

**Pune Institute of Computer Technology  
Dhankawadi, Pune**

**A SEMINAR REPORT  
ON**

Using Data Sampling for Credit Card Fraud Detection on  
imbalanced Dataset.

**SUBMITTED BY**

**Aamir Miyajiwala**

Roll No. 31401

Class TE 4

**Under the guidance of**  
Prof. A.A.Chandorkar



**DEPARTMENT OF COMPUTER ENGINEERING  
Academic Year 2020-21**



DEPARTMENT OF COMPUTER ENGINEERING  
**Pune Institute of Computer Technology**  
**Dhankawadi, Pune-43**

## **CERTIFICATE**

This is to certify that the Seminar report entitled

**“Using Data Sampling for Credit Card Fraud Detection  
on imbalanced Dataset.”**

Submitted by  
Aamir Miyajiwala      Roll No. 31401

has satisfactorily completed a seminar report under the guidance of  
Prof. A.A.Chandorkar towards the partial fulfillment of third year  
Computer Engineering Semester II, Academic Year 2020-21 of  
Savitribai Phule Pune University.

Prof. A.A.Chandorkar  
Internal Guide

Prof. M.S.Takalikar  
Head  
Department of Computer Engineering

Place: PICT, Pune Date: 26/05/2021

## ACKNOWLEDGEMENT

I sincerely thank our Seminar Coordinator Prof. B.D.Zope and Head of Department Prof. M.S.Takalikar for their support.

I also sincerely convey my gratitude to my guide Prof. A.A.Chandorkar, Department of Computer Engineering for her constant support, providing all the help, motivation and encouragement from beginning till end to make this seminar a grand success.

# Contents

<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
<b>2</b>	<b>MOTIVATION</b>	<b>2</b>
<b>3</b>	<b>LITERATURE SURVEY</b>	<b>3</b>
<b>4</b>	<b>A SURVEY ON PAPERS</b>	<b>4</b>
4.1	Using Variational Auto Encoding in Credit Card Fraud Detection .	4
4.2	Influence of Optimizing XGBoost to handle Class Imbalance in Credit Card Fraud Detection Learning. . . . .	4
4.3	A Predictive Analytics Framework to Anomaly Detection. . . . .	4
4.4	Synthetic oversampling with the majority class: A new perspective on handling extreme imbalance. . . . .	5
<b>5</b>	<b>PROBLEM DEFINITION AND SCOPE</b>	<b>6</b>
5.1	Problem Definition . . . . .	6
5.2	Scope . . . . .	6
<b>6</b>	<b>DIFFERENT DATA SAMPLING TECHNIQUES.</b>	<b>7</b>
6.1	Oversampling . . . . .	7
6.1.1	SMOTE . . . . .	7
6.1.2	ADASYN . . . . .	7
6.2	Undersampling . . . . .	7
6.2.1	Random undersampling . . . . .	7
6.2.2	NearMiss . . . . .	7
<b>7</b>	<b>METHODOLOGY</b>	<b>8</b>
7.1	Workflow . . . . .	8
7.2	Mathematical model of SMOTE. . . . .	9
<b>8</b>	<b>Results</b>	<b>10</b>
8.1	Data . . . . .	10
8.2	Implementation Results . . . . .	10
<b>9</b>	<b>CONCLUSION</b>	<b>13</b>
	<b>References</b>	<b>14</b>

## List of Tables

1	Literature survey . . . . .	3
2	Classification Perfromance. . . . .	10
3	Review Log . . . . .	15

## List of Figures

1	Workflow of dealing with imbalanced dataset. . . . .	8
2	This is the sample count for each class.It can be observed that the number of non-fraudulent sample (class 0) is exceedingly greater than fraudulent samples (class 1) . . . . .	10
3	Performance with baseline model.Eventhough the accuracy is a 100 percent, the recall for class 1 is less. . . . .	10
4	Applying SMOTE Oversampling technique. . . . .	11
5	Performance after oversampling with baseline model. Here we can observe that the accuracy dipped a little but the recall value for the minority class has increased. . . . .	11
6	Applying Nearmiss undersampling technique . . . . .	12
7	Performance after undersampling with baseline model.Here, even though the recall value for minority class increased, the recall for majority class decreased significantly along with accuracy which shows that there has been some loss of information from majority class. . . . .	12

## Abstract

E-Commerce and the move towards cashless transactions has resulted in a significant increase in Debit/Credit Card usage. Consequently, this has resulted in a quantum increase in fraudulent transactions. Online fraudulent activities cost the global economy billions of dollars each year. Detecting and preventing anomaly cases like fraudulent activities is an important task for banks, insurance companies and small businesses.

One major problem in building a predictive model for anomaly detection , is the scarcity of anomalous or fraudulent data records in comparison to non-anomalous or non-fraudulent data which makes training imbalanced which will affect the classifier's performance in bad way making it biased.

This work explores the use of predictive analytics for detecting anomaly cases by mitigating the problem of imbalanced datasets using sampling and data augmentation techniques. This approach can be adapted to other domains also where anomalous data is scarce.

## Keywords

Predictive Analytics,imbalanced dataset, anomaly detection,credit card..

# 1 INTRODUCTION

In today's world online payments and transactions have increased significantly due to the rapid development of e-commerce. Consequently this has led to a significant increase in anomalous behaviour leading to fraudulent activities. Majority of these transactions are made using Credit or Debit cards which seem to be secure, however, fraudsters are becoming more and more intelligent and can access the financial records of customer's details internally without getting caught. This gives rise to the need of automated and effective fraud detection methods which leverage the use of Machine Learning and Predictive analytics.

Machine Learning algorithms are being used extensively to tackle this problem due to their ability to learn patterns of fraudulent data from historical records and detect them in future transactions. They work faster than humans can and are scalable i.e. they are adaptable for large datasets also. Due to these merits, most credit card fraud detection approaches make use of machine learning. At present, machine learning has many approaches to solve credit card fraud detection, including unsupervised, supervised and semi-supervised learning; out of which supervised learning is the most popular due to the rise in big data technology.

Frauds can be of different types and can take place in different ways and hence the classification models need to be dynamic in nature to detect the possibility of a fraud. Hence, during the learning phase of the predictive model we need to train it on various fraudulent observations for it to discern a pattern and perform efficiently. However, a significant problem faced while building a fraud detection system is the lack of fraudulent or anomalous data present in the dataset. Number of positive cases (fraudulent) is much lesser than number of negative cases which leads to the problem of imbalanced classification. This impacts the learning phase of the model drastically and therefore its performance. Imbalanced classification causes the model to become biased, i.e. it will be inclined to classify almost every transaction as a non-fraudulent transaction. As the model becomes biased it will treat potential fraudulent observations as noise and discard it. This contradicts the purpose of building a fraud detection model.

Hence, to build an efficient model, several data augmentation techniques need to be tried out and leveraged. Data augmentation is creating synthetic data from minority samples which mimic the behaviour of the minority samples. This will help in creating a balanced dataset and hence remove bias from the model improving its performance. Depending on the dataset one will need to experiment with different techniques to determine which works best.

## 2 MOTIVATION

In recent times, online payment has become very convenient and has been adopted widely especially because of the rapid rise in e-commerce. It facilitates contactless payment and can be carried out anywhere. Due to this significant amount of transactions take place each day and large amount of user information is recorded which is susceptible to data breaches by fraudsters, leading to fraudulent activity.

Anomalous and fraudulent data is getting common so detecting them has become a major task. Apart from normal available patterns in data new patterns are to be detected which brings a challenge for machine learning. Efficiency of detection and dynamic detection over data is also a challenge.

Thus, for effective counter measure for detecting or rather preventing fraud, we need to apply the right data augmentation technique for the model to recognize fraudulent activity and perform well during realtime detection.



### 3 LITERATURE SURVEY

The Following table shows the literature survey by comparing techniques propose in various references:

Table 1: Literature survey

No.	Publication	Technique	Model	Dataset
1	Using Variational Auto Encoding in Credit Card Fraud Detection	Oversampling using 1. SMOTE 2. GAN 3.Variational Autoencoder	Baseline ANN model	European Credit Card Holder
2	Influence of Optimizing XGBoost to handle Class Imbalance in Credit Card Fraud Detection Learning	Oversampling using: 1.SMOTE 2.ADASYN 3.NearMiss 4.SMOTETomek	XGBoost	1.IEEE-CIS 2.European Credit Card Holder.
3	A Predictive Analytics Framework to Anomaly Detection	1.Oversampling 2.Mini-Batch Under-Sampling (Proposed Method)	Random Forest with Feature selection techniques.	European Credit Card Holder
4	Synthetic oversampling with the majority class: A new perspective on handling extreme imbalance	1.SWIM (proposed method) 2.Random Oversampling 3.Random Undersampling	Naive Bayes Decision Trees SVM Multilayer Perceptron	26 Benchmark Datasets.

## 4 A SURVEY ON PAPERS

### 4.1 Using Variational Auto Encoding in Credit Card Fraud Detection

This paper proposes a novel oversampling method based on Variational automatic encoding, combined with deep learning techniques to solve the problem of imbalanced classification often encountered while building fraud detection systems. Their approach involves bridging the gap between the fraudulent and non-fraudulent category of data by developing a framework. The framework will use oversampling of fraudulent data into the original dataset thereby augmenting it. The framework contains a sampling module attached to a baseline neural network model. The sampling module contains GAN, VAE and SMOTE. The classifier is trained with data augmented using these three different models in the sampling module and the performance of each combination is assessed. Their experimental results show that the performance of the baseline classifier was the best when fraudulent data was oversampled using VAE where the increase in samples was 0.5 times the number of original fraudulent cases in the training set. It was also determined that the recall values of SMOTE and GAN were more than VAE and since this comes under the category of supervised learning, the VAE model could perform poorly when dealing with novel fraud data.

### 4.2 Influence of Optimizing XGBoost to handle Class Imbalance in Credit Card Fraud Detection Learning.

XGBoost is a very popular model and one of the favourites of the Kaggle community, used for classification and regression problems. XGBoost is an effective machine learning model, even on datasets where the class distribution is skewed. This paper focuses on optimizing XGBoost model (hence the name OXGBoost) to perform classification of fraudulent and non-fraudulent data without the use of resampling techniques. The parameters of XGBoost are tuned using RandomizedSearchCV. This optimized model is compared with normal XGBoost model combined with sampling techniques over two credit card fraud datasets. Experimental results show that there was no increase in performance when XGBoost was integrated with sampling techniques whereas the OXGBoost model had a boost in performance and could classification of skewed datasets without the need of data sampling techniques.

### 4.3 A Predictive Analytics Framework to Anomaly Detection.

This paper adopts the use of predictive analytics for anomaly or fraud detection. Predictive analytics is a branch of artificial intelligence which learns from historical data to predict future events and give insights. This paper proposes a framework based on predictive analysis which contains three components. First being the feature selection, second being the data sampling technique and third being the predictive model being used. The framework follows a pipeline where

once the data is collected and preprocessed, the framework feeds into the feature selection module, and after that into the predictive model module where the training takes place. During the training process, the problem of imbalanced classification is mitigated by performing data sampling techniques. This paper uses random oversampling and a novel method of Mini-Batch Under Sampling to balance the dataset. The feature selection methods are used with Random Forest along with different sampling techniques where it is found that oversampling gives the best result.

#### **4.4 Synthetic oversampling with the majority class: A new perspective on handling extreme imbalance.**

This paper starts with describing the problem of imbalanced classification and how it can be resolved using oversampling of minority class. However, in the case of extreme imbalance, the minority class is too sparse and rare, and hence, does not contain enough information to perform oversampling. For this reason, they explore the case of extreme imbalance and come up with a solution of utilizing the distributional information in majority class to augment the minority class. With this in mind they propose a novel method of oversampling called SWIM (Synthetic oversampling with the Majority). This unique method focuses on generating the minority instances in regions that have similar density with respect to the majority class and they should be neighbouring the original minority instances. SWIM was tested on 26 benchmark datasets from different domains and compared with alternative sampling techniques. Their experimental results showed that SWIM is superior to other alternative techniques and outperformed them in 23 out of the 26 datasets.

## 5 PROBLEM DEFINITION AND SCOPE

### 5.1 Problem Definition

To design a fraud detection system by using the suitable the right data sampling technique to prevent classification imbalance and increase efficiency of fraud detection.

### 5.2 Scope

Fraud Detection is an important study for research and various methods are being developed to tackle it. Due to the scarcity of fraudulent data in a dataset we need to perform operations on data like oversampling the minority class or undersampling the majority class for the model to recognize a pattern of fraudulent activity.

For the above purpose it is necessary to select the right data sampling technique and machine learning model to create a well balanced dataset and prevent the model from becoming biased.

## 6 DIFFERENT DATA SAMPLING TECHNIQUES.

### 6.1 Oversampling

Oversampling is used to generate examples of minority class in the dataset. It is also called as data augmentation.

#### 6.1.1 SMOTE

This is a type of data augmentation technique for the minority class and is referred to as the Synthetic Minority Oversampling Technique, or SMOTE for short. It is one of the most widely used approach for oversampling the minority class in a dataset. It works by selecting a random sample from the minority class and finds  $k$  nearest neighbours or that sample. Randomly a neighbour is selected out of the  $k$  neighbours and a synthetic example is created at a randomly selected point between the two examples in the feature space.

#### 6.1.2 ADASYN

The Adaptive Synthetic Sampling Method for Imbalanced Data, or ADASYN for short builds on the methodology of SMOTE, by shifting the importance of the classifications boundary to those minority classes which are difficult. ADASYN uses a weighted distribution for different minority class examples according to their level of difficulty in learning, where more synthetic data is generated for minority class examples that are harder to learn.

### 6.2 Undersampling

Undersampling is the technique used for reducing the number of samples of majority class present in the dataset.

#### 6.2.1 Random undersampling

This technique randomly removes samples from majority class, with or without replacement. This one of the earliest techniques used to mitigate the problem of imbalance classification. However, it may increase the variance of the classifier and may potentially discard useful or important samples.

#### 6.2.2 NearMiss

NearMiss is an under-sampling technique. It aims to balance class distribution by randomly eliminating majority class examples. When instances of two different classes are very close to each other, we remove the instances of the majority class to increase the spaces between the two classes. This helps in the classification process. To prevent problem of information loss in most under-sampling techniques, near-neighbor methods are widely used.

## 7 METHODOLOGY

### 7.1 Workflow

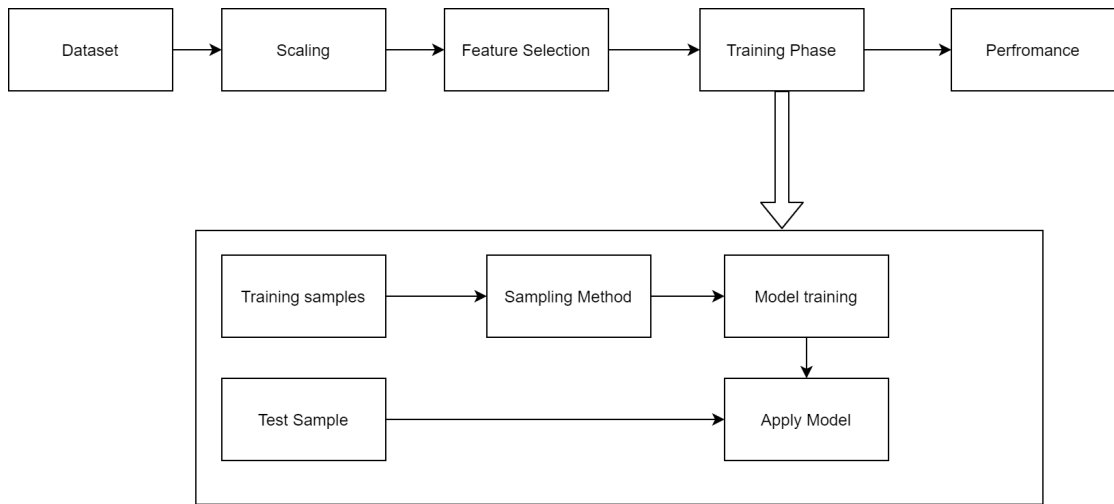


Figure 1: Workflow of dealing with imbalanced dataset.

## 7.2 Mathematical model of SMOTE.

$$S = \{A, x, N, x_k, x'\}$$

Let A be the set of minority class consisting of x data points.

**Step1:** For each  $x$ , the **k-nearest neighbors of  $x$**  are obtained by calculating the Euclidean distance between  $x$  and every other sample in set **A**

**Step2:** The rate N is set according to the imbalanced proportion. For each  $x$  in A, N examples (i.e  $x_1, x_2, x_3, \dots, x_N$ ) are randomly selected from its k-nearest neighbors, and they construct the set  $A_1$ .

**Step3:** For each example  $x_k$  in  $A_1$  ( $k=1,2,3,\dots,N$ ), the following formula is used to generate a new example:  $x' = x + \text{rand}(0,1) * (x - x_k)$   
in which  $\text{rand}(0,1)$  represents the random number between 0 and 1.

## 8 Results

### 8.1 Data

**Dataset** : European Credit Card Holder.

SNo.	Method	Class	F1-Score	Recall	Accuracy
1	No sampling+Classifier	0 (Non fraud)	1.00	1.00	1.00
		1 (fraud)	0.73	0.62	
2	SMOTE+Classifier	0 (Non fraud)	0.99	0.98	0.98
		1 (fraud)	0.11	0.92	
3	Nearmiss+Classifier	0 (Non fraud)	0.71	0.55	0.56
		1 (fraud)	0.01	0.95	

Table 2: Classification Performance.

### 8.2 Implementation Results

```
In [6]: data.Class.value_counts()
Out[6]: 0    284315
        1      492
        Name: Class, dtype: int64
```

Figure 2: This is the sample count for each class. It can be observed that the number of non-fraudulent sample (class 0) is exceedingly greater than fraudulent samples (class 1)

```
In [40]: lr = LogisticRegression()
lr.fit(X_train, y_train.ravel())
predictions = lr.predict(X_test)
print(classification_report(y_test, predictions))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	85296
1	0.88	0.62	0.73	147
accuracy			1.00	85443
macro avg	0.94	0.81	0.86	85443
weighted avg	1.00	1.00	1.00	85443

Figure 3: Performance with baseline model. Even though the accuracy is a 100 percent, the recall for class 1 is less.



```
In [14]: count_of_y1=sum(y_train==1)
count_of_y0=sum(y_train==0)
print(f"Before OverSampling, counts of label '1': {count_of_y1}")
print(f"Before OverSampling, counts of label '0': {count_of_y0} \n")

from imblearn.over_sampling import SMOTE
sm = SMOTE(random_state = 2)
X_train_res, y_train_res = sm.fit_resample(X_train, y_train.ravel())

shape_x=X_train_res.shape
shape_y=y_train_res.shape
print(f'After OverSampling, the shape of train_X: {shape_x}')
print(f'After OverSampling, the shape of train_y: {shape_y} \n')

count_after_y1=sum(y_train_res == 1)
count_after_y0=sum(y_train_res == 0)
print(f"After OverSampling, counts of label '1': {count_after_y1}")
print(f"After OverSampling, counts of label '0': {count_after_y0}")

Before OverSampling, counts of label '1': 345
Before OverSampling, counts of label '0': 199019

After OverSampling, the shape of train_X: (398038, 29)
After OverSampling, the shape of train_y: (398038,)

After OverSampling, counts of label '1': 199019
After OverSampling, counts of label '0': 199019
```

Figure 4: Applying SMOTE Oversampling technique.

```
In [15]: lr_oversampling = LogisticRegression()
lr_oversampling.fit(X_train_res, y_train_res.ravel())
predictions = lr1.predict(X_test)

# print classification report
print(classification_report(y_test, predictions))
```

	precision	recall	f1-score	support
0	1.00	0.98	0.99	85296
1	0.06	0.92	0.11	147
accuracy			0.98	85443
macro avg	0.53	0.95	0.55	85443
weighted avg	1.00	0.98	0.99	85443

Figure 5: Performance after oversampling with baseline model. Here we can observe that the accuracy dipped a little but the recall value for the minority class has increased.

---

```

In [17]: count_of_y1=sum(y_train==1)
count_of_y0=sum(y_train==0)
print(f"Before Undersampling, counts of label '1': {count_of_y1}")
print(f"Before Undersampling, counts of label '0': {count_of_y0} \n")

# apply near miss
from imblearn.under_sampling import NearMiss
nr = NearMiss()

X_train_miss, y_train_miss = nr.fit_resample(X_train, y_train.ravel())

shape_x=X_train_res.shape
shape_y=y_train_res.shape
print(f'After Undersampling, the shape of train_X: {shape_x}')
print(f'After Undersampling, the shape of train_y: {shape_y} \n')

count_after_y1=sum(y_train_res == 1)
count_after_y0=sum(y_train_res == 0)
print(f"After Undersampling, counts of label '1': {count_after_y1}")
print(f"After Undersampling, counts of label '0': {count_after_y0}")

Before Undersampling, counts of label '1': 345
Before Undersampling, counts of label '0': 199019

After Undersampling, the shape of train_X: (690, 29)
After Undersampling, the shape of train_y: (690,)

After Undersampling, counts of label '1': 345
After Undersampling, counts of label '0': 345

```

Figure 6: Applying Nearmiss undersampling technique

```

In [18]: lr_undersampling = LogisticRegression()
lr_undersampling.fit(X_train_miss, y_train_miss.ravel())
predictions = lr_undersampling.predict(X_test)

print(classification_report(y_test, predictions))

```

	precision	recall	f1-score	support
0	1.00	0.55	0.71	85296
1	0.00	0.95	0.01	147
accuracy			0.56	85443
macro avg	0.50	0.75	0.36	85443
weighted avg	1.00	0.56	0.71	85443

Figure 7: Performance after undersampling with baseline model. Here, even though the recall value for minority class increased, the recall for majority class decreased significantly along with accuracy which shows that there has been some loss of information from majority class.

## 9 CONCLUSION

Fraud detection is of utmost importance and while building a system one cannot afford to let the model have any bias towards one class. Hence, it is necessary to perform data sampling techniques to create a well balanced dataset to get rid of any bias. While there are different methods of oversampling and undersampling techniques, oversampling methods have been proved to give a better classification performance.

The scope of Fraud Detection is huge and data sampling can help the machine learning models learn different kinds of frauds by generating synthetic data mimicking fraudulent behaviour.

## References

- [1] S. Sharma, C. Bellinger, B. Krawczyk, O. Zaiane, and N. Japkowicz, "Synthetic oversampling with the majority class: A new perspective on handling extreme imbalance," in Proc. IEEE Int. Conf. Data Mining (ICDM), Singapore, Nov. 2018, pp. 447–456. .
- [2] H. Tingfei, C. Guangquan and H. Kuihua, "Using Variational Auto Encoding in Credit Card Fraud Detection," in IEEE Access, vol. 8, pp. 149841-149853, 2020, doi: 10.1109/ACCESS.2020.3015600.
- [3] C. V. Priscilla and D. P. Prabha, "Influence of Optimizing XGBoost to handle Class Imbalance in Credit Card Fraud Detection," 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), 2020, pp. 1309-1315, doi: 10.1109/ICSSIT48917.2020.9214206.
- [4] J. Wang, R. Martins de Moraes and A. Bari, "A Predictive Analytics Framework to Anomaly Detection," 2020 IEEE Sixth International Conference on Big Data Computing Service and Applications (BigDataService), 2020, pp. 104-108, doi: 10.1109/BigDataService49289.2020.00023.
- [5] Mohammed, Rafiq Ahmed, et al. "Scalable machine learning techniques for highly imbalanced credit card fraud detection: a comparative study." Pacific Rim International Conference on Artificial Intelligence. Springer, Cham, 2018.
- [6] T. Hasanin, T. M. Khoshgoftaar, J. L. Leevy, and N. Seliya, "Examining characteristics of predictive models with imbalanced big data," J. Big Data, vol. 6, no. 1, p. 69, Dec. 2019.

Table 3: Review Log

No.	Date	Points discussed	Suggestions given
1	28 <sup>th</sup> February 2021	Informed guide about the selected domain and shortlisted papers	Discuss the understanding of papers to finalize seminar topic and base paper
2	8 <sup>th</sup> March 2021	Presented the shortlisted papers and discussed their understanding with guide	Seminar guide Prof. A.A.Chandorkar approved a topic and gave a go-ahead.
3	16 <sup>th</sup> March 2021	Presented the literature review progress, discussed problem statement with seminar guide.	Corrections in the problem statement. The literature survey should be thorough.
4	11 <sup>th</sup> April 2021	Abstract Submission for guide's approval	Topic and abstract accepted by the seminar guide



### Document Information

Analyzed document	31401_Seminar_Report.pdf (D106344010)
Submitted	5/25/2021 1:23:00 PM
Submitted by	Arundhati
Submitter email	aachandorkar@pict.edu
Similarity	18%
Analysis address	aachandorkar.pict@analysis.orkund.com

### Sources included in the report

<b>SA</b>	<b>Pune Institute of Computer Technology / 31216_STC_Report.pdf</b> Document 31216_STC_Report.pdf (D67557011) Submitted by: yahandge@pict.edu Receiver: yahandge.pict@analysis.orkund.com		2
<b>SA</b>	<b>final.pdf</b> Document final.pdf (D76257386)		3
<b>W</b>	URL: <a href="https://thesis.cust.edu.pk/UploadedFiles/MCS181049-Asifa%20Kanwal.pdf">https://thesis.cust.edu.pk/UploadedFiles/MCS181049-Asifa%20Kanwal.pdf</a> Fetched: 12/11/2020 5:51:37 AM		4
<b>SA</b>	<b>Pune Institute of Computer Technology / 31166.pdf</b> Document 31166.pdf (D106344011) Submitted by: aachandorkar@pict.edu Receiver: aachandorkar.pict@analysis.orkund.com		1
<b>SA</b>	<b>Pune Institute of Computer Technology / STC_report_aniket pandey.pdf</b> Document STC_report_aniket pandey.pdf (D68357367) Submitted by: prrajmane@pict.edu Receiver: prrajmane.pict@analysis.orkund.com		1
<b>SA</b>	<b>Pune Institute of Computer Technology / 31111_Report.pdf</b> Document 31111_Report.pdf (D67797147) Submitted by: ddkadam@pict.edu Receiver: ddkadam.pict@analysis.orkund.com		1
<b>SA</b>	<b>Pune Institute of Computer Technology / Seminar_Report_Rushabh_Karnvat_31131.pdf</b> Document Seminar_Report_Rushabh_Karnvat_31131.pdf (D68977195) Submitted by: pijambhulkar@pict.edu Receiver: pijambhulkar.pict@analysis.orkund.com		1
<b>W</b>	URL: <a href="http://tharutechnologies.com/570-summon-peel/smote-undersampling.html">http://tharutechnologies.com/570-summon-peel/smote-undersampling.html</a> Fetched: 4/17/2021 9:41:40 AM		1
<b>W</b>	URL: <a href="https://krex.k-state.edu/dspace/bitstream/handle/2097/39692/NasikMuhammadNafi2019.pdf?sequence=5&amp;isAllowed=y">https://krex.k-state.edu/dspace/bitstream/handle/2097/39692/NasikMuhammadNafi2019.pdf?sequence=5&amp;isAllowed=y</a> Fetched: 10/21/2019 2:34:52 PM		1
<b>SA</b>	<b>ICDATA21_paper_14.pdf</b> Document ICDATA21_paper_14.pdf (D105861332)		2

1/13

## Guide's Approval.

### Approval for Submission of seminar report Inbox x



**aamir miyajiwala**

Hello ma'am, Please find attached the final seminar report with the urkund plagiarism report. Do we need to make any changes or can we submit the report? Thank



**aachandorkar**

to me ▾

Reviewed. OK. You can submit, send one final copy to me.

\*\*\*