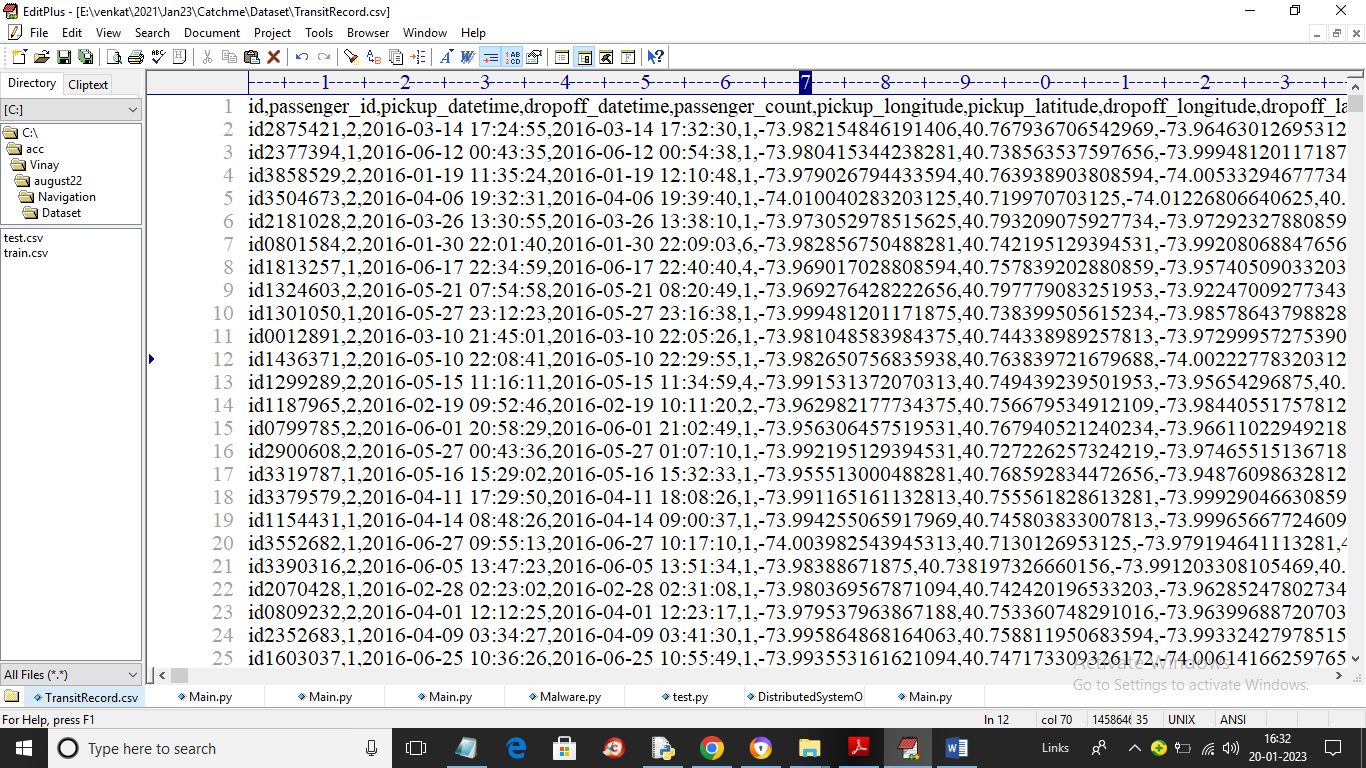
Catch Me If You Can: Detecting Pickpocket Suspects from Large-Scale Public Transit Records

Always while using public transport we will be in danger of pick pocketing and there is no direct way to identify pick pocketing person. So in propose paper author is using Automated Fare Collection and Transit Record dataset which consists passenger travelling routes and by analysing person travelling behaviour we can identify weather person is normal or pick pocketing.

In propose paper to detect person abnormal behaviour author employing Two-Step SVM which analysing person behaviour by using travelling frequency and Short Rides. If person has too many travelling frequency with short rides then that person will be pick pocketing suspect. So in this paper first we are applying Two-Step algorithm to identify person behaviour and then employing SVM algorithm on person behaviour to calculate suspect prediction Precision and FSCORE.

As existing algorithm we are using One Class SVM (OCS) algorithm to identify person behaviour and then training with SVM, Decision tree and Logistic Regression and then comparing OCS performance with propose Two-Step SMV.

To train algorithms author is using Transit passenger record dataset but this exact dataset not available on internet so we are using passenger dataset and in below screen we are showing dataset details



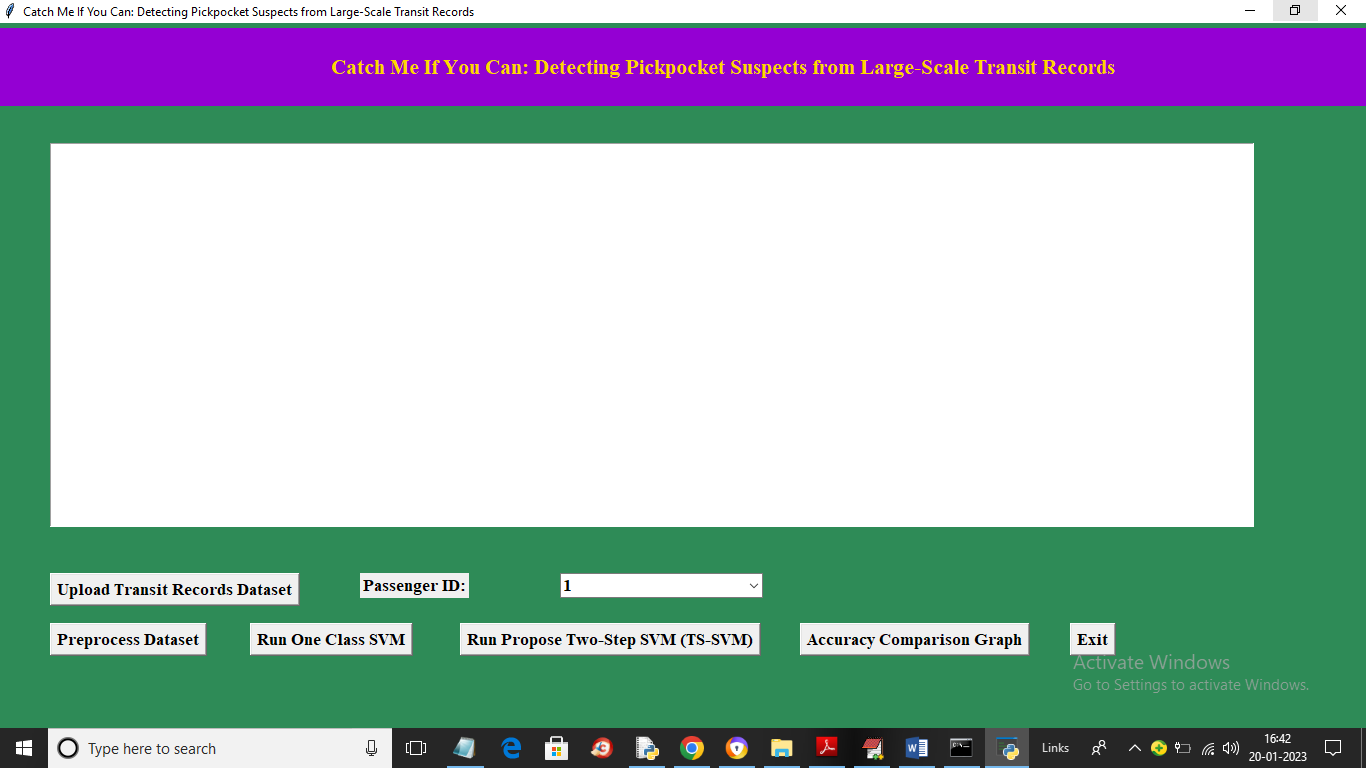
In above dataset screen first row contains dataset column names and remaining rows contains dataset values and this values has passenger traveling latitude and longitude with trip duration.

To implement this project we have designed following modules

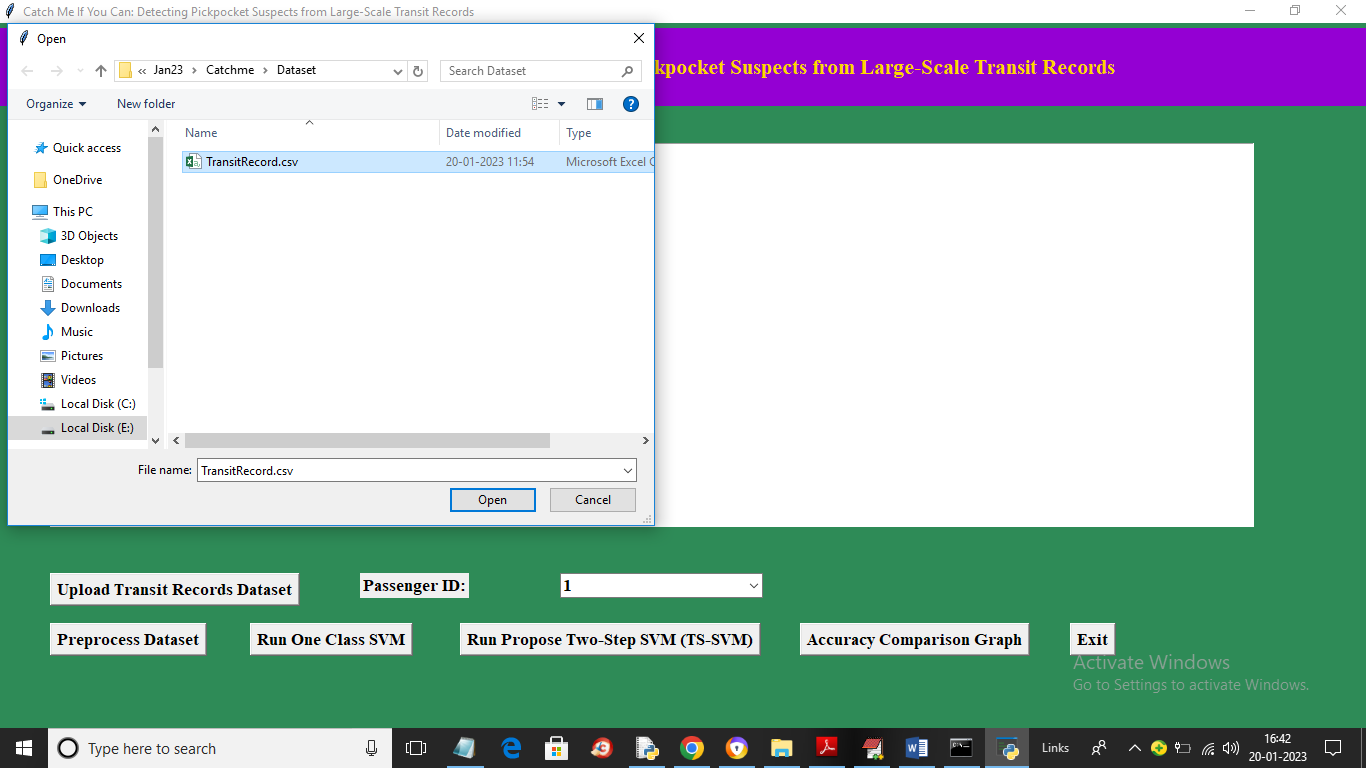
1. Upload Transit Records Dataset: using this module we will upload dataset to application
2. Preprocess Dataset: using this module we will select passenger ID from the drop down box and then application will read all rows from the dataset for selected passenger and then remove missing values and then normalize the values
3. Run One Class SVM: using this module we will apply OCS (one class SVM) to identify passenger behaviour and then apply SVM, Decision Tree and Logistic Regression to calculate suspect prediction precision and FSCORE values
4. Run Propose Two-Step SVM (TS-SVM): using this module we will apply TS-SVM (Two Step SVM) to identify passenger behaviour and then apply SVM, Decision Tree and Logistic Regression to calculate suspect prediction precision and FSCORE values
5. Accuracy Comparison Graph: using this module we will plot comparison graph between existing OCS and propose TS-SVM algorithms.

SCREEN SHOTS

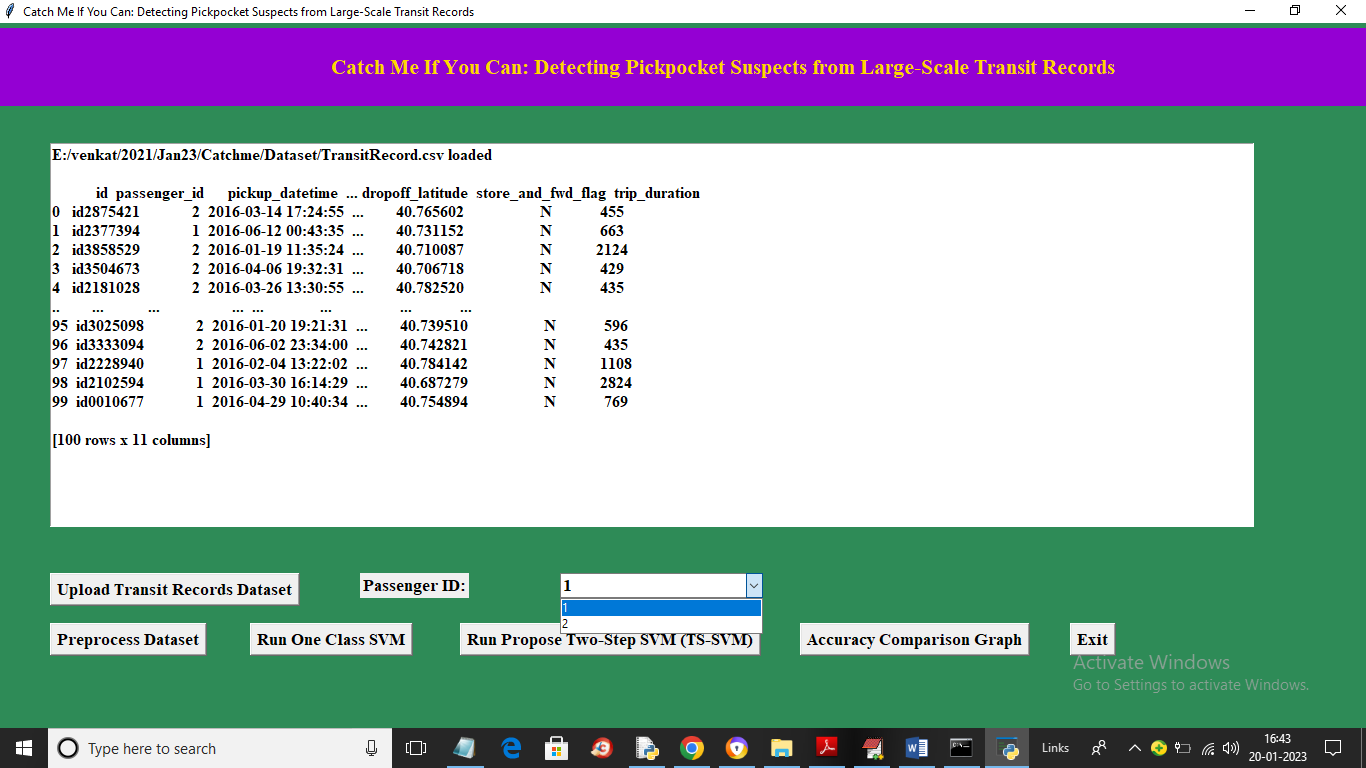
To run project double click on ‘run.bat’ file to get below screen



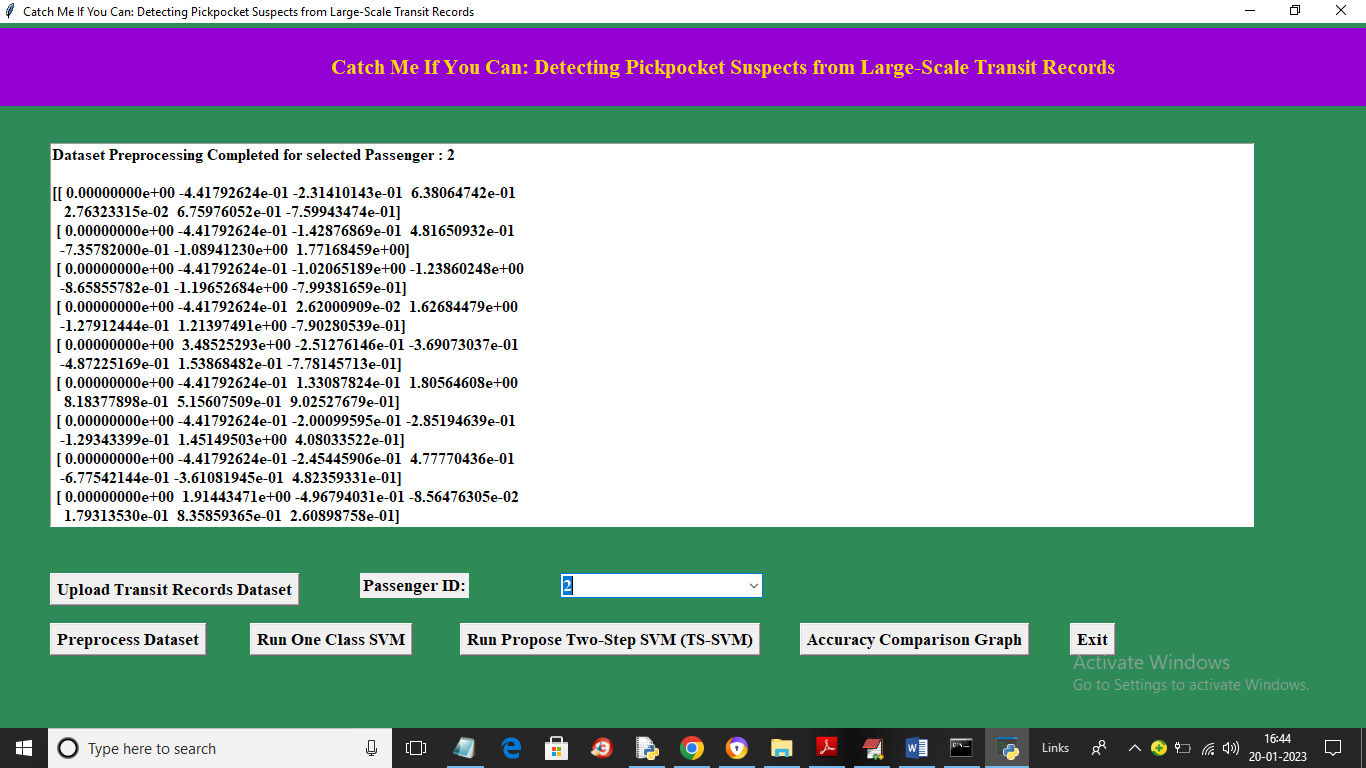
In above screen click on ‘Upload Transit Records Dataset’ button to upload dataset and get below screen



In above screen selecting and uploading dataset file and then click on ‘Open’ button to load dataset and get below output



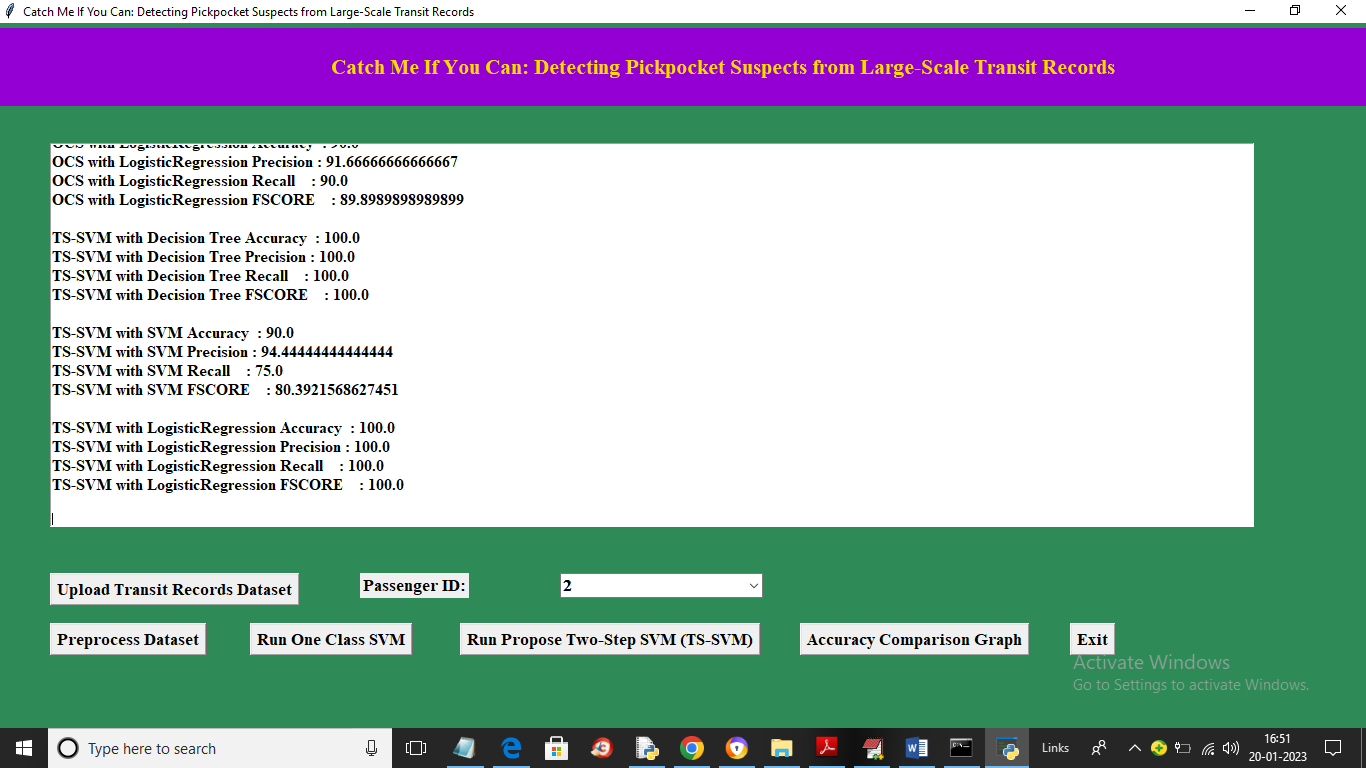
In above screen dataset loaded and now from drop down box select any passenger ID and then click on ‘Preprocess Dataset’ button to read all records from selected passenger and then normalize values to get below output



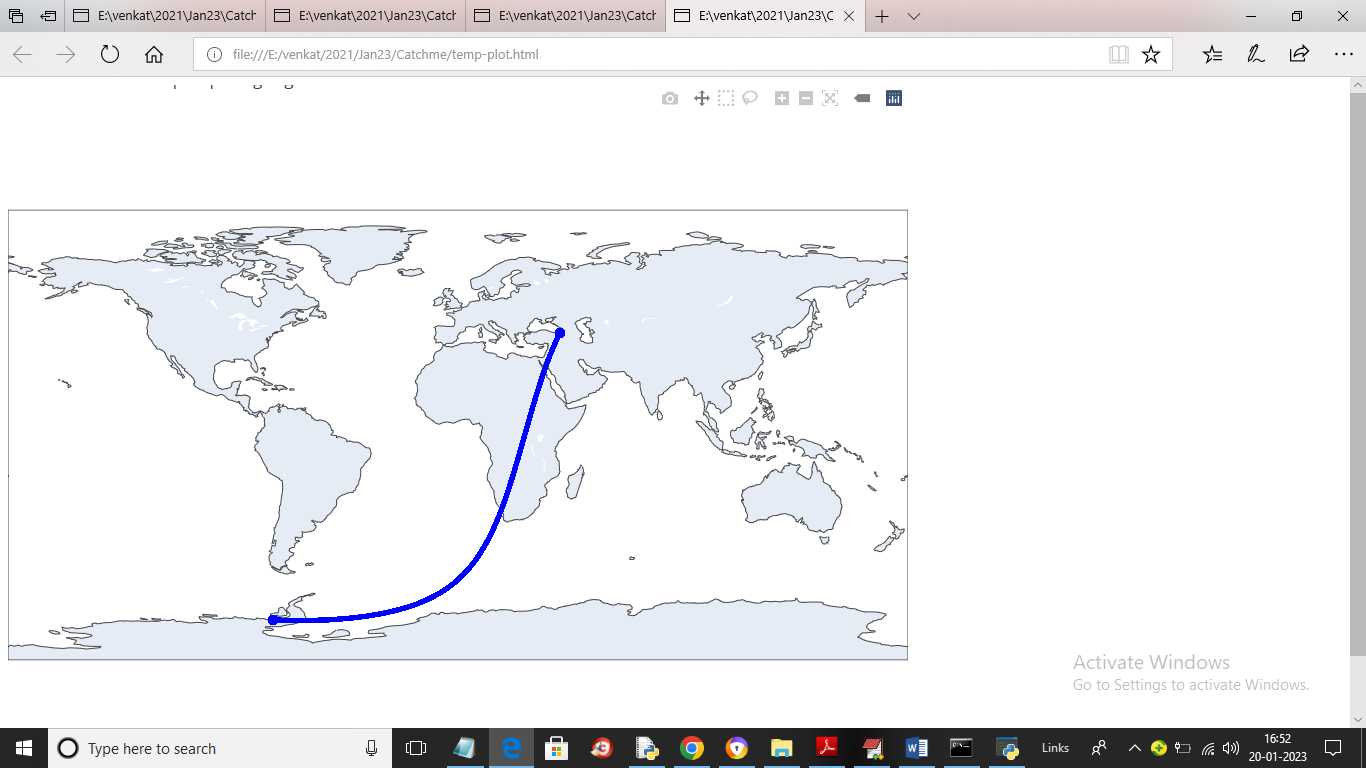
In above screen I selected passenger ID as 2 and then we got all normalize values for passenger 2 and now click on ‘Run One Class SVM’ button to get below output



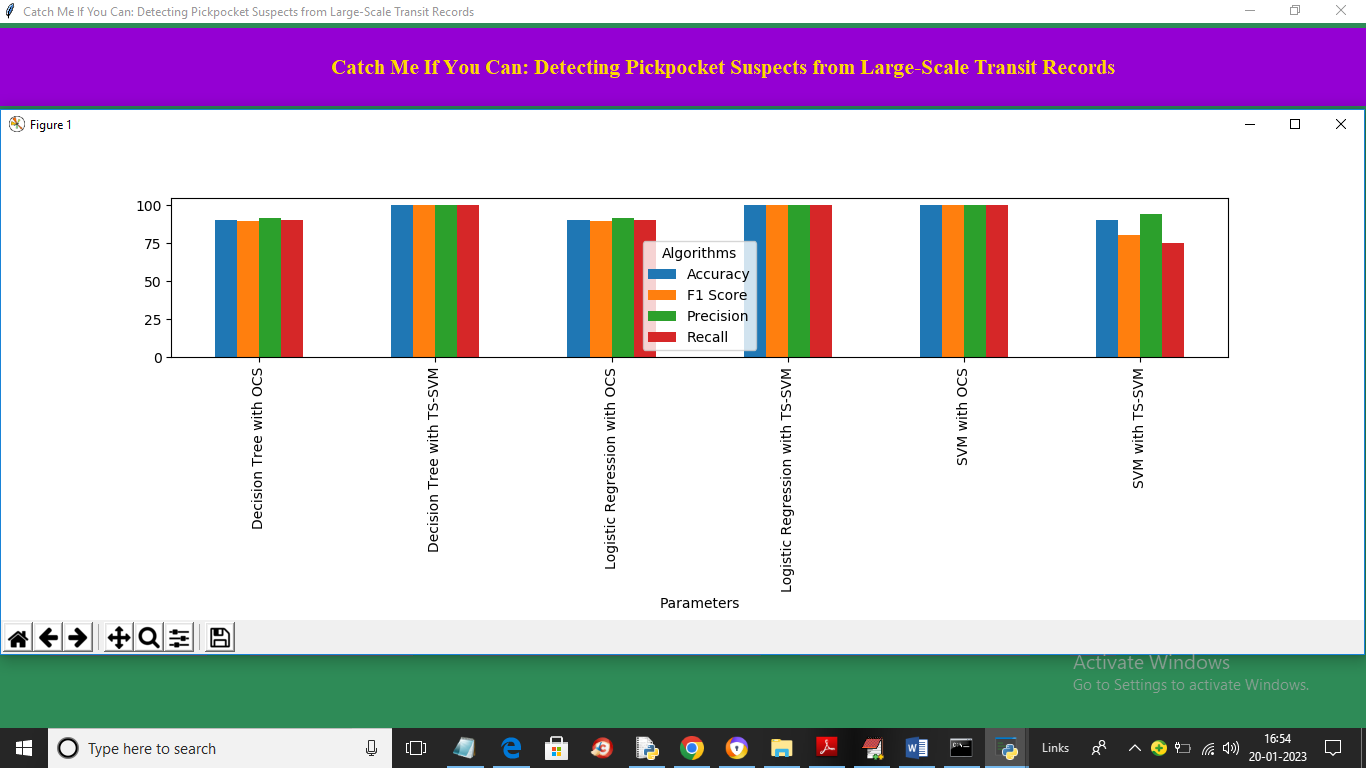
In above screen in first line we can see passenger 2 is pick pocket suspected and we can see OCS accuracy of decision tree as 90% and other metrics using Decision Tree, SVM and Logistic Regression and now click on ‘Run Propose Two-Step SVM (TS-SVM)’ button to calculate passenger behaviour and get below accuracy values



In above screen for TS-SVM accuracy of decision tree is 100% and we can see metric for all other algorithm and in below browser screen we can see passenger travel details



In above graph passenger travel to same route again and again so we can suspect him as Pick Pocket and now click on ‘Accuracy Comparison Graph’ button to get below output



In above graph x-axis represents algorithm names and y-axis represents accuracy and other metric in different colour bars and we can see propose TS-SVM got high performance or accuracy compare to existing OCS