INFORMATION SECURITY ANALYSIS AND AUDIT

CSE3501

NAME: KSHITIJ DHYANI

REGISTRATION NUMBER: 18BIT0131

FACULTY: Prof. SUMAIYA THASEEN

LAB: L31+L32

SLOT: G1

TOPIC: INTRUSION DETECTION USING MACHINE LEARNING

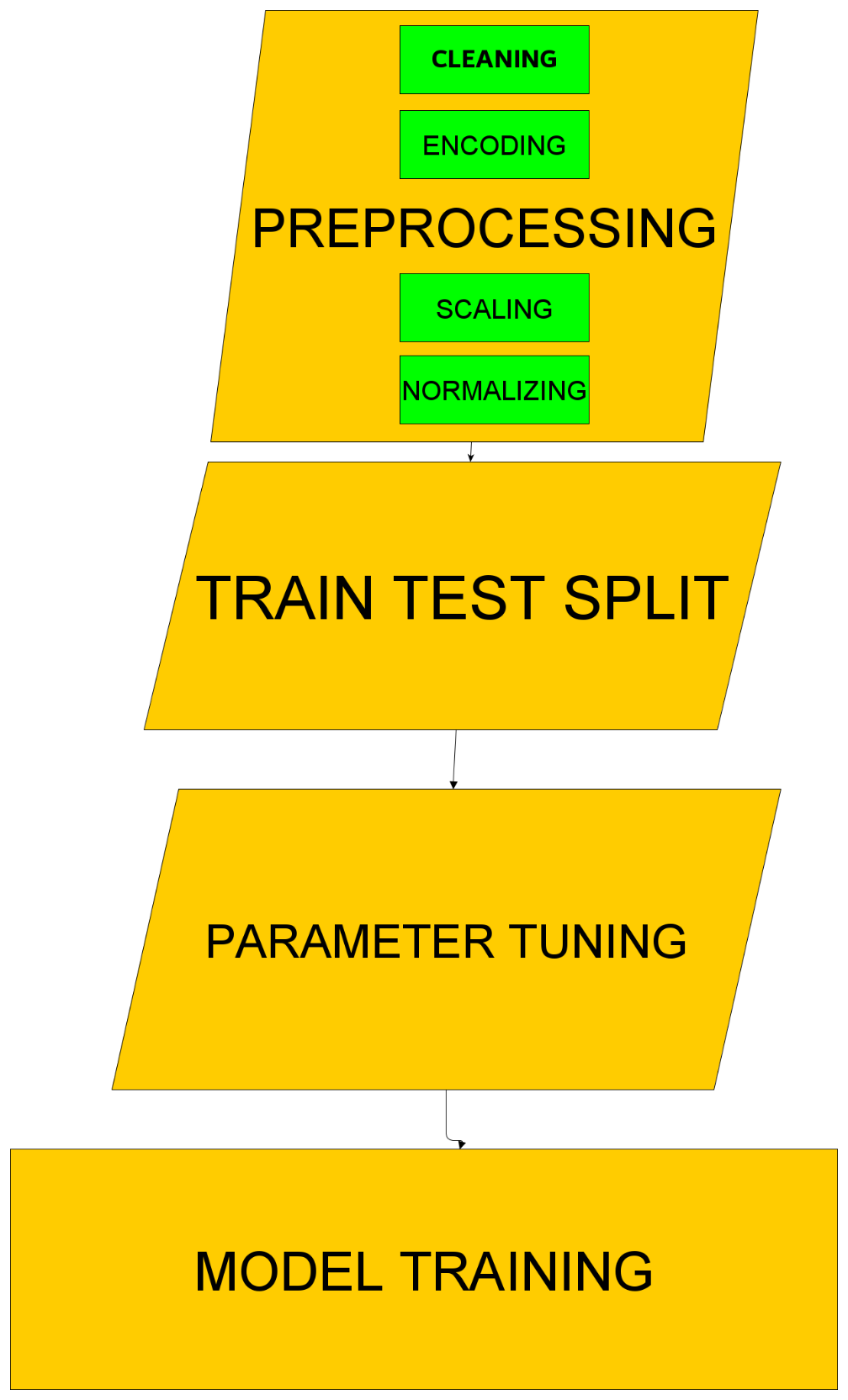
GITHUB REPOSITORY   
  
<https://github.com/wimpywarlord/darknet2020ML>  
  
THE ABOVE LINK HAS ALL THE CODE AND ALSO THE DATASET.

MY DATA SET WAS DARKNET 2020.

LINK TO THE DATA SET

<https://www.unb.ca/cic/datasets/darknet2020.html>

THE FLOW OF THE PROJECT



* **Design:**



* **Description:**
* **About the Dataset:**

In **CICDarknet2020** dataset, a two-layered approach is used to generate benign and darknet traffic at the first layer. The darknet traffic constitutes Audio-Stream, Browsing, Chat, Email, P2P, Transfer, Video-Stream and VOIP which is generated at the second layer. To generate the representative dataset, previously generated datasets, namely, [ISCXTor2016](https://www.unb.ca/cic/datasets/tor.html) and [ISCXVPN2016](https://www.unb.ca/cic/datasets/vpn.html), have been amalgamated and respective VPN and Tor traffic are combined in corresponding Darknet categories.

**No. of rows:** 1.4 Lacs approx..

**No. of columns:** 85

* First we will be importing all the python libraries and sklearn modules.
* **Importing the dataset:**

The dataset DARKNET 2020 can be downloaded from [***https://www.unb.ca/cic/datasets/darknet2020.html***](https://www.unb.ca/cic/datasets/darknet2020.html)

* **Data Cleaning Phase:**

In this phase:

* We will be handling the missing values and replacing them with the mean value of the respective column.
* Also the null and NaN values will also be replaced by the mean of the column.
* The columns also contain infinite values. To deal with such values, first we will find the row index of such values, then convert them into NaN and delete the entire row.
* **Data Pre-Processing Phase:**

**For binary classification:**

* The dataset is minimized by only including rows that have either benign or malicious classes i.e. Tor and Non-tor classes.
* This dataset is then balanced, that is, the number of rows of each class is made approximately equal by either over sampling or under sampling.
* Target variable is then separated from the dataframe and a new dataframe is created for the target variable. This target variable is then encode using Label Encoder to convert into integer from object.
* The dataset is then normalized using min-max scalar.
* The number of columns is large and we will not be needing all of them for classification. So we will be performing feature extraction using ranking technique to extract only the relevant columns and proceed further with only those columns.
* There are few columns like timestamp and flow-id that can be removed directly. Also columns like Src IP and Dst IP can be modified to be used further instead of deletion.

**For multi-class classification:**

The complete dataset is used in this case. Rest of the procedure remains same as the binary classification.

* **Model Training and Testing Phase:**

In this phase:

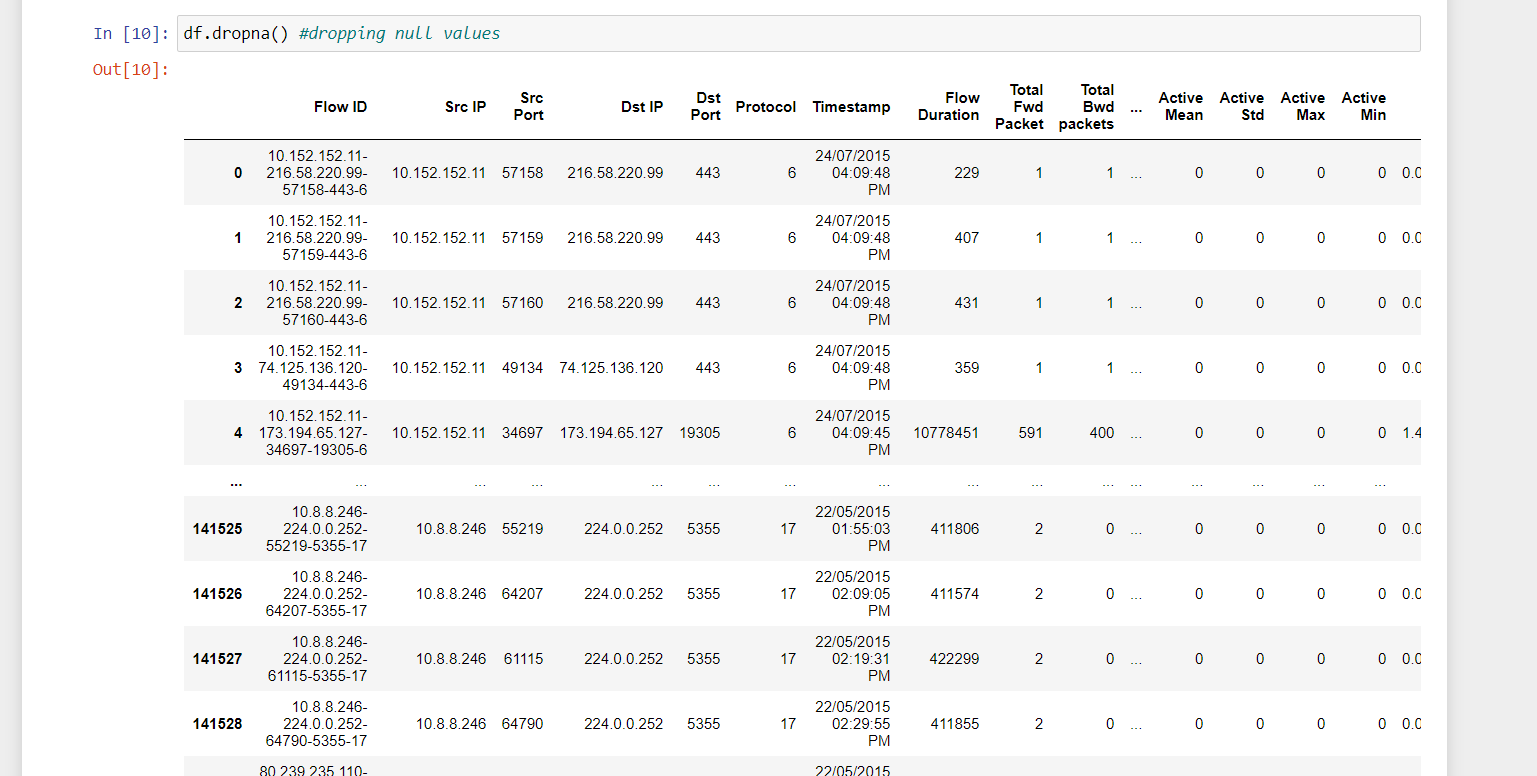
* First the dataset is split into training and testing dataset.
* Then we will be defining two classifiers: **Gradient Boosting Machine** and **AdaBoost**.
* Initially we will be passing random parameters and perform a baseline prediction. Classification Report is generated and accuracy is obtained. Confusion matrix is also obtained.
* Then we will be performing 10-Fold Cross Validation. After this we will be performing Hyperparameter tuning on both the models and tune three parameters each.

**The three best parameters will be obtained.**

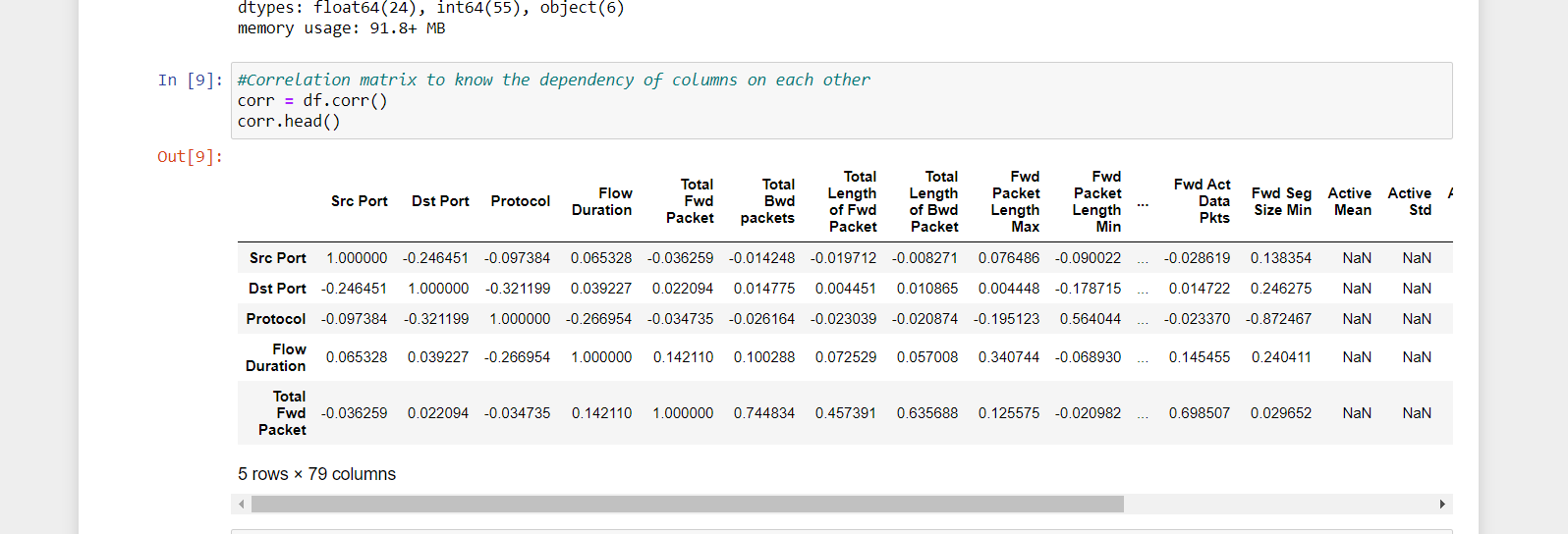
* We will again perform prediction on the test dataset using a new model and passing the three best parameters. Classification report is generated again. Accuracy is obtained.
* The evaluation metrics of this new tuned model is then compared with the baseline model.
* Graphs like roc\_auc curves for binary classification and heatmaps are plotted wherever required.
* Also comparison is done between both the classifiers to get the more efficient one.

DISCRIPTION

The project was to perform intrusion detection using machine learning, I was allotted the adaboost decision tree algorithm.  
  
I was given the darknet2020 dataset, which basically a very new dataset. It categorizes the data into TOR, NON-TOR , VPN and NON VPN users. TOR is browser which helps us acess the dark net. Dark Net is a untraceable internet, where all the malicious and illegal activities take place.

I started off by cleaning the data since some of the values were outliers and some were empty or corrupt. This was simply done by removing the nan values and empty rows and columns.   
  


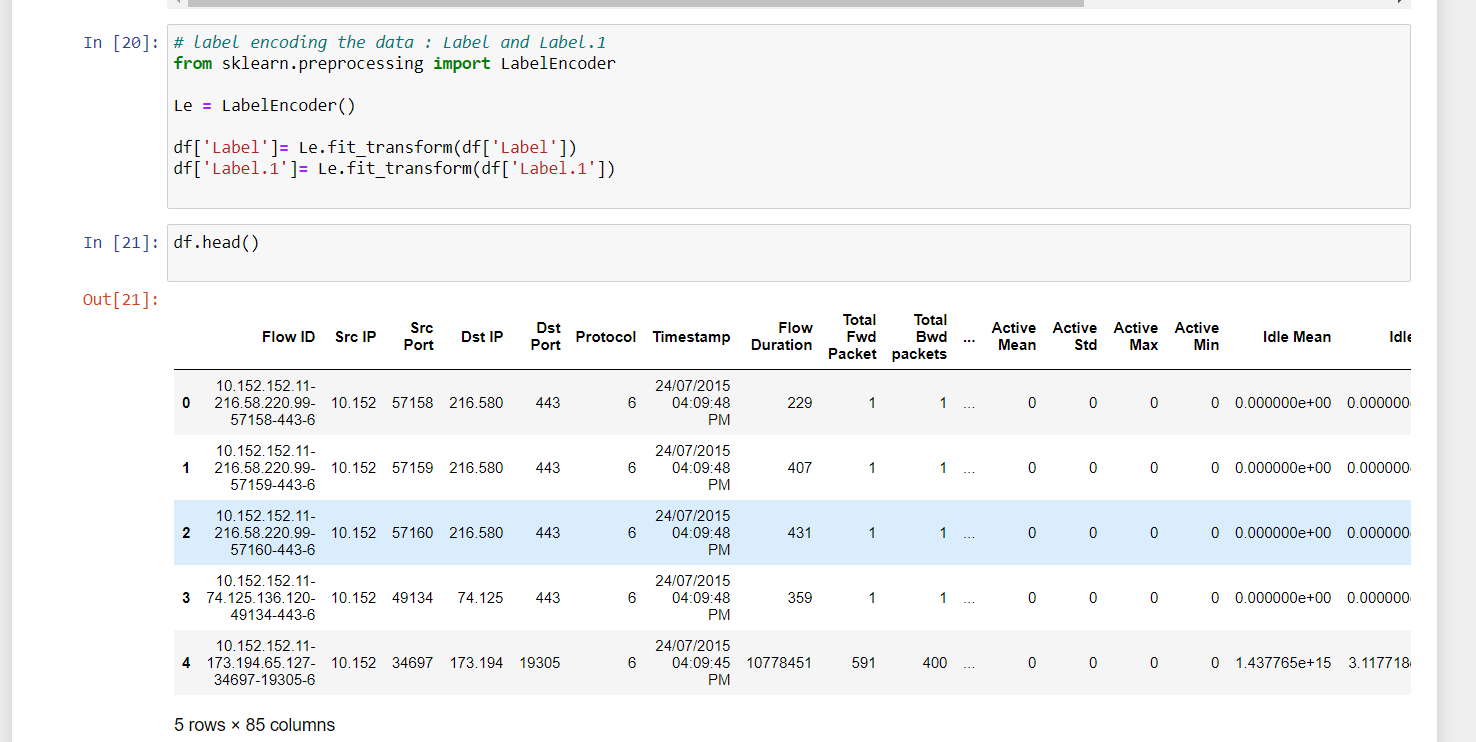
Then we started off by making a correlation matrix and analysing it. This helped us determine which fields were important for the output.



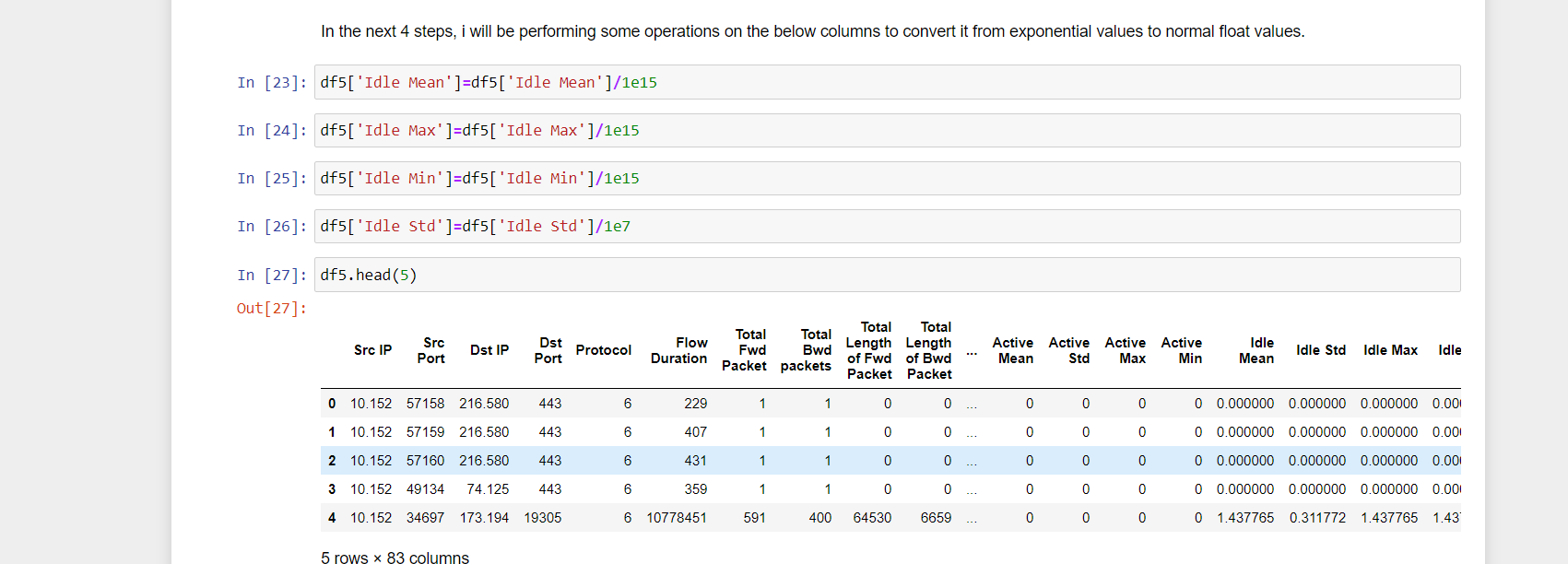
Next we went onto splice the IP address fields, since it was fitting in any of the datatypes.



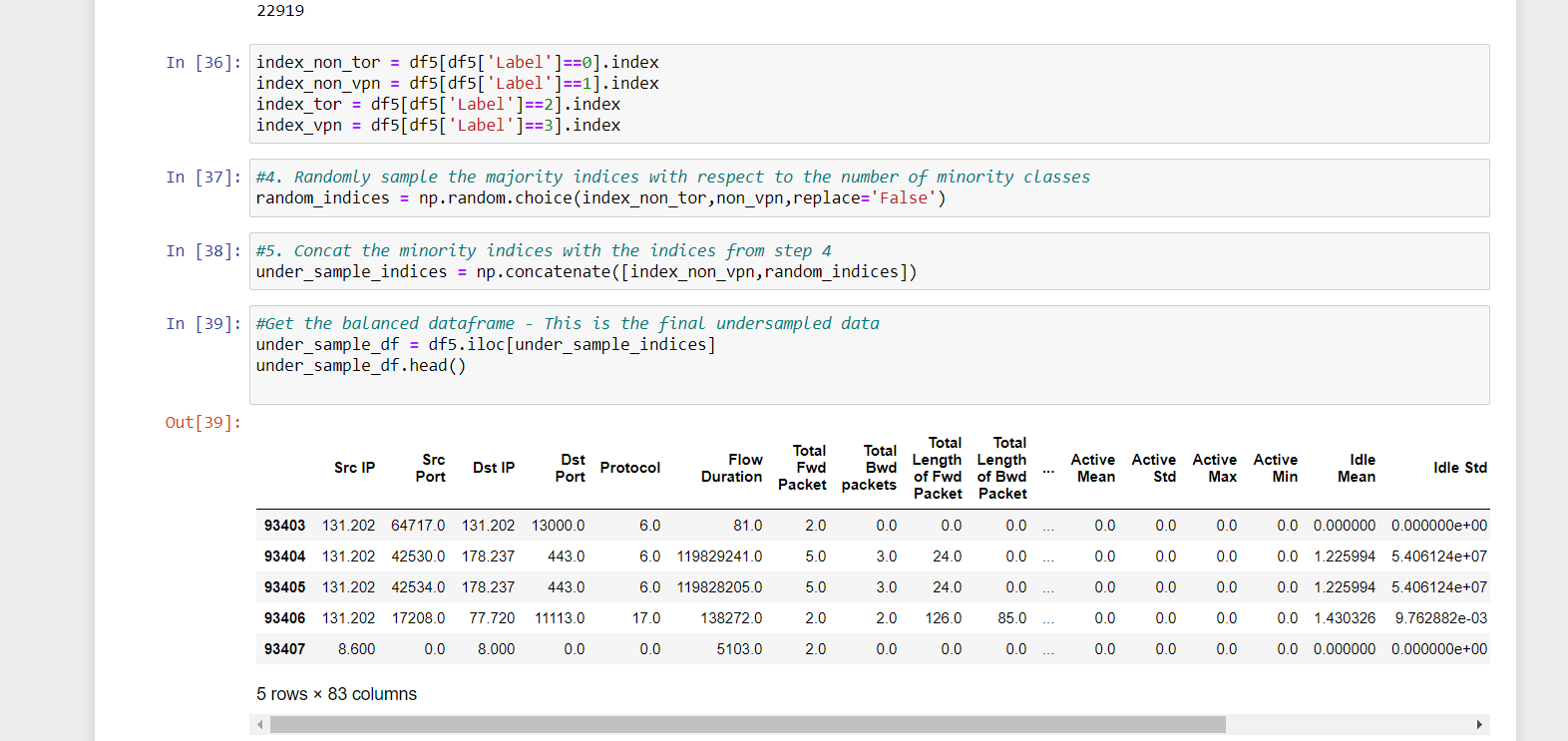
Then we encoded the Label field using, ONE HAUT ENCODING. This converted the label of non-tor,tor,vpn,non-vpn to 1s and 0s.



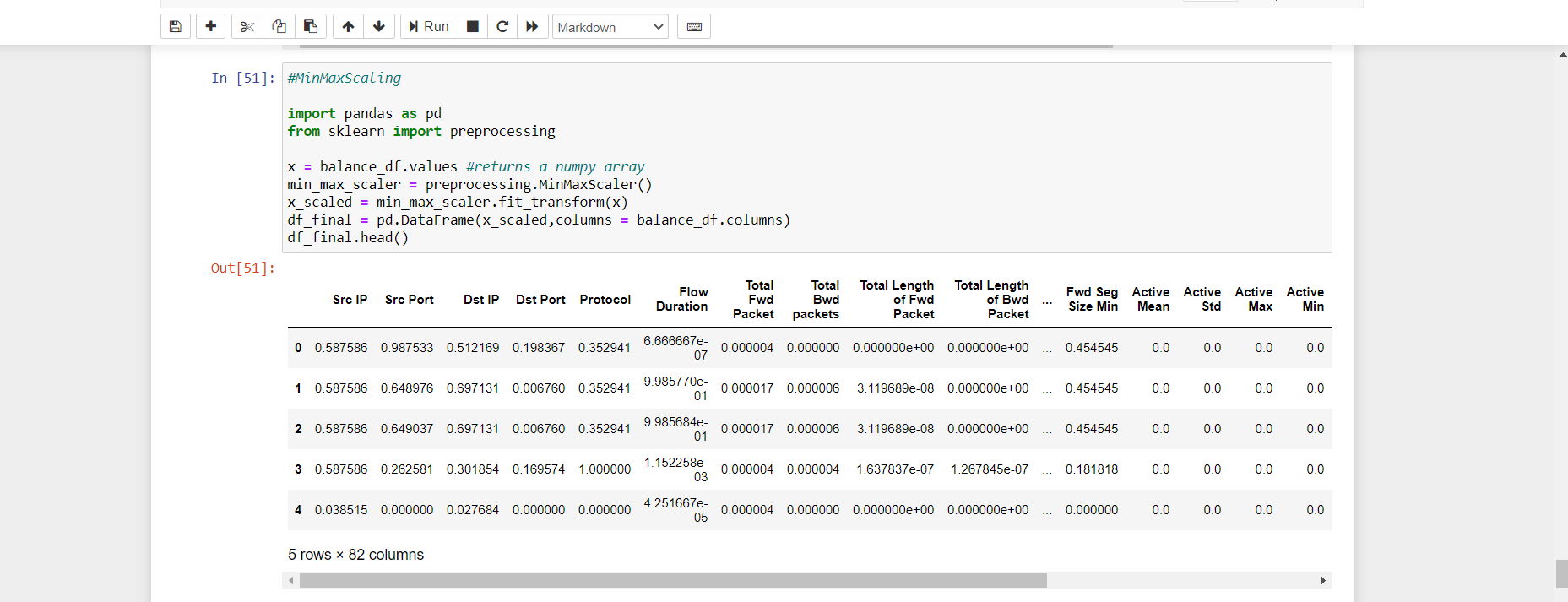
Since the dataset had values of infinite scale which is scale of 10^15. We had to divide these columns by 10^15.



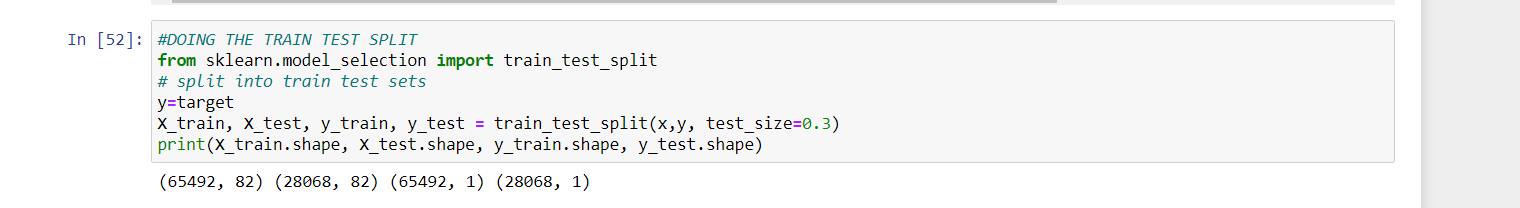
To make the machine learning algorithm not biased, we samples the data using under sampling and oversampling technique’s.



Then I went on to scale the data using minmax Scaling technique.



Then I did the train test split



Then I tuned 3 parameters.



Finally ran the machine learning decision tree adaboost model

