Detecting Ransomware Transaction

Elaf Alsaedi and Muntaha Jaber

TABLE OF CONTENTS

01

Introduction

What is Bitcoin?

- Virtual currency or a digital currency (is a type of money that is completely virtual)
- You can use it to buy products and services.
- Highly secure
- Users are anonymous

What is a Ransomware?

- Ransomware is a type of malware that threatens to publish the victim's personal data or block access to it unless a ransom is paid.
- The payment in Bitcoin

Why Attacker prefer bitcoin payment ?

- Unlike credit card payment, The transactions with bitcoin are completely anonymous and hard to trace.

Ransomware Attack Simulation

Example of a ransomware attack

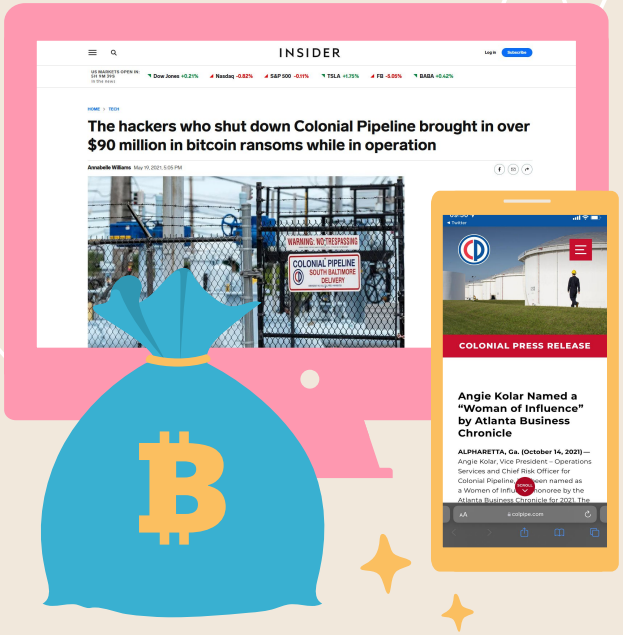
In May 19, 2021, hackers who shut down an oil company earned over

$90M

source

Colonial Pipeline Ransomware

- In May 9, The company was shut down for two days.
- Money was transferred to 47 bitcoin wallets.
- Attackers sent the company a decryption tool.
- By May 13, all the wallets were emptied and they couldn't trace it.



source

02

Problem Statement

Problem Statement

Binary classification problem

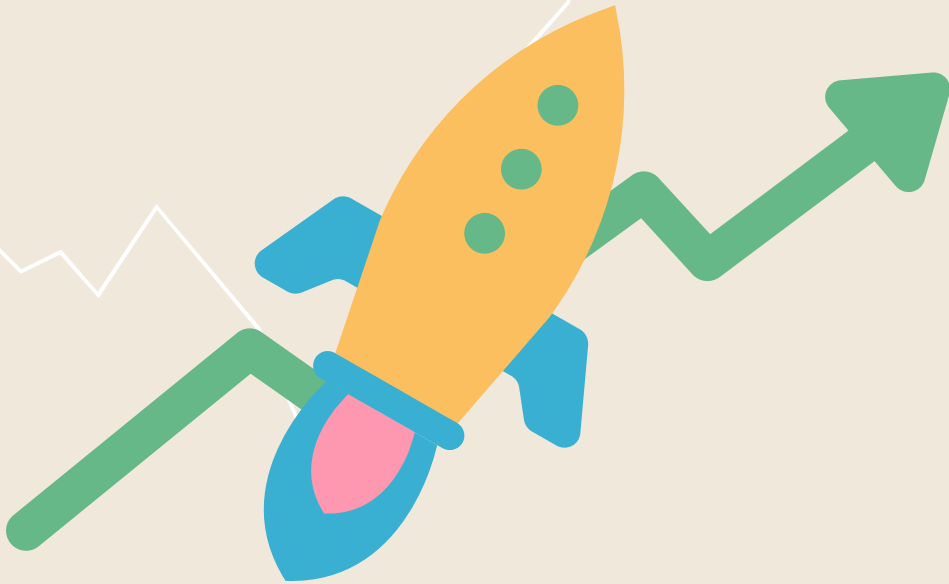
Build a classification model to detect whether the bitcoin transaction is a ransomware attack or not.

How can this model help people?

Detecting this kind of attack can protect organizations and individuals from ransomware attacks by blocking the attacker's bitcoin address associated with suspicious transactions.

03

Workflow & Tools



Workflow & Tools

(1) Data Acquisition

Data imported from **UCI.** have around **3 million** records

(2) Pre-processing

Preparing the data for the model

(3) Building the model

Training the model on the pre-processed data

(4) Evaluating

Scoring the model

Tools

Data

Pandas
Numpy
Sklearn
imbalance-learn

Model

Sklearn
mlxtend

Visualization

Matplotlip
Seaborn

04

DATA & EDA

Data Description

Who collected the data? paper

It was collected by domain expertises.

When it was collected?

The data was collected from 2009 to 2018.

What is the data size?

~3M records
With 41k records labeled as a ransom transaction

Features

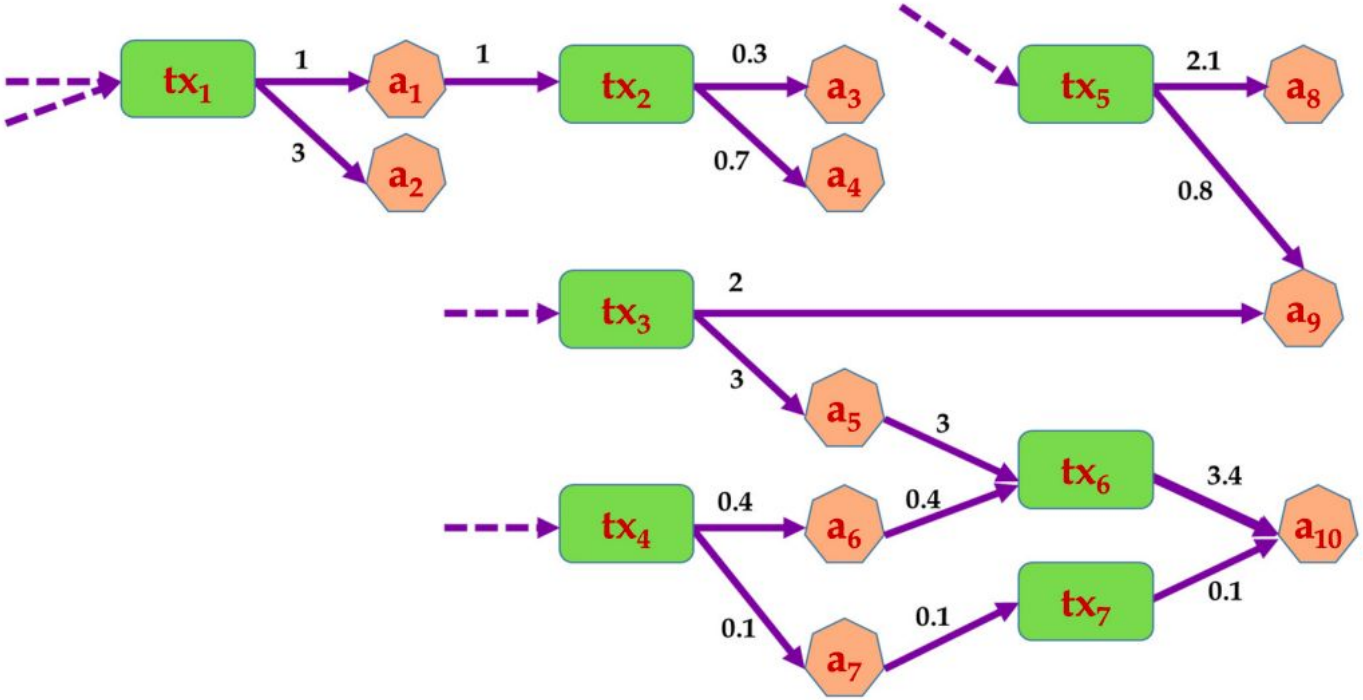


Figure 3. Sample Bitcoin Graph Features: network of 10 addresses and 7 transactions.

Source

Features

1. **Address**

   bitcoin transaction recipient.

2. **Year and day**

   Indicates the exact day and year of the attack.

3. **length**

   How many mixing rounds there was?

4. **Weight**

   Indicates the merge behavior. (amount of tranaction)

5. **count**

   Indicates the merge behavior. (number of transactions)

Features

8. looped

How many there was rounds until merging the coins.
Goes through these steps:

1. split their coins
2. move coins using different wallets
3. merge them in a single address.

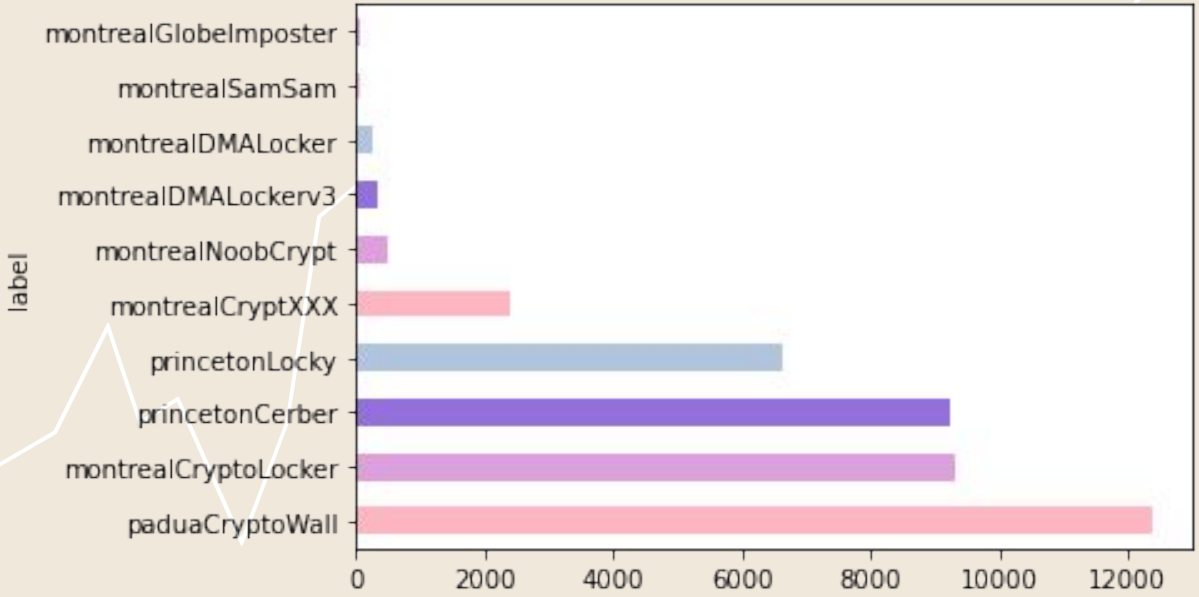
9. neighbors

The number of neighbors a transaction had.

10. income

Income in terms of Satoshi amount.

Ransomware Distribution



05

Model and results

Pre-Processing

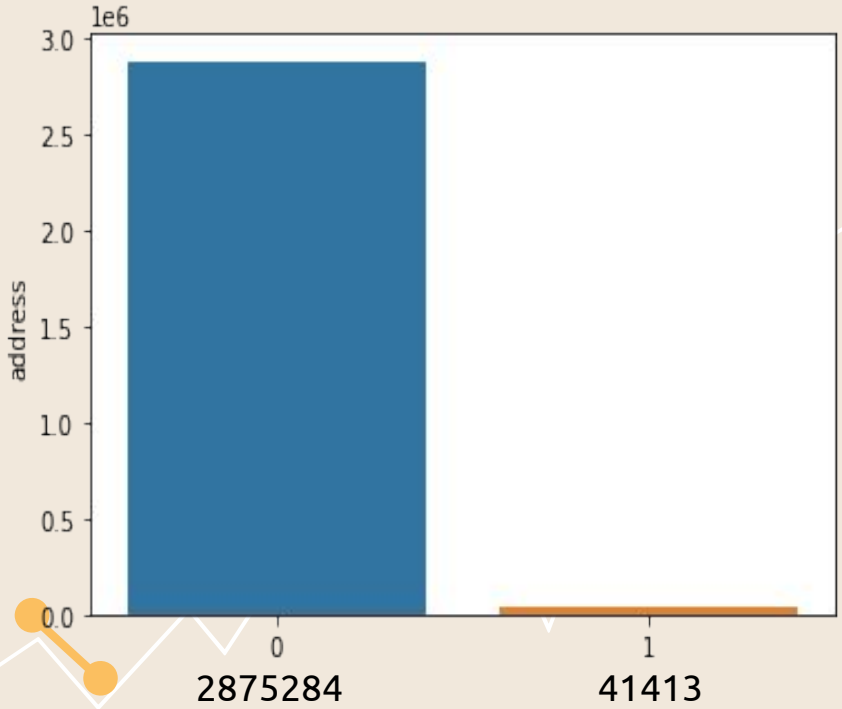
(1) Changing categorical data

(2) Handling unbalanced data

(3) Scaling

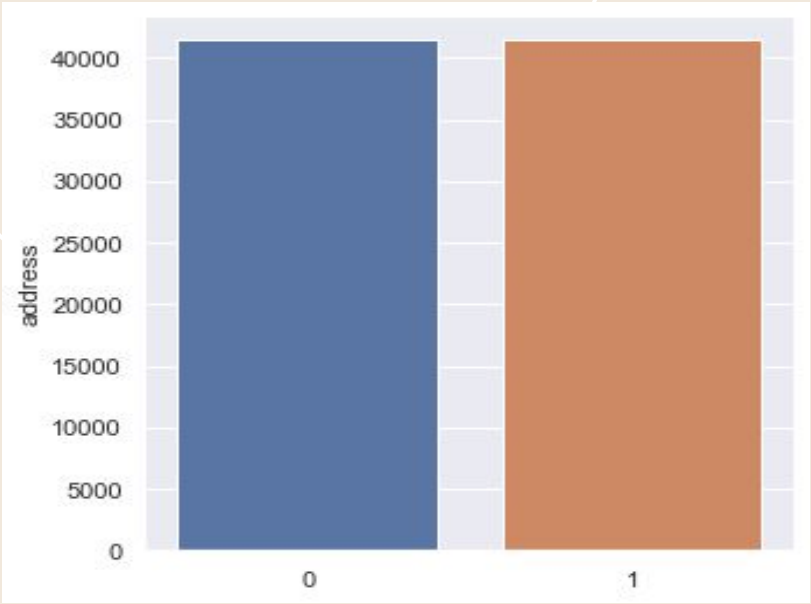
(4) Transformations

Handle Unbalanced Data

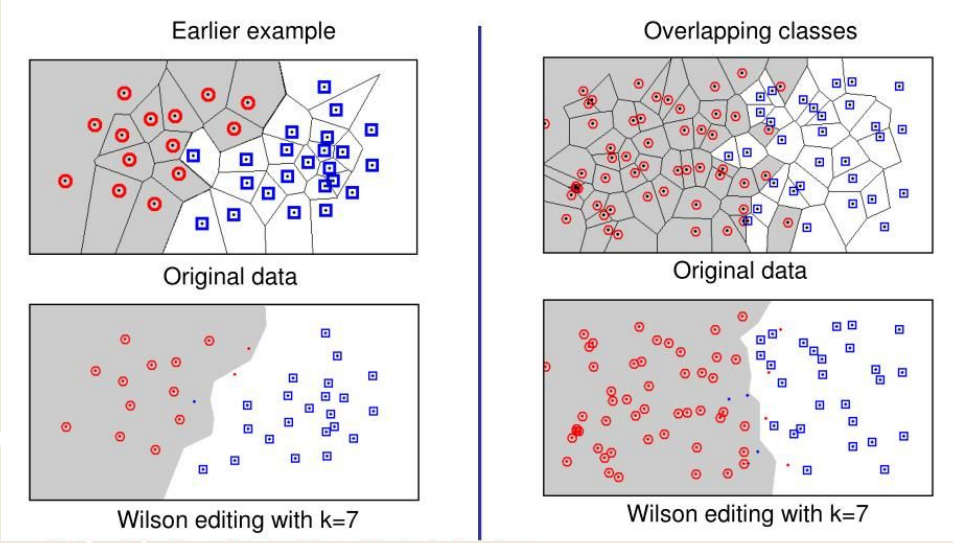


2875284

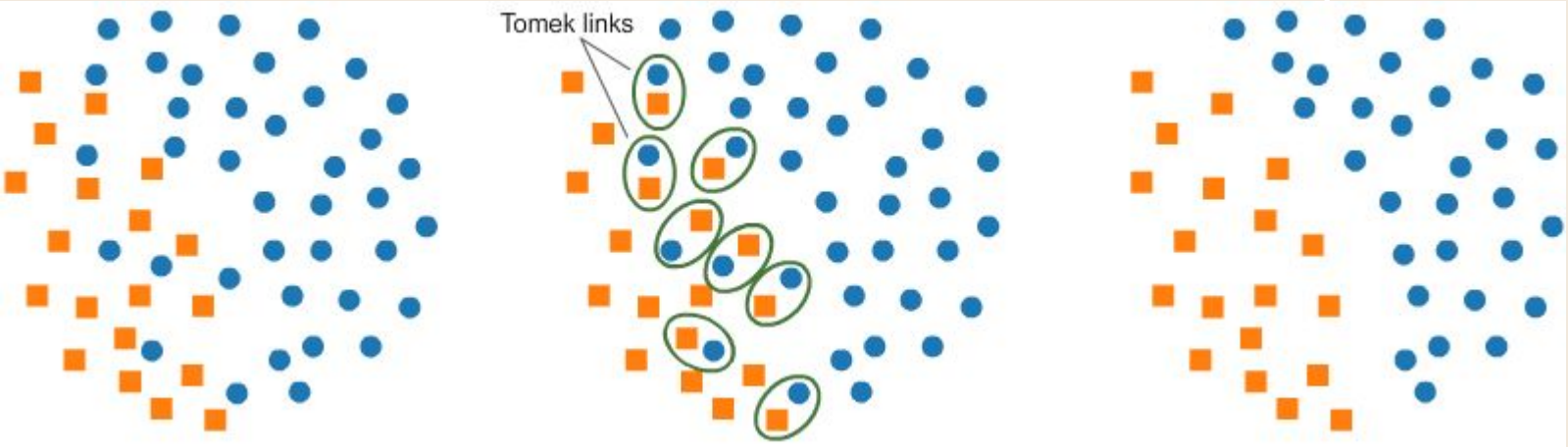
41413



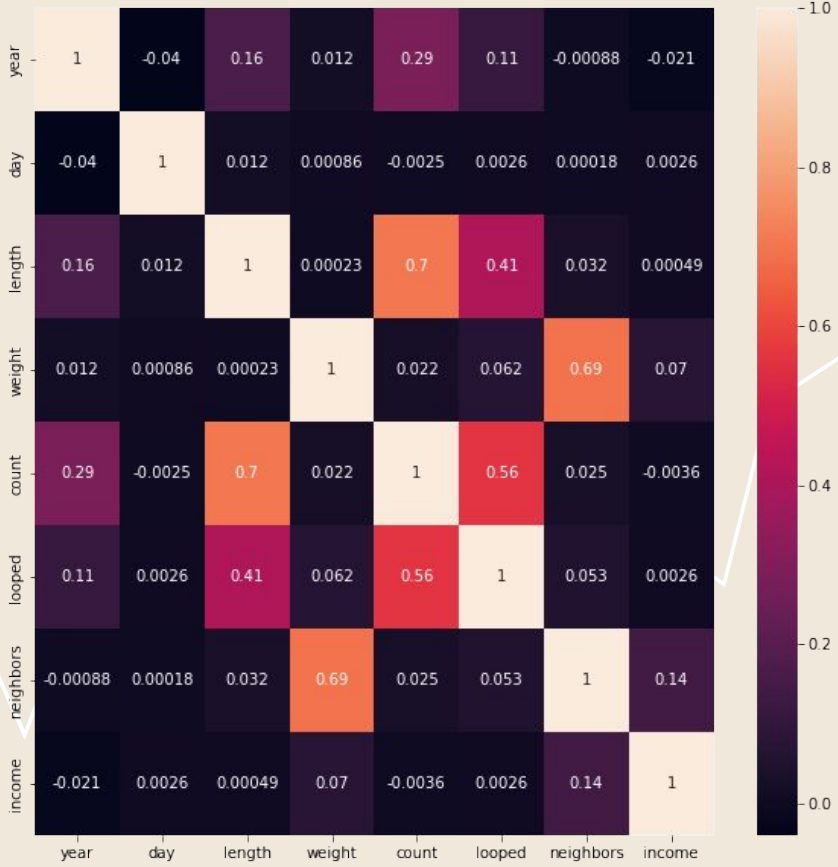
Undersampling with Wilson Editing



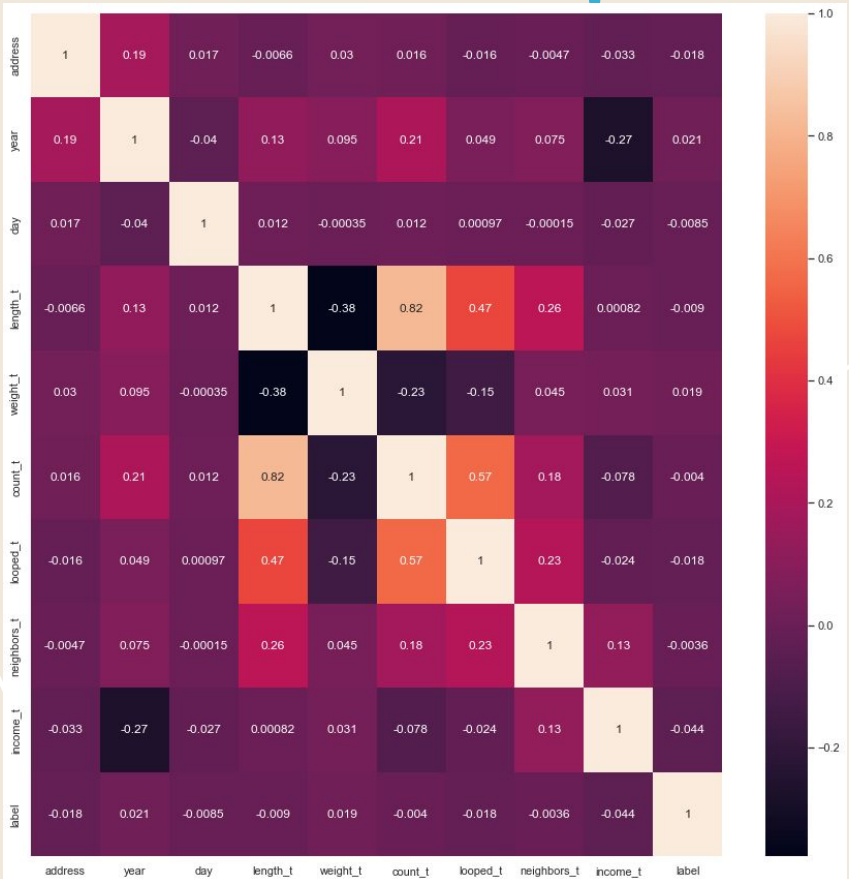
Undersampling with Tomek links



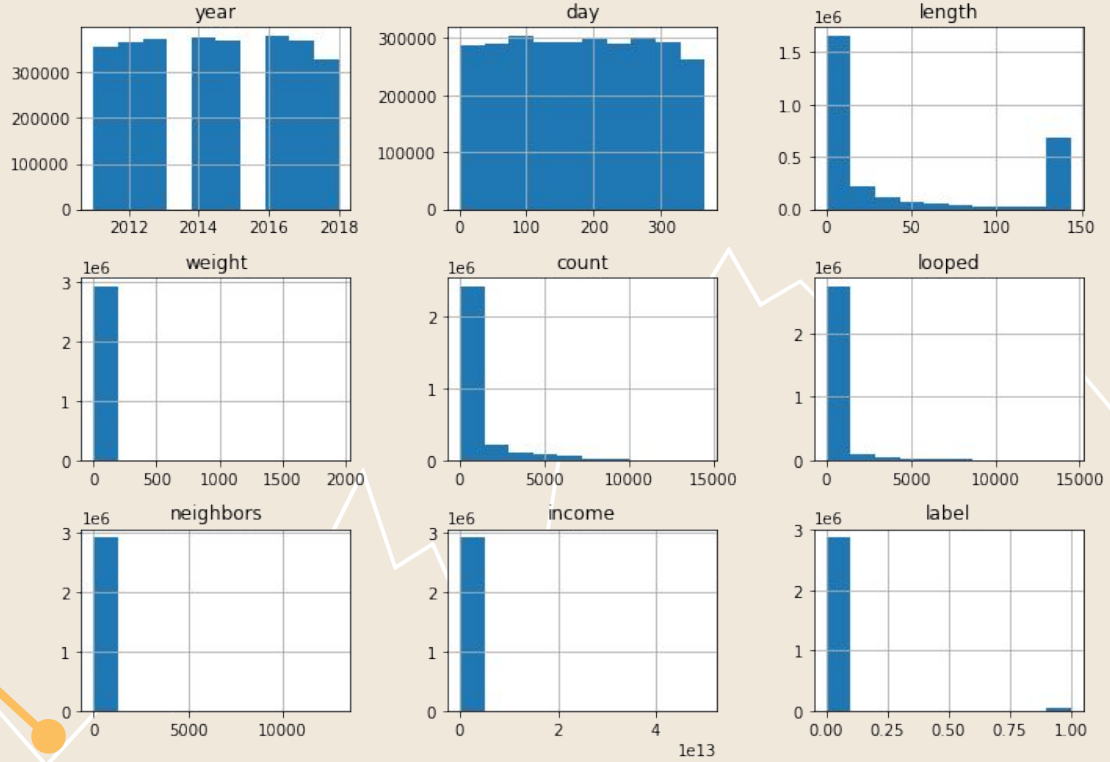
Correlation before data processing



Correlation after data processing

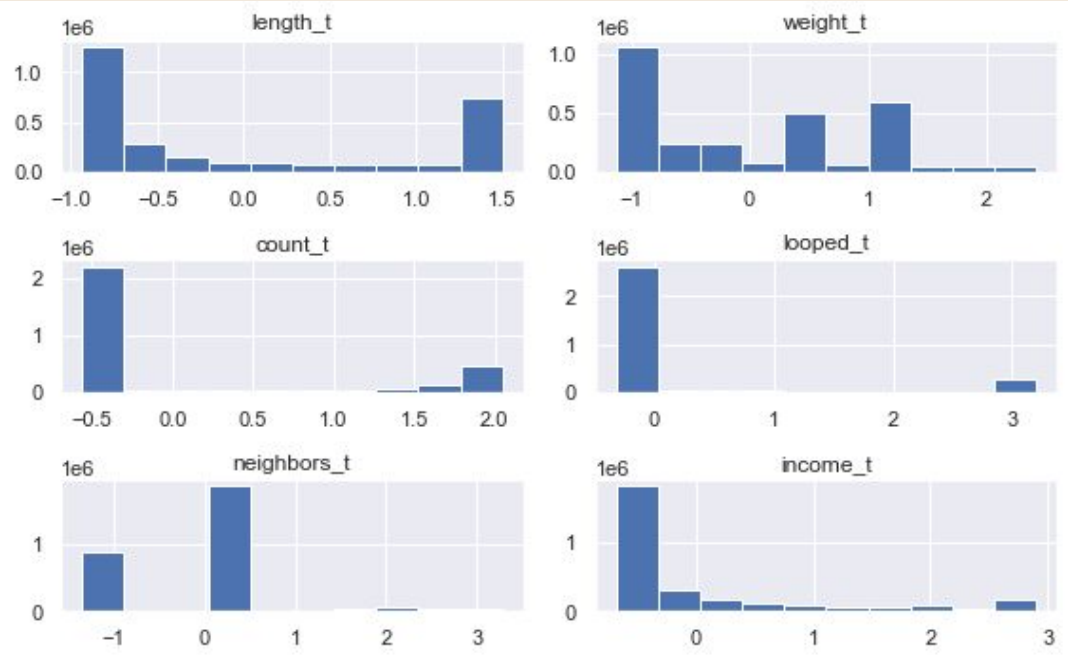


Transforming all Data



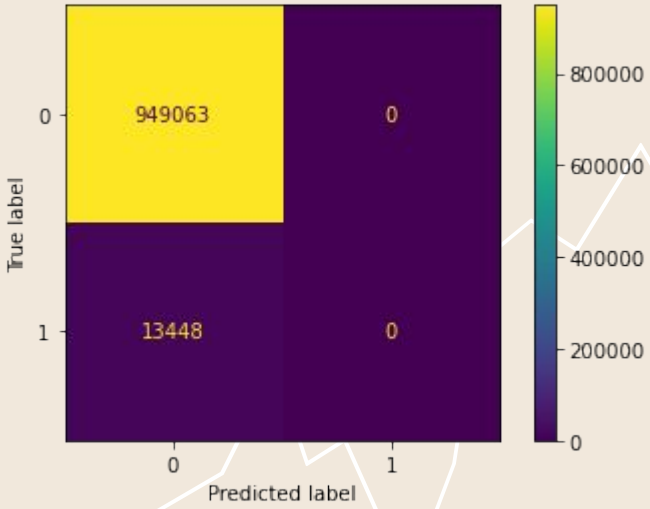
Data skewed to the left

Transforming all Data



Power Transformer (yoe johnson)

Experiment 1



Data:

unbalanced, skewed data

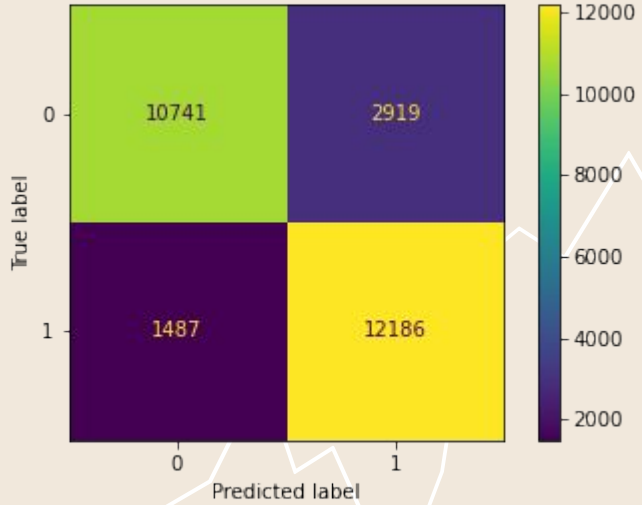
Model:

Logistic regression

Scoring:

Accuracy = 0.99
Precision = 0.97
Recall = 0.99
F1 = 0.98

Experiment 2



Data:

randomly undersampled and balanced, scaled with min-max

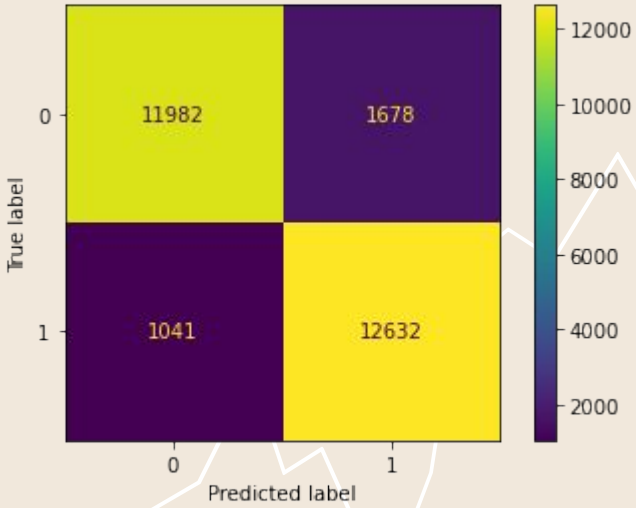
Model:

Gradient boost tree

Scoring:

Accuracy = 0.84
Precision = 0.81
Recall = 0.89
F1 = 0.85

Experiment 3



Data:

Undersampled (with TomekLinks and EditedNearestNeighbours) and unbalanced, scaled with min-max

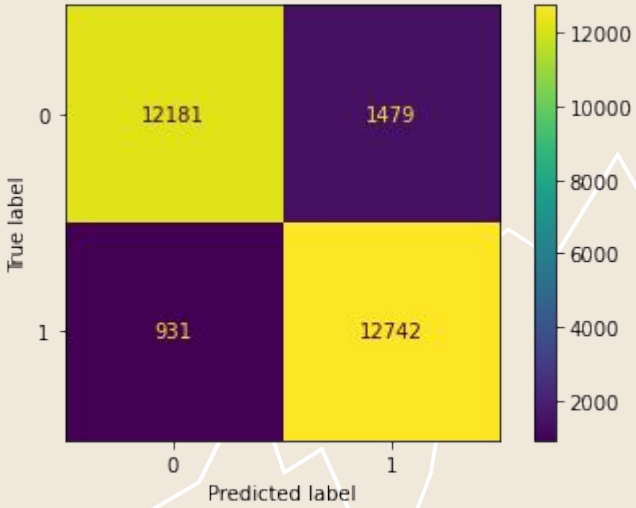
Model:

StackingClassifier with XGB, GBT and GNB

Scoring:

Accuracy = 0.9005
Precision = 0.8827
Recall = 0.9239
F-Score = 0.9028

Final Model



Data:

Undersampled (with TomekLinks and EditedNearestNeighbours) and unbalanced, scaled with min-max

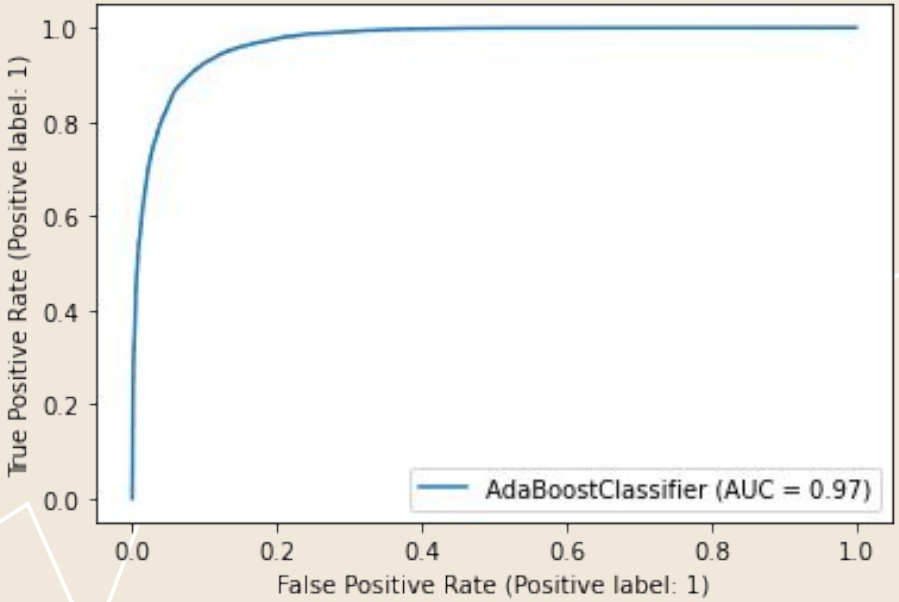
Model:

AdaBoosting with XGB, GBT and GNB

Scoring:

Accuracy = 0.9118
Precision = 0.896
Recall = 0.9319
F-Score = 0.9136

AdaBoost AUC



06

Conclusion

Final Result and Recommendations

Our best model was **AdaBoost** and has achieved the best score (F1-Score = 0.91), it and it can highly separate the two labels (AUC = 0.97)

For future work, we recommend:

- Collecting more updated data with more significant features like the time and targeted company information.

Thanks!

Any questions?