Detection of Ransomware Attacks Using Processor and Disk Usage Data

Ransom attackers will evade antivirus and then enter into victim system to execute malicious script which will encrypt and make entire system data unusable and to decrypt files back they will ask ransom from the victim and world has loss billions of dollars in ransom since the birth of internet network. Many existing techniques are available such as SYSTEM PROCESS MONITORING to identify and prevention of malicious script execution but this monitoring will impact system performance.

Another technique is to monitor files which are getting deleted or created to know malicious file but this technique also impact system performance and detection accuracy also not good enough.

To overcome from above issue author of this paper employing VMWARE on host system which will read Hardware Performance Counters (HPC) and IO EVENTS data and then applying this data on machine learning models to predict whether executing script is normal (benign) or Ransomware. Extracting HPC and IOEVENTS features using VMware will not affect system performance and machine learning models also able to predict Ransomware with more than 90% accuracy.

In propose paper author has experimented with various machine learning algorithms such as SVM, KNN, Decision Tree, Random Forest and XGBOOST and then employed two deep learning models called DNN and LSTM. In all algorithms Random Forest, XGBOOST is giving accuracy.

To train all algorithms author has publish HPC and IOEVENTS from different programs such as 7ZIP, AES and many more and this dataset can be downloaded from below link

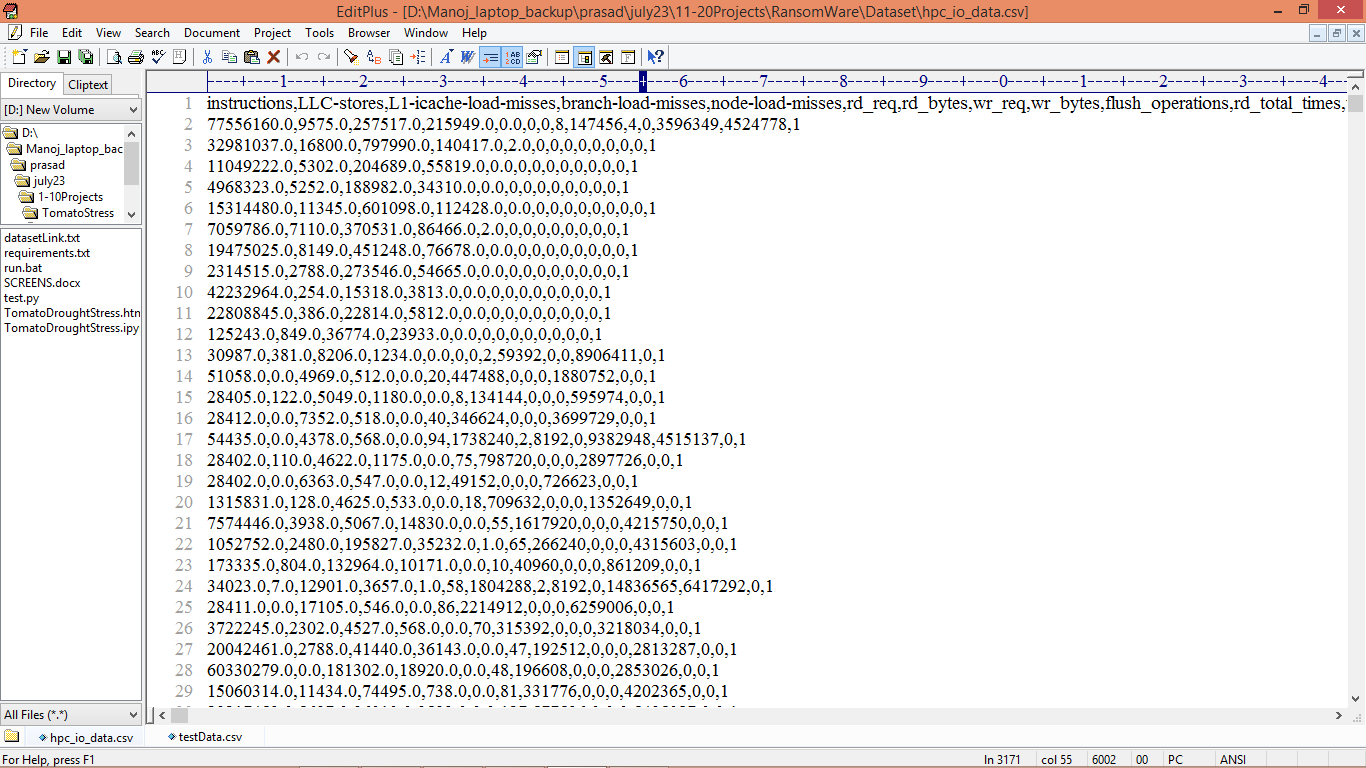
HPC dataset

https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/MA5UPP

IOEVENTS dataset

<https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/GHJFUT>

We have integrated both dataset and below screen showing combined dataset details



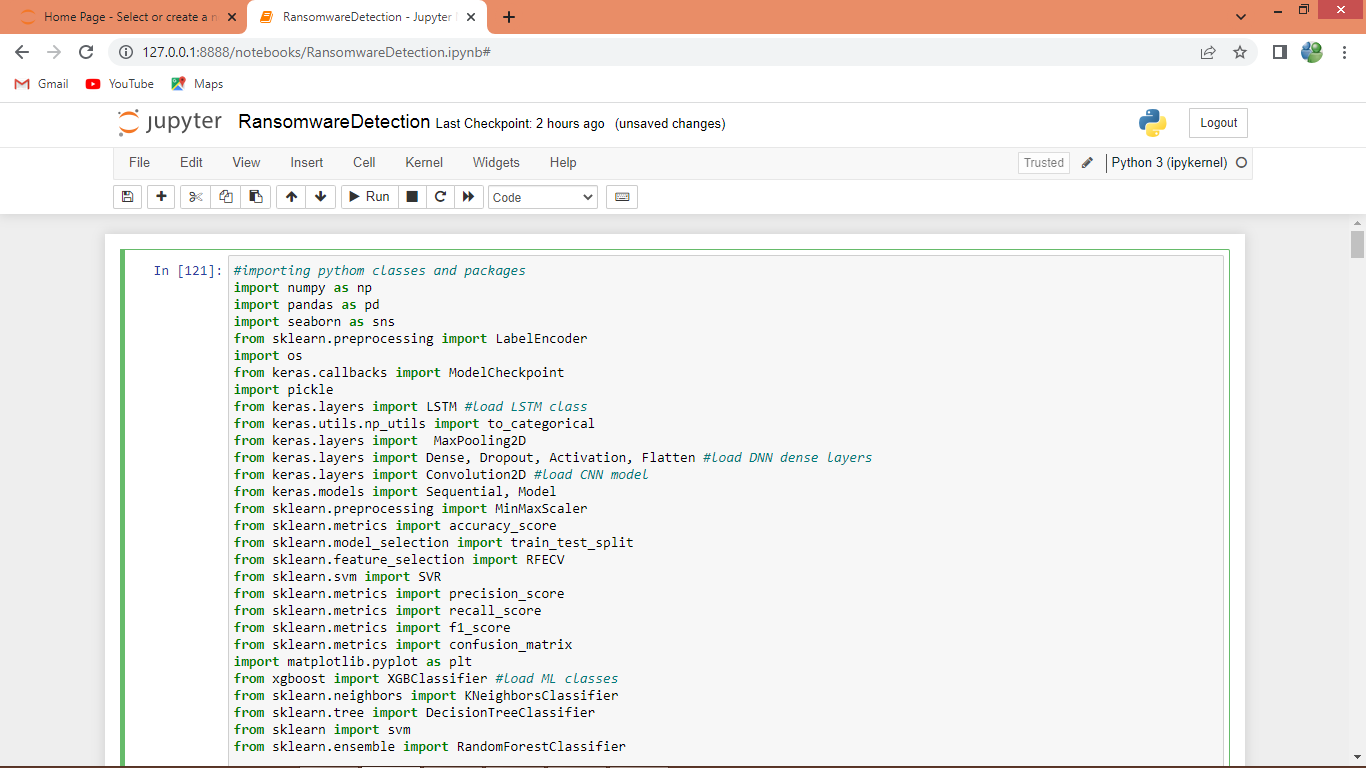
In above dataset screen first row represents dataset column names and remaining rows contains HPC and IOEVENTS features and in last column we have class label as 0 or 1 where 0 represents Benign program and 1 represents Ransomware program. So by using above dataset we are training and testing all algorithms performance.

Extension Concept

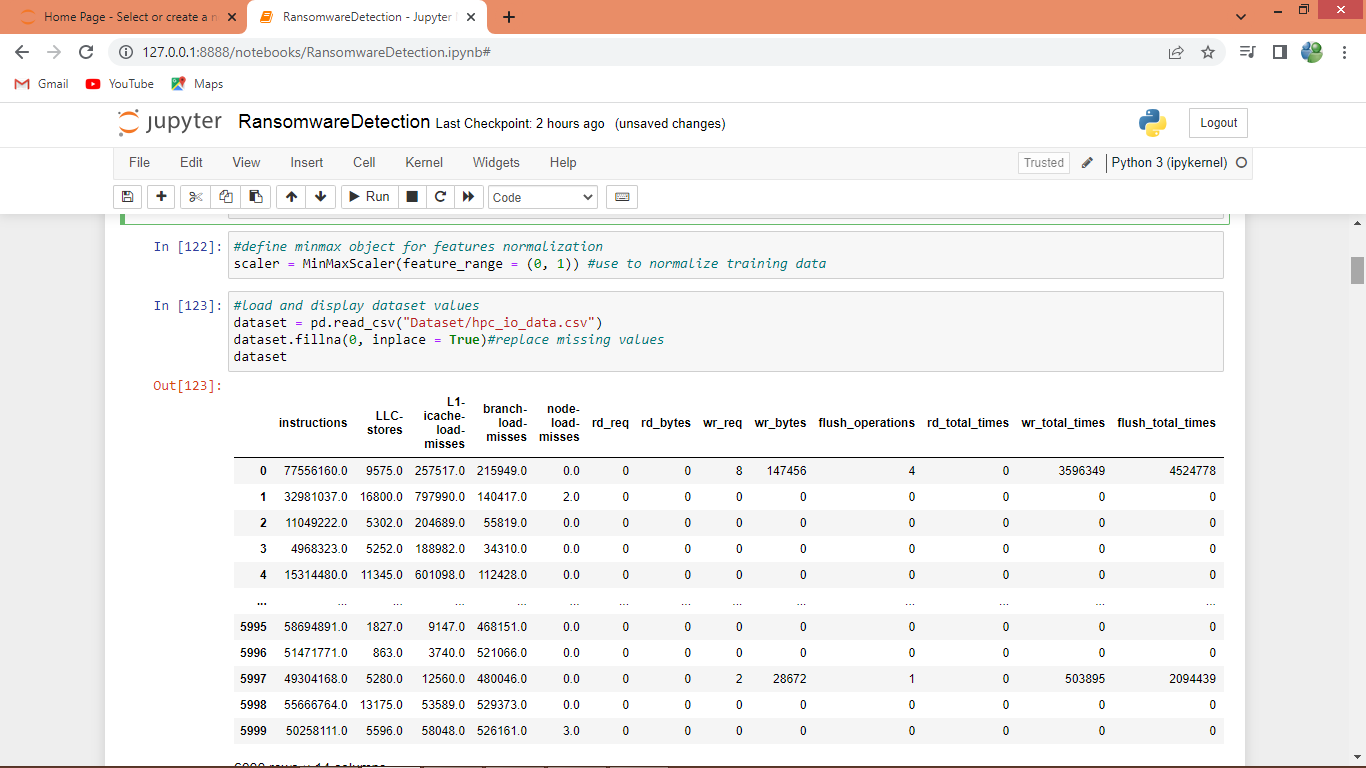
In propose work author has used all traditional algorithms but not used any advance features optimization algorithms so as extension we have experimented with CNN2D (2D convolution neural networks) which will optimized dataset features in multiple layers which help in getting relevant data for training and can improve detection accuracy.

SCREEN SHOTS

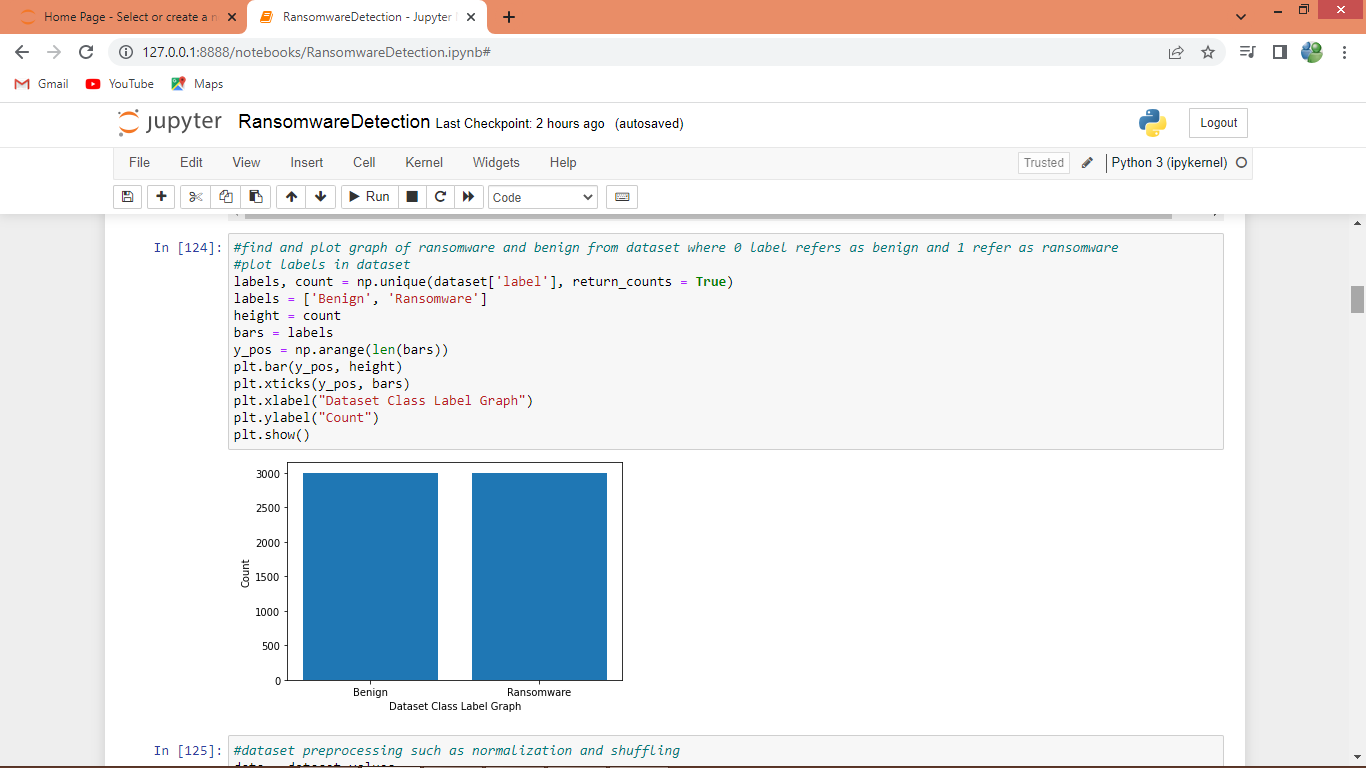
We have coded this project using JUPYTER notebook and below are the code and output screens with blue colour comments



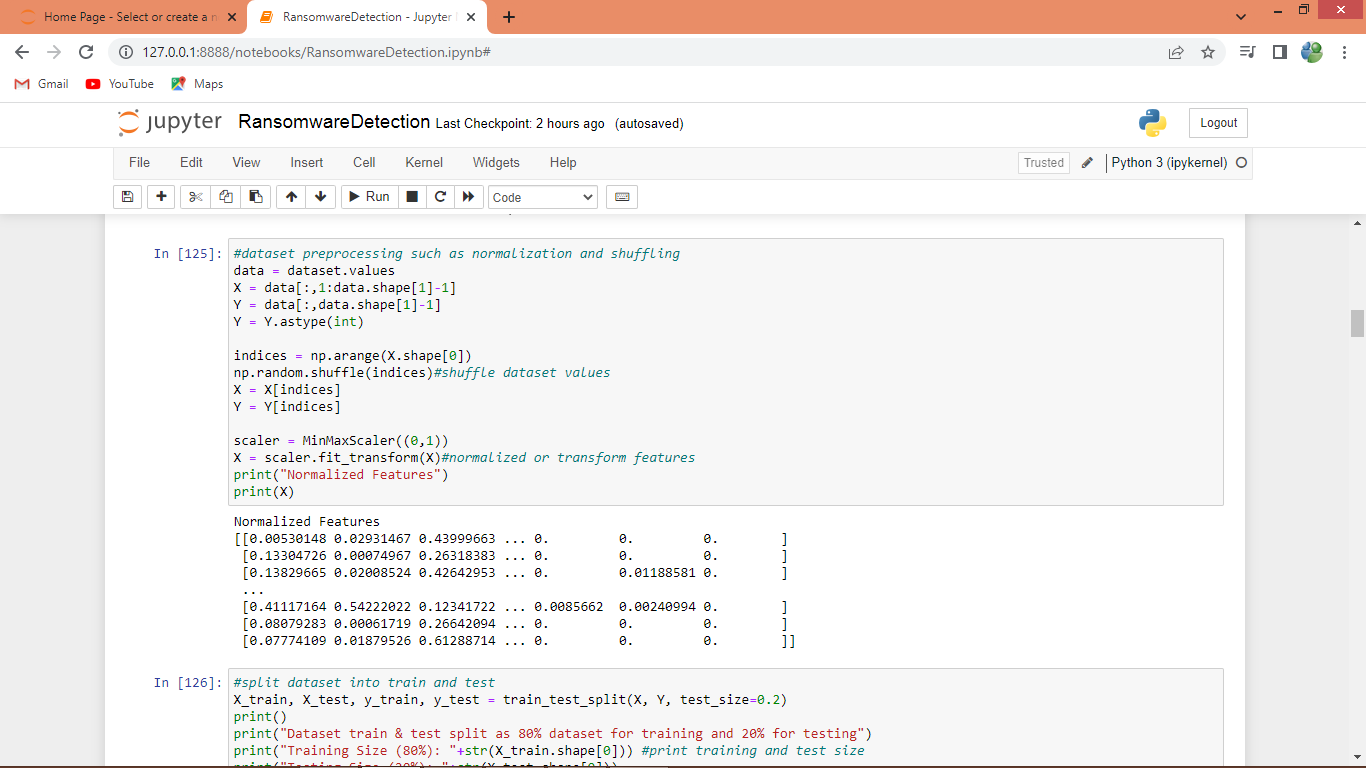
In above screen loading all packages and classes



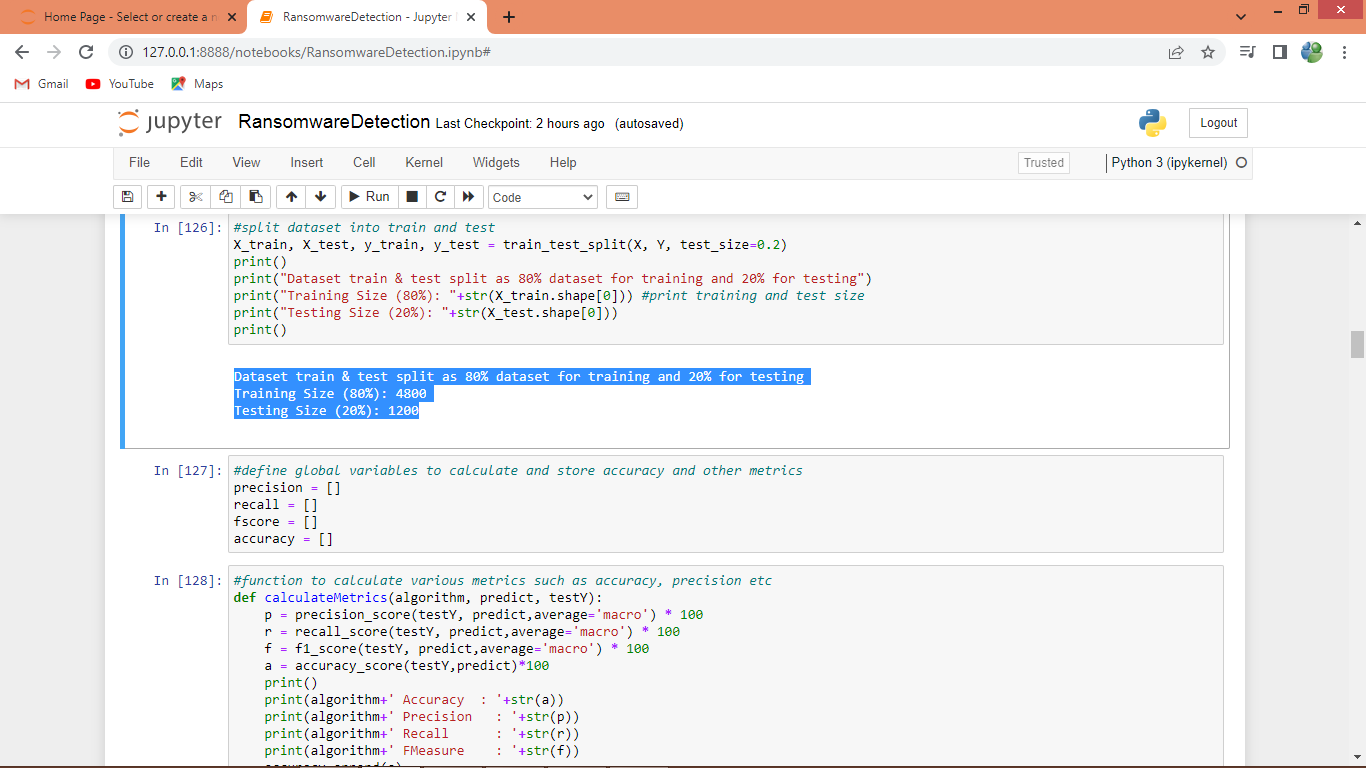
In above screen defining class to normalize features and then loading and displaying dataset values



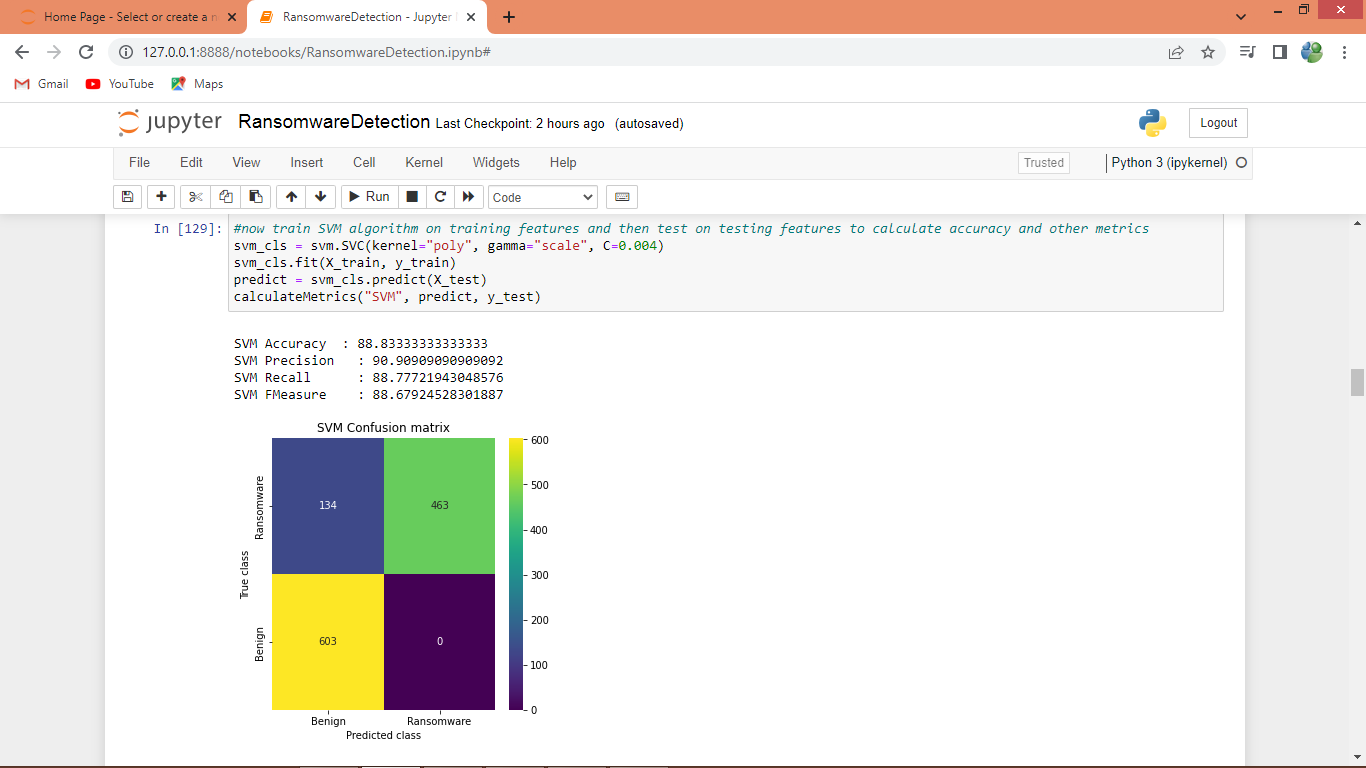
In above screen we are plotting graph of benign and Ransomware dataset size where x-axis represents class label and y-axis represents count



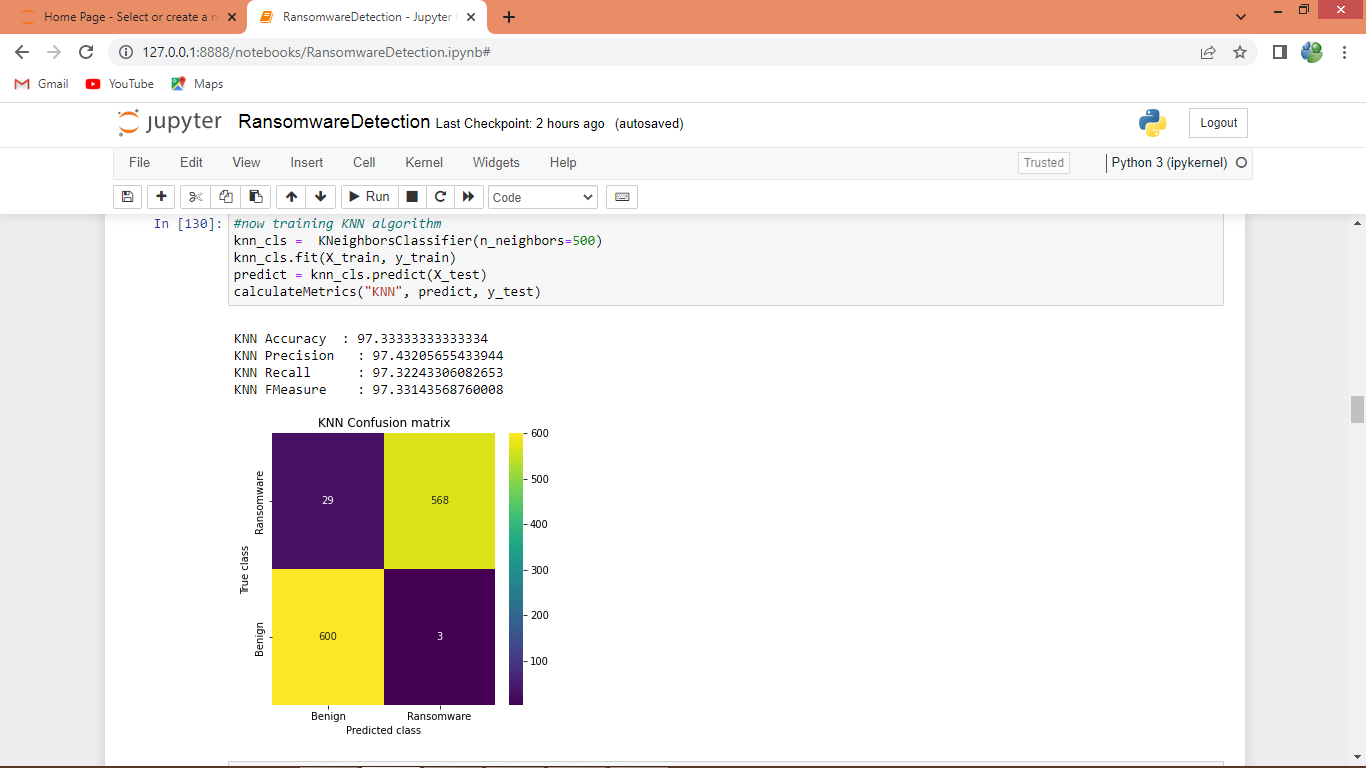
In above screen performing dataset pre-processing such as normalizing and shuffling and then displaying processed values



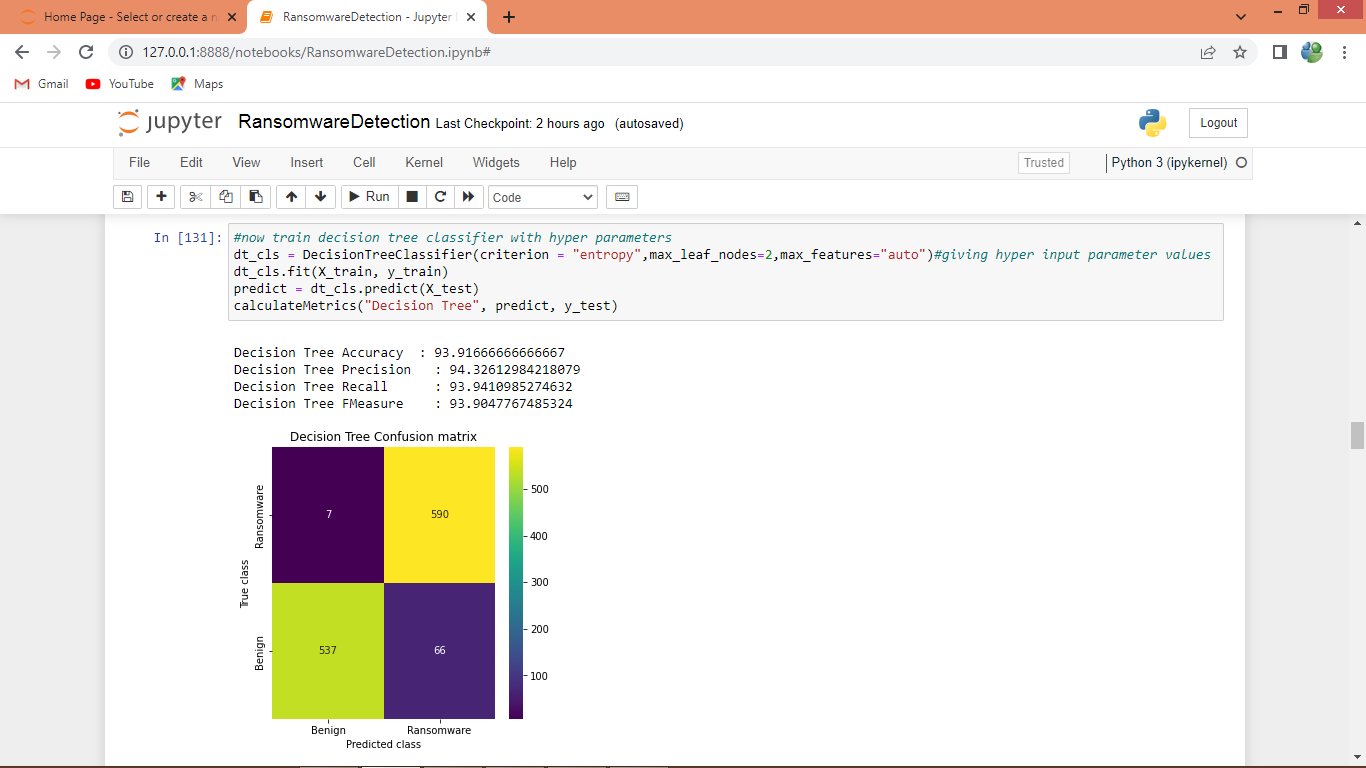
In above screen splitting dataset into train and test where application using 80% dataset for training and 20% for testing and then defined function to calculate accuracy and other metrics



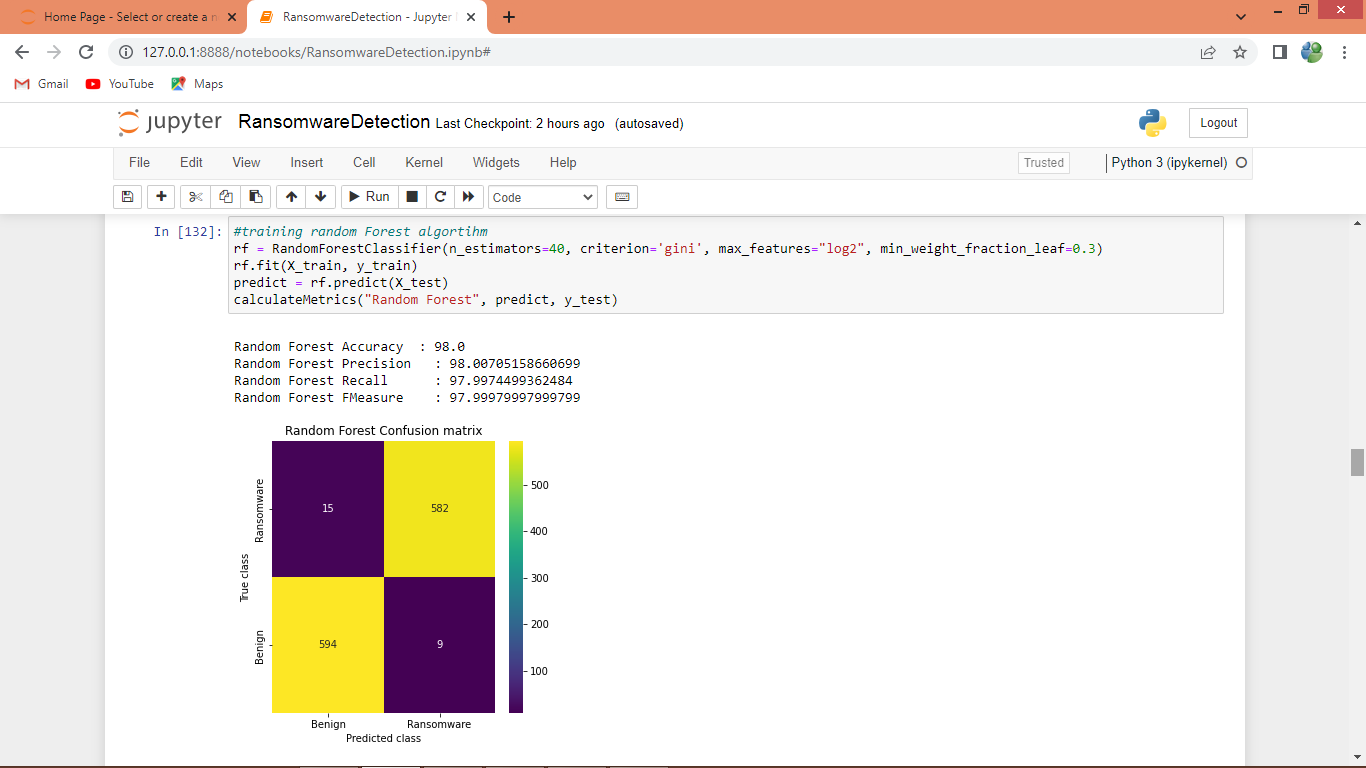
In above screen training SVM algorithm on 80% training data and then performing prediction on 20% test data and then calculating accuracy and other metrics. In above screen SVM got 88% accuracy and can see other metrics also and in confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where yellow and green boxes contains correct prediction count and blue boxes represents incorrect prediction count.



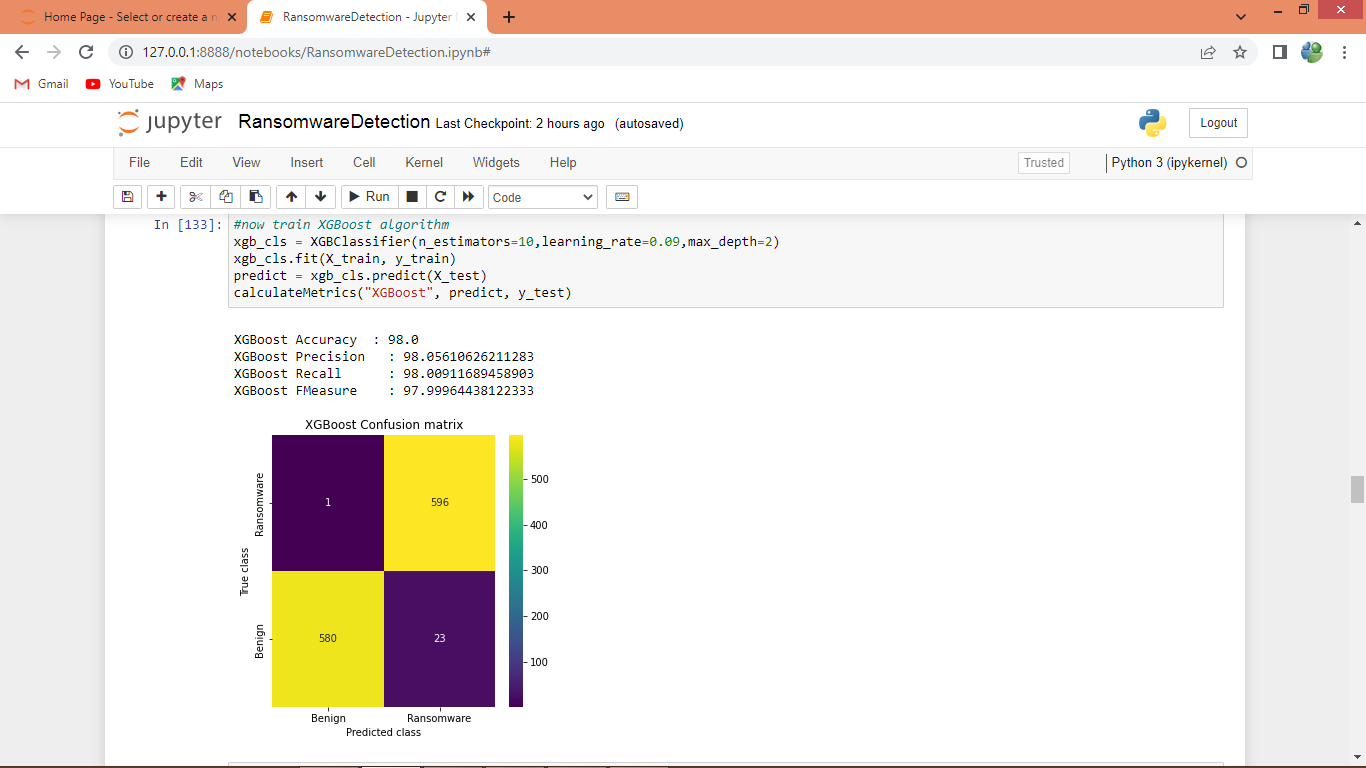
In above screen KNN got 97% accuracy



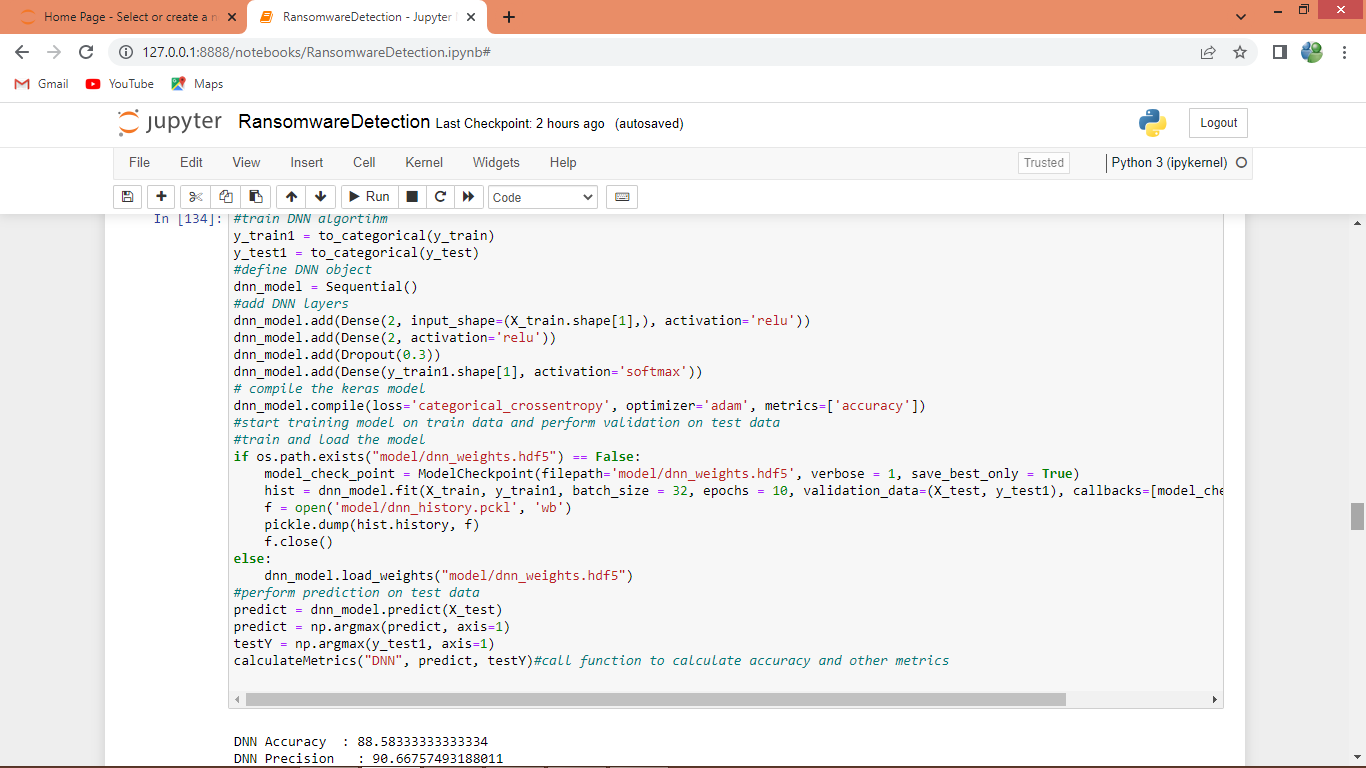
In above screen decision tree got 93% accuracy



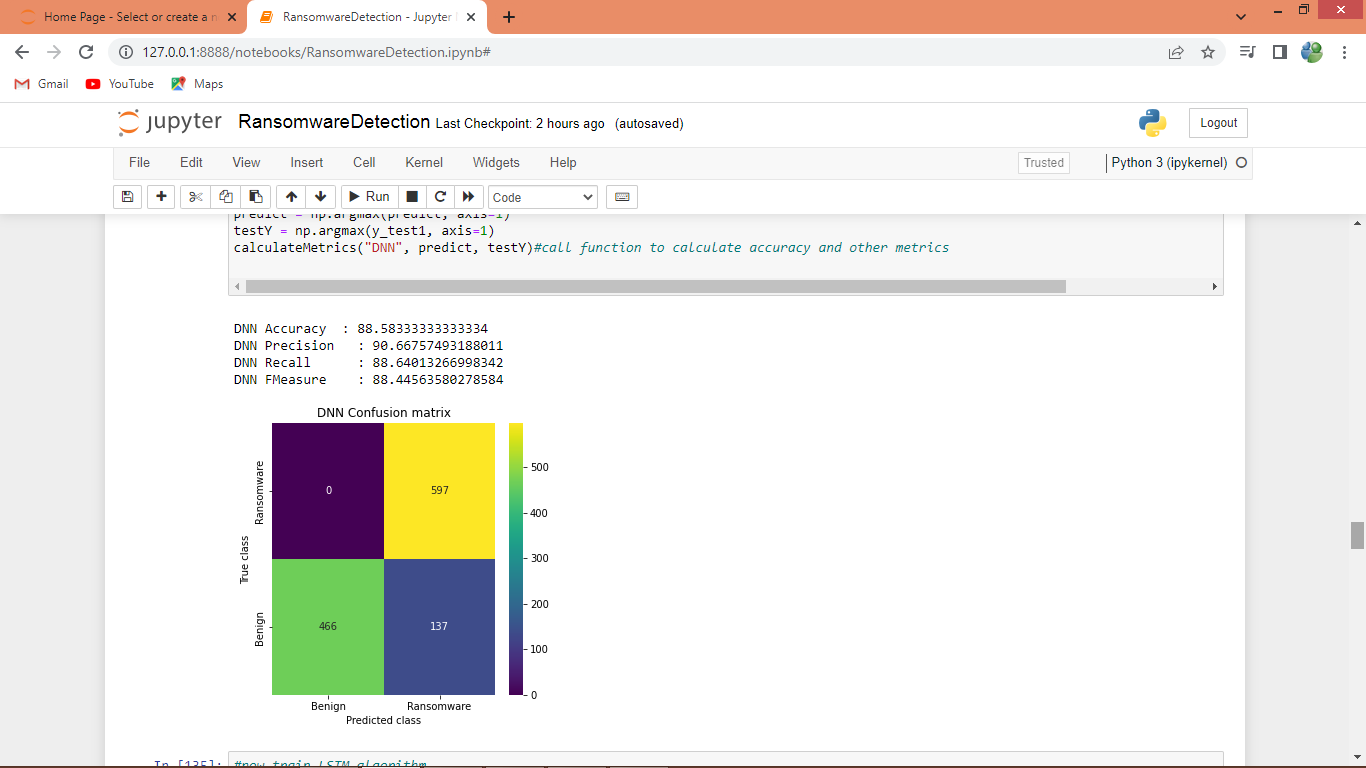
In above screen Random Forest got 98% accuracy



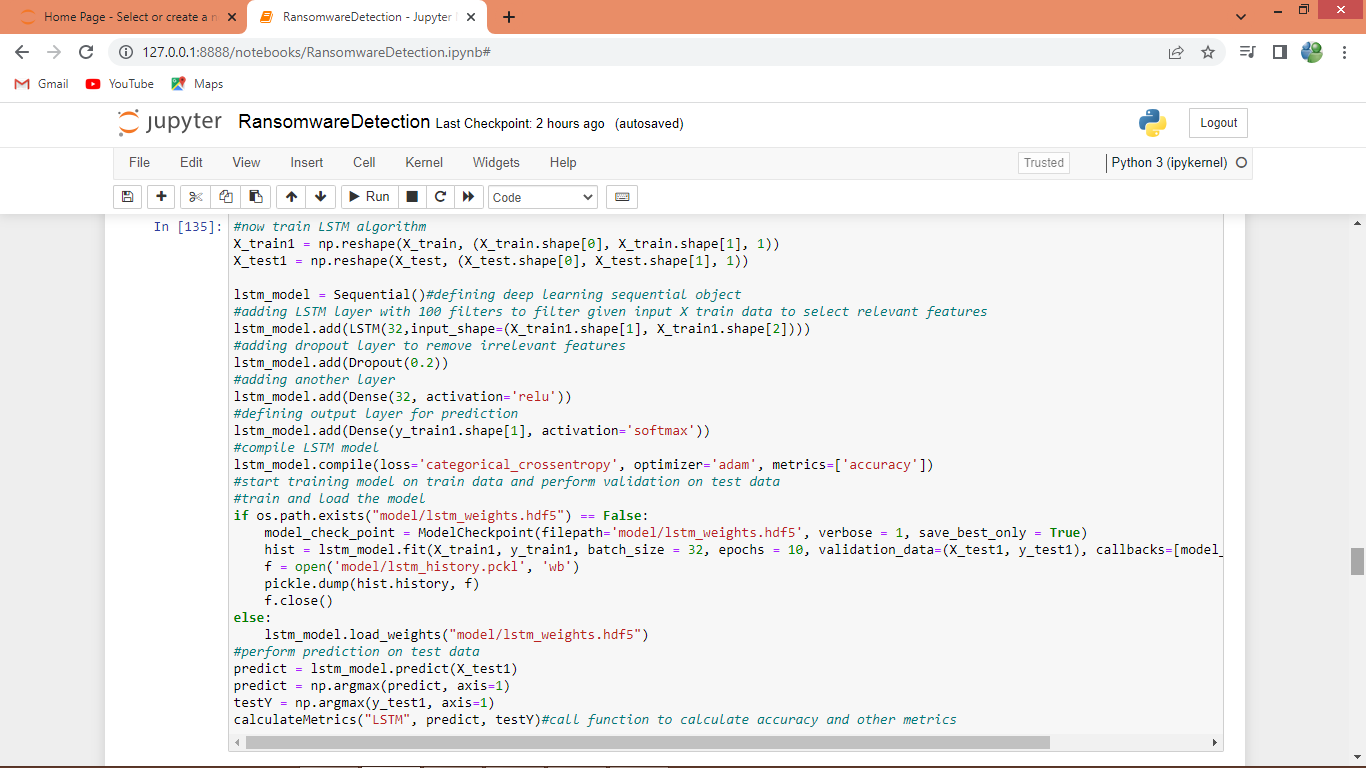
In above screen XGBOOST also got 98% accuracy



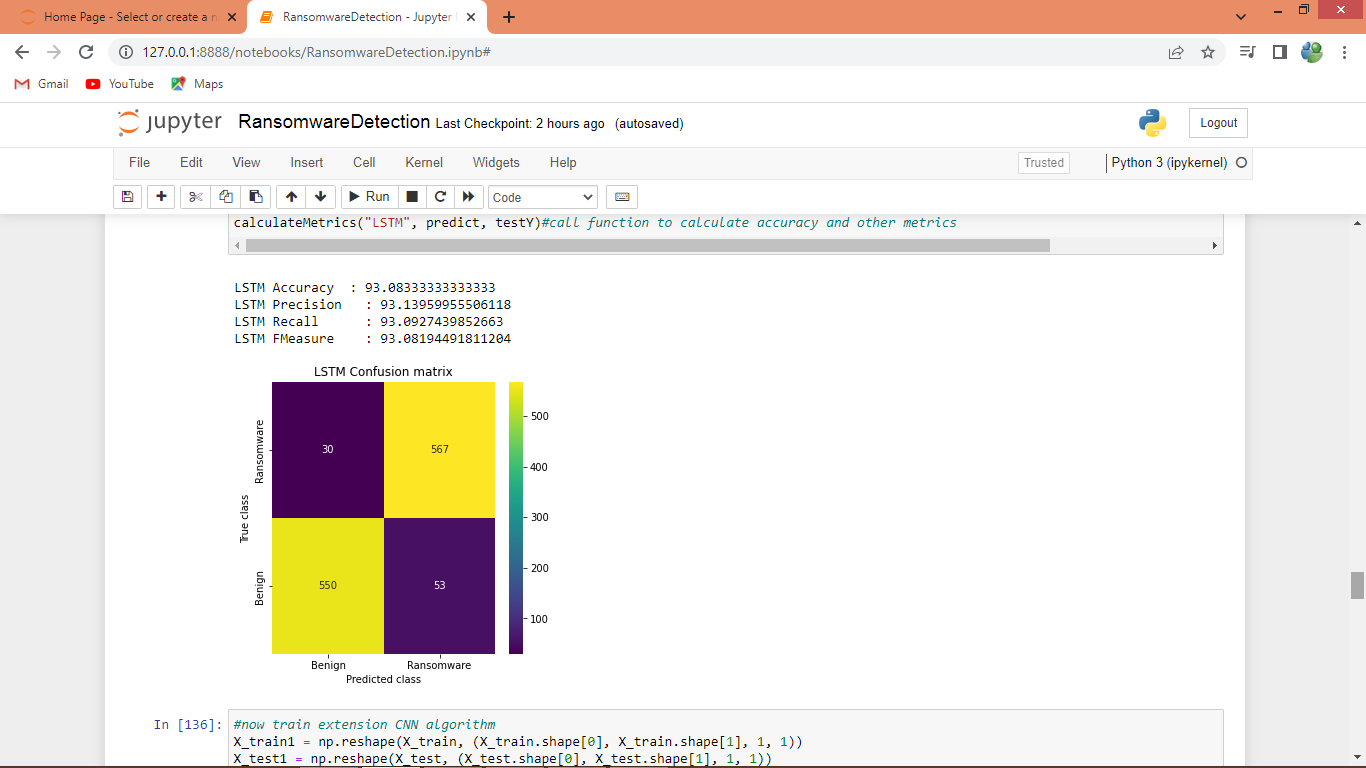
In above screen training DNN algorithm and below is the DNN output



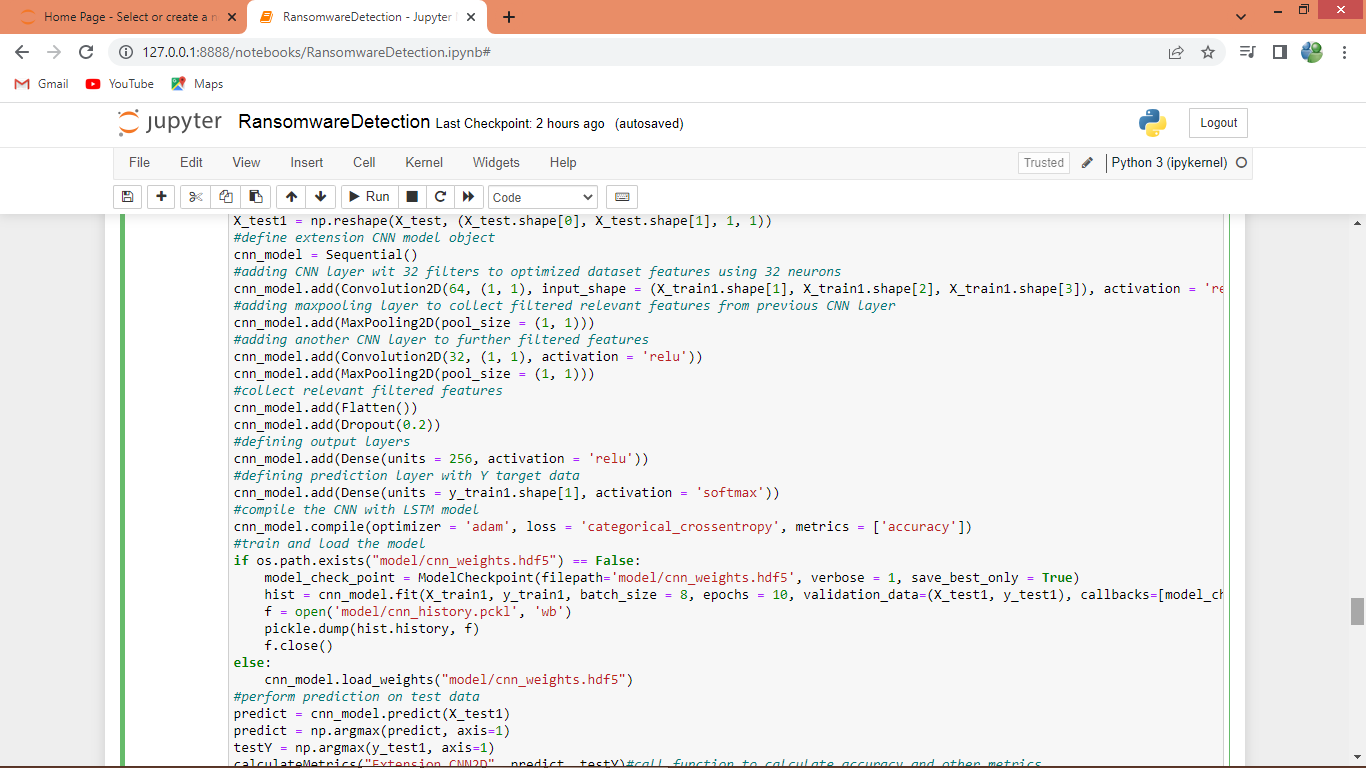
In above screen DNN got 88%accuracy



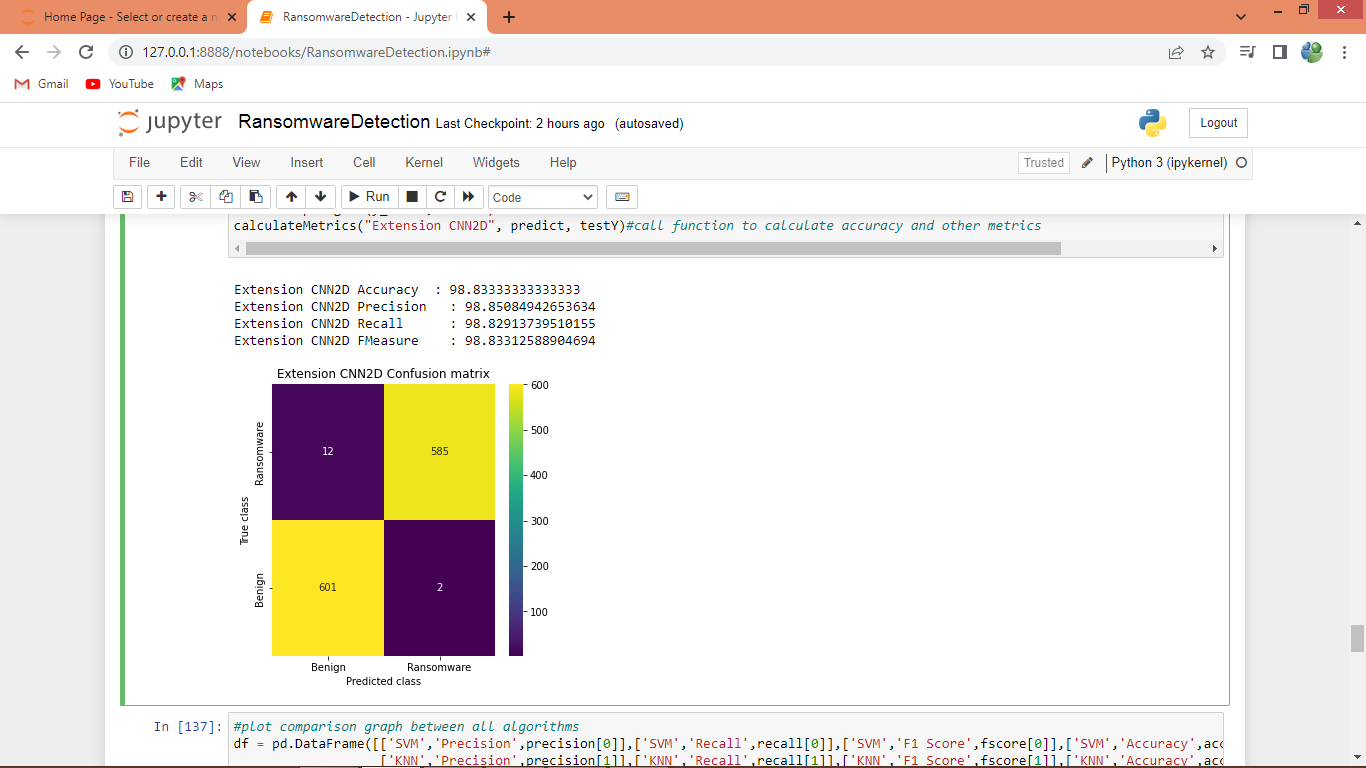
In above screen training LSTM algorithm and after executing above block will get below output



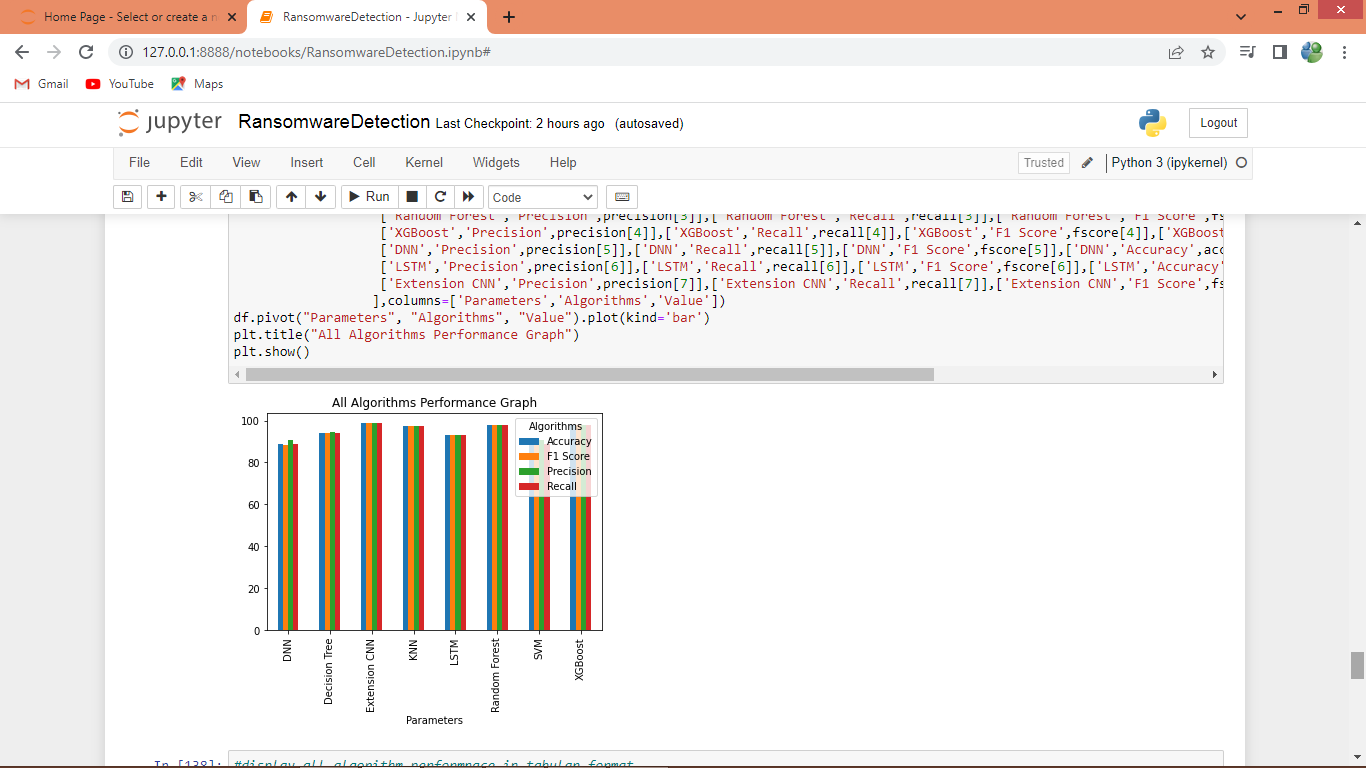
In above screen LSTM got 93% accuracy



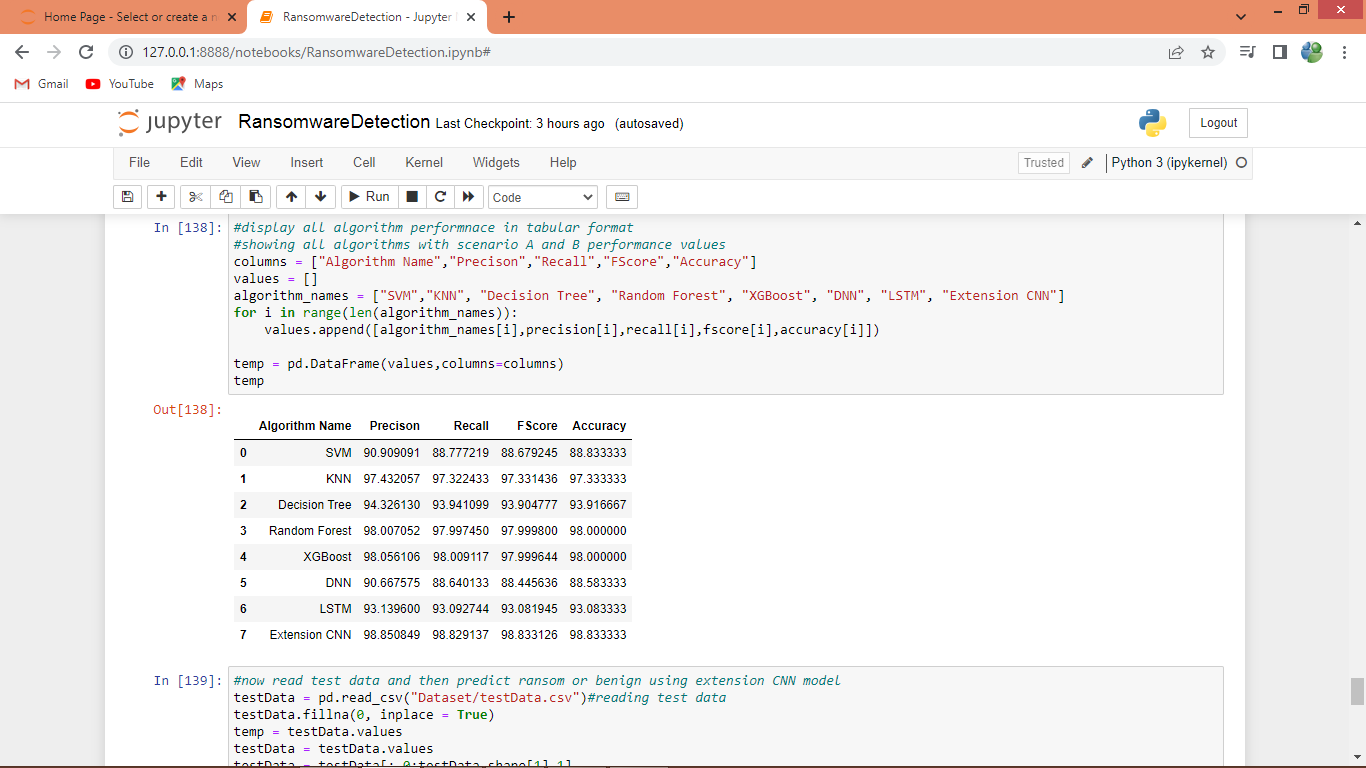
In above screen defining extension CNN2D algorithm and after executing this block will get below output



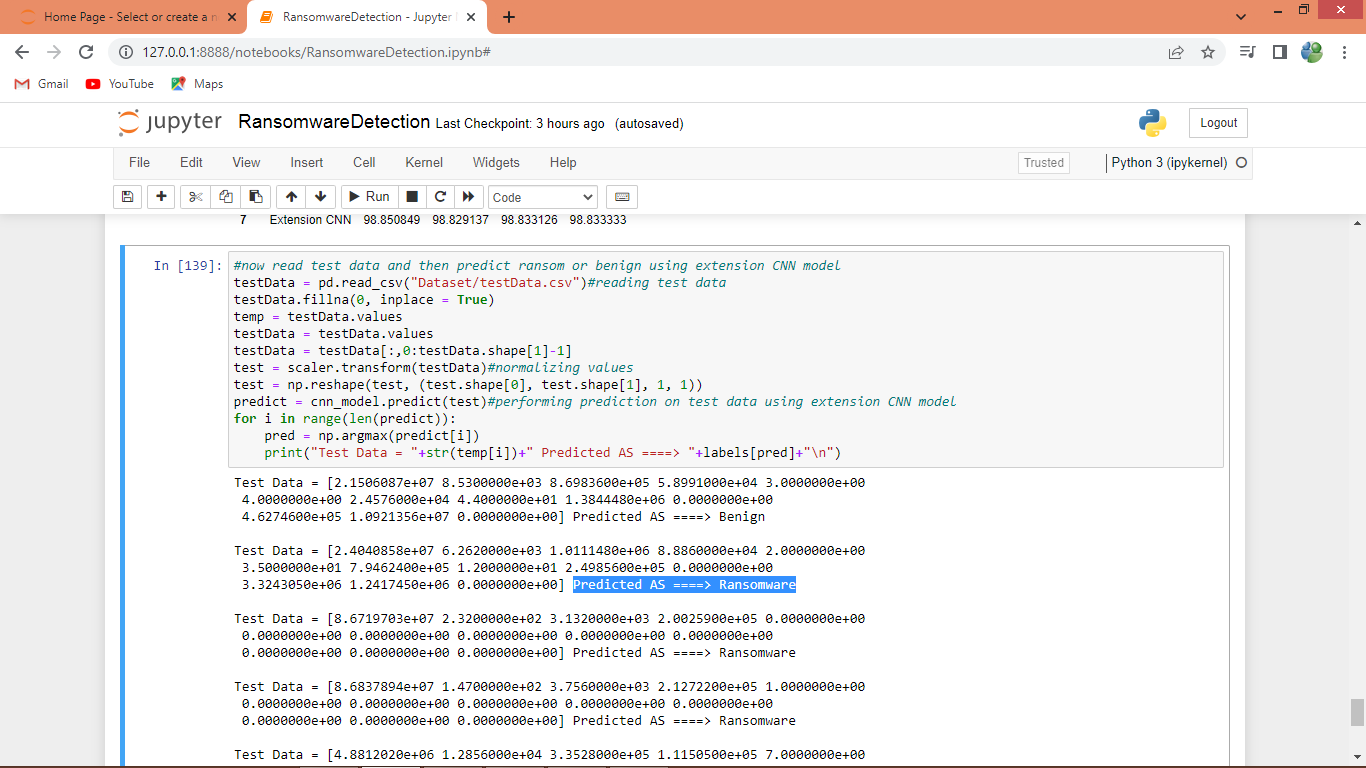
In above screen extension CNN2d got 98.83% accuracy and in all algorithms extension CNN got high accuracy



In above graph x-axis represents algorithm names and y-axis represents accuracy and other metrics in different colour bars and in all algorithms Extension CNN got high accuracy



In above screen displaying all algorithms performance in tabular format



In above screen reading test data and then using extension CNN algorithm object performing prediction on test data and then in output before arrow symbol =🡺 we can see Test data values and after arrow symbol we can see predicted values as ‘Ransomware or Benign’.