasets. The main areas to be search would be to investigate to performance of different sampling strategies on reflecting the sequential patterns of main time-based dataset, also those methods' success on representation of overall trends, seasonality and anomalies which exist in the time series data. The final aspect of the future work for sampling methods on time-based data, would be to compare the predictive performance of those sampled datasets with various sampling methods, by using time series forecasting models such as ARIMA, LSTM and clustering algorithms as we did for the classification, and clustering algorithms for data with binary target in this study.

BIBLIOGRAPHY

Abid, A., Zhang, M., Bagaria, V. K., & Zou, J. (2018). Exploring patterns enriched in a dataset with contrastive principal component analysis. *Nature Communications*, *9*.

Anderson, T. W. & Darling, D. A. (1952). Asymptotic Theory of Certain "Goodness of Fit" Criteria Based on Stochastic Processes. *The Annals of Mathematical Statistics*, *23*(2), 193 – 212.

Ayeldeen, H., Mahmood, M., & Hassanien, A. E. (2015). *Effective Classification and Categorization for Categorical Sets: Distance Similarity Measures*, volume 339, (pp. 359–368).

Bao, E., Xie, F., Song, C., & Song, D. (2019). Flas: fast and high-throughput algorithm for pacbio long-read self-correction. *Bioinformatics*, *35*(20), 3953–3960.

Becker, J. & Becker, A. (2023). Predictive accuracy index in evaluating the dataset shift (case study). *Procedia Computer Science*, *225*, 3342–3351. 27th International Conference on Knowledge Based and Intelligent Information and Engineering Sytems (KES 2023).

Blair, R. C. & Higgins, J. J. (1980). A comparison of the power of wilcoxon's rank-sum statistic to that of student'st statistic under various nonnormal distributions. *Journal of Educational Statistics*, *5*, 309 – 335.

Boriah, S., Chandola, V., & Kumar, V. (2008). Similarity measures for categorical data: A comparative evaluation. volume 30, (pp. 243–254).

Cuadras, C., Cuadras, D., & Greenacre, M. (2006). A comparison of different methods for representing categorical data. *Communications in Statistics-simulation and Computation - COMMUN STATIST-SIMULAT COMPUT*, *35*, 447–459.

Delacre, M., Lakens, D., & Leys, C. (2017). Why psychologists should by default use welch's t-test instead of student's t-test. *International Review of Social Psychology*, *30*, 92.

Dodge, Y. (2008). *The Concise Encyclopedia of Statistics*.

Franklin, J., Rassen, J., Ackermann, D., Bartels, D., & Schneeweiss, S. (2014). Metrics for covariate balance in cohort studies of causal effects. *Statistics in medicine*, *33*.

Gewers, F., Rodrigues Ferreira, G., Arruda, H., Silva, F., Comin, C., Amancio, D., & da F. Costa, L. (2018). Principal component analysis: A natural approach to data exploration.

He, H. & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263–1284.

Jr., F. J. M. (1951). The kolmogorov-smirnov test for goodness of fit. *Journal of the American Statistical Association*, *46*(253), 68–78.

Koutanaei, F. N., Sajedi, H., & Khanbabaei, M. (2015). A hybrid data mining model of feature selection algorithms and ensemble learning classifiers for credit scoring. *Journal of Retailing and Consumer Services*, *27*, 11–23.

Kullback, S. & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, *22*(1), 79–86.

Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research, (doi:10.1016/j.ejor.2015.05.030).*

Li, H., Cao, Y., Li, S., Zhao, J., & Sun, Y. (2020). Xgboost model and its application to personal credit evaluation. *IEEE Intelligent Systems*, *PP*, 1–1.

Lin, J. (1991). Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, *37*(1), 145–151.

Lipsmeyer, L. (2013). Lin, l. (2013). multiple dimensions of multitasking phenomenon. international journal of technology and human interaction, 9(1), 37-49. *International Journal of Technology and Human Interaction*, *9*, 37–49.

Maldonado, S., Perez, J., & Bravo, C. (2017). Cost-based feature selection for support vector machines - an application in credit scoring. *European Journal of Operational Research*, *261*.

Mann, H. B. & Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, *18*, 50–60.

Nalić, J., Martinović, G., & Žagar, D. (2020). New hybrid data mining model for credit scoring based on feature selection algorithm and ensemble classifiers. *Advanced Engineering Informatics*, *45*, 101130.

Nielsen, C. & Dane-Nielsen, H. (2014). Nielsen & dane-nielsen 2010.

Noether, G. E. (1992). *Introduction to Wilcoxon (1945) Individual Comparisons by Ranking Methods*, (pp. 191–195). New York, NY: Springer New York.

Peyré, G. & Cuturi, M. (2020). Computational optimal transport.

Potgieter, N., van Zyl, C., Schutte, W., & Lombard, F. (2023). The population resemblance statistic: A chi-square measure of fit for banking.

Reid, M. & Spencer, K. (2009). Use of principal components analysis (pca) on estuarine sediment datasets: The effect of data pre-treatment. *Environmental pollution (Barking, Essex : 1987)*, *157*, 2275–81.

Sahin, E. (2020). Assessing the predictive capability of ensemble tree methods for landslide susceptibility mapping using xgboost, gradient boosting machine, and random forest. *SN Applied Sciences*, *2*.

Serna, L., Vargas Cardona, H., González, P., Cárdenas-Peña, D., & Orozco Gutierrez, A. (2020). Classification of categorical data based on the chi-square dissimilarity and t-sne. *Computation*, *8*, 104.

Shaik, N. B., Jongkittinarukorn, K., & Bingi, K. (2024). Xgboost based enhanced predictive model for handling missing input parameters: A case study on gas turbine. *Case Studies in Chemical and Environmental Engineering*, *10*, 100775.

Shrestha, N. (2020). Detecting multicollinearity in regression analysis. *American Journal of Applied Mathematics and Statistics*, *8*, 39–42.

Sommerfeld, M. & Munk, A. (2017). Inference for empirical wasserstein distances on finite spaces.

Stanczyk, U. (2015). Weighting of features by sequential selection. *Studies in Computational Intelligence*, *584*, 71–90.

Su, W., Yuan, Y., & Zhu, M. (2015). A relationship between the average precision and the area under the roc curve. (pp. 349–352).

Taplin, R. & Hunt, C. (2019). The population accuracy index: A new measure of population stability for model monitoring. *Risks*, *7*.

Tyler, P., Du, H., Feng, M., Bai, R., Xu, Z., Horowitz, G., Stone, D., & Celi, L.

(2018). Assessment of intensive care unit laboratory values that differ from reference ranges and association with patient mortality and length of stay. *JAMA Network Open*, *1*, e184521.

Uncu, Ö. & Türksen, I. B. (2007). A novel feature selection approach: Combining feature wrappers and filters. *Inf. Sci.*, *177*, 449–466.

Wang, H., Nagy, J., Gilg, O., & Kuang, Y. (2012). Wang et al 2009 lemmings.

Wasserstein, R. L. & Lazar, N. A. (2016). The asa statement on p-values: Context, process, and purpose. *The American Statistician*, *70*(2), 129–133.

Xia, G.-S., Hu, J., Hu, F., Shi, B., Bai, X., Zhong, Y., Lu, X., & Zhang, L. (2017). Aid: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, *55*, 3965 – 3981.

Ye, J. (2014). Vector similarity measures of simplified neutrosophic sets and their application in multicriteria decision making. *International Journal of Fuzzy Systems*, *16*, 204–211.

Zhang, L., Luo, D., & Du, Y. (2021). Zhang et al., 2021 ear & hearing. *Ear and Hearing*, *42*, 258–270.